



# Administrator's Guide

## Databridge DMSII Client

Version 6.6

## **Legal Notices**

© Copyright 2019 Micro Focus or one of its affiliates.

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

### **Patents**

This Micro Focus software is protected by the following U.S. patents: 6983315, 7571180, 7836493, 8332489, and 8214884

### **Trademarks**

Micro Focus, the Micro Focus logo, and Reflection among others, are trademarks or registered trademarks of Micro Focus or its subsidiaries or affiliated companies in the United Kingdom, United States and other countries. RSA Secured and the RSA Secured logo are registered trademark of RSA Security Inc. All other trademarks, trade names, or company names referenced herein are used for identification only and are the property of their respective owners.

### **Third-Party Notices**

Third-party notices, including copyrights and software license texts, can be found in a 'thirdpartynotices' file located in the root directory of the software.

---

# Contents

<b>About This Guide</b>	<b>5</b>
<b>1 Introducing DMSII Client</b>	<b>7</b>
Introducing the DMSII Client	7
Understanding the Databridge Components	8
DMSII Accessroutines	8
Audit Trail	8
DMSII Database	8
Databridge Engine	9
Support Library	9
Databridge Server	9
DMSII Client Support for DMSII Structure Types	10
Data Sets	10
Data Items	10
Understanding How the DMSII Client Works	10
<b>2 Configuring and Starting the DMSII Client</b>	<b>13</b>
Configuring and Starting the DMSII Client	13
What to Do Next	14
Modifying the DMSII Client DASDL File	15
Sample DMSII Client DASDL Patch File	15
Modifying the DMSII Client Parameter File	18
DMSII Client Parameters	20
<b>3 Command Reference</b>	<b>25</b>
Starting and Stopping the DMSII Client	25
UPDATE	25
CLONE	25
STATUS	26
REGISTER	26
CLEAN	27
LOCATION	27
Stopping the DMSII Client	27
DMSII Client AX Commands	27
AX STATUS	27
AX QUIT	28
AX HELP	28
<b>4 Database Update and Reorganization</b>	<b>29</b>
When COMPILE is set to AUTO	29
When COMPILE is set to MANUAL	29

<b>5 Incorporating Existing Databases</b>	<b>31</b>
Incorporating an Existing Database .....	31
<b>6 Troubleshooting</b>	<b>33</b>
General Troubleshooting Procedures .....	33
Troubleshooting Table .....	33
<b>Glossary of Terms</b>	<b>35</b>

# About This Guide

This guide contains instructions for installing and configuring Attachmate Databridge DMSII Client. This preface includes information to help you use this guide.

To install, configure, and run Databridge, you should be a system administrator familiar with the following:

- ◆ Standard Unisys® operations for MCP-hosted mainframes such as the CS7xxx series, Libra series, ClearPath® NX/LX or A Series
- ◆ DMSII databases and Data And Structure Definition Language (DASDL)

## Conventions

The following conventions and terms may be used in this guide.

<b>This convention or term</b>	<b>Is used to indicate this</b>
<code>this type style</code>	text that you type, filenames and directory names, onscreen messages
<i>italic</i>	variables emphasis document titles
square brackets ( [ ] )	optional items in a command For example, [ true   false ]. (Do not type the brackets.)
pipe (   )	a choice between items in a command or parameter. When enclosed in braces ( { } ), the choice is mandatory.
UPPERCASE	DMSII data set and data item names
MCP server host mainframe	Unisys ClearPath NX, LX or A Series mainframe
DBEngine	Databridge Engine
DBEnterprise	Databridge Enterprise Server
DBServer	Databridge Server

## Abbreviations

The following abbreviations are used throughout this guide and are provided here for quick reference.

<b>Abbreviation</b>	<b>Name</b>
AA	Absolute Address
ABSN	Audit Block Serial Number
AFN	Audit File Number
API	Application Programming Interface
DASDL	Data and Structure Definition Language
DMSII	Data Management System II
IDX	Word Index (offset)
IPC	Inter-Process Communications
MCP	Master Control Program
RPC	Remote Procedure Call
SEG	30-word disk Segment
WFL	Work Flow Language

## Related Documentation

The following is a list of the documentation you might need to consult when using Databridge DMSII Client.

### Databridge Host

Before you can use Databridge DMSII Client, the Databridge Host database replication software must be installed and operating on the primary system. Complete instructions are in the *Databridge Installation Guide*, which is located in the DOCS folder on the Databridge installation image or DVD.

### Unisys Mainframe

You should have available documentation for the Unisys ClearPath NX/LX series or A Series, DMSII, WFL, and CANDE. This guide lists only general instructions. If you are not completely familiar with DMSII configuration, for example, you may need to refer to the Unisys documentation.

# 1 Introducing DMSII Client

This chapter explains what the Databridge DMSII Client is, how it works, and the different components of Databridge.

## In this Chapter

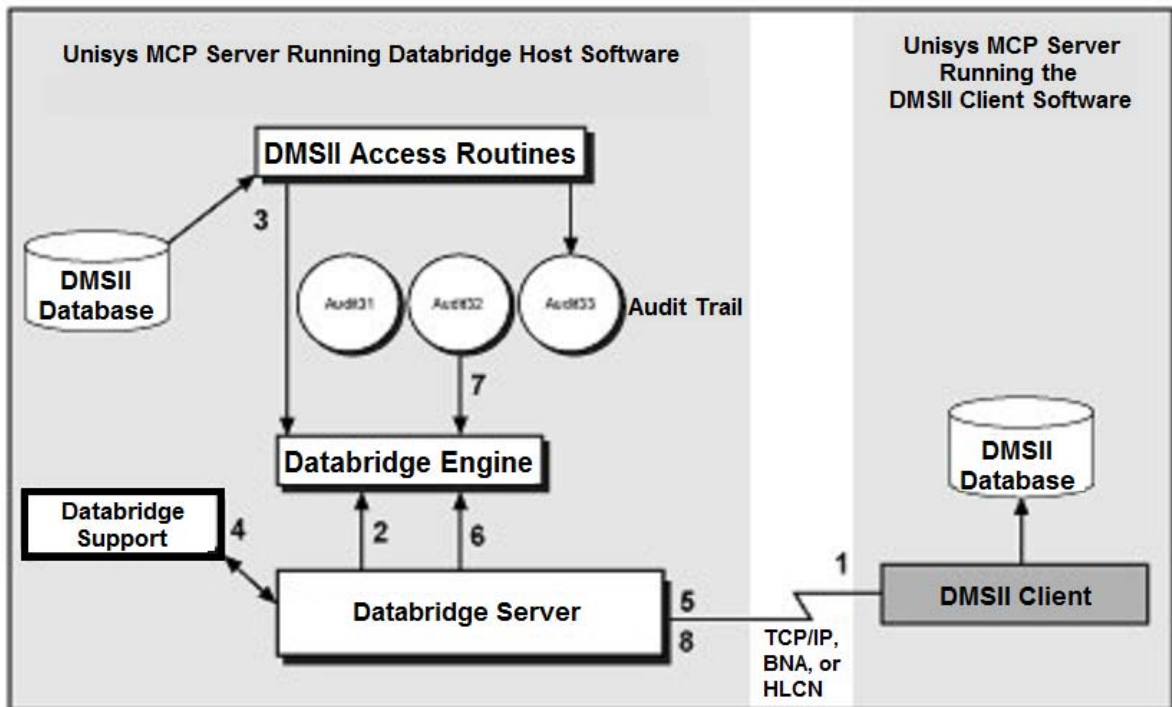
- ♦ “Introducing the DMSII Client” on page 7
- ♦ “Understanding the Databridge Components” on page 8
- ♦ “DMSII Client Support for DMSII Structure Types” on page 10
- ♦ “Understanding How the DMSII Client Works” on page 10

## Introducing the DMSII Client

The DMSII Client is a mainframe program that works with the Databridge host software to replicate (clone and then track) a DMSII database as another DMSII database. Updates to the client database occur as DMSII audit becomes available for the primary database.

DMSII Client runs on all Unisys MCP-hosted mainframes.

The following diagram shows how the DMSII Client works:



1. The DMSII Client calls Databridge Server to clone a specified database.
2. Databridge Server calls the Databridge Engine to clone the specified DMSII database.

3. The Engine accesses the specified DMSII database via the DMSII Accessroutines by calling routines in the Databridge DMSII Support library.
4. Databridge Server receives extracted records from the Engine and calls the filter in Databridge Support.
5. Databridge Server sends records that satisfy the filter to the DMSII Client, which stores them in the client database.
6. Databridge Server calls the Engine to retrieve updates that occurred during and after the extraction.
7. The Engine reads the audit trail and sends the information to the Databridge Server, which calls the filter in Databridge Support.
8. Databridge Server sends the changed information from the audit file to the DMSII Client.

## Understanding the Databridge Components

Each component of the previous diagram is explained in this section.

### DMSII Accessroutines

The Accessroutines program is a DMSII utility that controls access to the database, reads and writes records, and creates the audit trail.

### Audit Trail

The audit trail contains all of the audit files generated for a database. The audit trail consists of audit files named as follows:

*databasename* / AUDIT*nnnn*

where *databasename* is the name of the DMSII database, AUDIT is a literal, and *nnnn* is the audit file number (AFN), a number between 1 and 9999. For example, if you have a database named BANKDB, an audit file would be named similar to the following:

BANKDB/AUDIT7714

### DMSII Database

The DMSII database is the collection of files containing the data sets and sets. Records are stored in the data sets, and sets provide rapid, ordered, access to those data sets. One of the primary reasons to replicate a DMSII database is to provide data to end users with minimum impact on host resources. Therefore, the data you replicate should be the data that end users will request most often. Most sites do not clone entire DMSII databases.



## Databridge Engine

The Databridge Engine (DBEngine) is the main part of the Databridge replication software, which is explained in the *Databridge Host Administrator's Guide*. DBEngine is a host library program that retrieves structural information, layout information, and data from the DMSII database and passes the information to the Databridge Server.

## Support Library

The Databridge Support Library provides filtering and translation to the Databridge Server (as well as other Databridge Accessories). Once the Databridge Server receives data from DBEngine, it calls the Support Library to determine if the data should be replicated, and if so, the Databridge Server passes the data to the Support Library for formatting. In brief, the Support Library offers the following to the DMSII Client:

- ◆ Virtual data sets. Appear as normal data sets to the Databridge Server and the DMSII Client, even though they do not actually exist in the DMSII database.
- ◆ Data set filtering
- ◆ Column filtering
- ◆ Row filtering

For a complete explanation of data set, column, and row filtering, see the *Databridge Host Administrator's Guide*.

## Databridge Server

The Databridge Server (also referred to as DBServer) provides communications between the DMSII Client and DBEngine on the host. The Databridge Server calls all DBEngine functions for the DMSII Client. In general, Databridge Server responsibilities include the following:

- ◆ Communicating with the DMSII Client. If you configure the auto connect feature in both DBServer and the DMSII Client, Databridge Server initiates communication with the DMSII Client when audit files become available. If you do not use the auto connect feature, Databridge Server waits until it is called by the DMSII Client.
- ◆ Sending replicated DMSII data from DBEngine to the DMSII Client.
- ◆ Sending DMSII database layout information from DBEngine to the DMSII Client.

---

**NOTE:** Frequently, you can enhance performance by having the DMSII Client communicate with Databridge Enterprise Server instead of the Databridge Server. To configure this, in the DMSII Client parameter file, enter the IP address of Enterprise Server instead of the Databridge Server.

---

# DMSII Client Support for DMSII Structure Types

The DMSII Client supports data sets, remaps, and logical databases. Data sets in the client database will have additional data items that do not appear in the primary database. These data items make it easier to locate records during tracking.

The DMSII Client makes embedded data sets disjoint in the client database. Consequently, application programs designed for the primary database may need to be modified in order to compile and run properly against the client database.

## Data Sets

The DMSII Client supports all DMSII data set structure types except for the following.

- ◆ Data sets embedded within an ORDERED or COMPACT data set
- ◆ Embedded data sets if INDEPENDENTTRANS is reset
- ◆ Partitioned structures

---

**NOTE:** For information on DMSII structure types supported by Databridge Clients, see the *Databridge Client Administrator's Guide*.

---

## Data Items

The DMSII Client supports all types of DMSII data items except for the following:

- ◆ POPULATION
- ◆ COUNT
- ◆ FILLER
- ◆ AGGREGATE
- ◆ LINKS

# Understanding How the DMSII Client Works

Following is a simplified description of how the DMSII Client works. This description starts after all of the necessary Databridge database replication software and DMSII Client components are correctly installed and configured.

1. Run the DMSII Client WFL.

The DMSII Client contacts the Databridge Server for required database information to create the DASDL source code that it then compiles as the client database description. After this, the DMSII Client sends a request to the Databridge Server to clone the database.

2. The Databridge Server calls DBEngine to extract data from the specified database.

The Databridge Server then sends the database records to the DMSII Client.

3. The DMSII Client populates the client database and then either waits for additional audit information or terminates depending on the settings in the parameter file.
  - ◆ If the DMSII Client waits for additional audit information, it periodically contacts the Databridge Server to see if new updates are available.
  - ◆ If the DMSII Client terminates and it is registered for auto connect requests, the Databridge Server will initiate the DMSII Client when additional audit information becomes available.
4. As audit becomes available, the Databridge Server uses DBEngine to retrieve the after-images of modified records, which it then sends to the DMSII Client.
5. The DMSII Client updates the client database with the new updates and then either waits for additional audit information or terminates.



# 2 Configuring and Starting the DMSII Client

This chapter explains how to configure and start the DMSII Client on a Unisys MCP-hosted mainframe. For complete installation instructions, see “Installing the DMSII Client” in the *Databridge Installation Guide*.

For the DMSII Client to work properly, the mainframe must meet the system requirements for Databridge. (See the *Databridge Installation Guide* for a complete list of system requirements.)

- ◆ [“Configuring and Starting the DMSII Client” on page 13](#)
- ◆ [“Modifying the DMSII Client DASDL File” on page 15](#)
- ◆ [“Modifying the DMSII Client Parameter File” on page 18](#)

## Configuring and Starting the DMSII Client

Complete the following steps to configure and start the DMSII Client to clone the DMSII database.

---

**NOTE:** If you set COMPILE MANUAL (typically used when the DASDL and DMALGOL compilers are not available to the DMSII Client software) in the DMSII Client parameter file, there are several manual steps you must complete as documented in steps 6 through 13 below.

---

1. Get the DMSII Client DASDL patch file using CANDE, as follows:

```
GET PATCH/DATABRIDGE/DMSIICLIENT/SAMPLE/DASDL AS PATCH/  
DATABRIDGE/DMSIICLIENT/clientdbname/DASDL
```

where *clientdbname* is the name of the client database you are creating. We strongly recommend that the name of the client database be different from the name of the primary database so that there is no confusion as to which database you are using.

2. Modify and then save the DMSII Client DASDL patch file to reflect settings for your site. See [“Modifying the DMSII Client Parameter File” on page 18](#).
3. Get the DMSII Client parameter file using CANDE, as follows:

```
GET DATA/DMSIICLIENT/SAMPLE/CONTROL AS DATA/DMSIICLIENT/  
clientdbname/CONTROL
```

where *clientdbname* is the name of the client database you are creating.

4. Modify and then save the DMSII Client parameter file to reflect settings for your site. See [“Modifying the DMSII Client Parameter File” on page 18](#).
5. Start the DMSII Client by entering the following command:

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname")
```

where *clientdbname* is the name of the client database you are creating.

The DMSII Client will generate and compile the DASDL for the client database. The message that appears next depends on your setting for the COMPILE parameter in the DMSII Client parameter (step 2).

<b>If</b>	<b>Then</b>
COMPILE AUTO	<p>The DMSII Client displays the following message and then waits for the DMSupport library to compile:</p> <pre>Waiting for DMSupport compilation</pre> <p>After DMSupport is compiled, the client database will be initialized with empty structures and then the DMSII Client will clone the primary database.</p> <p>At this point, you are finished with this procedure. Go to “What to Do Next ” below.</p>
COMPILE MANUAL	<p>The DMSII Client displays the following message and then terminates:</p> <pre>Use WFL/DATABRIDGE/DMSIICLIENT/COMPILEDB to compile <i>clientdbname</i></pre> <p>Continue with step 6.</p>

6. Compile the client database by starting the following

```
START WFL/DATABRIDGE/DMSIICLIENT/COMPILEDB ("clientdbname")
```

where *clientdbname* is the name of the client database you are compiling.

7. When the COMPILEDB WFL is finished, if the compiled software is on a different pack or usercode than the DMSII Client, copy the following files to the DMSII Client usercode and pack:

```
DESCRIPTION/clientdbname  
DMSUPPORT/clientdbname  
RECONSTRUCT/clientdbname  
OBJECT/DATABRIDGE/DMSIICLIENT/LIB/clientdbname
```

8. Create empty database structures by running the following utility:

```
RUN SYSTEM/DMUTILITY ("DB= clientdbname INITIALIZE =")
```

9. Run the DMSII Client to clone the database by entering the following command:

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname")
```

## What to Do Next

The DMSII Client either waits for additional audit information or terminates, depending on the value of the MAXWAIT option in the parameter file. At this point, the client database is ready to use.

If you want to view the status of the DMSII Client, enter the AX STATUS command as described in [DMSII Client AX Commands \(page 27\)](#).

You can register the DMSII Client for auto connect so that Databridge Server initiates it when updates are available. To do this, enter the REGISTER command as described [Starting and Stopping the DMSII Client \(page 25\)](#).

## Modifying the DMSII Client DASDL File

The DMSII Client DASDL file (PATCH/DATABRIDGE/DMSIIClient/*clientdbname*/DASDL) is a skeleton source file for the client database. It provides information on where and how the client database will be created.

The DMSII Client generates a new DASDL source file by combining PATCH/DATABRIDGE/DMSIIClient/*clientdbname*/DASDL with the database layout information it receives from DBServer. The merged DASDL source is named SOURCE/DATABRIDGE/DMSIIClient/DASDL/*clientdbname*.

A sample DMSII Client DASDL patch file follows the procedure below.

---

**NOTE:** If you are completing step 2 in [“Configuring and Starting the DMSII Client” on page 13](#),” use the following information to configure your DMSII Client DASDL file. After you have configured the DMSII Client DASDL file, continue with step 3 above.

---

Modify the DASDL file settings by completing the following steps:

1. Set the FAMILYNAME attribute in the AUDIT TRAIL section.
2. For any primary-side sets you want declared on the client side, declare the following somewhere near the top in the DASDL file:

```
$ SET setname1 setname2 setname3 ...
```

---

**NOTE:** Remove any hyphens from the set name. For example, if you want to include a set called BANK-ID-SET, use \$ SET BANKIDSET.

---

3. For any primary-side subsets you want declared on the client side, declare one of the following somewhere near the top in the DASDL file:

```
$ SET ALLSUBSETS
```

—or—

```
$ SET subsetname1 subsetname2 subsetname3 ...
```

4. Change other parameters and audit trail attributes indicated in the comments of the DASDL file, if desired. If you decide to change any of these parameters and audit trail attributes, refer to your Unisys *DMSII Data and Structure Definition Language (DASDL) Programming Reference* manual for detailed information about the parameters and audit file attributes.

## Sample DMSII Client DASDL Patch File

Following is the source code from the sample DMSII Client DASDL patch file (PATCH/DATABRIDGE/DMSIIClient/*clientdbname*/DASDL) that you must modify by replacing the items in angle brackets with your entry.

```

DEFAULTS
(
  %- SECURITYGUARD    = <guardfiletitle>,
  CHECKSUM           = TRUE,
  %- DUMPSTAMP       = TRUE,
  DATASET
  (
    %- SECURITYGUARD    = <guardfiletitle>,
    BUFFERS           = 0 + 0 PER RANDOM USER
                      OR 2 PER SERIAL USER,
    REBLOCK           = TRUE,
    %- EXTENDED        = TRUE,
    POPULATIONINCR    = 100,
    POPULATIONWARN    = 0,
    LOCK TO MODIFY DETAILS
  )
  SET
  (
    %- SECURITYGUARD    = <guardfiletitle>,
    CHECKSUM           = TRUE
  )
);

%=====
%           Don't change any OPTIONS
%=====

OPTIONS
(
  $ SET OMIT = ^ JUMBODATABASE
    XL, % lots of structures
  $ POP OMIT % ^ JUMBODATABASE

  AUDIT,
  INDEPENDENTTRANS,
  REAPPLYCOMPLETED,

  ADDRESSCHECK           % ADDRESSCHECK must be set
);

%=====
%           Change PARAMETERS as desired for better performance.
%=====

PARAMETERS
(
  SYNCPOINT           = 100 TRANSACTIONS,
  CONTROLPOINT        = 20 SYNCPOINTS,
  ALLOWEDCORE         = 500000,
  OVERLAYGOAL         = 5
);

%=====

```



```

%          Change AUDIT TRAIL as desired for better performance.
%=====

AUDIT TRAIL
(

%=====
%          Set FAMILYNAME to the pack for the audit files.

%- FAMILYNAME          = <auditpack> COPY TO TAPE AND REMOVE,

%=====

AREAS                   = 10,
AREASIZE                = 1000 BLOCKS,
BLOCKSIZE               = 4080 WORDS,
UPDATE EOF              = 100 BLOCKS,
CHECKSUM                = TRUE
);

%=====
% Audit location dataset for keeping track of our position
% in the primary database.
%
%          *DO NOT MODIFY*
%=====

```

```

AUDITLOCATION DATASET
(
  AL-STRNUM
  $ SET OMIT = JUMBODATABASE
                NUMBER (4) INITIALVALUE 0;
  $ POP OMIT % JUMBODATABASE
  $ SET OMIT = ^ JUMBODATABASE
                NUMBER (5) INITIALVALUE 0;
  $ POP OMIT % ^ JUMBODATABASE
  AL-RECTYPE      NUMBER (3) INITIALVALUE 0;
  AL-MODE         NUMBER (1) INITIALVALUE 0;
  AL-AFN         NUMBER (4) INITIALVALUE 0;
  AL-ABSN        REAL (S11) INITIALVALUE 0;
  AL-SEG         REAL (S11) INITIALVALUE 0;
  AL-INX         REAL (S11) INITIALVALUE 0;
  AL-TIME        REAL          INITIALVALUE 0;
  AL-HOSTINFO    REAL          INITIALVALUE 0;
),

BLOCKSIZE = 30 RECORDS;

```

```

ALSET SET OF AUDITLOCATION
KEY
(
  AL-STRNUM,
  AL-RECTYPE
)

```

```

NO DUPLICATES;

%=====
%       Select predefined sets
%=====

% Put $ SET <setname> here for any desired sets from the
% primary database.

%$ SET CUSTNAMESET

% The following line is the "insertion point" where the
% generated structures will be inserted.

@ Insertion point -- Do NOT delete this line.

%=====
%       Logical databases and physical attributes
%=====

% Put any logical databases or physical
% attributes of specific structures here.

%=====
%                               End DMSII Client DASDL
%=====

```

## Modifying the DMSII Client Parameter File

The DMSII Client parameter file (DATA/DMSIICLIENT/*clientdbname*/CONTROL) is a SEQDATA file. It provides information on where and how to locate the Databridge Server SOURCE for the primary database.

---

**NOTE:** If you are completing step 4 in [“Configuring and Starting the DMSII Client” on page 13](#), use the following information to configure your DMSII Client parameter file. After you have configured the DMSII Client parameter file, continue with step 5.

---

The following is the sample DMSII Client parameter file that you must modify by replacing the items in angle brackets with your entry.

```

%-----
%
%
% Module: DATA/DMSIICLIENT/SAMPLE/CONTROL
%
% Project: Databridge
%
% Description: Databridge DMSII Client Sample Parameter File
%
% (c) Copyright 2019 Micro Focus or one of its affiliates.
%
%-----

% Name of the *** Client *** database.
% The client database files will be created on the
% <dbpack> family under the <usercode> usercode.

CLIENT DATABASE "(<usercode>)<clientdb> ON <dbpack>"

% Client database DMSUPPORT title.
% Default: (<usercode>)DMSUPPORT/<clientdb> ON <dbpack>

% DMSUPPORT "<DMSupporttitle>" % (Optional)

% Client database Reconstruct title.
% Default: (<usercode>)RECONSTRUCT/<clientdb>
% ON <dbpack>

% RECONSTRUCT "<Reconstructtitle>" % (Optional)

% DMSII software titles

DMCONTROL "SYSTEM/DMCONTROL ON DISK"
DMUTILITY "SYSTEM/DMUTILITY ON DISK"
DMALGOL "SYSTEM/DMALGOL ON DISK"
DASDL "SYSTEM/DASDL ON DISK"
BUILDREORG "SYSTEM/BUILDREORG ON DISK"
REORGSYM "DATABASE/REORGSYMBOLIC ON DISK"
PROPERTIES "DATABASE/PROPERTIES ON DISK"

% How to locate DBServer SOURCE ...

SOURCE <sourcename> % SOURCE name in DBServer parameter file
AT <host> % DBServer's hostname or IP address
VIA TCPIP % network protocol: TCPIP, HLCN, BNA
PORT <portnumber> % DBServer's port number, e.g. 11367

% example: SOURCE BANKDB AT PRODHOST VIA TCPIP PORT 11367

% When waiting for an audit file ...

RETRY 60 % seconds delay between retries
MAXWAIT FOREVER % max total seconds to wait
% (FOREVER means to retry indefinitely;
% DONTWAIT means do not retry at all)

```

```

        COMPILER AUTO           % automatically compile client database

% COMPILER MANUAL             % user is responsible for compiling
                             % the client database using
                             % WFL/DATABRIDGE/DMSIICLIENT/COMPILEDDB

        % Disable the following datasets ...

% DISABLE dataset1, dataset2, ...
% DISABLE ALL EXCEPT dataset1, dataset2, ...

% INSERT CLIENT TIMESTAMP % add timestamp data item to each
                             % record indicating last time updated
                             % by DMSII Client

% INSERT SERVER TIMESTAMP % add timestamp data item to each
                             % record indicating last time updated
                             % in the primary database

```

## DMSII Client Parameters

Note the following for the format of the DMSII Client parameter file:

- ◆ You can list the options in the parameter file in any order.
- ◆ You can list multiple options on a single line.
- ◆ You can split options across multiple lines.
- ◆ There is no termination character.
- ◆ There is no continuation character.
- ◆ The comment character is the percent sign (%). The comment character can appear anywhere on a line and anything after the comment character is ignored.
- ◆ If you name any entry the same as a parameter file keyword, enclose the name in “quotation marks.” For example, if you create a filter named SOURCE (which is also the name of a keyword in the DMSII Client parameter file), enclose SOURCE in “quotation marks” as follows:

```
FILTER "SOURCE"
```

The remainder of this section explains each option in the DMSII Client parameter file.

### CLIENT DATABASE “*title*”

**Required.** Enter the name you want to use for the client database. It is strongly recommend that the name of the client database be different from the name of the primary database so that there is no confusion as to which database you are using.

If you specify the optional usercode and pack family, they apply to the DMSII CONTROL file and the database files, as in this example:

```
CLIENT DATABASE "(DBRIDGE)BANKCLIENTDB ON DBPACK"
```

In this example:

- ◆ The generated client database files (CONTROL file, data sets, and sets) will have the DBRIDGE usercode and reside on DBPACK.
- ◆ By default, DMSUPPORT will have the DBRIDGE usercode and reside on DBPACK.
- ◆ The DESCRIPTION file will be on the database family under the usercode the DMSII Client is running under.

## DMSUPPORT “title”

**Optional.** If you want the DMSupport library for the client database to have a specific name, enter that name in this parameter. The default support library title is as follows:

```
(dbusercode)DMSUPPORT/clientdbname ON dbpack
```

## RECONSTRUCT “title”

**Optional.** If you want the Reconstruct program for the client database to have a specific name, enter that name in this parameter. The default program name is as follows:

```
(dbusercode)RECONSTRUCT/clientdbname ON dbpack
```

## DMSII Software Titles

**Optional.** The DMSII Software Titles parameter allows you to specify a different title for a variety of DMSII-related programs. This allows you to use nonstandard names for these programs, in addition to specifying usercodes and pack names, as in the following examples:

```
DMCONTROL      "(500)SYSTEM/DMCONTROL ON SYSPACK"  
DMUTILITY      "SYSTEM/500/DMUTILITY ON SYSPACK"  
DMALGOL        "*SYSTEM/DMALGOL ON SYSPACK"  
DASDL          "SYSTEM/DASDL ON SYSPACK"  
BUILDREORG     "SYSTEM/BUILDREORG ON SYSPACK"  
REORGSYM       "DATABASE/REORGSYMBOLIC ON DBPACK"  
PROPERTIES     "DATABASE/PROPERTIES ON DBPACK"
```

## SOURCE

**Required.** The SOURCE parameter is a four-part parameter that enables the DMSII Client to link up with DBServer. The syntax of the SOURCE parameter is as follows:

```
SOURCE sourcename AT host VIA protocol PORT portnumber
```

<b>Where</b>	<b>Is</b>
<i>sourcename</i>	The name of the SOURCE in the DBServer parameter file associated with the database you want to replicate.
<i>host</i>	The host name or the IP address where DBServer or DBEnterprise is running.  Note: The host name can contain multiple nodes separated by periods.
<i>protocol</i>	TCPIP
<i>portnumber</i>	The DBServer port number using TCP/IP.

## **RETRY *seconds***

**Optional.** The default is 60 seconds. Enter the number of seconds you want the DMSII Client to wait before it retries its available audit check. If more audit is unavailable, the DMSII Client will wait the number of seconds you specify before checking again.

## **MAXWAIT *seconds***

**Optional.** The default is FOREVER. Enter the maximum number of seconds you want the DMSII Client to wait for more audit to become available. Since the DMSII Client is designed to run continuously, you can use the default of FOREVER to indicate there is no limit to the waiting time. You can also enter DONTWAIT to indicate that the DMSII Client should immediately terminate when no updates are available.

When you enter a value, the DMSII Client will check for more audit every *nn* seconds, where *nn* is the value of the RETRY option. If the MAXWAIT time expires before more audit becomes available, the DMSII Client terminates. Once the DMSII Client terminates, you must restart it manually, unless you enabled the auto connect feature.

See REGISTER in [Starting and Stopping the DMSII Client \(page 25\)](#) for instructions on registering the DMSII Client for auto connect.

## **COMPILE { AUTO | MANUAL }**

**Optional.** The default is COMPILE AUTO. This parameter determines whether or not the DMSII Client will automatically compile the client database and support library when the client database is created or in the event of a reorganization. If the DASDL and DMALGOL compilers are not available to the DMSII Client software, use the COMPILE MANUAL option.

When you run the DMSII Client with the COMPILE MANUAL option set, the DMSII Client will generate the client database DASDL source and the patch files for the DMSII Client library program, and then terminate with the following message:

`"Use WFL/DATABRIDGE/DMSIICLIENT/COMPILEDDB to compile clientdbname"`

For instructions, see ["When COMPILE is set to MANUAL" on page 29.](#)

## INSERT CLIENT TIMESTAMP

**Optional.** This parameter adds a timestamp data item to each data set record that contains the time the DMSII Client last updated the record.

The default is no client timestamp.

## INSERT SERVER TIMESTAMP

**Optional.** This parameter adds a timestamp data item to each data set record that contains the time the application on the primary system last updated the record.

The default is no server timestamp.

## DISABLE

**Optional.** This parameter allows you to exclude specific data sets from being replicated to the client database. Use this parameter when you want to exclude a portion of a database from being replicated.

To exclude a portion of a database, enter the data sets that you want to exclude in this field, separated by commas, as in the following example:

```
DISABLE CUSTOMER, ACCOUNT, EMPLOYEES
```

When the client database is generated, any disabled data sets are omitted from the DASDL. If a data set is disabled in the parameter file after the initial client database is generated it remains in the client database but none of its updates are replicated.

## DISABLE ALL EXCEPT

**Optional.** This parameter allows you to exclude all data sets except the ones listed from being replicated to the client database. Use this parameter when you want to include only a portion of a database.

To replicate a portion of a database, enter the data sets that you want to replicate in this field, separated by commas, as in the following example:

```
DISABLE ALL EXCEPT CUSTOMER, ACCOUNT, EMPLOYEES
```

When the client database is generated, any disabled data sets are omitted from the DASDL.





# 3 Command Reference

This chapter explains the DMSII Client commands.

## In this Chapter

- ◆ “Starting and Stopping the DMSII Client” on page 25
- ◆ “DMSII Client AX Commands” on page 27

## Starting and Stopping the DMSII Client

This section describes how to start and stop the DMSII Client. For normal processing, use the `START WFL/DATABRIDGE/DMSIIClient ("clientdbname")` command. Depending on your processing needs, you might use one of the following optional second parameters. The DMSII Client defaults to the `UPDATE` command if you do not add a second parameter to the `START WFL`.

### Starting the DMSII Client Using an Optional Second Parameter

By adding an optional second parameter to the `START WFL/DATABRIDGE/DMSIIClient` command, you can change how the DMSII Client runs. Following are the optional parameters that you can use:

- ◆ `UPDATE [datasetlist]`
- ◆ `CLONE [datasetlist]`
- ◆ `STATUS [datasetlist]`
- ◆ `REGISTER port [password]`
- ◆ `CLEAN`
- ◆ `LOCATION afn absn seg inx`

## UPDATE

```
START WFL/DATABRIDGE/DMSIIClient ("clientdbname", "UPDATE [datasetlist]")
```

Use this command to process data set record updates. If you specify a list of data sets, only those data sets will be updated.

## CLONE

```
START WFL/DATABRIDGE/DMSIIClient ("clientdbname", "CLONE [datasetlist]")
```

Use this command to reclone data sets. If you specify a list of data sets, only those data sets will be recloned.

## STATUS

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname", "STATUS [datasetlist]")
```

Use this command to generate a report showing the data set structure number, name, mode (which can be normal, clone, fixup, purge, or reorg), and audit location. If you specify a list of data sets, only those data sets will appear in the report. The following is an example of status report:

```
                Databridge DMSII Client
                =====
Version 6.6.0.003 compiled Wednesday, April 19, 2016 @ 06:37:45
Started Tuesday, April 18,2017 @ 10:51:52

DMSIIClient Status Report for (DB66)BANKCLIENT ON DB as of Tuesday, January
26, 2016 @ 10:51:54
```

Str#	Name	Mode	AFN	ABSN	Segment	Index	Host info
	(common)	normal	9	1509	16155	10	B3BD2517E570
13	BANK	normal	9	1509	16155	10	B3BD2517E570
16	BRANCH	normal	9	1509	16155	10	B3BD2517E570
30	TELLER	normal	9	1509	16155	10	B3BD2517E570

```
                Databridge Client Information
                -----
System:          CS460:1122 MARS SSR 58.150.0453, microcode 11.609
Task:           40405/40406 50 (DB66) (DB66)OBJECT/DATABRIDGE/DMSIICLIENT
ON USER
Processor time: .34645 seconds
I/O time:       0.45 seconds
ReadyQ time:    0.05 seconds
Audit location: AFN=9, ABSN=1509, updates=0, errors=0
```

## REGISTER

```
START WFL/DATABRIDGE/DMSIICLIENT
("clientdbname", "REGISTER port [password]")
```

Use this command to register the DMSII Client to listen to the specified port for auto connect requests, which indicates that additional audit information becomes available. If additional audit file information is available, Databridge Server initiates the DMSIIClient, which automatically requests updates from the primary database and then updates the client database.

The port number must match the AUTO CONNECT port number specified in the DBServer parameter file (refer to Databridge Server in the *Databridge Host Administrator's Guide*).

The password must be the password you use to sign on to the usercode containing the DMSII Client software. If you do not supply a password, you must enter the following command from MARC or the ODT to modify the registration with the proper password so that the DMSII Client will run under the proper usercode:

```
NA REG MOD PROV DBCLIENTportPROV TASK ATTRIBUTES (VALUE=port;
TASKSTRING="clientdbname";USERCODE=usercode/password)
```

The REGISTER command creates a file called `DATA/DMSIICLIENT/clientdbname/AUTOCONNECT`, which contains the NA commands to register it. If you perform the REGISTER command under a privileged usercode, the DMSII Client automatically loads the file and registers it. If you are running under a nonprivileged usercode, enter the following from either the ODT or a privileged usercode in MARC:

```
NA LOAD (usercode)DATA/DMSIICLIENT/clientdbname/AUTOCONNECT ON familyname
```

where *usercode* is the nonprivileged usercode used to run the DMSII Client REGISTER command and *familyname* is the pack where you ran the DMSII Client REGISTER command.

## CLEAN

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname", "CLEAN")
```

Use this command to remove the client database and regenerate it from scratch.

## LOCATION

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname", "LOCATION afn absn seg inx")
```

Use this command to insert a specific audit location in the DMSII Client audit location data set. The next time the DMSII Client runs, it will start retrieving updates from this point.

This command is intended for sites that load a clone of the primary database using a method other than the DMSII Client.

---

**CAUTION:** The DMSII Client will not check the validity of the indicated audit location.

---

## Stopping the DMSII Client

In most cases, you do not need to stop the DMSII Client. However, if you do want to stop the DMSII Client, enter the following:

```
mixnumber AX QUIT
```

where *mixnumber* is the mix number of the DMSII Client. The DMSII Client will terminate at the next commit.

## DMSII Client AX Commands

You can enter the following AX commands anytime the DMSII Client is running.

### AX STATUS

To display the current AFN, ABSN, number of successful updates, and errors, enter the following:

```
mixnumber AX STATUS
```

where *mixnumber* is the mix number of the DMSII Client.

The following is an example of what appears when you enter an AX STATUS command:

```
DMSIIClient: AFN=38, ABSN=17877, updates=301, errors=0.
```

## AX QUIT

To terminate the DMSII Client at the end of the next transaction group, i.e. at the next commit, use the AX QUIT command. The following is an example of what appears when you enter an AX QUIT command:

```
DMSIIClient: Terminating at end of transaction group.  
DMSIIClient: AFN=38, ABSN=17877, updates=0, errors=0.  
DMSIIClient: [0007]: Active audit file 39 is unavailable.
```

To restart the DMSII Client see [“Starting and Stopping the DMSII Client” on page 25.](#)

## AX HELP

To display the AX commands that you can use with the DMSII Client, enter the following command:

```
mixnumber AX HELP
```

where *mixnumber* is the mix number of the DMSII Client.

The following is an example of what appears when you enter an AX HELP command:

```
DMSIIClient: Valid commands: STATUS, QUIT.
```

# 4 Database Update and Reorganization

This chapter contains information on what to do if there is a DASDL update or reorganization of the primary database. There are two ways to compile the client DASDL and the DMSII Client Library, depending on how you configured the COMPILE parameter.

## In this Chapter

- ◆ “When COMPILE is set to AUTO” on page 29
- ◆ “When COMPILE is set to MANUAL” on page 29

## When COMPILE is set to AUTO

If you set the COMPILE option to AUTO and a reorganization of the primary database occurs, the DMSII Client automatically generates the client database DASDL source and patch files for the DMSII Client support library, initiates several jobs to update the client database, and then displays the following:

```
Waiting for DMSupport compilation
```

After the DMSUPPORT library is compiled, the DMSII Client will compile and run a Reorganization program. The DMSII Client then resumes normal processing.

---

**NOTE:** The DASDL source generated by the DMSII Client is now named **SOURCE/DATABRIDGE/DMSIICLIENT/DASDL/databasename** instead of **SYMBOL/DATABRIDGE/DMSIICLIENT/DASDL/databasename**, as it was in older releases.

---

## When COMPILE is set to MANUAL

If you set the COMPILE option to MANUAL and a reorganization of the primary database occurs, the DMSII Client automatically does the following:

- ◆ Generates the client database DASDL source and patch files for the DMSII Client support library.
- ◆ Terminates with the following message:

```
"Use WFL/DATABRIDGE/DMSIICLIENT/COMPILEDB to compile clientdbname"
```

At this point, you must do the following:

1. Have the system administrator who has access to the DASDL and DMALGOL compilers start the following WFL to compile the client database and the DMSII Client support library:

```
START WFL/DATABRIDGE/DMSIICLIENT/COMPILEDB ("clientdbname")
```

2. After WFL/DATABRIDGE/DMSIICLIENT/COMPILEDB is finished, if the compiled software is on a different pack or usercode than the DMSII Client, copy the following files to the DMSII Client usercode and pack:

DESCRIPTION/*clientdbname*

DMSUPPORT/*clientdbname*

RECONSTRUCT/*clientdbname*

OBJECT/DATABRIDGE/DMSIICLIENT/LIB/*clientdbname*

3. Start the DMSII Client by entering the following command:

```
START WFL/DATABRIDGE/DMSIICLIENT ("clientdbname")
```

where *clientdbname* is the name of your client database. The DMSII Client updates the client database and then resumes normal processing.

# 5 Incorporating Existing Databases

This chapter contains information on how to use the DMSII Client to add its data sets to an existing database. Normally, when you run the DMSII Client the first time, it creates the client database. However, if you already have a database and you want to merge the information in that database with replicated information from the primary database, you can modify the DMSII Client to do so.

The DMSII Client will not access or modify any of the pre-existing data sets. Those data sets will be updated by the application software, independently of the DMSII Client. (The DMSII Client will update any data sets that it adds to the database.)

## In this Chapter

- ♦ [“Incorporating an Existing Database” on page 31](#)

## Incorporating an Existing Database

To have the DMSII Client add its data sets to an existing database, complete the following steps:

1. Use CANDE or another editor to copy the DASDL source file of the existing database, as follows

```
GET existing_database_source AS PATCH/DATABRIDGE/DMSIICLIENT/  
clientdbname/DASDL
```

2. Insert the following line after the OPTIONS parameter in the DASDL file you just created:

```
$ SET INITIALIZENEW AUTOREORG
```

3. Declare the following structures somewhere in the DASDL file:

```
AUDITLOCATION DATASET  
(  
  AL-STRNUM NUMBER (4) INITIALVALUE 0;  
  AL-RECTYPE NUMBER (3) INITIALVALUE 0;  
  AL-MODE NUMBER (1) INITIALVALUE 0;  
  AL-AFN NUMBER (4) INITIALVALUE 0;  
  AL-ABSN REAL (S11) INITIALVALUE 0;  
  AL-SEG REAL (S11) INITIALVALUE 0;  
  AL-INX REAL (S11) INITIALVALUE 0;  
  AL-TIME REAL INITIALVALUE 0;  
  AL-HOSTINFO REAL INITIALVALUE 0;  
) ,  
BLOCKSIZE = 30 RECORDS;  
ALSET SET OF AUDITLOCATION  
KEY  
(  
  AL-STRNUM,  
  AL-RECTYPE  
)  
NO DUPLICATES;
```

4. For any primary-side sets you want declared on the client side, declare the following somewhere above the insertion point (see step 6) in the DASDL file:

```
$ SET setname1 setname2 setname3 . . .
```

---

**NOTE:** Remove any hyphens from the set name. For example, if you want to include a set called BANK-ID-SET, use \$ SET BANKIDSET.

---

5. For any primary-side subsets you want declared on the client side, declare one of the following somewhere above the insertion point (see step 6) in the DASDL file:

```
$ SET ALLSUBSETS
```

—or—

```
$ SET subsetname1 subsetname2 subsetname3...
```

6. Type the following line where you want the declarations for the primary-side structures to appear, which must be somewhere after the \$ SET *setname ...*, and \$ SET *subsetname ...* lines, if present.

```
@ Insertion point -- Do NOT delete this line.
```

7. Save the DASDL.

### What to Do Next

Configure and then start the DMSII Client as described in [“Configuring and Starting the DMSII Client” on page 13](#). Make sure that you use the existing database name as the client database name.



# 6 Troubleshooting

This appendix contains troubleshooting information.

## In this Chapter

- ◆ “General Troubleshooting Procedures” on page 33
- ◆ “Troubleshooting Table” on page 33

## General Troubleshooting Procedures

If you have problems running the DMSII Client, complete the following steps:

1. Make sure your system meets the minimum hardware and software requirements necessary to use the product. See the Installation Requirements in the *Databridge Installation Guide's* chapter on Installing the DMSII Client for this information.
2. If your primary and client databases are on separate mainframes, ensure that the mainframes are communicating.
  - ◆ Try the PING function.
3. Is Databridge Server running on the primary system? Check the mix to make sure. If it is not running, refer to Databridge Server in the *Databridge Host Administrator's Guide* for instructions on how to start it.
4. Verify that your entries in the DMSII Client parameter file match your entries in the DBServer parameter file. See “[Modifying the DMSII Client Parameter File](#)” on page 18 for information about setting the DMSII Client parameters. Refer to the *Databridge Host Administrator's Guide* for instructions on configuring the DBServer parameter file.
5. Check the status of the DMSII Client. See “[AX STATUS](#)” on page 27 for instructions.
6. Resolve any errors that you receive. If you are receiving error messages that you don't understand, see the *Databridge Message Guide* for help resolving these messages.
7. If you cannot identify and solve the problem without assistance, contact your product distributor. Call from a location where you have access to the problem mainframe.
8. Contact Micro Focus Customer Care for Databridge:  
<https://support.microfocus.com/contact/databridge.html>

## Troubleshooting Table

The following table lists common problems and their solutions.  
For a complete list of DMSII Client errors, see the *Databridge Message Guide*.

## Problem

The DMSII Client gets a PORT file I/O error and terminates

Reorganization of the original database

The DMSII Client terminates with the following message:

```
mixnumber DBDMSIIClient:
```

```
>> [4002] Unable to connect to  
port nnnn at_hostname:
```

```
The OPEN operation failed because  
an abort was sent by the  
correspondent endpoint <<
```

**NOTE:** When this occurs after the initial clone, you do not have to repeat the initial clone.

## Solution

Start the Databridge Server. Refer to the *Databridge Host Administrator's Guide* for instructions on how to start it.

The Databridge Server must be running for the DMSII Client to contact it. If it's not running, the DMSII Client tries until it eventually gets a PORT file I/O error and terminates.

Additionally, make sure that the communication transport, such as TCP/IP, is running on both systems.

Perform the procedure described in [“Database Update and Reorganization” on page 29](#).

Check the following:

- ◆ Verify that the Databridge Server is running on the primary system. If not, refer to the *Databridge Host Administrator's Guide* for instructions on how to start it.
- ◆ Verify that the parameter settings in the DMSII Client parameter file match the parameter settings in the Databridge Server parameter file and vice versa. See [“Modifying the DMSII Client Parameter File” on page 18](#) for information about setting the DMSII Client parameters. Refer to the *Databridge Host Administrator's Guide* for information about setting the Databridge Server parameters.
- ◆ If you are using separate mainframes, verify that communication between the mainframes is functional.

Try the PING function.

# Glossary of Terms

**ABSN.** Audit block serial number—The audit block serial number is a 10-digit number that identifies an audit block.

**absolute address (AA) value.** AA is a DMSII term that stands for absolute address. An absolute address value is an A Series WORD (48-bits in length).

**accessories.** Databridge accessories access the services in DBEngine and DBSupport. Some of the accessories provided with Databridge are as follows:

- ◆ DBServer, which provides communication and DMSII database replication services to Databridge Clients.
- ◆ DBSpan, which produces a replication of one or more data sets into flat sequential disk files. DBSpan updates the cloned flat files by appending the changes to the end of the flat files (unlike DBSnapshot, which replaces the changed records).
- ◆ DBSnapshot, which produces a one-time replication of one or more data sets into flat sequential disk files or tape.
- ◆ DBInfo, which produces a report of your DMSII database timestamps, update levels, DMSII release levels, etc.
- ◆ DBLister, which produces a report of the layout of the structures in your DMSII database, including structure numbers and key sets.
- ◆ DBAuditTimer, which closes the current audit file when it is older than a specified length of time.

**Accessroutines.** The Accessroutines program is a DMSII library program that controls access to the database, reads and writes records, and creates the audit trail.

**AFN.** The audit file number is a four-digit number that identifies an audit file.

**audit file.** An audit file is created by the DMSII Accessroutines and contains the raw format of changes made to the DMSII database by update programs. Audit file records contain the deletes, creates, and modifies that were made to the various structures. Depending on the frequency of changes made to a database, the information in an audit file can span a few hours or several weeks.

Databridge uses the audit file for the raw data of each database change to exactly replicate the primary database. Databridge records the audit location (AFN, ABSN, SEG, IDX) between runs, so it can restart without losing any records.

**audit trail.** The audit trail contains all of the audit files generated for a database. The Databridge Engine reads the audit files to extract updates. It then passes the updates to the Client to be applied to the relational database. After the updates have been successfully extracted, the Client saves the state information, which includes the location in the audit trail from which the last group of updates for the data set were read.

**Batch Console.** The Batch Console automates routine Client tasks by allowing command files/shell scripts launched by the Client service to interact with the service.

**caching.** A process that filters files before they're requested by the Databridge Client. Caching allows Databridge Enterprise Server to send Client data requests quickly and without placing an additional resource burden on the mainframe.

**client system.** The client is the computer system that will receive DMSII records from the primary database. The client could be a Windows computer, a UNIX computer, or an MCP server. The client can have a relational or a DMSII database.

**cloning.** Cloning is the one-time process of generating a complete snapshot of a data set to another file. Cloning creates a static picture of a dynamic database. Databridge uses the DMSII data sets and the audit trail to ensure that the cloned data represents a synchronized snapshot of the data sets at a quiet point, even though other programs may be updating the database concurrently. Databridge clones only those data sets you specify.

Cloning is one phase of the database replication process. The other phase is tracking (or updating), which is the integration of database changes since the cloning.

**consolidated file.** A file created by Databridge Span that contains all replicated records from various data sets.

**CONTROL file.** The DMSII CONTROL file is the runtime analog of the DESCRIPTION file. The DESCRIPTION file is updated only when you compile a modified DASDL. The CONTROL file controls database interlock. It stores audit control information and verifies that all database data files are compatible by checking the database timestamp, version timestamp, and update level. The CONTROL file is updated each time anyone opens the database for updates. The CONTROL file contains timestamps for each data set (when the data set was defined, when the data set was updated). It contains parameters such as how much memory the Accessroutines can use and titles of software such as the DMSUPPORT library (DMSUPPORT/databasename).

Databridge uses the CONTROL file for the following information:

- ◆ Timestamps
- ◆ INDEPENDENTTRANS option
- ◆ AFN for the current audit file and ABSN for the current audit block
- ◆ Data set pack names
- ◆ Audit file pack name
- ◆ Database user code

**DASDL.** Data and Structure Definition Language (DASDL) is the language that defines DMSII databases. The DASDL must be compiled to create a DESCRIPTION file.

**data set.** A data set is a file structure in DMSII in which records are stored. It is similar to a table in a relational database. You can select the data sets you want to store in your replicated database.

**Databridge Director.** Databridge Director (also referred to as DBDirector) is a Windows Service installed with Enterprise Server that starts Enterprise Server whenever a connection request is received.

When you start your computer, DBDirector starts and reads the ListenPort registry value to determine which TCP/IP port communicates with Databridge Clients.

**Databridge Engine.** The Databridge Engine (also referred to as DBEngine) is a host library program that uses the DMSII Support Library to retrieve data records from the DMSII database for cloning.

**Databridge Server.** Databridge Server (also referred to as DBServer) is a Databridge Host accessory that responds to Databridge Client requests for DMSII data or DMSII layout information and provides communications between the following components:

- ◆ DBEngine and Databridge Enterprise Server
- ◆ DBEngine and the Databridge Client

---

**NOTE:** When Enterprise Server is used with the Databridge Client, Enterprise Server takes over much of the functionality of DBServer and DBEngine.

---

**DBClient.** A Client program that is launched by the service. DBClient handles the processing of DMSII data and updates the same as dbutility, except that it runs as a background run and uses the Client Console to display its output and interact with the operator.

**DBCIntCfgServer.** A program that handles all requests from the Client Console specific to a data source. These requests include updating the client configuration file, providing access to the client control tables, and handling the Client Configurator. Like DBClient, this program is run by the service as a background run.

**DBGenFormat.** A host utility that creates translation, filter, and format routines. The GenFormat utility interprets the GenFormat parameter file to generate ALGOL source code patches, which are included in the tailored Support Library.

**DESCRIPTION file.** The DESCRIPTION file contains the structural characteristics of a database, physically and logically. This file is created from the DASDL source by the DASDL compiler and contains the layout (physical description), timestamp, audit file size, update level, logical database definition, and any static information about the database. It contains information about the database, not the data itself.

There is only one current DESCRIPTION file for each DMSII database. Databridge must have access to the DESCRIPTION file before it can replicate a database. Additionally, Databridge uses the DESCRIPTION file information for consistency checks between the primary database and the secondary or replicated database.

The DESCRIPTION file corresponds to the schema in a relational database.

**direct disk.** A replication method that allows Databridge Enterprise Server to clone and track DMSII data sets without using any significant mainframe resources. Direct disk replication requires a SAN (Storage Area Network) or Logical Disks configured to make MCP disks visible in Windows.

**entry point.** A procedure in a library object.

**extraction.** Extraction is the process of reading through a data set sequentially and writing those records to a file (either a secondary database or flat file).

**file format conversion.** A type of DMSII reorganization affects file size values (for example, AREASIZE, BLOCKSIZE, or TABLESIZE), but it does not change the layout of the records in a DMSII database.

**filler substitution.** A DMSII filler substitution is a technique for avoiding a reorganization. It changes record layouts, but does not move records around.

**fixup records.** Changes that occur to the DMSII database while a clone is taking place.

**flat files.** A flat file is a plain text or mixed text and binary file which usually contains one record per line. Within the record, individual fields may be separated by delimiters, such as commas, or have a fixed length and be separated by padding. An example of a flat file is an address list that contains fields for *Name* and *Address*.

**garbage collection reorganization.** A garbage collection reorganization moves records around, but it doesn't change the layout of the DMSII database. Its primary function is to improve disk and/or I/O efficiency by eliminating the space occupied by deleted records. Optionally, a garbage collection reorganization reorders the remaining records in the same sequence as one of the sets.

**lag time.** The lag time is defined as the elapsed time between the time a record in the DMSII database is updated and the time where this update appears in the relational database. This value accounts for any difference between the clock on the mainframe and that on the client machine.

**Lister Accessory.** A Databridge Host accessory that produces a report of the layout of the structures in your DMSII database, including structure numbers and key sets.

**mutex.** A mutex is an operating system resource that is used to implement a critical section and prevent multiple processes from updating the same variables at the same time.

**null value.** The value defined in the DASDL to be NULL for a data item. If the DASDL does not explicitly specify a NULL value for a data item, the NULL value is all bits turned on.

**primary database.** This is the original DMSII database that resides on the host. Databridge replicates from the primary database to one or more client databases. The client databases can be another DMSII database or one of several relational databases. Compare this to the replicated (or secondary) database.

**quiet point (QPT).** A quiet point is a point in the audit trail when the DMSII database is quiet and no program is in transaction state. This can occur naturally, or it can be forced by a DMSII sync point.

**record format conversion.** A type of DMSII reorganization that occurs when a data set or set (group of keys) is reordered or reformatted. It indicates that changes were made to a data set format, or to data items, such as changing the length of an item, for example, BANK-ID NUMBER (10) to BANK-ID NUMBER (15).

**record serial number (RSN).** Record sequence numbers (RSN) are 48-bit quantities used by the Databridge Engine, in the case of DMSII XE, to uniquely identify a record. RSNs will always be used instead of AA values when available except for data sets having embedded data sets. RSNs are always static; they will not change after a garbage collection reorganization.

**replicated database.** The replicated database is the database that usually resides on the client machine and contains records cloned from the DMSII database. The replicated database is updated periodically with changes made to the primary (original) DMSII database. The periodic update (or tracking process) is explained later in this section. Compare this to the primary database.

**replication.** Replication is the ongoing process of cloning and tracking changes to a DMSII database.

**rollback.** A systematic restoration of the primary or secondary database to a previous state in which the problem or bad data is no longer found.

**secondary database.** The replicated database. The replicated database is the database that usually resides on the client machine and contains records cloned from the DMSII database. The replicated database is updated periodically with changes made to the primary (original) DMSII database. The periodic update (or tracking process) is explained later in this section. Compare this to the primary database.

**semaphores.** Operating system resources that are mainly used to implement thread synchronization and signaling.

**service.** The service (Windows) or daemon (UNIX) that automates most Client operations. It handles operator requests from the Client Console and routes all log and informational messages to the consoles.

**set.** An index into a data set. A set has an entry (key + pointer) for every record in the data set.

**state information.** Data that reflects information about the cloned data, such as the audit location and format level.

**structure.** A data set, set, subset, access, or remap. Each structure has a unique number called the structure number.

**subset.** An index into a data set. A subset does not necessarily have an entry (key + pointer) for every record in the data set. Subsets are used to access selected members of a data set and to represent relationships between data set records. Subsets typically contain fewer entries than normal sets.

An automatic subset is any subset that contains a WHERE clause and is maintained by DMSII.

A manual subset is any subset that is maintained by an application.

**Support Library.** A library that provides translation, formatting, and filtering to the DBServer and other accessories. After DBServer receives data from the Databridge Engine, it calls the Support Library to determine if the data should be replicated, and if so, passes the data to the Support Library for formatting.

**system library.** A library code file registered with the MCP on the host that is associated with a function name. Programs can link to the library code file by specifying the function name.

**table.** A data structure in the client database corresponding to a data set or remap in the host DMSII database.

**tracking.** Tracking is an ongoing process for propagating changes made to records in the DMSII primary database to the replicated database after the initial clone. The Databridge Engine performs extraction as well as tracking.

**undigits.** A NUMBER data item containing bit values from 10 to 15 in one or more digits. The digits in a NUMBER data item should contain values from 0 to 9; however, it is possible for the digits in NUMBER data item to contain values 0 to 15. Because values 10 to 15 are not valid digit values, the digits in NUMBER data items containing values from 10 to 15 are called undigits.

