ULTRIX/SQL

digital

ULTRIX/SQL NET User's Guide

ULTRIX

ULTRIX/SQL NET User's Guide

This manual describes how to use the network management component of ULTRIX/SQL.
ULTRIX/SQL Version 1.0 is based on INGRES Release 6.2.

The following are trademarks of Digital Equipment Corporation:

digital	DECUS	ULTRIX Worksystem Software
	DECwindows	VAX
CDA	DTIF	VAXstation
DDIF	MASSBUS	VMS
DDIS	MicroVAX	VMS/ULTRIX Connection
DEC	Q-bus	VT
DECnet	ULTRIX	XUI
DECstation	ULTRIX Mail Connection	

UNIX is a registered trademark of AT&T in the USA and other countries.

Network File System and NFS are trademarks of Sun Microsystems, Inc.

INGRES is a trademark of Ingres Corporation.

Contents

Preface

Purpose of this Document

Intended Audience

Organization of this Document

Compatibility with Remote Access to Rdb/VMS

Associated Documents

Conventions

References to Products

1 Accessing Remote Databases

2   Initializing ULTRIX/SQL NET

3   The ULTRIX/SQL NET Utility netu

A ULTRIX/SQL NET Commands and Utilities

Glossary

Index

Preface

___

Purpose of this Document

The *ULTRIX/SQL NET User's Guide* describes how to access remote databases over a network and how to use the ULTRIX/SQL NET management utility, **netu**.

The Guide also covers the way ULTRIX/SQL NET fits into the family of ULTRIX/SQL products. The ULTRIX/SQL family includes:

*   The ULTRIX/SQL relational database management system

*   ULTRIX/SQL NET, which connects you to databases on remote computer systems

*   Remote Access to Rdb/VMS, which connects you to Rdb/VMS databases

Intended Audience

This manual is intended for end-users of ULTRIX/SQL. It describes ULTRIX/SQL NET and explains how to use it to access local, remote, and Rdb/VMS databases on a network. It also explains how to use the **netu** utility to set up authorization to accounts on remote nodes.

This manual is also intended for the ULTRIX/SQL System Administrator and the Database Administrator. It explains how to install and maintain ULTRIX/SQL NET and how to define remote nodes to ULTRIX/SQL NET using the **netu** utility.

Organization of this Document

The *ULTRIX/SQL NET User's Guide* is divided into the following parts:

*   Chapter 1 explains how to access remote databases.

*   Chapter 2 shows system administrators how to initialize ULTRIX/SQL NET.

*   Chapter 3 shows system administrators how to use **netu** to define remote ULTRIX/SQL nodes and shows users how to define authorizations to their accounts on those nodes.

- Appendix A provides listings showing which operating system commands are and are not available for use with ULTRIX/SQL NET.

- The Glossary defines many of the terms used throughout this manual.

Compatibility with Remote Access to Rdb/VMS

This document provides examples of the syntax used to access Rdb/VMS databases. For more information, refer to your documentation for Remote Access to Rdb/VMS. Remote Access to Rdb/VMS is a VMS layered product installed on a VMS system running Rdb/VMS, which is connected to your ULTRIX/SQL system(s).

Associated Documents

The following associated manuals are included in your ULTRIX/SQL base system documentation set:

ULTRIX/SQL Database Administrator's Guide
ULTRIX/SQL NET User's Guide
ULTRIX/SQL Operations Guide
ULTRIX/SQL Reference Manual
ULTRIX/SQL Release Notes

Conventions

The following conventions are used to describe syntax in this manual:

- **Boldface** type is used to identify reserved words and required symbols and punctuation in syntax that must be typed as shown when used. Boldface is also used to indicate data types and key names. In sample code, boldface is used to indicate user input.

- Words in *italics* within text and syntax diagrams represent variable elements of syntax that are to be supplied by the program or the user. Italics are also used within text to introduce new terminology or to show emphasis.

- Double quotes (" ") within the general text indicate a specific value of a parameter. Double quotes (" ") and single quotes (' ') within syntax and in code examples have specific meanings within the context of SQL or a host programming language.

- Reserved words are shown in boldface, lowercase letters (except in host language examples, where embedded SQL statements appear in uppercase to distinguish them from the host language code). Although ULTRIX/SQL does not actually distinguish between uppercase and lowercase in reserved words, it does convert any uppercase letters to lowercase. This is true only for reserved words. Variables are case sensitive.

- This documentation uses generic keyboard key names. The key names on your particular keyboard may vary slightly from those used in this documentation. Key names joined by a hyphen (such as **Control-P**) indicate that the user is to press the named keys simultaneously.

- Syntax diagrams may continue over several lines. Line wraps and additional lines in statement and command line syntax are indented under the first line of the statement or command.

- Clauses enclosed in square brackets ([ ]) within syntax diagrams are optional.

- Clauses or arguments enclosed in braces ({ }) within syntax diagrams are optional and can be repeated zero or more times.

- Clauses or reserved words separated by vertical bars (|) within syntax diagrams indicate lists from which one element is to be chosen.

- Examples of code are separated from the text and are shown in a special, constant-width typeface.

- *Pseudocode*, a description of an operation without the actual code, is shown in *italics* within examples. This generic program code is used to clarify overall syntax structure without unnecessary detail.

- Within examples, the percent sign (%) represents an operating system prompt, although your system may use another customized prompt.]

References to Products

The ULTRIX/SQL documentation to which this manual belongs often refers to products by their abbreviated names:

- ULTRIX/SQL refers to ULTRIX/SQL database software and to its implementation of the SQL language. (Repetitive occurrences of ULTRIX/SQL have been shortened to SQL.)

- Rdb/VMS refers to VAX Rdb/VMS database software.

Accessing Remote Databases 1

1.1 Overview

This chapter explains how to access various kinds of remote databases. You can access:

- Remote ULTRIX/SQL databases with ULTRIX/SQL NET

- Remote Rdb/VMS databases with ULTRIX/SQL NET and Remote Access to Rdb/VMS product

This chapter assumes that you are familiar with accessing local databases. If you are not familiar with these operations, see the *ULTRIX/SQL Reference Manual.*

1.2 Before Starting ULTRIX/SQL

The type and location of the database you want to access determines which ULTRIX/SQL components you need:

- To access any ULTRIX/SQL database, you must have ULTRIX/SQL installed on your local computer.

- To access any database on a remote computer in a network, you must also have ULTRIX/SQL NET initialized on your local computer and each remote computer. See Chapter 2 for details.

  The nodes on the network must also have been defined by the ULTRIX/SQL System Administrator using the **netu** utility.

  You must also use the **netu** utility to define your authorization to your accounts on each remote node that contains a database you wish to access. See Chapter 3 to see how to use **netu** to define your authorization.

1.2.1 Setting Environment Variables

When ULTRIX/SQL was installed on your computer, some *environment variables* variables; setting were set that affect the operation of the program and the way data are handled. For example, there are variables that set the time zone and the type of currency symbols you use.

To use ULTRIX/SQL NET, you need to know the values given to these environment variables. See your ULTRIX/SQL System Administrator for this information.

1.2.2 Defining Your Terminal

One important environment variable, TERM_INGRES, defines the type of terminal you are using and enables you to configure your keyboard to your own personal preferences.

Normally, your ULTRIX/SQL System Administrator defines your terminal characteristics. However, you might need to redefine a terminal at a remote site or on a different machine in order to access one of your databases. For information on defining a terminal, refer to the *ULTRIX/SQL Reference Manual.*

1.3 ULTRIX/SQL NET Setup Requirements

Before you work with any database stored on another computer, certain ULTRIX/SQL NET initialization and setup requirements must be met.

1.3.1 Initialization Requirements

The computer on which you are entering commands and the remote computer that contains the database you want to access must be connected by a network.

ULTRIX/SQL NET must be initialized on your local node and on each remote node.

1.3.2 Defining Remote Nodes

Using the **netu** program, either you or the ULTRIX/SQL System Administrator must define all the remote nodes on the network that contain databases you want to access.

Defining a remote node gives it a *v_node* name. You use this *v_node* name to direct ULTRIX/SQL to the node that contains the database you want.

1.3.3 Defining Authorization to Access Remote Databases

Before you can access a database on a remote node, you *must*:

•   Have an account on the remote node (or access to a public guest account).

•   Use **netu** on your local node to define your authorization to your account on the remote node. (Alternatively, the ULTRIX/SQL System Administrator must define your access to a public guest account on that remote node.)

You need to define your authorization to your account on a specific remote node only once. Your authorization remains in effect until you delete it.

See Chapter 3 for a full description of the netu program and information on defining authorizations. Alternatively, you can consult your ULTRIX/SQL System Administrator for assistance.

1.4 Accessing Remote Databases

The remainder of this chapter discusses in detail how to access:

* Remote ULTRIX/SQL databases.

* Remote Rdb/VMS databases through ULTRIX/SQL NET and Remote Access to Rdb/VMS.

The general syntax for accessing a remote database is:

command[*v_node*::]*dbname*[*/server_type*]

Table 1-1: Access Syntax

Option	Function
command	Any command used to invoke an ULTRIX/SQL software tool, such as sql.
v_node::	The virtual node name of the remote node on which the database is located (followed by two colons). The *v_node* name implies the actual network node address and the network protocol, which are linked when the node name is defined with the netu utility. (See Chapter 3.)
dbname	The name of the database.
server_type	The type of server being accessed at the local or remote site. The server_type designated for the database management system server is **ingres**. For a list of the possible server types, see Table 1-2.

Table 1-2: ULTRIX/SQL Server Types

Server_Type	ULTRIX/SQL Server or Remote Access
ingres	ULTRIX/SQL database management system
rdb	Remote Access to Rdb/VMS

The default server type is defined by the environment variable II_GCNxx_SVR_TYPE. If this environment variable is not defined, the default is **ingres**.

1.5 Accessing a Remote ULTRIX/SQL Database

To access an ULTRIX/SQL database on a remote node through ULTRIX/SQL NET, you need to know the virtual name (*v_node*) of the remote node and the name of the database. ULTRIX/SQL NET handles the details of gaining access to data on a remote node.

1.5.1 Syntax for Accessing a Remote ULTRIX/SQL Database

The syntax for accessing a remote database is:

command v_node::dbname[/server_type]

You *must* specify the virtual node name (*v_node*) of the remote node on which the database is located. The *v_node* name must be followed by two colons.

There is no need to specify the type of server being accessed as the default; unless reset, the default is **ingres**.

1.5.1.1 Example

This example calls ULTRIX/SQL and accesses the ULTRIX/SQL database parts on remote node paris:parts.

```
sql paris::parts.
```

The only difference between accessing a local ULTRIX/SQL database and accessing a remote ULTRIX/SQL database is that you must specify the *v_node* name of the remote node.

1.5.1.2 Working with Databases on Remote Nodes

Once you have accessed the remote database, you can begin working with it exactly as you would with a database located on your own computer.

For example, your organization can have two nodes on a network, one for production tasks and the other for software development. Your ULTRIX/SQL System Administrator might designate the parts node as "parts" and the software development node as "devel."

If you normally work on the production node "prod," you would log in there and use any of the ULTRIX/SQL utilities to work on your databases.

To access the "proddb" database stored on the production node, you could execute the command:

```
sql proddb
```

Suppose you want to see the "devdb" database on the "devel" software development node. With ULTRIX/SQL NET you can access the database easily from the production node, using the command:

```
sql devel::devdb
```

ULTRIX/SQL NET requires only that you specify a remote virtual node name, followed by two colons, followed directly by the database name. All other aspects of using ULTRIX/SQL remain the same.

1.5.2 Commands Not Available Over the Network

Some utilities typically used by database administrators cannot be used over the network. These include **accessdb**, **createdb**, and **rollforwarddb**. Appendix A lists both the ULTRIX/SQL commands and utilities supported by ULTRIX/SQL NET and those that cannot be used on remote databases.

1.6 Accessing a Remote Rdb/VMS Database

You can access a remote Rdb/VMS database through ULTRIX/SQL NET and the Remote Access to Rdb/VMS product. The Remote Access to Rdb/VMS product resides on the VMS node.

1.6.1 Syntax for Accessing Remote Rdb/VMS Databases

The syntax for accessing a remote Rdb/VMS database through the Remote Access to Rdb/VMS is:

command v_node::dbname/rdb

You must specify the virtual node name (*v_node*) of the remote node on which the database is located. The *v_node* name must be followed by two colons.

1.6.2 Example

To call ULTRIX/SQL and access the Rdb/VMS personnel database on a remote node whose *v_node* name is "london," type the following command:

```
sql london::personnel/rdb
```

In this case, you must specify both the remote *v_node* name of the remote node and the *server_type* (that is, rdb).

1.7 Database Access Summary

You use a common syntax to access:

- Remote ULTRIX/SQL databases with ULTRIX/SQL NET

- Remote Rdb/VMS databases with ULTRIX/SQL NET and Remote Access to Rdb/VMS

1.7.1 Syntax

The syntax for accessing a remote database is:

commandv_node::dbname[/server_type]

1.7.2 Local ULTRIX/SQL Syntax

sql *dbname*

isql *dbname*

1.7.3 Remote ULTRIX/SQL Syntax

sql *v_node::dbname*

isql *v_node::dbname*

1.7.4 Remote Rdb/VMS Syntax

sql *v_node::dbname*/**rdb**

isql *v_node::dbname*/**rdb**

Initializing ULTRIX/SQL NET   2

2.1 Overview

This chapter is intended for the ULTRIX/SQL System Administrator who initializes ULTRIX/SQL NET. You should read the chapter thoroughly before you begin the initialization to ensure that you have all the information you need.

To initialize ULTRIX/SQL NET, you must:

- Ensure that network software is installed on your local node and each remote node that has an ULTRIX/SQL database to which you want remote access

- Initialize ULTRIX/SQL NET on your local node

- Initialize ULTRIX/SQL NET on each remote node you intend to access

To enable users to access databases on remote nodes, you must:

- Establish ULTRIX/SQL user accounts and operating system accounts at your local node and at each remote node.

- At each local node, define all the remote nodes your users intend to access. Use the network utility program **netu** (described in Chapter 3) to define remote nodes.

- Run **netu** at each user's local node and define users' global authorizations to global "guest accounts" at each remote node.

Note

Users can run **netu** and define their authorizations to each of their private remote node accounts. An ULTRIX/SQL System Administrator cannot define a user's authorization to a private account on a remote node. Only the individual user can do this.

The remainder of this chapter describes how to initialize ULTRIX/SQL NET.

2.2 Initializing and Testing Your Network

Before you install ULTRIX/SQL NET, check the documentation for your network software to ensure that your network is installed correctly and that you can connect from your local node to each remote node.

To install ULTRIX/SQL NET for ULTRIX with TCP/IP as its network protocol, you must ensure that the /etc/hosts file (or other list of network host addresses) contains the network addresses of all hosts that are to use ULTRIX/SQL NET. Note that ULTRIX/SQL NET does *not* accept host names containing embedded dashes or periods.

You can establish aliases in the /etc/hosts file. Refer to your ULTRIX operating system documentation for details.

Test TCP/IP fully before initializing ULTRIX/SQL NET by completing an **rlogin, ftp** or **telnet** command and connecting to another node on the network.

If you plan to use DECnet communications protocol, make sure DECnet is installed and initialized on each required network node. DECnet can be used for some nodes and TCP/IP for others.

2.3 Initializing ULTRIX/SQL NET on Your Local Node

If you are installing ULTRIX/SQL for the first time, follow the standard ULTRIX/SQL installation procedures in your *ULTRIX/SQL Installation Guide*. To initialize ULTRIX/SQL use the instructions in the *ULTRIX/SQL Operations Guide*.

2.3.1 ULTRIX/SQL NET Initialization Procedure

During the ULTRIX/SQL initialization procedure, you will be prompted for certain information specific to ULTRIX/SQL NET.

2.3.1.1 Maximum Inbound and Outbound Connections

You will be prompted for the maximum number of inbound connection sessions and the maximum number of outbound connection sessions with the following prompts:

```
-------------------------------------------
| You will now be asked to specify the number of inbound and
| outbound ULTRIX/SQL NET sessions.
-------------------------------------------

Enter the maximum number (>1) of inbound connected ULTRIX/SQL NET
sessions for node_a <12>:

Enter the maximum number (>1) of outbound connected ULTRIX/SQL NET
sessions for node_b <12>:
```

The default number of inbound connected sessions is 12. The default number of outbound connected sessions is 12. One session means either a user on your local node connecting to a remote node through ULTRIX/SQL NET, or a user on a remote node connecting to your local node.

If any additional connection attempts are made above the total of the two numbers specified, the user attempting to make the connection will receive a "Failed to connect ..." type of error message. However, do not enter a total number greater than you need. Allowing many connections will affect your local allocation of resources.

You can amend these figures once you have set them during installation without running the build procedure again provided you do not exceed the original total number of outbound and inbound connections.

To amend the number of connections, alter the II_GCCxx_OBMAX "nn" and II_GCCxx_IBMAX "nn" environment variables.

2.3.1.2 Amending the Number of ULTRIX/SQL NET Connections

To amend the number of incoming and outgoing connections, you may use the **ingunset** and **ingsetenv** utilities.

The syntax for resetting the inbound connections is as follows:

ingsetenv II_GCC[xx]_IBMAX [nn]

where xx is the two-character installation code for II_INSTALLATION and [nn] is the maximum number of inbound connections.

The syntax for resetting the outbound connections is as follows:

ingsetenv II_GCC[xx]_OBMAX [nn]

where xx is the two-character installation code for II_INSTALLATION and [nn] is the maximum number of outbound connections.

If you alter these variables after running the communications server, you must stop the communications server and restart it for these new limits to take effect.

If the total number of inbound and outbound connections specified in these environment variables is greater than the value specified during the build process, you need to first run **iistartup -init** to establish the appropriate resource levels.

2.3.1.3 ULTRIX/SQL NET Listen Addresses

During the initialization process, you are informed of the ULTRIX/SQL NET communications server listen addresses. The ULTRIX/SQL NET communications server listen address is dependent on the underlying network software type.

The ULTRIX/SQL NET communications server listen addresses are set by the II_GCC_TCP_IP and II_GCC_DECNET environment variables. The values of these variables are respectively the TCP/IP and DECnet port addresses.

To see the value of the port addresses, enter:

```
ingprenv  |grep II_GCC_TCP_IP

ingprenv  |grep II_GCC_DECNET
```

If you have defined this environment variable, ULTRIX/SQL NET uses its value, which must be a number greater than 1023. If you do not define this environment variable, ULTRIX/SQL NET uses the ASCII value of the two-letter installation code for II_INSTALLATION. The default value value of II_INSTALLATION is II. Its decimal ASCII equivalent is 7373.

To find out the value of an installation identifier, log in to that node and enter:

```
ingprenv |grep II_INSTALLATION
```

The following type of message is displayed:

```
"II_INSTALLATION"="II"
```

ULTRIX/SQL NET uses the decimal ASCII value of this string.

Depending on the network protocol you are using with ULTRIX/SQL NET, you receive protocol-specific messages and prompts. With TCP/IP and DECnet, during the initialization procedure you receive the following message:

```
Listen Address(es):
II_GCC_DECNET_II;exportII_GCC_DECNET;II_GCC_TCP_IP=II;exportII_GCC_TCP_IP

The TCP/IP port number is 7373
```

2.3.2 Starting ULTRIX/SQL NET

Once you have installed ULTRIX/SQL NET, it will start automatically whenever you start ULTRIX/SQL.

If ULTRIX/SQL NET is stopped (that is, if the ULTRIX/SQL NET communications server is shut down), you can start it again by using the **iistartup** command at the system prompt. Only the ULTRIX/SQL System Administrator can start the communications server.

2.3.3 Stopping ULTRIX/SQL NET

To stop the ULTRIX/SQL NET communications server, run the **netu** utility, specifying the communications server ID as the **netu** startup parameter, and choose the **Stop** option. Refer to Chapter 3 for a complete explanation of how to use **netu**.

2.4 ULTRIX/SQL NET Error Logging

By default, when you install ULTRIX/SQL NET, error messages and communications server startup/shutdown information messages, together with database management system server error messages, are logged in the $II_SYSTEM/sql/files... errlog.log file. Normal status messages from the communications server are not logged.

You can change the level of error and status message logging by altering the environment variable II_GCC[xx]_LOGLVL using **ingsetenv**. Only the ULTRIX/SQL System Administrator can alter this environment variable.

By defining the environment variable II_GCC[xx]_LOG to a file name, you can direct messages to both the errlog.log file and the file you have defined.

2.4.1 II_GCC[xx]_LOGLVL

II_GCC[xx]_LOGLVL determines different levels of informational and error logging. It is defined to an integer value. The default value is 4 and is set when you initialize ULTRIX/SQL NET.

The value of *xx* is the two-letter installation code defined in II_INSTALLATION.

If you define II_GCC[*xx*]_LOGLVL to a higher level, the errors and messages logged are cumulative (that is, you get all the messages appropriate to that level and to all lower levels).

The following table shows the different levels.

Table 3-1: IIGCC[*xx*]_LOGLVL Levels

Value	Description
0	SILENT No error messages Startup messages only
4	GCC START/STOP status messages Fatal GCC Errors that cause the GCC process to stop Connection-specific errors that cause a specific connection to be broken
6	Connection setup and termination messages for the Application Layer and Transport Layer (AL/TL)

The Layers referred to are the layers in the OSI/ISO model: Application, Presentation, Session, Transport, Network, Data Link, and Physical.

If you set II_GCC[*xx*]_LOGLVL to a higher value than level 4, normal communications server (GCC) status messages will be interleaved with GCC and database management system error messages in the errlog.log file. You must set II_GCC[XX]_LOGLVL before starting the communications server.

2.4.2   II_GCC[*xx*]_LOG

If you define II_GCC[*xx*]_LOG to a valid file name and II_GCC[*xx*]_LOGLVL remains at default level 4, errors and level 4 messages also are logged to that specified file as well as to errlog.log.

It is recommended that the file name be similar to iigcc[*xx*].log_*node name* where the value of *xx* is the two-letter installation code defined in II_INSTALLATION.

You should define II_GCC[xx]_LOG to a file name *before* running the communications server (that is, before running ULTRIX/SQL NET).

If you set II_GCC[xx]_LOG to a valid file name and *in addition* set II_GCC[xx]LOGLVL to a higher than the default value, GCC error messages *and* normal status messages appropriate to that level and all lower levels will be logged *both* in the errlog.log file and the file defined to II_GCC[xx]LOG.

Note

Please turn on message logging to obtain information from these log files.

2.4.3 Network Protocol Error Messages

The names of ULTRIX/SQL-supported network software protocols are set up automatically for you during the initialization procedure. When you start the communications server, ULTRIX/SQL NET seeks to initialize each of the network protocols in turn. Whenever one is not found, a message is sent to the ULTRIX/SQL error file, errlog.log. ULTRIX/SQL NET is simply advising you that no connection with that particular network software protocol can be made.

2.5 Initializing ULTRIX/SQL NET on Remote Nodes

Initializing ULTRIX/SQL NET at remote nodes is the same as a local node initialization. Simply log on to each remote node in turn and follow the steps outlined above according to the protocol you will be using.

Depending on your organization, the person who initializes ULTRIX/SQL NET on a remote installation might vary. Normally it will be the ULTRIX/SQL System Administrator for that installation.

2.6 Establishing User Accounts on Local and Remote Nodes

After initializing ULTRIX/SQL NET on each local and remote node, the ULTRIX/SQL System Administrator for each remote node must establish operating system accounts and ULTRIX/SQL user access at each remote node. It is assumed that you have previously set up operating system user accounts and ULTRIX/SQL user access at your own local node.

To establish ULTRIX/SQL user access, the ULTRIX/SQL System Administrator can use the **accessdb** program. The **accessdb** program is an ULTRIX/SQL forms-based program that lets you update user and database information in the ULTRIX/SQL master database, the **iidbdb**.

You *cannot* use **accessdb** with ULTRIX/SQL NET on one node to establish ULTRIX/SQL access on another node. The ULTRIX/SQL System Administrator for the remote node must establish accounts locally. In this way, local autonomy over database access is maintained.

2.7 Additional Setup Requirements

This section discusses additional requirements that you must complete before you can use ULTRIX/SQL NET.

2.7.1 Privileges

You must be the ULTRIX/SQL System Administrator to set up and start the communications server.

Refer to the *ULTRIX/SQL Operations Guide* for further information about user privileges, user accounts, and quotas.

2.7.2 Defining Remote Nodes and User Authorization

At the definition and authorization stage, your ULTRIX/SQL NET installation is ready to receive incoming connection requests from other nodes. However, before your local users can make outgoing connection requests—that is, before users can access databases on remote nodes—two further steps must be taken:

- All remote nodes must be defined. This definition is known as the remote node's virtual node, or *v_node*, name.

- User authorization to those remote nodes must be defined.

The system administrator and individual users define remote nodes and define authorization to those remote nodes using the ULTRIX/SQL NET utility program **netu** described in Chapter 3.

2.7.2.1 Remote Nodes and User Authorization Defined by the DBA

The ULTRIX/SQL System Administrator defines remote nodes with names to be used by all users. These are known as *global* virtual node (global *v_node*) names. The System Administrator also defines users' authorizations to any public "guest" accounts on remote nodes.

To perform these tasks, the System Administrator must run the **netu** program at each local node and define all the commonly accessible remote nodes that users intend to access. For example, if you have installed ULTRIX/SQL NET on three nodes, at each node you must define the other two nodes.

The definition of each remote node includes its *v_node* name, its actual node name or network address, and the network software type linking the nodes on the network. This information ensures that, when a user on the local node tries to access a database on a remote node, ULTRIX/SQL NET knows where to go and how to get there.

If there are public "guest" accounts at any of the remote nodes, the ULTRIX/SQL System Administrator also must use **netu** to define global (public) authorization to those remote public accounts.

2.7.2.2 Remote Nodes/User Authorization Defined by the User

Individual users also can define remote nodes. Although most users simply will use the global *v_node* name defined by the System Administrator, individual users can define remote nodes with their own private remote *v_node* names.

Individual users *must* define their own authorization to their remote private accounts. At a remote node, a user will have an account (that is, a user name or login) and a password. At the user's local node, the user must run **netu** and define his or her remote node user name and password. The System Administrator *cannot* define user authorization to a user's private account on behalf of that user.

2.7.3 Setting Default Nodes

You may set a default value for a remote node. The value of your local node is automatically set for you.

2.7.3.1 II_GCNxx_RMT_VNODE

As the ULTRIX/SQL System Administrator, you might want to direct all users' remote database requests to a specific node and eliminate the need for users to specify a *v_node* name when accessing databases on that remote node.

To do this, you must use **ingsetenv** and define the environment variable II_GCNxx_RMT_VNODE. For example, to direct all queries to a node named *haven* (assuming II_INSTALLATION = II), type:

```
ingsetenv II_GCNII_RMT_VNODE haven
```

This defines the default virtual node name of a remote node, where "II" is the value of II_INSTALLATION on the local node.

Set II_GCNxx_RMT_VNODE if you want users to connect to a remote node by default. Then users can access the remote database as if it were a local database. They need not specify the remote *v_node* name when accessing a database on a remote node.

When a user specifies a dbname, ULTRIX/SQL automatically places this logical before the database name and ULTRIX/SQL NET directs all queries to the specified default remote node.

If the user includes a *v_node* name in the query, the default is overridden and the query will be directed to the remote node the user specifies.

2.7.3.2 II_GCNxx_LCL_VNODE

The local *v_node* is set automatically every time you run **iistartup**.

The local node has an environment variable, II_GCNxx_LCL_VNODE, that is automatically set during the initialization procedure.

This specifies the virtual local node name of the current local installation, where "xx" is the value of II_INSTALLATION on the local node.

ULTRIX/SQL uses II_GCNxx_LCL_VNODE if a user specifies the local node name as if it were a remote node when attempting to access a database on the local node.

2.8 Summary of Environment Variables

The following table lists the environment variables used by ULTRIX/SQL NET.

Table 2-2: Environment Variables

Variable	Function
II_GCCxx_OBMAX	Sets the maximum number of outbound NET connections.
II_GCCxx_IBMAX	Sets the maximum number of inbound NET connections.
II_GCC[xx].LOG	Defines a specified error logging file.
II_GCC[xx].LOGLVL	Defines the level of detail of errors logged.
II_GCC_TCP_PORT	Specifies the TCP/IP port address.
II_INSTALLATION	Specifies the installation code for an ULTRIX/SQL installation.
II_GCNxx_LCL_VNODE	Defines the local node.
II_GCNxx_RMT_VNODE	Defines the default remote node.

The ULTRIX/SQL NET Utility netu   3

3.1 Overview

The ULTRIX/SQL NET network management utility, **netu**, performs a variety of network administration functions. The **netu** program has a user interface that prompts you for commands and information associated with the operation you want to perform.

Any ULTRIX user can invoke **netu** and add and delete private remote node definitions and private remote authorization entries. However, to execute global (public) operations such as defining global virtual node names, the user must be one with ULTRIX/SQL superuser status, such as the ULTRIX/SQL System Administrator.

The **netu** program has three operations:

*   Stopping the Database Communication Server shuts down the ULTRIX/SQL NET communications server. This operation can be done only by an individual with ULTRIX/SQL superuser status.

*   Modifying remote node information allows users to add, delete, and retrieve information about remote nodes. Only an ULTRIX/SQL superuser may modify global entries.

*   Modifying remote authorization information allows an ULTRIX/SQL superuser to define global access to specific remote nodes. This operation also enables individual users to define private authorization to specific remote nodes. Users can add, delete, or retrieve information about remote authorization.

The ULTRIX/SQL System Administrator normally will run **netu** at each node on the network where ULTRIX/SQL NET is installed and define:

*   Commonly accessible remote nodes on the network with global virtual node (*v_node*) names

*   Global authorization to any guest accounts at those globally-defined remote nodes

3.1.1 Completing the netu Work Sheet

Before you use **netu,** you should read this entire chapter to ensure that you have all the information you need. There is a work sheet at the end of the chapter. You may wish to complete it as you gather all your data. If you need to, make additional copies of the work sheet.

To define a remote node, you need to gather certain information about that remote node. To define a user authorization, you need the *v_node* name of the remote node and information about the account you want to define.

3.1.2 Information Needed to Define Remote Nodes

Before you start to use **netu** to define remote nodes, you need the following node-addressing information for each remote node:

- Server listen address

- *v_node* names

- Reserved ULTRIX/SQL name for your network software type

- Remote node addresses

- Remote communication server listen addresses

3.1.2.1 V_node Names

The ULTRIX/SQL System Administrator should decide on the global *v_node* names that are going to be used for each remote node. Global *v_node* names are node names to which all users will have access. The System Administrator determines such names by associating a meaningful name with the actual name of a remote node. All users can use this name to access the remote node. Using global names gives you greater flexibility in designing ULTRIX/SQL applications and in providing easier access to remote nodes.

Individual users can also decide on any private names they might want to give to particular remote nodes. However, only the user defining the private name can use it to access that remote node.

3.1.2.2 Reserved ULTRIX/SQL Names for Network Software Types

You identify your network software type to ULTRIX/SQL NET using a specific ULTRIX/SQL reserved name for that type of network software.

Valid reserved ULTRIX/SQL names for network software types (case-insensitive) are:

decnet	DECnet protocol
tcp_ip	ULTRIX TCP/IP protocol

3.1.2.3 Remote Node Addresses or Names

You need to know the actual addresses or names of each remote node. The syntax of the address is dependent on the type of network software you are using. Normally, the name of the remote node (as defined at the remote node) is sufficient for the node address. Your Network Administrator specifies this name when the network software is installed.

ULTRIX/SQL node addresses are dependent on the underlying network software. TCP/IP node addresses are simply the ULTRIX host names for the node.

For most TCP/IP installations, you can see a list of the names of all the nodes in the network by looking at the /etc/hosts file. At the operating system prompt, type **cat /etc/hosts**. A list of all the nodes on the network is displayed.

Alternatively, you can ask your Network Administrator for the names of the nodes on the network.

3.1.2.4 Remote Communication Server Listen Addresses

The remote ULTRIX/SQL NET communications server listen address is determined during the ULTRIX/SQL installation procedure.

The syntax of the listen address is dependent on the network protocol you are using.

The TCP/IP listen address for an ULTRIX/SQL installation is the decimal value of the two-letter installation code (value of II_INSTALLATION). The value must be greater than 1023.

To see the value of the port address, type:

```
ingprenv |grep II_GCC_TCP_IP
```

The identity, or value, of an installation can be seen by logging in to that node and typing:

```
ingprenv |grep II_INSTALLATION
```

The following type of message is displayed:

```
"II_INSTALLATION"="II"
```

3.1.3 Information Needed to Define Remote Authorization

Before you start to use **netu** to define user authorization at remote nodes, you need the following user information:

* Remote *v_node* name

* Remote user name

* Remote password

3.1.3.1 Remote v_node Name

The remote *v_node* name is the descriptive virtual node name of the remote node as described above. If it is a global *v_node* name, it will already have been defined by the ULTRIX/SQL System Administrator.

3.1.3.2 Remote User Name

The remote user name is the authorized name of the user (or public guest account) recognized by the operating system at one of the remote nodes.

3.1.3.3 Remote Password

The remote password is the user's password at the remote node. A user not only must have a valid account at a remote node, but must define himself or herself with **netu** at the local node as being authorized to access that remote node. Then, to access the remote node, the user logs on to his or her local node. When the user contacts the remote node through ULTRIX/SQL NET, the remote node validates the authorization and allows the user to query a database there.

If you are using Remote Access to Rdb/VMS, the user name or password need not be the same as your login to that node. Check with the database administrator for the database management system concerned to ascertain the user name and password you require.

<div align="center">

Note

You need only to define your authorization to a remote node once. You do not need to do this every time you access that remote node. Authorization is established permanently until you delete it.

</div>

3.2 Accessing netu

To start **netu**, at the operating system prompt enter:

```
netu
```

To use the Stop option, specify the communications server id as a parameter when you start netu. At the operating system prompt, enter netu *yyyy*, where *yyyy* is the value of the GCA_ADDRESS. You can obtain this value by running **iinamu** and entering **show comsvr**. (The value you must use is the number displayed in the third column.)

3.2.1 The netu Top-level Menu

After you start **netu**, the top-level menu is displayed:

```
Select one of the following:

     S{top}             - Stop Database Communication Server

     N{ode}             - Modify Node Entry

     A{uthorization     - Modify Remote Authorization Entry

     E{xit}             - Exit

NETU>
```

3.2.2 netu User Input

The following table summarizes the behavior of the netu utility:

Table 3-1: Displays and User Options for the netu Utility

Display or User Action	Description
NETU>	The **netu** prompt, NETU>, indicates that user input is required.
add, del, show, exit	To choose one of these operations, type the first letter of your choice or the whole word.
User name prompts	Certain **netu** options prompt for a user name. This name is the operating system authorized user name or *login*.
Wildcard character	When you are removing information using the del(ete) option or retrieving information using the show option, you can use the asterisk (*) as a *wildcard character* to mean "delete all" or "show all."
Aborting an Operation	To abort any function or operation, press **Escape** and then **Return**.
Exiting netu	To exit the **netu** program, type E and press **Return**.
(value):	On an input line, any value shown in parentheses is a default value. The **netu** program makes no syntax check on the values you enter. Any errors can result in unpredictable behavior by ULTRIX/SQL NET. Please ensure that your entries are accurate.

Prompt example:

```
Enter Private or Global (P):
```

By pressing **Return,** you choose the default, Private, and a status message appears. For the example, the following message would then appear:

```
Default value: P
```

If no default is displayed when netu prompts for input, you *must* enter a value. If you do not make an entry, the following error message appears:

```
Value required or enter <Esc> to exit.
```

3.3 Stopping ULTRIX/SQL NET

Taking the menu choice Stop shuts down the ULTRIX/SQL NET communications server. Only a user with ULTRIX/SQL superuser status, such as the ULTRIX/SQL System Administrator, can do this.

You can use the Stop option only if you specified the communication server ID as a parameter when you started netu.

When you choose the Stop option, the communications server closes all communications links on the local node. ULTRIX/SQL continues to function on your local node.

1. To choose Stop ULTRIX/SQL NET, call up the top-level **netu** menu.

2. At the NETU prompt, type S and press **Return**

   The following message appears:

   ```
   Stop Database Communication Server: (Yes/No):
   ```

3. Type N to return to the **netu** main menu or Y to stop ULTRIX/SQL NET.

   After a short time the ULTRIX/SQL NET communications server shuts down and you return to the **netu** main menu.

To choose another **netu** operation, enter the appropriate letter and press **Return**.

To exit the **netu** program, type E and press **Return**.

Refer to Chapter 2 to see how to restart ULTRIX/SQL NET.

3.4 Modifying a Node Entry

The Modify Node Entry option allows you to:

- Add remote node definitions (**add**).

- Delete remote node definitions (**del**). You do not delete the actual node name or address, just your local definition of that remote node name.

- Display a list of all the nodes you have defined (**show**).

To choose one of these operations, type the first letter of your choice or the whole word.

You should note that you are not naming the remote nodes when you use **netu**. The actual network node names are specified by the Network Administrator at the time the network software is installed or when that remote node is added to the network. The **netu** program simply allows you to define each remote node at your local node with a name that is meaningful to you. The name is a virtual node (*v_node*) name.

The virtual node name you define is mapped to the actual name or address of the remote node by the other information you specify: the connecting network software type, the actual remote node name or address, and the ULTRIX/SQL NET communications server listen address at the remote node.

3.4.1 Adding Remote Node Definitions

To add a remote node definition, first call up the **netu** top-level menu. You will see the **netu** prompt, NETU>.

1. To choose Modify Node Entry, type N and press **Return**.

The following command line appears:

```
Enter operation (add,del,show,exit):
```

2. Select the add operation by typing A or Add and pressing **Return**.

The following prompt lines appear in sequence:

```
Enter Private or Global (P):

Enter the remote v_node name:

Enter the network software type(TCP/IP):

Enter the remote node address(node):

Enter the remote Database Communication Server Listen address(II):
```

3. Complete the entries, pressing **Return** after each one.

No wildcard entry is allowed in an add operation.

You can press **Return** to enter a default value shown in parentheses. If no default is shown, you *must* enter a value, or press **Escape** and then **Return** to exit.

If the name entered for a *v_node* is found in the local /etc/hosts file, that name will be provided as a default for the remote node address. After you complete the last entry, you return automatically to the "Enter the operation (add, del, show, exit):" prompt.

To define another remote node, repeat steps 2 and 3 in the preceding list.

To quit without making an entry, press **Escape** and then **Return**. You return to the **netu** main menu.

To exit the **netu** program, type E and press **Return**, or press **Escape** and then **Return**.

<div align="center">Note</div>

Be sure that your entries are accurate. The netu program makes no syntax check on the values you enter. You will not know if you have made an erroneous entry until you try to run an ULTRIX/SQL NET operation on a database on a remote node and fail to make a connection with that remote node.

You may only have one entry for any *v_node* in either of the Private or Global tables. If you add a node that is already listed, the previous entry will be automatically deleted.

The following table describes each of the prompt lines displayed by the Modify Node Entry operation.

Table 3-2: Modify Node Entry Prompts

Prompt	Description
Private or Global (P,G)	This option determines the accessibility of the node name. The default entry is P for Private. Only an ULTRIX/SQL superuser may select the Global option. If a user other than an ULTRIX/SQL superuser tries to define or access a Global entry, an error is generated after all other entries are complete.
	If a v_node name is made Global, any user can use that v_node name. If a v_node name is Private, only the user who defined that name can use that v_node name.
	It might happen that a Private v_node name is defined to one node and the same name, but with Global privileges, is defined to another node. In this situation, if the user who defined the Private v_node name uses that v_node name, ULTRIX/SQL NET will access the remote node defined by the private definition.
	If any other person uses that name, ULTRIX/SQL NET will access the remote node defined by the global v_node definition.
	Private definitions take precedence over Global definitions for the user who defined that private entry. Private definitions have no effect on Global definitions.
remote v_node name	This option specifies the descriptive virtual name you must assign to the remote node. (You can use the actual name of the remote node as defined by the Network Administrator.) Users must use this v_node variable when accessing a database on a remote node.
	You can enter any valid node name. The name cannot exceed 15 characters. The name may not include a double colon (::), a slash (/), or a comma (,). The name is not case-sensitive.
	If this v_node name is Private, the name must be unique within each user's list of Private v_node names. If the v_node name is global, the name must be unique within the list of global v_node names.
	You might want to use a virtual node name that relates to the actual remote node name. (See the section "remote node address" that follows.) For example, if the actual remote node is named "rabbit" you can give the node a virtual name of "rabbit," "vrabbit," or "rabbit1."
	Alternatively, you might want to use a virtual node name that is functionally related to your organization. For example, you can define your production machines in New York and Boston as prod_n_york or prod_boston. (See the Remote Node Address prompt that follows.)

Prompt	Description
network software type	This is the ULTRIX/SQL reserved name of the network software type on the *remote* node linking your local and remote nodes. Note that you specify the network software type on the remote node.
	Valid reserved ULTRIX/SQL names (case-insensitive) for network software types are:
	decnet DECnet protocol **tcp_ip** ULTRIX TCP/IP Internet protocol
remote node address	This option specifies the actual name of the remote node as known to the underlying network software.
	The address cannot exceed 15 characters.
	The syntax depends on the network software type specified. This address is first specified when the network software is installed. To obtain this node address, ask the Network Administrator or refer to the section "Remote Node Addresses or Names" earlier in this chapter.
remote ULTRIX/SQL NET server listen address	This option specifies the remote ULTRIX/SQL NET communications server listen address of the server at the specified remote node.
	The address cannot exceed 15 characters.
	A default address displays. The syntax of the address depends on the protocol type specified. See the section titled "Remote Communications Server Listen Addresses" earlier in this chapter for information on how to specify this address.

3.4.2 Deleting Remote Node Information

You can delete a definition of a single remote node or definitions of several nodes at the same time.

1. From the **netu** main menu, type N and press **Return**

   The following command line appears:

   ```
   Enter operation (add, del, show, exit):
   ```

2. Type D or del and press **Return**

   The following prompt lines appear in sequence:

   ```
   Enter Private or Global (P):
   ```

   ```
   Enter the remote v_node name:
   ```

3. Complete the entries, pressing **Return** after each response.

After you respond to the last prompt, the node name definition is deleted.

To delete a single node name definition, be sure that the response for each prompt line matches the node definition.

You can use the asterisk (*) wildcard character or an empty input line to delete multiple node definitions. You must enter a value on all input lines that do not display a default. After you press **Return** at the last prompt, all node name definitions that match your criteria are deleted. The system displays the number of definitions deleted. If you do not match any definition with exact information or wildcards, no entries are deleted.

To avoid problems, check on the definition before you delete it. You can use the **show** command to review definitions before deleting them. (See the section "Retrieving Remote Node Information" that follows.)

Note

Only the ULTRIX/SQL superuser can delete Global entries.

After the last entry is completed, you are returned automatically to the "Enter operation (add, del, show, exit):" prompt.

To delete another remote node definition, type del and repeat steps 2 and 3 in the preceding list.

To quit without making an entry, press **Escape** and then **Return**. You return to the **netu** main menu.

To exit the **netu** program, type E, and press **Return**, or press **Escape** and then **Return**.

3.4.3 Retrieving Remote Node Information

Using the **show** operation, you can display a list of all the remote nodes you have defined or display information about a single node you have defined.

1. At the **netu** main menu, type N and press **Return**.

    The following command line appears:

    ```
    Enter operation (add,del,show,exit):
    ```

2. Type S or show and press **Return**.

    The following prompt lines appear in sequence:

    ```
    Enter Private or Global (P):

    Enter the remote v_node name:
    ```

3. Complete the entries, pressing **Return** after each response.

To retrieve information about a single node, specify an exact match at each prompt line.

To retrieve information about several nodes, use the asterisk (*) wildcard character. All node name definitions will be retrieved.

Retrieved definitions are displayed as shown in the following table:

Table 3-3: Sample Remote Node Definitions

Global:	V_NODE	Net Software	Node Address	Listen Address
	london	tcp_ip	uk1	II
	rome	tcp_ip	italy2	II
	n_york	tcp_ip	usa1	II

Global:	V_NODE	Net Software	Node Address	Listen Address
	london	decnet	uk1	II
	rome	decnet	rome	II
	s_fran	decnet	s_fran	II

You *cannot* change any node name definition in this format.

After the definition or list of definitions is displayed, you are returned automatically to the "Enter operation (add, del, show, exit):" prompt.

To retrieve another node definition, repeat steps 2 and 3 in the preceding list.

To quit without making an entry, press Escape and then **Return**. You return to the **netu** main menu.

To exit the **netu** program, type E, and press **Return**, or press **Escape** and then **Return**.

You normally use the Modify Remote Authorization option in **netu** to add user authorization to each remote node after defining the remote node. Please refer to the section entitled "Modifying Remote Authorization Entry" for further details.

3.4.4 Changing Remote Node Information

To correct an erroneous remote node definition or revise a remote node definition because of network configuration changes, you must delete the old definition and then add the new definition.

You must return to the "Enter operation (add, del, show, exit):" prompt, delete the present definition using the **del** operation, and then define a new node name using the **add** operation. Alternatively, if a new entry for an existing v_node is entered, it will replace the previous entry.

3.5 Modifying Remote Authorization Entry

The Modifying Remote Authorization Entry option allows you to:

- Add definitions of user authorization to remote nodes (**add**).

- Delete definitions of authorization to remote nodes (**del**). You do not delete the actual user name and password at the remote node, only your local definition of that remote authorization on your local node.

- Display a list of all the authorizations you have defined (**show**).

To choose an **add, del, show, exit** operation, enter the first letter of your choice or the whole word.

Note

Only the ULTRIX/SQL System Administrator can define global authorization entries. Only private users can define authorization to private accounts.

3.5.1 Defining User's Authorization to Remote Nodes

To choose the Modify Remote Authorization Entry option, first call up the **netu** main menu, and then:

1. At the NETU> prompt, type A and press **Return**

    The following command line appears:

    ```
    Enter operation (add, del, show, exit):
    ```

2. Type A or add and press **Return**

    The following prompt lines appear in sequence:

    ```
    Enter Private or Global (P):

    Enter the remote v_node name:

    Enter remote User Name:

    Enter the remote Password (*):

    Repeat the remote Password (*):
    ```

3. Complete the entries, pressing **Return** after each one.

After the last entry is completed, you are returned automatically to the "Enter operation (add, del, show, exit):" prompt.

To authorize access to another remote node, repeat steps 2 and 3 in the preceding list.

To quit without making an entry, press Escape and then **Return**. You return to the **netu** main menu.

To exit the **netu** program, type E and press **Return**, or press **Escape** and then **Return.**

The following table describes the prompt lines displayed by the Modify Remote Authorization Entry operation.

Table 3-4: Modify Remote Authorization Prompts

Prompt	Description
Private or Global (P,G)	This option determines the accessibility of the user's account on the remote node. The default entry is P for Private. Only an individual with ULTRIX/SQL superuser privileges, such as the ULTRIX/SQL System Administrator, can select the Global option. If any other user selects this option, an error is generated after all other entries are completed.
remote v_node name	This option specifies the descriptive virtual name assigned to the remote node. This v_node name should already have been defined by the ULTRIX/SQL System Administrator (or the individual user if it is a private v_node name) using the "N - Modify Node Entry" operation. To see the list of v_node names, choose Modify Node Entry from the netu main menu and then type show and press Return.
remote user name	This option specifies the operating system-determined user name (*login*) at the remote node. The remote user name can be a global or public account name, for example, "guest."
Remote Password	This option specifies the user's password (or public account password) at the remote node. The default displayed is the null password indicate by an asterisk (*). If an account on a remote node has no password, enter an asterisk (*). You must enter a value. Your password is not displayed.
Repeat Remote Password	This option ensures that the previously input password is correct.

Make sure that your entries are accurate. The netu program makes no syntax check on the values you enter. You will not know if an entry is wrong until you run an ULTRIX/SQL NET operation on a database on a remote node and, for example, fail to make a connection with that remote node.

Note

Each user may only have one entry for any *v_node* in either of the Private or Global tables. If you add a *v_node* entry that is already listed, the previous entry will be automatically deleted.

3.5.2 Deleting Remote Authorization Information

You can delete the local definition of a user's authorization to a remote node. However, only the local definition is deleted, not the actual user name and password at the remote node.

1. From the **netu** main menu, at the NETU> prompt type A and press **Return**.

   The following command line appears:

   ```
   Enter operation (add, del, show, exit):
   ```

2. Type D or del and press **Return**.

   The following prompt lines appear in sequence:

   ```
   Enter Private or Global (P):

   Enter the remote v_node name:

   Enter the remote User Name:

   Enter the remote Password(*):
   ```

3. Complete the entries, pressing **Return** after each response.

   To delete authorization to a single node, your response at each prompt line must match the authorization definition.

   Make sure that your entries are accurate. If you do not make an exact match, no entry will be deleted. To avoid this problem, check the node name using the **show** operation before you attempt to delete authorization to any node. (See the section "Retrieving Remote Authorization Information" that follows.)

4. Press **Return** after you complete the last entry. The number of deletions is shown.

   You are returned automatically to the "Enter operation (add, del, show, exit):" prompt.

   To cancel authorization to another node, type d or del and repeat steps 2 and 3 in the preceding list.

   To quit without making an entry, press Escape and then **Return**. You return to the **netu** main menu.

   To exit the **netu** program, type E and press **Return**.

   ### Note

   Only the ULTRIX/SQL superuser can delete Global entries. All users should be careful when using the asterisk (*) wildcard to delete multiple entries. For example, if you enter nothing, but just press **Return** after the "Enter the remote *v_node* name:" prompt, all authorization entries for all nodes will be deleted from the current user's Private or Global table.

3.5.3 Retrieving Remote Authorization Information

You can use the **show** operation to see a list of authorizations to remote nodes you have defined or display information about a single authorization you have defined.

1. Choose "A-Modify Remote Authorization Entry" from the **netu** main menu and press **Return**.

   The following command line appears:

   ```
   Enter operation (add, del, show, exit):
   ```

2. Type S or show and press **Return**

   The following prompt lines appear in sequence:

   ```
   Enter Private or Global (P):

   Enter the remote v_node name:
   ```

3. Complete the entries, pressing **Return** after each response.

To retrieve information about a single node, your response at each prompt line must match the node definition.

To retrieve information about multiple nodes, use the wildcard asterisk character (*) when prompted for a *v_node* name. You cannot use the wildcard character when prompted for Private or Global (the only valid response is P or G or simply **Return** to accept the default value shown). The information retrieved appears in the following table:

Table 3-5: Sample Remote Authorization Information

Private:	V_NODE	User Name
	london	janetd
	rome	janetd
	n_york	janetd
	s_fran	janetd

You cannot correct any information in this format. To correct an erroneous entry or revise an entry because of network configuration changes, you must return to the "Enter operation (add, del, show, exit):" prompt, delete the present authorization definition, and then add a new definition.

After the definition or list of definitions displays, you are returned automatically to the "Enter operation (add, del, show, exit):" prompt.

To retrieve additional authorization information, repeat steps 2 and 3 in the preceding list.

To quit without making an entry, press **Escape** and then **Return**. You return to the **netu** main menu.

To exit the **netu** program, type E and press **Return**, or press **Escape** and then **Return**.

3.5.4 Changing Remote Authorization Information

To correct an erroneous remote authorization entry or revise a remote authorization entry because of network configuration changes, you delete the old definition and then add the new definition.

You must return to the "Enter operation (add, del, show, exit):" prompt, delete the present authorization definition using the del operation, and then define the new authorization using the **add** operation. Alternatively, if any new entry for an existing authorization is entered, it will replace the previous entry.

3.6 A Sample netu Session

Assume a simple network with three nodes: node_a, node_b, and node_c. The network is shown below with the users and accounts on each node.

Figure 3-1: Simple Network with Three Nodes
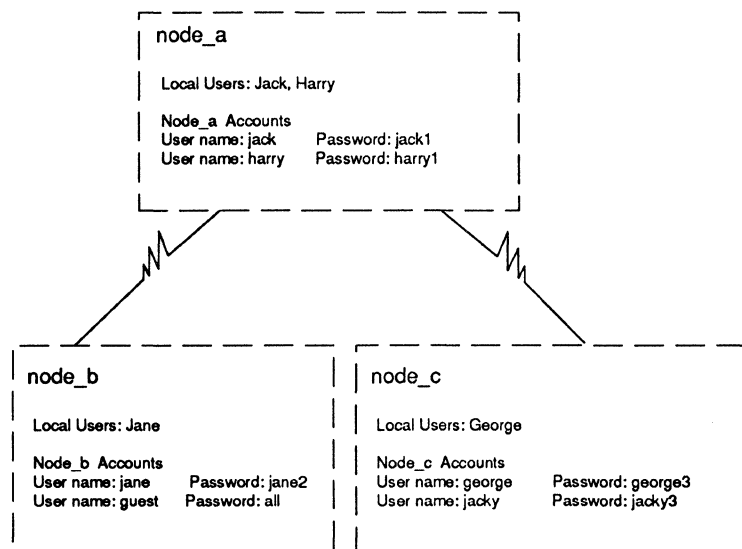
- To implement ULTRIX/SQL NET on this network, you need the following definitions:

- To enable Jack and Harry on node_a to access node_b through the public "guest" account, the ULTRIX/SQL System Administrator must run **netu** at node_a to define both node_b and the global authorization to node_b's "guest" account.

- For Jack on node_a to access node_c through his own private "Jacky" account, he must define node_c and his private authorization to his "jacky" account on node_c. (Remember, individual users cannot define authorization to global accounts, only to their own private accounts.)

Jack could request the System Administrator to define node_c as a *global v_node*. But Jack would still have to define his private authorization to his private "jacky" account.

- To enable George on node_c to access node_b through the public "guest" account, the ULTRIX/SQL System Administrator must run **netu** *at node_c* and define node_b and define the global authorization to node_b's "guest" account.

Note the following limitations of this ULTRIX/SQL NET configuration:

- Jane cannot access databases on either node_a or node_c, as there is no public account defined for those nodes and she has no private account on either node.

- Similarly, George on node_c cannot access databases on node_a as there is no public account and he has no private account there.

- Also, Harry on node_a cannot access node_c because there is no public guest account and he has no private account there.

The following sample script shows how the ULTRIX/SQL System Administrator defines each remote node and global authorization to the guest account on node_b, and how each user must define his or her private authorization to each remote node that holds that user's private account.

3.6.1 ULTRIX Sample netu Script

The sample scripts in the following sections show the **netu** session required to enable:

- Jack and Harry to access node_b from node_a

- Jack also to access node_c from node_a

- George to access node_b from node_c

In the following examples, the operating system prompt is indicated by a percent (%) sign. On some ULTRIX systems, your own customized prompt may display. Sample user responses are shown in **boldface**.

3.6.2 Using netu at node_a

The ULTRIX/SQL System Administrator at node_a defines node_b as a *global v_node* name and defines global authorization to the public "guest" account on node_b. The ULTRIX/SQL System Administrator also defines node_c as a *global v_node* name. However, Jack must run **netu** again at node_a and define his own private authorization to his private node_c "jacky" account.

The ULTRIX/SQL System Administrator first defines global authorizations.

%

% netu Return

```
Select one of the following:

    S - Stop Database Communication Server
    N - Modify Node Entry
    A - Modify Remote Authorization Entry
    E - Exit
```

NETU> **N Return**

```
Modify Node Entry:
    Enter operation (add, del, show, exit): add Return

    Enter Private or Global (P): g Return
    Enter the remote v_node name: node_b Return
    Enter the network software type (TCP_IP): Return
    Default value: TCP_IP
    Enter the remote node address: node_b <Return>
    Enter the remote Database Communication Server Listen address (II):Return
 Default value: II

Modify Node Entry:
    Enter operation (add, del, show): Return

Select one of the following:
    S - Stop Database Communication Server
    N - Modify Node Entry
    A - Modify Remote Authorization Entry
    E - Exit
```

NETU> **A Return**

```
Modify Remote Authorization Entry:
    Enter operation (add, del, show, exit): add Return

    Enter Private or Global (P): g Return
    Enter the remote v_node name: node_b Return
    Enter the remote User Name: guest Return
    Enter the Password (*): (no display) Return
    Repeat the Passqord (*): (no display) Return

Modify Remote authorization Entry:
    Enter operation (add, del, show): <Return>

Select one of the following:
    S - Stop Database Communication Server
    N - Modify Node Entry
    A - Modify Remote Authorization Entry
    E - Exit>
```

NETU> **E**

%

The ULTRIX/SQL System Administrator has defined node_c as a global *v_node*.
However, Jack still has to run **netu** at node_a, and define his private authorization
to his private node_c "jacky" account.

```
%

% netu Return

Select one of the following:

        S - Stop Database Communication Server
        N - Modify Node Entry
        A - Modify Remote Authorization Entry
        E - Exit

NETU> A Return

Modify Remote Authorization Entry:
        Enter operation (add, del, show, exit): add Return

        Enter Private or Global (P): Return
        Default value: P
        Enter the remote v_node name: node_c Return
        Enter the remote User Name: jacky Return
        Enter the remote Password (*): jacky3 (does not display) Return
        Repeat the remote Password (*): jacky3 (does not display) Return

Modify Remote Authorization Entry:
        Enter operation (add, del, show, exit): Return

Select one of the following:

        S - Stop Database Communication Server
        N - Modify Node Entry
        A - Modify Remote Authorization Entry
        E - Exit

NETU> E Return

%
```

3.6.3 Using netu at node_b

As Jane at node_b has no account on either node_a or node_c, you need not run **netu** at node_b at this stage. When Jane establishes accounts at node_a and node_c, or when global accounts are set up at those nodes, you have to run **netu** at node_b to define node_a and node_c. Also, Jane will have to run **netu** to define her authorization to those nodes.

However, as the ULTRIX/SQL System Administrator, you might want to run **netu** at node_b to define node_a and node_c with globally-accessible *v_node* names at this time. You can do so even though there are no node_b users with accounts at those nodes.

In the future, node_b users might establish accounts on node_a and node_c and want to access databases on those remote nodes from node_b. At that time, users will need to run **netu** at node_b only to define their private authorization to their private accounts through the already globally-defined *v_node* names for node_a and node_c.

```
%

% netu Return

Select one of the following:
     S - Stop Database Communication Server
     N - Modify Node Entry
     A - Modify Remote Authorization Entry
     E - Exit

NETU> N Return

Modify Node Entry:
     Enter operation (add, del, show, exit): add Return

     Enter Private or Global (P): g Return
     Enter the remote v_node name: node_a Return
     Enter the network software type(TCP_IP): decnet Return
     Default value: decnet
     Enter the remote node address: (node_a) Return
     Enter the remote Database Communication Server Listen address (II): Return
     Default value: II

Modify Node Entry:
     Enter operation (add,del,show): add Return

     Enter Private or Global (P): g Return
     Enter the remote v_node name: node_c Return
     Enter the network software type(TCP_IP): decnet Return
     Default value: decnet
     Enter the remote node address: (node_c) Return
     Enter the remote Database Communication Server Listen address (II): Return
     Default value: II

Modify Node Entry:
     Enter operation (add, del, show, exit): Return

Select one of the following:
     S - Stop Database Communication Server
     N - Modify Node Entry
     A - Modify Remote Authorization Entry
     E - Exit

NETU> E Return

%
```

3.7 The netu Work Sheets

The work sheets in this section show the information you need to define remote
nodes and define user authorization to remote nodes. Three sample entries have
been made:

- A DECnet test installation with a global *v_node* defined by the ULTRIX/SQL
  System Administrator

- A TCP/IP installation with a global *v_node* defined by the ULTRIX/SQL
  System Administrator

- A user's private authorization defined to that node

For security, do not write down any passwords. The password on the authorization work sheet appears simply as a reminder that you need the passwords to complete a **netu** session successfully.

Table 3-6: Work Sheet for Defining Remote Nodes

P or G	Remote v_node name	Network software type	Remote node name	Com Server Listen address
g	euro1	decnet	london	II
g	euro1	tcp_ip	london	II

Table 3-7: Work Sheet for Defining Authorization to Remote Nodes

P or G	Remote v_node name	Remote user name	Remote password
p	euro1	tester	

ULTRIX/SQL NET Commands and Utilities   A

A.1   Commands Available in ULTRIX/SQL NET

The following ULTRIX/SQL commands can be run against a remote database using ULTRIX/SQL NET:

isql	sql
optimizedb	unloaddb

A.2   Commands Not Available in ULTRIX/SQL NET

The following commands and utilities may *not* be run against remote databases:

accessdb	lockstat	rungcc
createdb	logdump	rungcn
destroydb	mkrungcc	statdump
finddbs	netu	sysmod
iimonitor	rcpconfig	verifydb
iinamu	rundbms	

Glossary

This section provides definitions for many of the terms used throughout this manual to discuss ULTRIX/SQL NET.

Heterogeneous Communication—Communication between computers with different operating systems and/or architectures.

Homogeneous Communication—Communication between computers with the same operating system and architecture.

Local Database—A database stored on a single computer that does not rely on information stored on any other computer.

netu—The ULTRIX/SQL NET network utility program that enables you to add, delete, and view virtual remote node definitions on the network, and to define user authorizations to accounts on remote nodes.

Network—A collection of computers that can communicate with each other using a common network protocol. Each computer on the network is known as a node.

Network Administrator—The individual responsible for maintaining the network (often the same person as the operating System Administrator).

Network Protocol—The set of rules specific to communications and network software that permits data to be exchanged within networks and between networks. Sometimes referred to simply as protocol.

Network Software—The software that enables computers to communicate on a network. The network software maintains the network protocol.

Node—A computer that is connected to a network. Also known as a "host." Each node of a network is identified by a unique name or address. The local node is your own computer. Remote nodes are all the other computers on the network.

Node Address—The address that uniquely identifies a node in a particular network. The syntax of the node address depends on the particular network protocol.

Node Name—A synonym for the node address. The name identifies a node in a particular network.

Operating System Administrator—The individual responsible for installing, controlling, and maintaining an operating system.

System-Level Command—A command you issue when you see the prompt character displayed by your operating system.

ULTRIX/SQL Database Administrator (DBA)—The person who creates and manages a local database. The Database Administrator controls access to the database's data, coordinates the data's upkeep, and performs regular database backups. The person who creates a database automatically becomes its DBA.

ULTRIX/SQL NET—The ULTRIX/SQL component that provides remote access to multiple computers (nodes) across a network without regard to computer architecture, operating system, or network protocol.

ULTRIX/SQL NET Communications Server—A daemon process that enables communication between nodes across a variety of communications network protocols.

ULTRIX/SQL NET Communications Server Listen Address—The name or address that uniquely identifies a specific communications server on a node.

ULTRIX/SQL System Administrator—The individual responsible for initializing and managing the ULTRIX/SQL system software. The ULTRIX/SQL System Administrator decides who can use ULTRIX/SQL and what functions each user can perform. The ULTRIX/SQL System Administrator has superuser status.

Virtual Node Name (*v_node* name)—Using the netu program, this is the name you assign to a remote node on the network. You access a remote node by its *v_node* name, not the actual remote node name. The actual remote node name or node address is assigned by the Network Administrator when your network is first installed or a node is added to your network.

Index

S

Sample session
    in netu, 3-16
Server
    specifying in database access syntax, 1-3
Server address, 3-9
Syntax
    for database access, 1-3, 1-5
    for Rdb/VMS database access, 1-5
    for remote database access, 1-4

T

TCP/IP
    listen address, 2-4
Terminals
    defining, 1-2

U

ULTRIX/SQL, 2-2
    calling, 1-1
ULTRIX/SQL NET
    accessing remote databases through, 1-4
    commands available, A-1
    commands not available, A-1
    connections, 2-2
    initializing, 2-1 to 2-6, 2-6
    initializing locally, 2-2
    starting, 2-4
    stop with netu, 3-5
    stopping, 2-4

V

V_node, 2-8, 3-8
Variables
    environment, 1-1
    server_type, 1-3
    v_node, 2-8, 3-8
Virtual node name, 2-8, 3-8

W

Wildcard character
    asterisk, 3-10, 3-15
    in netu, 3-5, 3-10, 3-15
Work sheet
    in netu, 3-20

How to Order Additional Documentation

Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

Electronic Orders

To place an order at the Electronic Store, dial 800-234-1998 using a 1200- or 2400-baud modem from anywhere in the USA, Canada, or Puerto Rico. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Your Location	Call	Contact
Continental USA, Alaska, or Hawaii	800-DIGITAL	Digital Equipment Corporation P.O. Box CS2008 Nashua, New Hampshire 03061
Puerto Rico	809-754-7575	Local Digital Subsidiary
Canada	800-267-6215	Digital Equipment of Canada Attn: DECdirect Operations KAO2/2 P.O. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6
International	————	Local Digital subsidiary or approved distributor
Internal*	————	SSB Order Processing - WMO/E15 *or* Software Supply Business Digital Equipment Corporation Westminster, Massachusetts 01473

* For internal orders, you must submit an Internal Software Order Form (EN-01740-07).