Digital SNA
Data Transfer Facility
for OpenVMS

Management

Part Number: AA-JM78F-TE

May 1996

This document provides information needed to manage the Digital SNA
Data Transfer Facility (DTF) product.

Revision/Update Information:	This is a revised manual.
Operating System and Version:	OpenVMS VAX Versions 6.1, 6.2, or 7.0 OpenVMS Alpha Versions 6.1, 6.2, or 7.0
Software Version:	Digital SNA Data Transfer Facility for OpenVMS, Version 3.2

Contents

3   Maintaining the OpenVMS/DTF File Definition Database

A   Messages

B   Maintaining the OpenVMS/DTF Proxy Database

Index

Figures

Tables

Preface

The Digital SNA Data Transfer Facility (DTF) software is a Digital Equipment
Corporation product that allows you to transfer files between IBM VM and
MVS systems in an SNA network and systems in a Digital network. To use
the DTF access routine, you must first install the appropriate versions of the
software and hardware that you plan to use. Refer to the Software Product
Description (SPD) for this information.

The DTF access routine consists of two major parts: the IBM-resident software
(DTF for IBM) and the Digital-resident software (OpenVMS/DTF). This manual
describes all the commands that OpenVMS system managers need to use the
SNA DTF Configuration (SNADTFCFG) utility to configure the OpenVMS/DTF
server.

Intended Audience

This manual is designed for system managers who configure and manage the
OpenVMS/DTF product.

Changes and New Features

The Digital SNA Data Transfer Facility for OpenVMS, Version 3.2 differs from
the Version 3.1 product only in that it includes support for utilizing TCP/IP to
communicate between TE and the Gateways (Domain and/or Peer Server).

The informatin relevant to TCP/IP transport support include:

- SNA_TCP_PORT logical
- SNA_TRANSPORT_ORDER logical
- Specifying TCP/IP hostnames

SNA_TCP_PORT Logical

The SNA_TCP_PORT logical refers to the remote connection TCP/IP port. The default connection TCP/IP port number is 108. For example, if you want the remote connection TCP/IP port number to be 1234, you can enter the following command line:

```
$ define SNA_TCP_PORT 1234
```

If you want the remote connection TCP/IP port to be made to a service defined and enabled in the UCX database; for example *service_name*, you can enter the following command line:

```
$ define SNA_TCP_PORT service_name
```

SNA_TRANSPORT_ORDER Logical

The SNA_TRANSPORT_ORDER logical refers to a transport list, which is used in automatic selection of transports. Connections are attempted once for each transport in the list until either a successful connection is made, or an error is returned when all transports in the list fail to connect.

For example, if you want the software to try the DECnet transport and if this fails then to try the TCP/IP transport, you can enter the following command line:

```
$ define SNA_TRANSPORT_ORDER "decnet, tcp"
```

If you want the software to try the TCP/IP transport and if this fails then to try the DECnet transport, you can enter the following command line:

```
$ define SNA_TRANSPORT_ORDER "tcp, decnet"
```

If you want the software to never try the DECnet transport and to try only the TCP/IP transport, you can enter the following command line:

```
$ define SNA_TRANSPORT_ORDER "nodecnet, tcp"
```

If you want the software to never try the TCP/IP transport and to try only the DECnet transport, you can enter the following command line:

```
$ define SNA_TRANSPORT_ORDER "decnet, notcp"
```

―――――――――――――――― **Note** ――――――――――――――――

If the SNA_TRANSPORT_ORDER logical is not defined, the default transport order for OpenVMS Alpha will be decnet, tcp; and the default transport order for OpenVMS VAX will be local, decnet, tcp.

――――――――――――――――――――――――――――――――――――――

Specifying TCP/IP Hostnames

If you want to specify a full path hostname, the hostname must be enclosed in a pair of double-quotes; for example, "foo.bar.company.com".

If you want the TCP/IP transport to be used as the preferred transport, without specifying a TCP/IP full path hostname, then define the SNA_TRANSPORT_ORDER with "tcp" as the first element in the transport list.

If the hostname ends with a single full-colon (":"), then the TCP/IP transport will be used; for example, "foo:" or foo:.

_____ **Note** _____

If you specify a double full-colon ("::"), you force the DECnet transport to be used; for example, "foo::" or foo::.

_____

Document Structure

This document is divided into the following three chapters and two appendixes.

Chapter 1	Describes the OpenVMS/DTF configuration concepts that a system manager must understand to configure OpenVMS/DTF.
Chapter 2	Describes how to maintain the OpenVMS/DTF server database.
Chapter 3	Describes how to maintain the file definition database.
Appendix A	Describes all OpenVMS/DTF configuration utility (SNADTFCFG) messages.
Appendix B	Describes how to maintain the OpenVMS/DTF proxy database.

The postage-prepaid Reader's Comments form on the last page of this manual is for your critical evaluation to assist us in preparing future documentation.

Associated Documents

The Digital SNA Data Transfer Facility software documentation consists of the following manuals:

- _Digital SNA Data Transfer Facility for OpenVMS Installation_

- _Digital SNA Data Transfer Facility for OpenVMS Management_

- _Digital SNA Data Transfer Facility for OpenVMS Problem Solving and Messages_

- *Digital SNA Data Transfer Facility for OpenVMS Use*

You should have the following Digital documents available for reference when you use the Digital SNA Data Transfer Facility software:

- *Common Data Dictionary Summary Description*
- *Common Data Dictionary User's Guide*
- *Common Data Dictionary Utilities Reference*
- *Common Data Dictionary Data Definition Language Reference*
- *Digital SNA Domain Gateway Installation*
- *Digital SNA Domain Gateway Management*
- *Digital SNA Domain Gateway Guide to IBM Resource Definition*
- *DECnet SNA Gateway Problem Determination Guide*
- *DECnet SNA Gateway-CT Installation*
- *DECnet SNA Gateway-CT Problem Solving (OpenVMS & ULTRIX)*
- *DECnet SNA Gateway-CT Management (OpenVMS)*
- *DECnet SNA Gateway-CT Guide to IBM Parameters*
- *DECnet SNA Gateway-ST Installation*
- *DECnet SNA Gateway-ST Problem Solving (OpenVMS)*
- *DECnet SNA Gateway-ST Guide to IBM Parameters*
- *DECnet SNA Gateway Management for OpenVMS*
- *Digital Peer Server Installation and Configuration*
- *Digital Peer Server Management*
- *Digital Peer Server Network Control Language Reference*
- *Digital Peer Server Guide to IBM Resource Definition*
- *OpenVMS SNA Installation*
- *OpenVMS SNA Problem Solving*
- *OpenVMS SNA Guide to IBM Parameters*
- *OpenVMS SNA Management*
- *OpenVMS SNA Problem Determination Guide*

OpenVMS client users may also find the following Digital OpenVMS manuals to be helpful:

- *OpenVMS User's Manual*

- *OpenVMS Record Management Utilities Reference Manual*

- *DEC DATATRIEVE User's Guide*

- *OpenVMS DCL Dictionary*

- *VMS DCL Concepts Manual*

- *Guide to OpenVMS File Applications*

ULTRIX client users may also find the following Digital DECnet-ULTRIX manual to be helpful:

- *DECnet-ULTRIX Use*

DOS client users may also find the following PATHWORKS for DOS manual to be helpful:

- *PATHWORKS for DOS DECnet User's Guide*

OS/2 client users may also find the following PATHWORKS for OS/2 manual to be helpful:

- *PATHWORKS for OS/2 Utilities Guide*

IBM client users may also find the following IBM manuals to be helpful:

- *JCL Reference*

- *CMS Primer*

- *IBM Access Method Services*

The following IBM manuals provide additional reference information that could be helpful in problem determination:

- *DFP: Access Method Services Reference*

- *IBM VTAM Customization*

- *MVS JCL Reference Manual*

- *MVS Service Aids Manual*

- *MVS System Commands*

- *MVS System Messages*

- *VTAM Operator Commands*

- *VM Planning and Reference*

- *VM/SP6 Connectivity, Planning, and Administration*

- *VM/SP5 TSAF*

Acronyms

The following acronyms are used throughout this manual:

ASCII	American Standard Code for Information Interchange
DCL	Digital Command Language
DMCS	Digital Multinational Character Set
DTF	Digital SNA Data Transfer Facility software
EBCDIC	Extended Binary Coded Decimal Interchange Code
ESDS	Entry Sequenced Data Set
GDG	Generation Data Group
HSM	IBM's Hierarchical Storage Manager
IBM MVS	IBM's Multiple Virtual Storage
IBM SNA	IBM's Systems Network Architecture
IBM VM	IBM's Virtual Machine
ISPF	Interactive System Productivity Facility
KSDS	Key Sequenced Data Set
MS-DOS	Microsoft Corporation's DOS operating system
NFT	Network File Transfer utility
PDS	Partitioned Data Set
REXX	Restructured Extended Executor
RRDS	Relative Record Data Set
SMS	IBM's Storage Management Subsystem
SNADTFCFG	SNA DTF Configuration utility
TRANSFER/DTF	SNA DTF Transfer utility
TSO	IBM's Time Sharing Option
ULTRIX	Digital's UNIX-based operating system
OpenVMS/DTF	Digital-resident Data Transfer Facility software
OpenVMS	The OpenVMS operating system
VSAM	IBM's Virtual Storage Access Method

VTAM   IBM's Virtual Telecommunications Access Method

Conventions Used in this Document

The following conventions are used throughout this manual:

Special type   System output and user input appears in this type.

UPPERCASE   Uppercase letters in command lines indicate keywords that must be
                  entered. You can enter keywords in either uppercase or lowercase.
                  You can abbreviate command keywords to the first three characters
                  or their minimum unique abbreviation.

lowercase italics   Lowercase italics in command syntax or examples indicate variables
                  for which either the user or the system supplies a value.

CTRL/*x*   Hold down the CTRL key and press the key specified by *x*.

[ ]   Square brackets in command syntax statements indicate that the
                  enclosed value is optional. You can enter none or one. Default
                  values apply for unspecified options. (Do not type the brackets when
                  you enter a command.)

Return   Press the RETURN key. The RETURN key, which you must press to
                  execute commands, is assumed in command examples and therefore
                  is not always shown in command displays.

VM   Instructions or comments that apply to the IBM/VM operating
                  system only.

MVS   Instructions or comments that apply to the IBM/MVS operating
                  system only.

1

Configuring the OpenVMS/DTF Server After Installation

Before proceeding with the tasks described in this chapter, you should install the OpenVMS/DTF server software on the nodes you have selected to be OpenVMS/DTF servers. You can also optionally install the OpenVMS/DTF utilities software on additional OpenVMS client nodes. Since the OpenVMS/DTF server software includes the OpenVMS/DTF utilities software, do not install the utilities software on your server nodes. For information about installing OpenVMS/DTF software, refer to the *Digital SNA Data Transfer Facility for OpenVMS Installation* manual.

After you have successfully installed OpenVMS/DTF software, you should perform the following tasks as described in this chapter:

- Configure the OpenVMS/DTF software.

- Set the OpenVMS/DTF operating parameters.

1.1 Configuring the OpenVMS/DTF Software

Before users can transfer files between Digital and IBM systems, you must perform the following configuration tasks to set up the OpenVMS/DTF databases:

- Set up the OpenVMS/DTF server account database.

- Authorize access to server accounts.

- Optionally, set up the OpenVMS/DTF file definition database.

- Optionally, set up the OpenVMS/DTF proxy database.

1.1.1 Invoking and Exiting from the SNADTFCFG Utility

To invoke the SNADTFCFG utility enter the following command:

```
$ MCR SNADTFCFG
```

The SNADTFCFG utility then returns the following prompt:

```
DTFCFG>
```

Once you receive this prompt, you can enter SNADTFCFG commands. To exit from the SNADTFCFG utility, type EXIT or press CTRL/Z after the DTFCFG> prompt.

1.1.2 Naming a Default Server Node

The SNADTFCFG utility allows you to manage multiple OpenVMS/DTF servers from a single node. To name an OpenVMS/DTF server node other than the local node as the default server node, use the USE [NO]NODE command. The node you specify is used as the default node when you are performing operations with other SNADTFCFG commands. If you do not define a default server node, the default is the local node.

The default OpenVMS/DTF server node is set with the USE NODE command and can be cleared with the USE NONODE command. The USE NONODE command returns the default node to the local node. You can override the default node at any time by simply including a node name with the access control information when you specify a server account or a file definition.

USE [NO]NODE [*node*["*access-control*"]::]

Note

You cannot use the NONODE keyword with the *node* parameter.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node at which records are added, modified, displayed, or removed from any of the databases. If the double colon (::) is not provided, SNADTFCFG appends a double colon to the node name.

The *access-control* parameter specifies the name and password of an account with read and write access to all the OpenVMS/DTF databases. The account name is usually SNADTF$MGR.

<hr>

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

<hr>

Example

```
DTFCFG> USE NODE BUFFALO"SNADTF$MGR SECRET"
```

This example defines BUFFALO as the default server node for other SNADTFCFG commands (ADD, MODIFY, REMOVE, and SHOW). The access control information within double quotation marks (" ") consists of an account name (SNADTF$MGR) and password (SECRET) for an account with read and write access to all the OpenVMS/DTF databases.

1.1.3 Setting Up the OpenVMS/DTF Server Account Database

An OpenVMS/DTF server account database is automatically created with initial server accounts (SNADTF and SNASERVER) as part of the OpenVMS/DTF server installation procedure. After installation you should set up the server account database by modifying the SNADTF and SNASERVER server accounts using the MODIFY SERVER_ACCOUNT command. You can then add additional server accounts with the ADD SERVER_ACCOUNT command.

You use server accounts to specify the information OpenVMS/DTF requires to access an IBM system. You will need to set up several server accounts on a single OpenVMS/DTF server node if your server node will be doing one or more of the following:

- Accessing more than one IBM system

- Using more than one Digital SNA Gateway to access a single IBM system

- Using more than one copy of DTF for IBM on a single IBM system

- Using more than one physical unit (PU) or logical unit (LU) on a Digital SNA Gateway

- Using more than one access name on a single PU or LU on the Digital SNA Gateway

All communications between DECnet nodes or TCP/IP hosts and IBM systems are defined in terms of two types of sessions: server sessions and transfer sessions. Server sessions route transfer requests received from IBM DTF users to DECnet nodes or TCP/IP hosts capable of handling the requests. Transfer sessions pass the actual data to and from DECnet nodes or TCP/IP hosts and DTF for IBM software.

You should create different server accounts for each session type. The server
session should have an IBM request/response unit (RU) size of at least 800
bytes, while transfer sessions can have varying RU sizes depending upon the
application. You can change the RU sizes through the IBM logon mode tables
used for your server account. Section 1.2 describes how to modify your system
startup file so that you can specify a different server account for the transfer
session.

Each server account allows you to specify information about the qualifiers
described in Table 1–1.

1.1.3.1 Adding and Modifying Entries in the Server Account Database

To modify the SNADTF and SNASERVER server accounts, use the MODIFY
SERVER_ACCOUNT command:

MODIFY SERVER_ACCOUNT[/*qualifiers*] [*node*["*access-control*"]::]*server-
account-name*

To add a new server account name to the server account database, use the
ADD SERVER_ACCOUNT command:

ADD SERVER_ACCOUNT[/*qualifiers*] [*node*["*access-control*"]::]*server-account-
name*

The parameters and qualifiers for the ADD and MODIFY commands are
identical and are described in the sections that follow.

For more detailed information about each of these commands and their
parameters and qualifiers, see Chapter 2.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node on which the
server account database resides. If you omit the *node* parameter from the
ADD SERVER_ACCOUNT command, the server node defined by the last USE
NODE command is selected by default. If you have not issued a USE NODE
command, then your current node is selected.

The *access-control* parameter specifies the name and password of an account on
the server node with read and write access to all the databases. The account
name is usually SNADTF$MGR.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database
for the *access-control* parameter.

_____

server-account-name
The name for the server account in the OpenVMS/DTF server account database.

Command Qualifiers

Table 1–1 Qualifiers Used for Server Account Commands

/ACCESS_NAME=*access-name*	Specifies the Digital SNA Gateway, OpenVMS SNA (OpenVMS VAX Version 6.1 and Version 6.2 only) access name. See the Digital SNA Gateway manager for this name.
/APPLICATION_NAME=*application-name*	Specifies the DTF for IBM application name. See the IBM system programmer for this name.
/FILE_DEFINITION=*file-definition*	Specifies the name of the default file definition to use for this server account. See Section 1.1.6 for more information.
/GATEWAY=*gateway-node*	Specifies the SNA Gateway DECnet node name, TCP/IP host name, or OpenVMS SNA name used to access the IBM system.
/[NO]IBM_PREFIX	Specifies a 0 to 4 character prefix string that will be used by DTF to prefix the IBM user name when using IBM-initiated proxy. See Chapter 2 for more information.
/[NO]LOG	Specifies if an information message confirming the operation is desired.
/LOGON_MODE=*logon-mode*	Specifies the IBM logon mode value. See the IBM system programmer for this value.
/NODE=*dtf-file-service-node* VM	Specifies the IBM system's physical node name in a Transparent Services Access Facility (TSAF) collection. See the IBM system programmer for this value.
/NULL=*vm-null-qualifier*	Specifies the response OpenVMS/DTF should use when it encounters null records during a transfer to VM systems. See Chapter 2 for more information.

(continued on next page)

Table 1–1 (Cont.)   Qualifiers Used for Server Account Commands

/PU=*pu-name* | *lu-name*	Specifies the physical unit (PU) to use on the DECnet SNA Gateway, or OpenVMS SNA. Specifies the logical unit (LU) to use on Digital SNA Domain Gateway or Digital SNA Peer Server. See the SNA Gateway manager for this name.
/SERVICE_MACHINE=*virtual-machine-name* VM	Specifies the name of the VM virtual machine running the DTF for IBM software. See the IBM system programmer for this value.
/SESSION_ADDRESS=*session*	Specifies the logical unit (LU) to use on the DECnet SNA Gateway, or OpenVMS SNA. See the Digital SNA Gateway manager for this name. This information is ignored when connecting to a Digital SNA Domain Gateway or Digital SNA Peer Server. See the /PU qualifier for information on how to specify an LU to use in these gateways.
/[NO]TRANSLATION=*translation-location*	Specifies the location (either LOCAL or REMOTE) where data translation should occur. See Chapter 2 for more information.

Example 1
```
DTFCFG> MODIFY SERVER_ACCOUNT/GATEWAY=SNAGWY/ACCESS_NAME=DTF-
_DTFCFG> /APPLICATION=DTFV302/PU=SNA-0/LOGON=R8129/NULL=IGNORE -
_DTFCFG> SNASERVER
```

This example modifies entry SNASERVER in the server account database on the local node. The /GATEWAY qualifier specifies that the node SNAGWY provides access to the IBM system referenced by SNASERVER. The /ACCESS_NAME qualifier specifies that the access name DTF on the SNA Gateway contains the proper information for accessing the IBM system. The /APPLICATION, /PU, and /LOGON qualifiers specify information about the access name. The /NULL qualifier specifies that the default action will be to ignore null records during transfers to VM systems. The OpenVMS/DTF software uses this information when starting the server session SNASERVER.

Example 2
```
DTFCFG> MODIFY SERVER_ACCOUNT/GATEWAY=SNAGWY/ACCESS_NAME=DTF-
_DTFCFG> /APPLICATION=DTFV302/PU=HOIDEOIE/LOGON=R8129/NULL=IGNORE -
_DTFCFG> SNASERVER
```

This example is just like Example 1, except an LU name is specified in the /PU qualifier.

1.1.4 Creating OpenVMS Accounts for DTF Server Accounts

Every OpenVMS/DTF server account used for a transfer session must contain
a corresponding entry in the OpenVMS user authorization file (UAF).
After adding a server account, you must also add the UAF record with the
Authorize utility. You can use the following sequence of commands to invoke
the Authorize utility:

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
```

The system responds with the following prompt:

```
UAF>
```

Then enter either of the following commands.

```
UAF> COPY SNADTF new-server-acct-name/NOADD_IDENTIFIER/NOPASSWORD
```

```
UAF> COPY SNADTF new-server-acct-name/NOADD_ID/PASSWORD=new-pass
```

You must supply the new server account name and you can supply an optional
password. All the default authorization characteristics are then copied from
the default SNADTF server account to your new server account. Refer to the
OpenVMS System Management Utilities Reference manual for more information
about using the Authorize utility.

If you decide to add a password to the server account, you should use the
/PASSWORD qualifier. If you do not specify the /NOADD_IDENTIFIER
qualifier an error message is reported which you can ignore.

If the server account is not given a password, all users on any OpenVMS/DTF
client node or server node can use DTF. However, they can access IBM-resident
files only if they know the IBM file's correct access control.

1.1.5 IBM-Initiated Proxy

IBM-initiated proxy allows the IBM user to submit a request with the
DECnet file specifications without specifying the DECnet password. Refer
to the chapter on data security in the *Digital SNA Data Transfer Facility for
OpenVMS Use* for more information on IBM-initiated proxy. For information
on adding proxy entries, refer to the Authorize Utility in *OpenVMS System
Management Utilities Reference* manual.

1.1.5.1 How Does IBM-Initiated Proxy Work?

The proxy concept is based on a pseudo-IBM user name constructed from
an optional prefix string attached to the actual IBM user name. The prefix
string can be 0 to 4 characters long. The prefix is the differentiator between
an IBM user name and an OpenVMS user name that happen to be the same.
For example, an IBM user name CURRIE might become the pseudo-IBM user
name, IBM_CURRIE.

You can specify IBM prefixes on a server-by-server basis using the ADD
SERVER command. See Chapter 2 for more information on IBM prefixes.

OpenVMS/DTF uses the pseudo-IBM user name in the following ways.

With DTF Running on the Client Node

1. The dispatcher user name changes to the pseudo-IBM name before
   connecting to the SNARCOPY object on the client node.

2. If there is a DECnet proxy entry, and SNARCOPY is running on the client
   node, the connect to the client SNARCOPY succeeds and the transfer runs
   to completion.

Without DTF Running on the Client Node

1. Access fails when the SNADTF$DISPATCHER tries to connect to the
    SNARCOPY object on the client node.

2. The dispatcher invokes SNARCOPY from the server node through
    DTF$SERVER.

3. SNARCOPY on the server node runs and changes its user name to the
    pseudo-IBM user name. (The node name in the request will now be the
    server node name, and the user name will be the pseudo-IBM user name,
    server-node::*pseudo-ibm-name*.)

4. If there is a DECnet proxy entry on the client node, SNARCOPY on the
    server node is able to access the files on the client node and the copy runs
    to completion.

Note

Before proxy can work, the user must ask the OpenVMS system
manager of the remote DECnet system to enter a proxy entry coming
from the *server-node*::*pseudo-account-name* and accessing the local
account name. This one proxy entry will work regardless of whether
SNARCOPY is on the remote node or not.

1.1.5.2 Example of an IBM-Initiated Transfer Request with DTF

Figure 1–1 illustrates a request made from a user on an IBM client system for
DTF services. The remote DECnet node is running DTF. Note that the DECnet
SNA Gateway can also represent a Digital SNA Domain Gateway or Digital
SNA Peer Server, where you can use TCP/IP interchangeably with DECnet.

Figure 1–1 IBM-Initiated Access: DECnet Client Node Running DTF
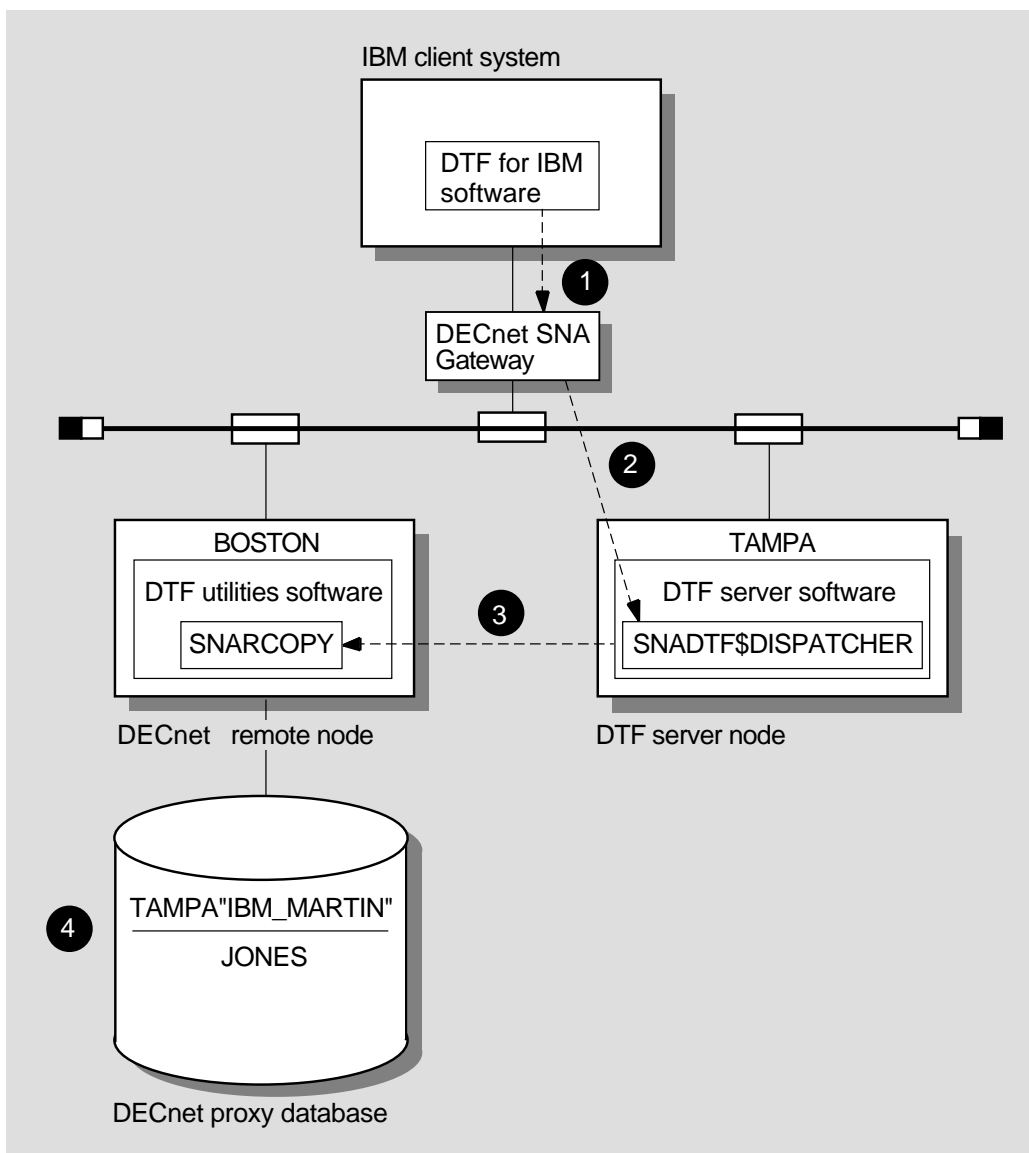
LKG−5493−93R

1. IBM user MARTIN issues a DTF command to access files owned by BOSTON"JONES".

2. The SNADTF$DISPATCHER on server node TAMPA receives the request.

3. The dispatcher, SNADTF$DISPATCHER, connects to the SNARCOPY object on BOSTON by using the user name IBM_MARTIN. (For this example, assume the prefix was set to IBM_.) BOSTON receives the connection to the SNARCOPY object from remote user TAMPA::IBM_MARTIN.

4. DECnet proxy checks whether user TAMPA::IBM_MARTIN is set up for access to JONES. If the proxy entry is defined, SNARCOPY runs under the JONES account.

Node BOSTON requires the following OpenVMS proxy entry:

```
$ MCR AUTHORIZE
UAF>  ADD/PROXY TAMPA::IBM_MARTIN JONES
```

1.1.5.3 Example of an IBM-Initiated Transfer Request

Figure 1–2 illustrates a request made from an IBM user for DTF services. The remote DECnet node is not running DTF. Note that the DECnet SNA Gateway can also represent a Digital SNA Domain Gateway or Digital SNA Peer Server, where you can use TCP/IP interchangeably with DECnet.

Figure 1–2 IBM-Initiated Access: DECnet Client Node Not Running DTF
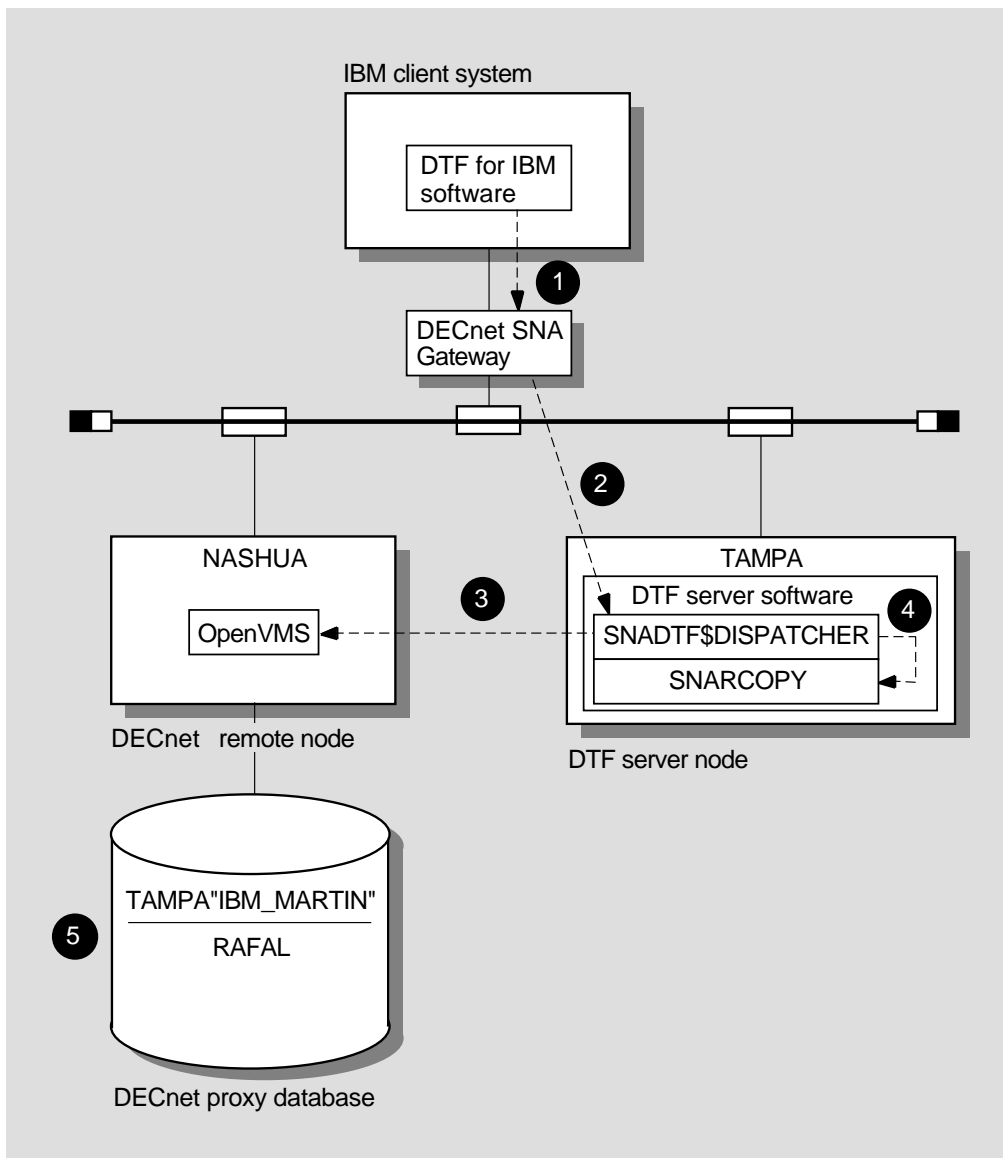
LKG−5632−93R

1. IBM user MARTIN issues a DTF command to access files owned by NASHUA"RAFAL".

2. The SNADTF$DISPATCHER on server node TAMPA receives the request.

3. SNADTF$DISPATCHER changes its name to the IBM user name and tries to connect to the SNARCOPY object on NASHUA. SNARCOPY is not running on NASHUA. The connection fails.

4. SNADTF$DISPATCHER connects to the SNARCOPY object on server node TAMPA. SNARCOPY changes its name to the IBM user name. SNARCOPY uses the user name IBM_MARTIN to complete the request. (For this example, assume the prefix was set to IBM_.)

5. NASHUA sees that TAMPA::IBM_MARTIN is trying to access a file in the RAFAL account. DECnet proxy checks if user TAMPA::MARTIN is set up for access to RAFAL. If the proxy entry is defined, the request continues.

Node NASHUA requires the following OpenVMS proxy entry:

```
$ MCR AUTHORIZE
UAF>  ADD/PROXY TAMPA::IBM_MARTIN RAFAL
```

1.1.6 Setting Up the File Definition Database

A file definition database is automatically created as part of the OpenVMS/DTF server installation procedure. After installation, you may wish to add file definitions with the ADD FILE_DEFINITION command.

A file definition provides users with a short cut for specifying IBM file creation parameters. A file definition consists of a collection of default IBM file creation parameters under a single file definition name. Users can specify the file definition name instead of entering individual IBM file specification qualifiers when transferring files with a COPY command. If users include file specification qualifiers and a file definition name with a COPY command, the qualifiers they include override the parameters specified in the file definition.

Refer to the *Digital SNA Data Transfer Facility for OpenVMS Use* for information on the file creation parameters and for information on the IBM file specification qualifiers.

1.1.6.1 Adding Entries to the File Definition Database

To add a new entry to the file definition database, use the ADD FILE_ DEFINITION command:

ADD FILE_DEFINITION[*/qualifier*] [*node*["*access-control*"]::] *file-definition-name input-file-spec*

or ADD FILE_DEFINITION[*/qualifier*] [*node*["*access-control*"]::] *file-definition-name ibm-file-spec-quals*

For more detailed information about this command and its parameters and qualifiers, see Chapter 3.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the file definition database resides. If you omit the *node* parameter, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

---

file-definition-name
The user-assigned name of the file definition in the definition database.

input-file-spec
The name of the file used to define file creation parameters.

ibm-file-spec-quals
Any of the IBM file specification qualifiers described in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual with the exception of those qualifiers that specify passwords.

Command Qualifier
The /LOG qualifier used to request informational messages about the operation.

Example
```
DTFCFG> ADD FILE_DEFINITION PERSONNEL_FILE PERSONNEL
```

This example creates the PERSONNEL_FILE file definition record in the file definition database on the default node (the node defined by the last USE NODE command). The DTF file definition attributes are read from the PERSONNEL.SNADTF$FDL file in the current directory.

1.2 Setting DTF Operating Parameters

To set the DTF operating parameters, you must edit your system startup file and change the parameters associated with the following OpenVMS/DTF startup command procedures:

- SNADTF$STARTUP_SERVER.COM (for server nodes)

- SNADTF$STARTUP_CLIENT.COM (for client nodes)

After editing the system startup file, you must either invoke the OpenVMS/DTF command procedure using the same parameters or reboot the system.

_____ **Note** _____

When DTF is installed on an OpenVMS cluster, you must run the same DTF startup procedure on all the members of the cluster.

_____

1.2.1 DTF Queues

DTF execution queues are used for recoverable file transfers. Only one recoverable file transfer per queue is allowed at a time. To have concurrent recoverable file transfers, you must start a number of execution queues. The DTF generic queue allows users to treat the DTF execution queues as a single entity. For example, if a user submits a job to the DTF generic queue, the request would be executed by the next available DTF execution queue.

Use the DTF system startup procedures to create and initialize new queues. You can create new queues by altering the parameters passed to the DTF startup command files in your system startup file.

After a queue is created and initialized, you can use the SET QUEUE command to alter queue parameters. The SHOW QUEUE command displays the name and characteristics of the queues after they have been initialized.

1.2.2 The OpenVMS/DTF Server Operating Parameters

The OpenVMSQINSTAL procedure prompts you to add the following line to your system startup procedure:

```
@SYS$STARTUP:SNADTF$STARTUP_SERVER [p1] [p2] [p3] [p4] [p5] [p6] [p7] [p8]
```

where

[*p1*]	specifies the device where the OpenVMS/DTF server work file directory [SNADTF] is located. The default for this parameter is SYS$COMMON:.
[*p2*]	specifies all the OpenVMS/DTF execution queues that you want started at system startup time.
	Note that in an OpenVMS cluster environment, you can also specify the node for each execution queue by using the following format for this parameter:
	[*node::*]*queue-name*[,[*node::*]*queue-name*]
	If this parameter is not specified, an execution queue named SNADTF$QUEUE is initialized and started.
[*p3*]	specifies a YES or NO response to indicate whether or not a generic OpenVMS/DTF queue called SNADTF$QUEUE is started at system startup time. If you specify a null *p2*, then parameter *p3* is ignored. The default value YES specifies that you want to start and initialize the generic queue SNADTF$QUEUE. To use the generic queue, however, you must specify the values for the parameter *p2* that will be used for your execution queues. If you specify NO for *p3* and a value for *p2*, then the SNADTF$QUEUE is initialized and started as an **execution** queue.

[*p4*]   specifies a list of server account names used for server sessions by
              SNADTF$DISPATCHER for IBM-initiated transfers. This list must have a
              one-to-one correspondence with the list in *p6*. The default for this parameter
              is the SNASERVER server account name.

[*p5*]   specifies the time interval that passes before the dispatcher attempts
              to connect to an IBM host after a server session is disconnected. This
              parameter is specified in OpenVMSQ delta time. The default for this
              parameter is 00:15:00.

[*p6*]   specifies a list of server account names to be used for IBM-initiated transfer
              sessions. This list must have a one-to-one correspondence with the list in *p4*.
              The default for this parameter is SNADTF.

[*p7*]   specifies a YES or NO response to indicate whether or not the OpenVMS/DTF
              software should send a cluster alias or system-specific node name to IBM
              for DTF when the server session starts. A YES indicates the cluster alias
              should be used; a NO indicates the system-specific node name should be
              used. If not specified, a cluster alias is used.

[*p8*]   specifies the translation table file name to be used by OpenVMS/DTF for
              translating from ASCII to EBCDIC and from EBCDIC to ASCII. If not
              specified, the default translation table is used. This translation table
              controls how commands over the server sessions are translated; it does
              not affect text translation over the transfer sessions.

The following example shows how you might specify parameters *p2*, *p3*, and
p4.

```
@SYS$STARTUP:SNADTF$STARTUP_SERVER "" DTFQUE01,DTFQUE02 YES DISP1,DISP2
```

In this example, the double quotation marks specify that the [SNADTF]
directory is located on the default device SYS$COMMON:. Two execution
queues (DTFQUE01 and DTFQUE02) as well as a generic queue
(SNADTF$QUEUE) are started. The generic queue SNADTF$QUEUE
supplies jobs to the two OpenVMS/DTF execution queues DTFQUE01 and
DTFQUE02.

Two server sessions are started using information from server accounts DISP1
and DISP2. Finally, because parameters *p5* through *p8* are not specified, the
dispatcher retry interval defaults to 15 minutes, IBM-initiated transfers will
use the SNASERVER account for the server session and the SNADTF account
for the transfer session, IBM-initiated transfers will use the cluster alias, and
the default ASCII-to-EBCDIC translation table is used.

Option to Disable Connect to Client SNARCOPY

During an IBM-initiated transfer request, the DTF server software attempts to connect to the client node's SNARCOPY object to get the copy to run on the client node. However, if SNARCOPY is not running on the client node, the copy will run on the server node. This results in a security violation logged on the client node.

You have the option to disable connects to the client node on nonrecoverable transfers. The /NOCONNECT qualifier in the START SERVER command in the SNADTF$DISPATCHER utility activates the disable connect option. A security violation will not be logged if /NOCONNECT is specified.

To disable the /NOCONNECT qualifier to the remote DECnet node's DTF, uncomment the line in the SNADTF$STARTUP_SERVER file which defines the qualifier as /NOCONNECT.

Note

If a recoverable transfer was specified, DTF ignores the /NOCONNECT qualifier and issues a connect to the client node.

1.2.3 The OpenVMS/DTF Client Operation Parameters

The VMSINSTAL procedure reminds you to add the following line to your system startup procedure:

```
@SYS$STARTUP:SNADTF$STARTUP_CLIENT [p1] [p2] [p3]
```

where

[*p1*]   specifies the device on which the OpenVMS/DTF work file directory [SNADTF] is located. The default for this parameter is SYS$COMMON:.

[*p2*]   specifies all the OpenVMS/DTF execution queues that you want started at system startup time.

Note that in an OpenVMS cluster environment, you can also specify the node for each execution queue by using the following format for this parameter:

[*node::*]*queue-name*[,[*node::*]*queue-name*]

If this parameter is not specified, then an execution queue named SNADTF$QUEUE is initialized and started.

[*p3*]   specifies a YES or NO response to indicate whether or not a generic
         OpenVMS/DTF queue (SNADTF$QUEUE) is started at system startup time.
         If you specify a null value for *p2*, then parameter *p3* is ignored. The default
         value YES specifies that you want to start and initialize the generic queue
         SNADTF$QUEUE. To use the generic queue, however, you must specify the
         values for the parameter *p2* that will be used for your execution queues.
         If you specify NO for *p3* and a value for *p2*, then the SNADTF$QUEUE is
         initialized and started as an **execution** queue.

The following example shows how you might specify parameter *p2*:

```
@SYS$STARTUP:SNADTF$STARTUP_CLIENT SYS$COMMON: DTFQUE01,DTFQUE02 YES
```

In this example, the [SNADTF] directory is located on the default device
SYS$COMMON:. Two execution queues (DTFQUE01 and DTFQUE02) as
well as a generic queue (SNADTF$QUEUE) are started. The generic queue
SNADTF$QUEUE supplies jobs to the OpenVMS/DTF execution queues
DTFQUE01 and DTFQUE02.

2

Maintaining the OpenVMS/DTF Server Account Database

The server account database is used by OpenVMS/DTF to obtain information on how to set up the session. The server account name defines the parameters that users need to access DTF for IBM software. A server account database containing server account names resides on each OpenVMS/DTF server node.

The SNADTFCFG utility is used to manage the server account database. You can add new entries, display entries, modify entries, and remove entries as necessary. The following sections describe these operations.

2.1 Adding and Modifying Entries in the Server Account Database

To add a new server account name to the server account database, use the ADD SERVER_ACCOUNT command:

ADD SERVER_ACCOUNT[/*qualifiers*] [*node*["*access-control*"]::]*server-account-name*

To modify the characteristics of a server account name in the server account database, use the MODIFY SERVER_ACCOUNT command:

MODIFY SERVER_ACCOUNT[/*qualifiers*] [*node*["*access-control*"]::]*server-account-name*

Any values for qualifiers not specified with the MODIFY SERVER_ACCOUNT command remain the same.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node on which the server account database resides. If you omit the *node* parameter from the ADD SERVER_ACCOUNT or MODIFY SERVER_ACCOUNT command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

_____

server-account-name
The name for the OpenVMS/DTF server account in the server account database.

Command Qualifiers
/ACCESS_NAME=*access-name*
Specifies a 1- to 16-character SNA Gateway, or OpenVMS SNA access name. This access name represents a list of parameters required for accessing an IBM system. This list includes the PU name, LU list, logon mode, and application name. See the SNA Gateway manager for this name. This qualifier is required if access is provided by OpenVMS SNA.

/APPLICATION_NAME=*application-name*
Specifies a 1- to 8-character DTF/VTAM application name. This qualifier overrides the application name specified with the /ACCESS_NAME qualifier. See the appropriate IBM system programmer for this name.

/FILE_DEFINITION=*file-definition*
Specifies the 1- to 32-character name of the file definition entry that will be used as the default file definition if users do not specify one. See Section 1.1.6 for more information. This qualifier is ignored for server sessions.

/[NO]IBM_PREFIX=*prefix-string*
Specifies a 0- to 4-character prefix string that will be used by DTF to prefix the IBM user name when using IBM-initiated proxy. The server node initiates the proxy access from the concatenation of this IBM prefix and the IBM user's user name. This prefix allows the differentiation between users with the same user name but on different IBM systems, and also between users with the same user name on the IBM system and on the OpenVMS server node. No prefix is used when the default, /NOIBM_PREFIX, is specified. Refer to Chapter 1 for more information on managing IBM-initiated proxy. Refer to *Digital SNA Data Transfer Facility for OpenVMS Use* for information on using IBM-initiated proxy.

/GATEWAY=*gateway-node*
Specifies the 1- to 80-character node name of the node users use to access the
IBM system. You can specify either a SNA Gateway DECnet node, an SNA
Gateway TCP/IP host name, or an OpenVMS SNA node name. To specify an
OpenVMS SNA node name, type 0 for the *gateway-node* parameter.

/[NO]LOG
Displays informational messages about the operation.

/LOGON_MODE=*logon-mode*
Specifies a 1- to 8-character logon mode table entry. The /LOGON_MODE
qualifier overrides the logon mode entry specified with the /ACCESS_NAME
qualifier. See the appropriate IBM system programmer for this name.

/NODE=*dtf-file-service-node* VM
Specifies a 1- to 8-character node in a Transparent Services Access Facility
(TSAF) collection. See the appropriate IBM system programmer for this node
name.

/NULL:*option*
Specifies the action to take when a null record is to be written to an IBM file.
You can choose from the following options:

value	specifies a number or character that should be stored in place of the null record. The value must be specified as one of the following:

	%C*char*	specifies that the indicated character should be stored in place of the null record.
	%D*number*	specifies that the indicated decimal number should be stored in place of the null record. The number must be between -128 and +127 inclusive.
	%X*number*	specifies that the indicated hexadecimal number should be stored in place of the null record. The number must be between 0 and FF inclusive.
	%O*number*	specifies that the indicated octal number should be stored in place of the null record. The number must be between 0 and 277 inclusive.

SPACE	specifies that a single space should be stored in place of the null record.
IGNORE	specifies that the null record should be ignored (that is, not copied).
NONE VM	specifies no special processing will occur. An error will be returned if a null record is encountered.

NONE $\boxed{\text{MVS}}$ specifies no special processing will occur. The record will be accepted as is.

REJECT $\boxed{\text{MVS}}$ specifies that an error should be returned if a null is encountered.

Usage Notes:

- If you do not specify a value, the default specified in the server account database will be used.

- Applies only to non-VSAM files.

- This qualifier is valid only for DECnet to IBM transfers. It will be ignored when copying files from IBM to Digital.

- $\boxed{\text{VM}}$ Null records cannot be written to CMS files. If a null record is encountered, an error will be returned.

/PU=*pu-name* | *lu-name*
Specifies a 1- to 8-character PU name for a PU between the SNA network and either a DECnet SNA Gateway node, or an OpenVMS SNA node, or an LU name for an LU between the SNA network and either a Digital SNA Domain Gateway or a Digital SNA Peer Server. The /PU qualifier overrides the PU or LU specified with the /ACCESS_NAME qualifier. See the SNA Gateway manager for this information.

/SERVICE MACHINE=*virtual-machine-name* $\boxed{\text{VM}}$
Specifies the 1- to 8-character name of the virtual machine which handles file access for this transfer. See the appropriate IBM system programmer for this name.

/SESSION_ADDRESS=*session*
Specifies a number between 1 and 255 inclusive that is the LU number for the transfer session. If you do not specify the /SESSION qualifier, a free LU is selected from the range specified with the /ACCESS_NAME qualifier. See the SNA Gateway manager for this information. When connecting to a Digital SNA Domain Gateway or Digital SNA Peer Server this qualifier is ignored.

---------------------- **Note** ----------------------

This qualifier is not recommended for transfer sessions. This qualifier overrides any range of sessions specified for the SNA Gateway, OpenVMS SNA access name. Specifying this qualifier for a transfer session will restrict the server account to one user and will not allow certain file access operations that require more than one session address.

/[NO]TRANSLATION=*translation-location*
Specifies where data translation is to be performed.

LOCAL   specifies that translation is performed on the OpenVMS server
                   node.

REMOTE   specifies that translation is performed on the IBM node.

/NOTRANSLATION specifies that no data translation is performed.

This qualifier is ignored for server sessions. The default is /TRANSLATION=LOCAL.

ADD SERVER_ACCOUNT Examples

```
DTFCFG> ADD SERVER_ACCOUNT DOPEY"SNADTF$MGR PASSWORD"::SERVER_017-
_DTFCFG> /GATEWAY=SNAGWY/ACCESS=DTF/NULL=IGNORE/TRANSLATION=REMOTE
```

This example creates entry SERVER_017 in the server account database on
node DOPEY. SNADTF$MGR is created at installation time. It is a good idea
to use this account in situations similar to this because the account has access
to the files and databases in the SNADTF$MANAGER area. The /GATEWAY
qualifier stores the value SNAGWY as part of the new server account record
(SERVER_017). The /ACCESS qualifier stores value DTF in this record. If
the transfer is to a VM system, NULL records are ignored and the translation
occurs on the DTF for IBM side. The OpenVMS/DTF server software uses
these values to make a connection to the IBM host system when using server
account SERVER_017.

```
DTFCFG> ADD SERVER_ACCOUNT CORPORATE/GATEWAY=GWY001/ACCESS=CICS-
_DTFCFG> /PU=SNA-0/APPLICATION=SNADTF
```

This example creates entry CORPORATE in the server account database on
the default node (the node defined by the last USE NODE command). The
/GATEWAY, /ACCESS_NAME, /PU, and /APPLICATION qualifiers store values
GWY001, CICS, SNA-0, and SNADTF, respectively, as part of the new server
account CORPORATE.

MODIFY SERVER_ACCOUNT Examples

```
DTFCFG> MODIFY SERVER_ACCOUNT SMILEY"SNADTF$MGR PASSW"::BIG_XFER-
_DTFCFG> /ACCESS_NAME=XCICS/APPLICATION=SNADTF3/SESSION=1
```

This example modifies the server account record for the OpenVMS/DTF server
account BIG_XFER on node SMILEY. The access name becomes XCICS, the
application name becomes SNADTF3, and the session address becomes 1. DTF
uses these new values the next time the server software attempts to connect to
the IBM system using the BIG_XFER server account.

```
DTFCFG> MODIFY SERVER_ACCOUNT STAR/APPLICATION
```

This example modifies the server account record for the OpenVMS/DTF server account STAR on the default node. The /APPLICATION qualifier erases that record's application name field.

2.2 Displaying Server Account Name Information

To display the characteristics of a specified server account name from the server account database specified on the server node, use the SHOW SERVER_ ACCOUNT command:

SHOW SERVER_ACCOUNT [*node*["*access-control*"]::]*server-account-name*

Command Parameters

[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node on which the server account database resides. If you omit the *node* parameter from the SHOW SERVER_ACCOUNT command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

_____

server-account-name
The server account name for the OpenVMS/DTF server account in the server account database. The server account name you specify can include wildcard characters. If you choose to use wildcards, information on all servers with matching wildcards is displayed. A percent sign (%) indicates a single wildcard character, and an asterisk (*) indicates a wildcard string. See the description of the OpenVMS RTL routine STR$MATCH_WILD in the *OpenVMS DCL Dictionary* for more information about wildcards.

Examples

```
DTFCFG> SHOW SERVER_ACCOUNT CORPORATE

Server account:   CORPORATE
Access Name:     CICS        PU:          SNA-0          Node:
Application:     SNADTF      Logon Mode:  FT008192       Session:
Translation:     NONE        Null:        NONE           Service m/c:
File Definition:                                         IBM Prefix:
Gateway:         GWY001::
```

This example displays the record for server account CORPORATE from the database of the default node.

```
DTFCFG> SHOW SERVER_ACCOUNT FILLY::*DTF*

Server account:   DTFMVS
Access Name:     CICS        PU:          SNA-0          Node:
Application:     DTFPROD     Logon Mode:  FT02048        Session:
Translation:     LOCAL       Null:        NONE           Service m/c:
File Definition:                                         IBM Prefix:
Gateway:         GWY003::

Server account:   DTFMVS1
Access Name:     CICS        PU:          SNA-0          Node:
Application:     DTFTST      Logon Mode:  FT02048        Session:
Translation:     LOCAL       Null:        NONE           Service m/c:
File Definition:                                         IBM Prefix: MVS_
Gateway:         GWY003::

Server account:   DTFVM
Access Name:     CICS        PU:          SNA-0          Node:
Application:     VMPROD      Logon Mode:  FT001024       Session:
Translation:     NONE        Null:        NONE           Service m/c:VM1
File Definition:                                         IBM Prefix: VM_
Gateway:         GWY003::

Server account:   SNADTF
Access Name:     CICS        PU:          SNA-0          Node:     VM2
Application:     VMPROD      Logon Mode:  FT001024       Session:
Translation:     REMOTE      Null:        IGNORE         Service m/c:VM3
File Definition:                                         IBM Prefix:
Gateway:         GWY003::
```

This example displays records for all server accounts on node FILLY that contain DTF in the server account name.

2.3 Removing a Server Account Name

To remove a server account name from the server account database on a
specified OpenVMS/DTF server node, use the REMOVE SERVER_ACCOUNT
command:

REMOVE SERVER_ACCOUNT[/*qualifier*] [*node*["*access-control*"]::]*server-
account-name*

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node on which the
server account database resides. If you omit the *node* parameter from the
REMOVE SERVER_ACCOUNT command, the server node defined by the last
USE NODE command is selected by default. If you have not issued a USE
NODE command, your current node is selected.

The *access-control* parameter specifies the name and password of an account on
the server node with read and write access to all the databases. The account
name is usually SNADTF$MGR.

------------------------------ **Note** ------------------------------

Do not use an account in the OpenVMS/DTF server account database
for the *access-control* parameter.

----

server-account-name
The server account name for the OpenVMS/DTF server account in the server
account database.

Command Qualifier
/[NO]LOG
Provides the option of displaying informational messages about the operation.

Examples
```
DTFCFG> REMOVE SERVER_ACCOUNT SERVER_017
```

This example deletes server account record SERVER_017 from the server
account database on the default node.

```
DTFCFG> REMOVE SERVER_ACCOUNT GLEE"SNADTF$MGR PASSWORD"::SNADTF_2
```

This example deletes server account record SNADTF_2 from the server account
database on node GLEE.

3

Maintaining the OpenVMS/DTF File Definition Database

A file definition database resides on each OpenVMS/DTF server node and contains the file definitions that are used to access files on the IBM system.

3.1 Adding Entries to the File Definition Database

To add a new entry to the file definition database, use the ADD FILE_ DEFINITION command:

ADD FILE_DEFINITION[/*qualifiers*] [*node*["*access-control*"]::]*file-definition-name input-file-spec*

or

ADD FILE_DEFINITION[/*qualifiers*] [*node*["*access-control*"]::]*file-definition-name ibm-file-spec-quals*

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the file definition database resides. If you omit the *node* parameter from the ADD FILE_DEFINITION command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

_____

file-definition-name
Specifies the user-assigned name of the file definition in the definition database.

input-file-spec
The name of the file used to define file creation parameters. This file can be used for all supported file types, including VSAM files. See the *Digital SNA Data Transfer Facility for OpenVMS Use* for information on setting up this file.

ibm-file-spec-quals
Specifies IBM file specification qualifiers. These qualifiers are described in the DTF File Specifications chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

Note

The qualifiers that can be specified in the ADD FILE_DEFINITION command are limited to those in the following list:

/ALIGNMENT	/BATCHID	/BLOCK_SIZE
/CATALOG	/CLASS	/DENSITY
/DIRECTORY_BLOCKS	/HSMRECALL	/LABEL
/MDADDRESS	/MRS	/NULL
/OWNERID	/RECORD_DEFINITION	/RELEASE
/RETENTION_PERIOD	/SECURITY_DATA	/SEQUENCE_NUMBER
/SINGLE	/SMSDCLASS	/SMSMCLASS
/SMSSCLASS	/SPANNED	/SUPERSEDE
/TRANSLATE	/UNIT	/USERID
/VOLUME		

Command Qualifier
/[NO]LOG
Provides the option of displaying informational messages about the operation.

ADD FILE_DEFINITION Examples

```
DTFCFG> ADD FILE_DEFINITION PERSONNEL_FILE PERSONNEL
```

This example creates the PERSONNEL_FILE file definition record in the
file definition database on the default node (the node defined by the last
USE NODE command). The DTF file definition attributes are read from the
PERSONNEL.SNADTF$FDL file in the current directory.

```
DTFCFG> ADD FILE_DEFINITION/LOG RAWDATA TAPE_SPECS.TXT
%SNADTF-I-ADDFIL, file definition RAWDATA added
```

This example creates the RAWDATA file definition record in the file definition
database on the default node. The /LOG qualifier indicates that a message
should be displayed indicating that the file definition has been created. The
DTF file definition attributes are read from the TAPE_SPECS.TXT file in the
current directory.

```
DTFCFG> ADD FILE_DEFINITION SWNODE"SNADTF$MGR SECRET"::SWD_DATA-
_DTFCFG> SWD_DATA
```

This example creates the SWD_DATA file definition record in the file definition
database on node SWNODE. The DTF file definition attributes are read from
the SWD_DATA.SNADTF$FDL file in the current directory.

```
DTFCFG> ADD FILE_DEFINITION NON_VSAM1/NOSUPERSEDE/MDADDRESS=193
```

This example creates a file definition record NON_VSAM1 in the file definition
database on the default node (the node defined by the last USE NODE
command). The values for the various file creation parameters are specified on
the command line. Files created using the NON_VSAM1 file definition will not
be superseded and will be placed on the minidisk with an address of 193.

3.2 Displaying File Definition Attributes

To display the attributes of a single file definition or a group of file definitions
in the file definition database on a specified node, use the SHOW FILE_
DEFINITION command:

SHOW FILE_DEFINITION[/*qualifiers*] [*node*["*access-control*"]::]*file-definition-
name*

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the file
definition database resides. If you omit the *node* parameter from the ADD
FILE_DEFINITION command, the server node defined by the last USE
NODE command is selected by default. If you have not issued a USE NODE
command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

---

file-definition-name
Specifies the user-assigned name of the file definition in the definition database.

The file definition you specify can include wildcard characters. If you choose to use wildcards, information about all file definitions with matching wildcards is displayed. A percent sign (%) indicates a single wildcard character, and an asterisk (*) indicates a wildcard string. Refer to the description of the OpenVMS RTL routine STR$MATCH_WILD in the *OpenVMS DCL Dictionary* for more information about wildcards.

Command Qualifiers
/BRIEF
Displays only the file definition names. By default, the entire contents of a file definition are displayed.

/OUTPUT[=*file-specification*]
The name of the file where the output of the SHOW FILE_DEFINITION command is to be placed. If the /OUTPUT qualifier is used without an argument, SYS$OUTPUT is used as the default.

SHOW FILE_DEFINITION Examples
```
DTFCFG> SHOW FILE_DEFINITION/BRIEF *DAT*

!
!    File definition:
!
RAWDATA     SWD_DATA
```

This example displays the file definition names in the database containing the string DAT. The /BRIEF qualifier causes only the names to be displayed, rather than the entire file definition.

```
DTFCFG> SHOW FILE_DEFINITION/OUTPUT=VSAM_ESDS VSAM_ESDS_FDL
```

This example displays the VSAM_ESDS_FDL file definition. The /OUTPUT qualifier directs the output from this command to the VSAM_ESDS.SNADTF$FDL file in the current directory.

```
DTFCFG> SHOW FILE_DEFINITION PERSONNEL_FILE


!
!   File definition: PERSONNEL_FILE
!
BASE
     RECORD_DEFINITION           PERSONNEL_RECORD
     RELEASE                     NO
     SINGLE                      NO
     SPANNED                     NO
     SUPERSEDE                   NO
     TRANSLATE                   YES
     UNIT                        DISK01
     USERID                      PERONL
     VOLUMES                     VOL1
     VSAM_SEQUENTIAL             NO
```

This example displays the PERSONNEL_FILE file definition on the user's terminal.

3.3 Modifying a File Definition

The process for modifying file definitions depends on whether you are modifying the BASE section or an ALTERNATE_INDEX section. The following two subsections describe the modify process for each instance.

3.3.1 Modifying the File Definition BASE Section

To modify entries in the file definition's BASE section, use the MODIFY FILE_DEFINITION command:

MODIFY FILE_DEFINITION[/*qualifier*] [*node*["*access-control*"]::]*file-definition-name ibm-file-spec-quals*

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the file definition database resides. If you omit the *node* parameter from the MODIFY FILE_DEFINITION command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the databases. The account name is usually SNADTF$MGR.

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

---

file-definition-name
Specifies the user-assigned name of the file definition in the definition database.

ibm-file-spec-quals
Specifies IBM file specification qualifiers. These qualifiers are described in the DTF File Specifications chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

Note

The qualifiers that can be specified in the MODIFY FILE_DEFINITION command are limited to those in the following list:

/ALIGNMENT	/BATCHID	/BLOCK_SIZE
/CATALOG	/CLASS	/DENSITY
/DIRECTORY_BLOCKS	/HSMRECALL	/LABEL
/MDADDRESS	/MRS	/NULL
/OWNERID	/RECORD_DEFINITION	/RELEASE
/RETENTION_PERIOD	/SECURITY_DATA	/SEQUENCE_ NUMBER
/SINGLE	/SMSDCLASS	/SMSMCLASS
/SMSSCLASS	/SPANNED	/SUPERSEDE
/TRANSLATE	/UNIT	/USERID
/VOLUME		

Command Qualifier

/[NO]LOG

Provides the option of displaying informational messages about the operation.

MODIFY FILE_DEFINITION Examples

```
DTFCFG> MODIFY FILE_DEFINITION PERSONNEL_FILE PERSONNEL
```

This example creates the PERSONNEL_FILE file definition record in the
file definition database on the default node (the node defined by the last
USE NODE command). The DTF file definition attributes are read from the
PERSONNEL.SNADTF$FDL file in the current directory.

```
DTFCFG> MODIFY FILE_DEFINITION NON_VSAM1/TRANSLATE/NOSUPERSEDE-
_DTFCFG> /MDADDRESS=195
```

This example updates the NON_VSAM1 file definition. Files created using the
NON_VSAM1 file definition will now use DMCS/EBCDIC data translation, will
not be superseded, and will be on the minidisk with address 195.

```
DTFCFG> MODIFY FILE_DEFINITION MVS_VSAM/USERID=PAYROLL-
_DTFCFG> RETENTION_PERIOD=1000/TRANSLATE
```

This example alters the values associated with the RETENTION_PERIOD and
USERID attributes and allows data translation for the file definition MVS_
VSAM. Note that all of these changes occur only in the BASE section of the file
definition.

3.3.2 Modifying File Definition ALTERNATE_INDEX Sections

To modify file definition ALTERNATE_INDEX sections, do the following:

1. Use the SHOW FILE_DEFINITION command with the /OUTPUT qualifier.
   This creates a copy of the file definition you want to modify.

2. Edit this copy of the file and make your changes.

3. Use the REMOVE FILE_DEFINITION command to remove the file
   definition you want modified.

4. Use the ADD FILE_DEFINITION command to replace the file definition
   with your modified file.

SHOW FILE_DEFINITION Example

```
$ RUN SYS$SYSTEM:SNADTFCFG
DTFCFG> SHOW FILE_DEFINITION/OUTPUT=NEW_CHANGES.DAT PERSONNEL_FILE
DTFCFG> EXIT
$ EDIT NEW_CHANGES.DAT
    .
    .
    .
* EXIT
$ RUN SYS$SYSTEM:SNADTFCFG
DTFCFG> REMOVE FILE_DEFINITION PERSONNEL_FILE
DTFCFG> ADD FILE_DEFINITION PERSONNEL_FILE NEW_CHANGES.DAT
DTFCFG> EXIT
```

In this example, PERSONNEL_FILE is the file definition you want to modify;
NEW_CHANGES.DAT is the copy of the file definition that you use to make
your actual changes.

3.4 Removing an Entry from the File Definition Database

To remove a file definition from the file definition database on a specified server
node, use the REMOVE FILE_DEFINITION command:

REMOVE FILE_DEFINITION [*node*["*access-control*"]::]*file-definition-name*

Command Parameters

[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the
file definition database resides. If you omit the *node* parameter from the
REMOVE FILE_DEFINITION command, the server node defined by the last
USE NODE command is selected by default. If you have not issued a USE
NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an
account on the server node with read and write access to all the databases.
The account name is usually SNADTF$MGR.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database
for the *access-control* parameter.

_____

file-definition-name
Specifies the user-assigned name of the file definition in the definition
database.

REMOVE FILE_DEFINITION Examples

```
DTFCFG> REMOVE FILE_DEFINITION ACCOUNTING_FILE
```

This example deletes the record called ACCOUNTING_FILE from the file definition database on the current node.

```
DTFCFG> REMOVE FILE_DEFINITION MOOSE"SNADTF$MGR SECRET"::POP_FILE
```

This example deletes the record called POP_FILE from the file definition database on node MOOSE. The access control information within double quotation marks (" ") consists of an account name (SNADTF$MGR) and password (SECRET) for an account with read and write access to the file definition database.

A
Messages

This appendix lists and describes all OpenVMS/DTF SNADTFCFG messages. All OpenVMS/DTF messages have the following format:

facility-l-ident, text

where

facility   is the name of the facility or program that generates the message. All the messages in this appendix have the facility code SNADTF.

l   is a severity level indicator with one of the following values:

Code	Meaning
S	Success—the action completed successfully.
I	Information—you do not need to take any further action.
W	Warning—OpenVMS/DTF may have performed some of your request.
E	Error—the output is incorrect. You may need to refer to the problem determination guide for your interconnect system or report the problem to Digital using a Software Performance Report (SPR).
F	Fatal, or severe, error—OpenVMS/DTF cannot continue with your request. You may need to refer to the problem determination guide for your interconnect system or report the problem to Digital using a Software Performance Report (SPR).

ident   is an abbreviation of the message text.

text   is the explanation of the message.

A sample OpenVMS/DTF message follows:

```
%SNADTF-I-ADDFIL, file definition 'DTF1' added
```

A.1 OpenVMS/DTF Messages

The following messages are listed alphabetically according to the abbreviated message text (*ident*).

ACCROUFAI, error from Gateway access routine, gateway unknown or unreachable,

Explanation: SNA Gateway is unknown or unreachable; Transport list (defined by SNA_TRANSPORT_ORDER logical) is defined incorrectly or Gateway/Host Name specified does not support transport selected; or TCP/IP Port (defined by SNA_TCP_PORT logical) does not match the remote connection TCP/IP Port.

User Action: Check the SNA Gateway, the SNA_TRANSPORT_ORDER logical, or the SNA_TCP_PORT logical.

ADDFIL, file definition *file-definition* added

Explanation: File definition *file-definition* was added to the file definition database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

ADDPRO, proxy account *proxy-account* added

Explanation: The proxy account was added to the proxy database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

ADDSER, server account *server-account* added

Explanation: Server account *server-account* was added to the server account database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

BADIDXSEQ, alternate index name *index-name* is out of sequence

Explanation: An alternate index name is out of sequence. When alternate index names are specified for a VSAM indexed file, all names must be in ascending EBCDIC sequence. The major difference between the two character coding schemes has to do with the relationship between numeric and alphabetic characters. Numeric characters appear after alphabetic characters in the EBCDIC collating sequence, while numeric characters appear before alphabetic characters in the ASCII collating sequence.

User Action: Modify the file definition input file so that the alternate index names are in EBCDIC collating sequence order.

BUGCHK,   internal error detected in *routine-name* at PC *xxxxxxxx*

Explanation: An internal error has been detected in the OpenVMS/DTF software. This error is fatal and causes the running image to exit immediately. The occurrence of this message implies that the OpenVMS/DTF software is operating incorrectly.

User Action: Record this error message and all subsequent error messages associated with this error. You should gather information on all the circumstances surrounding this error, and then report the problem to Digital by submitting a Software Performance Report (SPR) or by calling the Digital support hotline.

CLIRET,   unexpected value returned by routine CLI$PRESENT

Explanation: This is a secondary message to the BUGCHK message.

User Action: See BUGCHK.

COMMEXC,   failed to execute command successfully

Explanation: The SNADTFCFG utility failed to execute a command successfully.

User Action: Examine the subsequent error messages to determine why the command could not be executed.

CONVERMIS,   SNADTF$MANAGER:SNADTF$SERVER_ACCOUNT.DAT is incompatible

Explanation: The server name database file on the OpenVMS/DTF server node is incompatible with the running version of the OpenVMS/DTF product.

User Action: Delete the SNADTF$SERVER_ACCOUNT.DAT;* file from the SNADTF$MANAGER directory on the OpenVMS/DTF server node. Use the SNADTFCFG ADD SERVER_ACCOUNT command to create a new server account database file.

CREATEACC,   if *server-account* is used for file transfers, add an OpenVMS account

Explanation: This message is reported when a new server account is created on an OpenVMS/DTF server node. If the new server account is used for transfer sessions and not for server sessions, an OpenVMS account must be created on the OpenVMS/DTF server node using the OpenVMS AUTHORIZE utility.

User Action: Use the OpenVMS AUTHORIZE COPY command to create an OpenVMS user account entry with the same name as the

OpenVMS/DTF server account. You need this account to copy the existing SNADTF UAF record into the new UAF record.

ERRCLOCON, error closing server account database *file-spec*

Explanation: The server account database file could not be closed.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRCLOFIL, error closing file definition database *file-spec*

Explanation: The file definition database file could not be closed.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRCLOOUT, error closing output file *file-spec*

Explanation: The output file could not be closed.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRCLOPRO, error closing proxy database *file-spec*

Explanation: The proxy database file could not be closed.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRDELPRO, error deleting proxy database *file-spec*

Explanation: The proxy record could not be deleted from the proxy database file.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERROPECON, error opening server account database *file-spec*

Explanation: The server account database file could not be opened.

User Action: Examine the subsequent error messages to determine the exact cause of the error. If the problem occurred because the file could not be found, use the SNADTFCFG ADD SERVER_ACCOUNT command to create a new version of the file on the OpenVMS/DTF server node.

ERROPEFIL, error opening file definition database *file-spec*

Explanation: The file definition database file could not be opened.

User Action: Examine the subsequent error messages to determine the exact cause of the error. If the problem occurred because the file could not be found, use the SNADTFCFG ADD FILE_DEFINITION command to create a new version of the file on the OpenVMS/DTF server node.

ERROPEPRO, error opening proxy database *file-spec*

Explanation: The proxy database file could not be opened.

User Action: Examine the subsequent error messages to determine the exact cause of the error. If the problem occurred because the file could not be found, use the SNADTFCFG ENABLE PROXY command to create a new version of the file on the OpenVMS/DTF server node.

ERRPARVSA, error parsing VSAM definition file

Explanation: The VSAM definition file cannot be parsed.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRREAPRO, error reading proxy database *file-spec*

Explanation: A proxy record could not be read from the proxy database file.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRSUBMITTING, error encountered while submitting *file-spec*

Explanation: An error was encountered while submitting the indicated file to the DTF queue.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRWRIOUT, error writing to output file *file-spec*

Explanation: A line could not be written to the output file.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

ERRWRIPRO, error writing proxy database *file-spec*

Explanation: A proxy record could not be written to the proxy database file.

User Action: Examine the subsequent error messages to determine the exact cause of the error.

FAITRALOG, failed to translate logical name *logical-name*

Explanation: The logical name *logical-name* could not be translated.

User Action: Examine the subsequent error messages to determine why the specified logical name could not be translated.

FILALREXI, file definition *file-definition* already exists

Explanation: The file definition *file-definition* cannot be added to the file definition database on the OpenVMS/DTF server node because a file definition of the same name already exists.

User Action: Modify the existing file definition, or use the ADD FILE_ DEFINITION command to create a new and different file definition name.

FILDEFCOR, file definition *file-definition* is corrupt

Explanation: This error occurs when attempting to read a file definition from the file definition database. This error indicates that there is an internal consistency error for the specified file definition. This situation can occur if an ADD FILE_DEFINITION command is interrupted (for example, by the user pressing the CTRL/C sequence).

User Action: Delete the corrupted file definition. Use the ADD FILE_ DEFINITION command to re-enter the file definition into the database.

FILEREM, error removing file definition from *file-spec*

Explanation: The REMOVE FILE_DEFINITION command failed to remove a file definition from the file definition database *file-spec* that resides on the OpenVMS/DTF server node.

User Action: Examine the subsequent error messages to determine why the REMOVE command failed.

FILVERMIS, SNADTF$MANAGER:SNADTF$FILE_DEFINITION.DAT is
incompatible

Explanation: The file definition database file on the OpenVMS/DTF
server node is incompatible with the running version of the OpenVMS/DTF
product.

User Action: Delete the SNADTF$FILE_DEFINITION.DAT;* file from
the SNADTF$MANAGER directory on the OpenVMS/DTF server node.
Use the SNADTFCFG ADD FILE_DEFINITION command to create a new
file definition database file.

IDXNAMMIS, an alternate index name is missing

Explanation: If one alternate index name is specified in a file definition
input file, then all alternate index names must be specified. This error
occurs if there are more indexes specified than index names.

User Action: Modify the file definition input file either to remove all
alternate index names or to specify an alternate index name for all
alternate indexes.

INVNODNAM, *node-name* is an invalid node name

Explanation: An invalid node name was specified in an SNADTFCFG
command.

User Action: Reissue the command specifying a valid DECnet node or
TCP/IP host name.

INVQUAL, invalid value specified for qualifier /*qualifier-name*

Explanation: An invalid value was specified for this IBM file specification
qualifier.

User Action: Correct the command that specified the incorrect value and
reenter the command.

KEYWRDCON, keyword conflict in file definition input

Explanation: Two mutually exclusive keywords were specified in a DTF
file definition file.

User Action: Examine the subsequent error message to determine which
keywords conflict.

MODPRO, proxy account *proxy-account* modified

Explanation: The proxy account in the proxy database on the
OpenVMS/DTF server node was modified.

User Action: None. This is an informational message.

MODSER, server account *server-account* modified

Explanation: Server account *server-account* on the OpenVMS/DTF server node was modified.

User Action: None. This is an informational message.

NOMATFIL, no file definition found that matched specification

Explanation: This error message occurs if you issue an SNADTFCFG SHOW FILE_DEFINITION command using a wildcard file definition, and there are no matching definitions in the OpenVMS/DTF server node file definition database.

User Action: Reissue the command using a more generic wildcard specification (for example: SHOW FILE_DEFINITION *).

NOMATPRO, no proxy account found that matched specification

Explanation: This error message occurs if you issue an SNADTFCFG SHOW PROXY command using a wildcard file definition, and there are no matching definitions or the file is empty, in the OpenVMS/DTF server node proxy database.

User Action: Reissue the command using a more generic wildcard specification (for example: SHOW PROXY *).

NOMATSER, no server account found that matched specification

Explanation: This error message occurs if you issue an SNADTFCFG SHOW SERVER_ACCOUNT command using a wildcard file definition, and there are no matching server accounts in the OpenVMS/DTF server node file server account database.

User Action: Reissue the command using a more generic wildcard specification (for example: SHOW FILE SERVER_ACCOUNT *).

NOSUCHFIL, file definition *file-definition* does not exist

Explanation: The file definition *file-definition* has not been defined in the file definition database on the OpenVMS/DTF server node.

User Action: Add the file definition to the file definition database or use an existing file definition name.

NOSUCHPRO, proxy account *proxy-account* does not exist

Explanation: The proxy account has not been defined in the proxy database on the OpenVMS/DTF server node.

User Action: Create the account or use an existing account.

NOSUCHSER,   server account *server-account* does not exist

Explanation: The server account *server-account* has not been defined in the file definition database on the OpenVMS/DTF server node.

User Action: Create the account or use an account name that has been defined.

PROALREXI,   proxy account *proxy-account* already exists

Explanation: The proxy account *proxy-account* cannot be added to the proxy account database on the OpenVMS/DTF server node because a proxy account of the same name already exists.

User Action: Use the MODIFY PROXY command to modify the existing proxy account, or use the ADD PROXY command to create a new proxy account with a different name.

PRODISABLED,   proxy database is disabled

Explanation: The proxy database has been disabled.

User Action: None. This is an informational message.

PROENABLED,   proxy database is enabled

Explanation: The proxy database has been enabled.

User Action: None. This is an informational message.

PROXYDIS,   proxy database is disabled

Explanation: The proxy database has been disabled.

User Action: If you want to enable the proxy database, use the ENABLE PROXY command.

REMFIL,   file definition *file-definition* removed

Explanation: File definition *file-definition* was removed from the file definition database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

REMPRO,   proxy account *proxy-account* removed

Explanation: The proxy account was removed from the proxy database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

REMSER, server account *server-account* removed

Explanation: Server account *server-account* was removed from the server account database on the OpenVMS/DTF server node.

User Action: None. This is an informational message.

REQUAL, IBM userid and server account qualifiers are required

Explanation: IBM user ID and server account qualifiers are required qualifiers.

User Action: Specify the IBM user ID and server account name qualifiers in the command.

RETEXP, RETENTION_PERIOD and EXPIRATION_DATE are mutually exclusive

Explanation: Both the RETENTION_PERIOD and EXPIRATION_DATE attributes were used in a file definition. These two keywords are mutually exclusive.

User Action: Modify the DTF file definition file to remove one of the two keywords.

SERALREXI, server account *server-account* already exists

Explanation: The server account *server-account* cannot be added to the server account database on the OpenVMS/DTF server node because a server account of the same name already exists.

User Action: Use the MODIFY SERVER_ACCOUNT command to modify the existing server account definition, or use the ADD SERVER_ACCOUNT command to create a new server account definition with a different name.

SERVREM, error removing server account from *server-account*

Explanation: The REMOVE SERVER_ACCOUNT command failed to remove a server account definition from the server account database *server-account* that resides on the OpenVMS/DTF server node.

User Action: Examine the subsequent error messages to determine why the REMOVE command failed.

TOOLONG, the value specified for /*qualifier-name* is too long

Explanation: The value specified for the qualifier was too long.

User Action: Retype the command specifying a correct value for this qualifier.

UNERETSYS,   unexpected return code from a system routine

Explanation: This is a secondary message to the BUGCHK message.

User Action: See BUGCHK.

UNIREU,   UNIQUE and REUSE are mutually exclusive

Explanation: The UNIQUE keyword was used for the COMPONENT_ ALLOCATION attribute and the YES keyword was used for the REUSE attribute. These two keywords are mutually exclusive.

User Action: Modify the file definition input file to alter the value for either the COMPONENT_ALLOCATION or REUSE attribute.

UNRKEY,   *keyword* is not a recognized keyword

Explanation: This is a secondary message to the BUGCHK message.

User Action: See BUGCHK.

UNRKEYVAL,   *value* is not a recognized keyword value

Explanation: This is a secondary message to the BUGCHK message.

User Action: See BUGCHK.

UNRVERB,   unrecognized command verb

Explanation: This is a secondary message to the BUGCHK message.

User Action: See BUGCHK.

VALREQ,   missing keyword value - supply all required values

Explanation: A value must be specified for a keyword in the DTF file definition file.

User Action: Modify the DTF file definition file and supply a value for the keyword.

VALTOOLONG,   the value specified for keyword *keyword-name* is too long

Explanation: The value specified for the keyword was too long.

User Action: Modify the command specifying a correct value for this qualifier.

B

Maintaining the OpenVMS/DTF Proxy Database

The proxy database lets you store IBM security information for an OpenVMS, ULTRIX, RSX-11M/M-PLUS, OS/2, or MS-DOS user. If an authorized DECnet user does not include IBM security information in the access request, the proxy database is checked to see if security information is stored for that user. If it is, that information is included in the access request to the IBM system.

You can use the SNADTFCFG utility to add, modify, display, and remove proxy database entries. To access the proxy database, you must invoke the SNADTFCFG utility from an account that has read and write access to the proxy database and has the SYSPRV privilege. The following sections describe these operations.

_____ **Note** _____

For IBM MVS DTF clients, the IBM system programmer can also set up a proxy database using a technique similar to that used for the OpenVMS/DTF server proxy database. You may want to consult with the IBM system programmer to determine a mutually agreeable course of action concerning proxies. See the *Digital SNA Data Transfer Facility for OpenVMS Use* manual for more information about the interplay between the two proxy databases if both are created.

_____

B.1 Enabling the Proxy Database

To enable the proxy database, use the ENABLE PROXY command:

ENABLE PROXY [*node*["*access-control*"]::]

This command enables the database and creates the database if one is not already available.

Command Parameter

[node["access-control"]::]

The *node* parameter specifies the OpenVMS/DTF server node where the proxy database resides. This parameter must be specified if the proxy database is on a node other than the one you are currently on. If the *node* parameter is omitted from the ENABLE PROXY command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the OpenVMS/DTF databases and with the SYSPRV privilege.

---------------------------------- **Note** ----------------------------------

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

B.1.1 Automatic Update of the Proxy Database

The OpenVMS/DTF server software updates the DTF proxy database when a valid password is specified on the command line and it is different from the password in the database. This allows users to update the proxy database automatically when accessing an IBM file using DTF.

To update the passwords stored in the proxy database automatically, users should specify both the /USER and /PASSWORD qualifiers or both the /USER qualifier and any of the minidisk password qualifiers. If the specified password is different from the one stored in the database and the access works using the new password, then the new password is stored in the database.

This function was built into the product as an ease-of-use feature for users who specify updated passwords.

B.1.2 Option to Disable Automatic Update to the Proxy Database

The option to disable automatic update of passwords was added to facilitate those users who typically do not store IBM passwords. However, this option will allow users to obtain other security information included in an access request sent to the IBM system.

To disable automatic updating of the proxy database, specify the /NOAUTOMATIC qualifier on the ENABLE PROXY command line. To enable the feature, specify the /AUTOMATIC qualifier on the ENABLE PROXY command line. The feature is enabled by default.

B.2 Disabling the Proxy Database

To disable the proxy database, use the DISABLE PROXY command:

DISABLE PROXY [*node*["*access-control*"]::]

The records in the database remain stored there, but they cannot be used as long as the proxy database is disabled.

Command Parameter
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the proxy database resides. This parameter must be specified if the proxy database is on a node other than the one you are currently on. If the *node* parameter is omitted from the DISABLE PROXY command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the OpenVMS/DTF databases and with the SYSPRV privilege.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

_____

DISABLE PROXY Examples
```
DTFCFG> DISABLE PROXY
```

This example disables the proxy database on the current node or the node defined by the last USE NODE command.

```
DTFCFG> DISABLE PROXY RABBIT"SNADTF$MGR PASSWORD"::
```

This example disables the proxy database on node RABBIT.

B.3 Adding and Modifying Entries in the Proxy Database

To add a new entry to the proxy database, use the ADD PROXY command:

ADD PROXY[/*qualifiers*] [*node*["*access-control*"]::] *proxy-account-name*/SERVER_ACCOUNT=*server-account* /USERID=*ibm-userid*

The first entry for any node/user/server account triple becomes the default record in the proxy database. The /SERVER_ACCOUNT and /USERID qualifiers must be specified when using this command.

To change a user's IBM password or minidisk passwords, use the MODIFY PROXY command.:

MODIFY PROXY[/*qualifiers*] [*node*["*access-control*"]::] *proxy-account-name*/SERVER_ACCOUNT=*server-account* /USERID=*ibm-userid*

The /SERVER_ACCOUNT and /USERID qualifiers are required to specify the proxy entry to be modified. In addition, if there are multiple entries for a node/user/server account triple, you must also specify the /NODE qualifier to specify the proxy entry to be modified. You cannot modify the IBM user ID, the name of the server, the nodename, the minidisk address, or the minidisk owner ID. If you need to modify any of these fields, remove the entry and add a new entry with the modified information.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the proxy database resides. This parameter must be specified if the proxy database is on a node other than the one you are currently on. If the *node* parameter is omitted from the ADD PROXY or MODIFY PROXY command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the OpenVMS/DTF databases and with the SYSPRV privilege.

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

---

proxy-account-name
Specifies the name of the user in the proxy database.

Command Qualifiers
/DEFAULT
Specifies this record is the default. If a default record was previously specified for this node/user/server account triple, the previous default record is changed to a non-default record and the record specified with this qualifier becomes the new default record.

/[NO]LOG
Provides the option of displaying informational messages about the operation.

/MDADDRESS=*disk-address* VM
Specifies the address of the minidisk using up to 4 hexadecimal digits. For
more information on minidisk addresses see the DTF File Specifications
chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

/MDMPASSWORD=*multi-password* VM
Specifies the multiwrite password for the minidisk using up to 8 characters.
For more information on minidisk passwords, see the DTF File Specifications
chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

/MDRPASSWORD=*read-password* VM
Specifies the read password for the minidisk using up to 8 characters. For
more information on minidisk passwords, see the DTF File Specifications
chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

/MDWPASSWORD=*write-password* VM
Specifies the write password for the minidisk using up to 8 characters. For
more information on minidisk passwords, see the DTF File Specifications
chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

/NODE=*client-node*
The remote client node from which the user is accessing OpenVMS/DTF. If
available, the cluster alias name is used as the default. Otherwise, the name
of the node where the database is being updated is used as the default for this
qualifier.

/OWNERID=*owner-id* VM
Specifies the 1- to 8-character minidisk owner as specified in the VM directory.
For more information on minidisk owner IDs, see the DTF File Specifications
chapter in the *Digital SNA Data Transfer Facility for OpenVMS Use* manual.

/PASSWORD=*ibm-password*
The 1- to 8-character IBM password for a particular DTF user and the
corresponding IBM user ID.

_____ **Note** _____

You do not need to include the IBM password if the proxy database on
the IBM client is being used for access verification.

_____

/SERVER_ACCOUNT=*server-account*
The server account entry in the server account database. This is a required
qualifier.

/USERID=*ibm-userid*

The 1- to 8-character IBM user ID used by a particular DTF user. This is a required qualifier.

ADD PROXY Examples

```
DTFCFG> ADD PROXY DOE/USERID=JANE/PASSWORD=SECRET-
_DTFCFG> /SERVER=SNADTF/NODE=JOHN
```

This example adds OpenVMS user DOE on node JOHN, whose IBM user ID is JANE and IBM password is SECRET. When user DOE (logged onto node JOHN) accesses the IBM system through DTF server account SNADTF, the MVS user ID JANE and password SECRET will be used to gain access to the IBM system.

```
DTFCFG> ADD PROXY RABBIT"SNADTF$MGR SECRET"::LOU/USERID=LOU-
_DTFCFG> /PASS=IBMPASS/SERVER=IBMS1/NODE=VEENA
```

This example adds into the database that is located on node RABBIT the OpenVMS user LOU on node VEENA and the corresponding IBM user ID LOU and IBM password IBMPASS using the server IBMS1. When user LOU (logged onto node VEENA) accesses the IBM system through the DTF server account IBMS1 on node RABBIT, the MVS user ID LOU and password IBMPASS will be used to gain access to the IBM system.

```
DTFCFG> ADD PROXY CANTER/USERID=CANTER/PASSWORD=SECRET/SERVER=VMDTF-
_DTFCFG> /NODE=BOSTON/MDADDRESS=191/MDRPASSWORD=sight-
_DTFCFG> /MDWPASSWORD=ecrit
```

This example creates an entry for OpenVMS user CANTER on node BOSTON to provide access to IBM user ID CANTER on a node pointed to by server account VMDTF. Also provided are the multi, read, and write minidisk passwords for the 191 disk. These passwords are valid only if the IBM account is on a VM system.

MODIFY PROXY Example

```
DTFCFG> MODIFY PROXY MIKE/USER=MIKE/SER=SNADTF/PASS=NEWPAS
```

This example changes the password of the OpenVMS user MIKE to NEWPAS. The IBM user ID is MIKE and the server account is SNADTF.

B.4 Displaying Entries in the Proxy Database

To display DTF security information, use the SHOW PROXY command:

SHOW PROXY[/*qualifiers*] [*node*["*access-control*"]::]*proxy-account-name*

This command shows whether the proxy database is enabled or disabled and displays the user names, node names, server accounts, IBM user IDs, and last modification dates for the requested users. The IBM passwords are not displayed.

Command Parameters

[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the proxy database resides. This parameter must be specified if the database is on a node other than the one you are currently on. If the *node* parameter is omitted from the SHOW PROXY command, the server node defined by the last USE NODE command is selected by default. If you have not issued a USE NODE command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an account on the server node with read and write access to all the OpenVMS/DTF databases and with the SYSPRV privilege.

Note

Do not use an account in the OpenVMS/DTF server account database for the *access-control* parameter.

---

proxy-account-name
The name of the DTF user in the proxy database. You can use wildcards to show all the users in the proxy database or for a particular server, node, or IBM user ID. You can also display information specifically for default or non-default records.

Command Qualifiers

/DEFAULT
Displays information for default proxy records.

/NODE=*client-node*
The remote client node from which the user is accessing OpenVMS/DTF. The default node used is the cluster alias name, if present, or the node name where the database is being updated.

/SERVER_ACCOUNT=*server-account*
The server account entry in the server account database.

/STATUS

Displays the status of the proxy database only. You can use this qualifier to:

- Check whether the proxy database is enabled or disabled.

- Determine the status of the proxy database on a remote node by specifying the node and access control parameters for that node.

- Check whether automatic update of the DTF proxy database is enabled or disabled.

Note

Do not specify the *proxy-account-name* parameter when using this qualifier; the parameter will be ignored.

If you do not specify any parameters, the status of the proxy database for the current node is displayed.

/USERID=*ibm-userid*

The IBM user ID for a particular DTF user.

SHOW PROXY Examples

```
DTFCFG> SHOW PROXY *

DRAGO::GARY                                                    (D)
Server: DTFVM          IBM Userid: CANTER
Mdaddress: 191         Ownerid:    KNIGHT      25-JUL-1990 15:26

Server: DTFVM          IBM Userid: CANTER
Mdaddress: 193         Ownerid:    KNIGHT      25-JUL-1990 15:26

DRAGO::PETERS                                                  (D)
Server: SNADTF         IBM Userid: PETE
Mdaddress:             Ownerid:                25-JUL-1990 15:31

Server: SNADTF         IBM Userid: PYRLL
Mdaddress:             Ownerid:                25-JUL-1990 15:32
```

This example shows that OpenVMS user GARY on node DRAGO has proxy access to the 191 and 193 disk with owner ID KNIGHT. Use the minidisk address and owner ID when specifying access to a VM system. This example also shows that DRAGO::PETERS has proxy access to the IBM account PETE and PYRLL. The default proxies are noted by the (D).

```
DTFCFG> SHOW PROXY/STATUS
%SNADTF-I-PROENABLED, proxy database is enabled
```

This example shows you the proxy database status on the default node.

```
DTFCFG> SHOW PROXY/STATUS GLEE"SNADTF$MGR PASSWORD"::
%SNADTF-I-PRODISABLED, proxy database is disabled
```

This example shows you the proxy database status on node GLEE.

B.5 Removing an Entry from the Proxy Database

To remove entries from the proxy database use the REMOVE PROXY
command:

REMOVE PROXY[/*qualifiers*] [*node*["*access-control*"]::] *proxy-account-
name*/SERVER_ACCOUNT=*server-account* /USERID=*ibm-userid*

The /SERVER_ACCOUNT and /USERID qualifiers must be specified when
using this command. To remove records that are not defaults, you must also
specify the node name.

Command Parameters
[node["access-control"]::]
The *node* parameter specifies the OpenVMS/DTF server node where the proxy
database resides. This parameter must be specified if the database is on a
node other than the one you are currently on. If the *node* parameter is omitted
from the REMOVE PROXY command, the server node defined by the last USE
NODE command is selected by default. If you have not issued a USE NODE
command, then your current node is selected.

The *access-control* parameter specifies the account name and password of an
account on the server node with read and write access to all the OpenVMS/DTF
databases and with the SYSPRV privilege.

_____ **Note** _____

Do not use an account in the OpenVMS/DTF server account database
for the *access-control* parameter.

_____

proxy-account-name
The name of the user in the proxy database.

Command Qualifiers

/NODE=*client-node*
The remote client node where the user is logged on. If available, the cluster alias name is used as the default. Otherwise, the name of the node where the database is being updated is used as the default value for this qualifier.

/SERVER_ACCOUNT=*server-account*
The server account entry in the server account database.

/USERID=*ibm-userid*
The IBM user ID for a particular DTF user.

REMOVE PROXY Example

```
DTFCFG> REMOVE PROXY MIKE/USERID=MIKE/SER=SNADTF
```

This example removes the proxy record MIKE from the database. The IBM user ID is MIKE and the server account is SNADTF.

Index