MICRO FOCUS

Installation Guide

Databridge Installation Guide

Version 6.5

Legal Notices

Patents

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

Trademarks

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.

1 Databridge Installation Guide

Use this guide to install, configure, and run Databridge 6.5. Included are instructions for installing and upgrading all of the Micro Focus Databridge Host, Enterprise Server, and Client products, including FileXtract, Plus, Twin, DMSII Client, and Cluster. Also included are a list of new features, system requirements, and installation results for these products.

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)

What's New In This Release

Databridge 6.5 includes the following changes.

All-Product Features

- This version is qualified to run on DMSII is 59.1 (MCP 18).
- Microsoft SQL Server 2016 support
- Windows cluster support

Databridge Host

- Software license keys (access codes) are stored in a separate file and monitored using the new DBLicenseSupport library. This prevents overwriting the DBEngine control file when installing a new release.
- License keys have an associated limit on the number of hosts that can use the product.
- The new DBLicenseManager program displays licensed products and installs product keys.
- WFL programs are installed with a node of SAMPLE in the title to avoid overwriting customer-modified WFLs. The SAMPLE files are automatically duplicated with the standard WFL name for first-time installations.
- The WFL/DATABRIDGE/INCLUDE/SSRTITLES file allows you to specify non-standard titles and locations for the DMSII software and compilers for each system software release. This avoids the need to customize individual WFLs or to perform manual compiles.
- All database access is through the DBDMSIISupport library, which is automatically compiled using WFL/DATABRIDGE/COMP for each database and change in update level.
- Unaudited databases may be cloned.
- If the new DBEngine parameter file option DYNAMIC NAMES is set the Extract Workers will be named EXTRACT/WORKER*x*/*databasename*/*structurenumber*.
- The new DBEngine parameter file option MANUAL COMPILE prevents automatic recompiles of DBSupport and DBDMSIISupport, which is useful if the DMALGOL compiler is not available.
- In most cases, there are no forced audit switches at the end of a clone.
- DBEngine displays its compile options. For example,

DBEngine: Compile options: STATS DEBUG

- The AX STATUS command displays any STOP conditions.
- DBServer returns an error if its port is already in use. This prevents multiple copies of DBServer from sharing the same port.
- DBServer prevents denial of service attacks from tying up Workers.
- DBSupport uses less memory and allows a large number of filters in large databases.
- DBTwin and DMSII Client support the AUTHORIZED USERCODE option in a DBServer SOURCE, which allows only certain usercodes to access the source.
- The DBTwin status report shows version and parameter settings.
- Databridge version 6.5 will be the final release that supports DBTanker.

Databridge Client

- The Client can operate without using stored procedures, while still using host variables. This mode of operation improves performance and makes the Client less vulnerable to problems with triggers that are sometimes used to capture updates to certain tables. You can enable this mode globally and change it for individual data sets using the Client Configurator.
- The Client Manager Service can operate in a clustered Windows Server environment. The service starts and stops automatically on the corresponding node of the cluster when a node switch occurs. The service remembers the state of the data sources, and it restarts any runs that were aborted as a result of the cluster switch, thus making the switch totally automatic.
- A special mode allows you to handle DMSII reorganizations that cause large tables to be altered. In SQL Server, rename and copy the table using “SELECT INTO”. In Oracle, rename and copy using “Copy Table As Select” (CTAS). These techniques are faster than the normal way the Client handles such situations and do not involve the log, thus avoiding the possibility of running out of disk space. The only problem with this technique is that you need to have enough disk space to hold a second copy of the table while the operation is in progress.
- If you cannot use the above method (i.e. don’t have enough disk space), we have also enhanced the way initial values are applied to newly-added columns. Now the updates are done in batches instead of in one giant step to avoid the log file getting too big. The batch size for these updates is configurable, defaulting to 10,000 updates. We also extended this technique to the cleanup scripts.
- The Client protects against a rogue user running potentially harmful scripts from a directory he or she has access to. We do this by only running scripts that reside inside the Client working directory, which is protected by file security.
- The Windows Clients now allow you to configure up to 4 bulk loader threads which could speed up the data extraction process on a high-end server machines. The default number of bulk loader threads is 1.
- When the base structure of a REMAP is filtered out, the Client sets active=1 for the first REMAP of the base structure, thus allowing the REMAP to be replicated. This is also extended to embedded data sets where the parent data set is filtered out, while its REMAP is not.
- You can now force the Client to stop when the audit file access method of Databridge Enterprise switches away from the designated mode (normally direct disk). In the past, we just issued a WARNING and continued processing audits. This change gives the user the option to make it stop automatically when this happens.
- When upgrading the Client software, you no longer must use the migrate utility, provided your software is 6.1 SP3 or newer. The upgrade can use the same working directory and binary configuration files as the old Clients -- no need to do anything else. If the Client Control Tables are incompatible with the old software, the Client Manager Service automatically runs dbfixup. This makes it extremely simple to do an upgrade. All you have to do is install the new software

and specify the same working directory as the old software in the Options tab of the setup program. The updatepath program now includes an checkbox to start the Client Manager service in the last dialog. You will need enable it to start the service.

- You can configure the Client to strip common prefixes from all item names without writing cumbersome user scripts. Many DMSII databases use the dataset name (or an abbreviated form it) as a common prefix, which is considered arcane in a relational database.
- You can store GUIDs represented as ALPHA(36) items in DMSII as columns of data type UNIQUEIDENTIFIER when using the SQL Server client.
- We enhanced the Client to display the correct exit code in many situations which used to display the somewhat meaningless catch-all exit code of 2099, “Internal Error”. We also added a few new exit codes to the Client.
- We significantly reduced the number of messages written to the screen and the log file during data extraction for Windows Clients, as they served no practical purpose other than creating clutter on the screen and the log file. We now show the “creating temp file” message only the first time we get a record for the table. We no longer show the starting bcp messages. Instead we use one message that indicates that the temporary file size threshold has been reached and we are starting bulk loader operations.
- We enhanced the Client logging for runs launched by the Client Manager Service so that all logging for operations that lock the data source get written to the main log file. This makes the log easier to follow, as you no longer need to look in two places.
- 64-bit counters are now used for statistics. This eliminates any possibility of overflow during full clones.
- MISER database customers no longer have to set the parameter `suppress_dup_warnings` to true to prevent real errors from being suppressed. The client automatically does this for the history virtual datasets that would otherwise generate bogus warnings.
- The Databridge Client Manager Service is no longer vulnerable to short messages, which would cause it to hang. The timeout period, which defaults to 2 seconds is configurable.
- The Databridge Client Manager Service times out rogue connections that send no input, thus preventing a denial of service attack from succeeding.
- The Databridge Client Manager Service times out runs when they do not reconnect within the timeout period. This avoids hanging the console when a Client run encounters a problem that causes it to exit before reconnecting to the service.
- The Client Console for the AIX and HP-UX platforms now uses GTK rather than motif, while the AIX console is available in 64-bit form.
- The console operator may now add command line options for runs initiated from the console.
- The Client Console can customize and enforce stricter password rules. Options include a minimum length for passwords, requiring certain types of characters in a password, and the ability to force users to change their passwords on first use.
- We implemented a Flat File client that works like DBSPAN. This client uses an ODBC database to hold the Client Control Tables.
- The "auto_reclone" parameter has been deprecated, as this feature did not work correctly with online garbage collection reorgs. The two options that are available are to stop the client when a garbage collection is detected, or let the client deselect the affected data sets and reclone them later when all the reorgs are completed. The -y option for the process command makes it easy to reclone these data sets without having to do anything else.
- The "max_wait_secs" parameter now accepts two values, the second of which is optional and defaults to 60. When the second parameter is non-zero, it enables code that breaks up long wait-and-retry loops, implemented by the Engine alone, into smaller wait-and-retry loops in the Engine with the client repeatedly issuing DBWait remote procedure calls until the period of

inactivity exceed the value specified by the first value. For example, you can break up a 1 hour loop into 60 one minute loops, which ensures that the line does not go idle for long periods of time.

- The index suffixes for the Oracle client now can use the environment variable $(INDEX_NAME), which gets replaced by the actual index name when the suffix is applied. See the section on "SQL Statement Suffixes" in Appendix C of the Client Administrator's Guide.
- The client implements row filtering for secondary tables derived from items (or GROUPS) that have OCCURS clauses that are not flattened. This is done by using a text file to specify the filtering criteria for such tables using a SQL like syntax.
- The client implements the flattening of unsigned NUMBER items and ALPHA items with OCCURS clauses to a CHAR of VARCHAR column instead of making each occurrence of the item into a separate column. This process is referred to as flattening the OCCURS to a string. You can have fixed format strings and CSV format strings where the delimiter can be selected.
- The DMSII time handling code now supports NUMBER(12) items that represents time in the format "HHMMSSmmmmmm", where the last 6 digits are the fractional part of the seconds.
- The SQL Server client can now handle table names that are TRANSACT-SQL reserved words by enclosing them in square brackets in all SQL statements it uses. In order to maintain backward compatibility we implemented the configuration parameter "bracket_tabnames" that enables this feature.
- The -k option was added to force the Client to drop tables rather than run cleanup scripts in a multi-source environments when recloning datasets. This is designed to be used after a DMSII reorganization that requires certain data sets to be recloned. Using the -k option for the first such clones makes the process a lot simpler, as manually dropping the tables to get them recreated with the new layouts are no longer required.

Databridge Enterprise

- The COPY command line can now specify the HIDE option to prevent displaying the progress dialog. Copies from the MCP environment to the Windows environment can also specify the OVERWRITE option, which will automatically overwrite an existing Windows file. If HIDE is specified without OVERWRITE, existing files will be unchanged.
- The list of files copied is now consolidated in a single file called "`Files copied.txt`" in the same subdirectory as the file transfer log files for a particular host (either TO or FROM).
- The client user ID is passed to DBServer for verification when using the AUTHORIZED USERCODE option.
- Network transmissions are buffered for improved throughput.
- DBEnterprise prevents denial of service attacks from causing copies of DBEnterprise to hang.
- DBEnterprise uses the COMMIT DURING IDLE DATABASE setting from the DBEngine parameter file.
- DBEnterprise supports Windows clustered systems.
- The Disk Properties dialog shows the base pack serial number.

Security Guidelines

To protect your data, make sure that you implement the security guidelines recommended by Unisys and the manufacturer for your relational database.

For	To decrease security risk
Databridge Host	Install Databridge Host to the Unisys mainframe under its own privileged usercode. This is preferable to installing under the usercode of the DMSII database. If you install and run Databridge under multiple usercodes, you must install a copy of DBEngine under each of those usercodes. Alternatively, you can establish DBEngine as a system library (SL).
Databridge Client	File security helps to protect Client operations by restricting access to the working directory and its files and subdirectories. Access is limited to the administrators and the system account, and to designated users (or on Windows, user groups). In Windows, you can enable file security via the option **Enable File Security** at installation, or by running the program `setfilesecurity.exe` as Administrator after installation. From the **UserID** box and the **DBAdministrator** box, specify users who can run Databridge Client and modify the working directory and its files. User IDs specified in the **DBOperator** box can monitor operations by reading the log files, but cannot run Databridge Client. In UNIX, you restrict file access via the umask entry in the file `globalprofile.ini`. See “Install Databridge Client on UNIX” on page 33.
Databridge Enterprise Server	File security is automatically enabled when you install Databridge Enterprise. File security protects the configuration and log files by restricting privileges on the PC to users who need to modify the configuration. By default, any user in the Administrators group can modify the configuration files and delete the read-only log files. If you create a user group in Windows called DBEAdmins, only the users in that group will be able to alter the configuration files and delete log files. Likewise, if you create a group called DBEUsers, only the users in that group will be able to view the configuration. For more information, see "Implementing File Security" in the Databridge Enterprise Server Help.

System Requirements

To install and run Databridge 6.5, your system must meet the following hardware and software requirements. Specific requirements vary based on your hardware and other software components present.

Databridge Host	Unisys mainframe system with an MCP level SSR 53.1 through 59.1 DMSII or DMSII XL software (including the DMALGOL compiler) DMSII database DESCRIPTION, CONTROL, DMSUPPORT library, and audit files

Databridge Enterprise Server	ClearPath PC with Logical disks or MCP disks (VSS disks in MCP format) -or- Windows PC that meets the minimum requirements of its operating system, which is one of the following: ◆ Windows 7 ◆ Windows Server 2012 R2 (CORE mode must be disabled for installation and configuration) ◆ Windows Server 2012 ◆ Windows Server 2008 R2 ◆ Windows Server 2008 Direct Disk replication (recommended) requires read-only access to MCP disks on a storage area network (SAN) TCP/IP transport To view the product Help, a supported Internet browser (such as, Internet Explorer, Firefox, or Google Chrome) is required. In addition, JavaScript must be enabled in the browser settings to navigate and search Help.
Databridge Client Console	You can install Client Console to the same machine as the Client; however, this can drain resources and impact the performance of your relational database. Instead, we recommend that you install the Client Console to a machine that connects to the Client machine via TCP/IP transport. Requires Java Runtime Environment (JRE). Download and install the latest version from Oracle (http://www.oracle.com). For more information, see “Java JRE Installation” on page 24.
Databridge Client	**NOTE:** Disk space requirements for replicated DMSII data are not included here. For best results, use a RAID disk array and store the client files on a separate disk from the database storage. **NOTE:** Memory requirements do not include the database requirements when running the Client in the server that houses the relational database (consult your database documentation for these). The numbers are for a stand-alone client machine that connects to a remote database server. If you run the console in the same machine as the Client, it will need an additional 1-2 GB of memory depending on how many data sources you have and how long you let it run. All log information is saved in memory.

Client - Windows

Unisys ES7000

-or-

Pentium PC processor 3 GHz or higher (multiple CPU configuration recommended)

2 GB of RAM (4 GB recommended)

100 GB of disk space **in addition to** disk space for the relational database built from DMSII data)

TCP/IP transport

One of the following operating systems:

- Windows Server 2016
- Windows Server 2012 R2 (CORE mode must be disabled for installation and console cannot be run locally)
- Windows Server 2012
- Windows Server 2008 R2
- Windows Server 2008
- Windows 10
- Windows 8.1
- Windows 8
- Windows 7

One of the following databases:

- Microsoft SQL Server 2016
- Microsoft SQL Server 2014
- Microsoft SQL Server 2012
- Microsoft SQL Server 2008 R2 (up to SP1)
- Microsoft SQL Server 2005 (up to SP4)
- Oracle 11g, and 12c

Client - UNIX and Linux	One of the following systems: ◆ Sun Microsystems SPARCstation running Solaris 10 or later, and Oracle 11g Release 2, or 12c ◆ Hewlett Packard Itanium with HP-UX 11i v3 (B11.31) running Oracle 11g Release 2, or 12c ◆ IBM pSeries running AIX 5.3L or later (AIX 6.1 for Oracle 12c), and Oracle 11g Release 2, or 12c ◆ Intel X-86 with Red Hat Enterprise Linux Release 4 (Release 5 for Oracle 12c) or later, SUSE Linux Enterprise Server 11 SP1 or later, or UBUNTU Linux 7.1 or later running Oracle 11g Release 2, or 12c 2 GB of RAM (4 GB recommended) 100 GB of free disk space for installation (in addition to disk space for the relational database built from DMSII data) TCP/IP transport **NOTE:** Oracle 12c clients only exist as 64-bit programs. **NOTE:** If you are using Oracle 11g, we strongly recommend that you upgrade Oracle 11g release 2, as the client is designed for this version of Oracle.

Related Documentation

When using Databridge, you may need to consult the following resources.

Databridge product documentation	On the Databridge installation image, the DOCS folder contains guides for installation, error codes, and administrator's guides for each Databridge product. These documents require Adobe Reader for viewing, which you can download from the Adobe website (http://get.adobe.com/reader/). This documentation, and current technical notes, is also available on the Micro Focus support site (http://support.attachmate.com/manuals/databridge.html). Documentation for Databridge Enterprise Server and Databridge Client Console is also available from the **Help** menu. A modern browser is required for viewing this documentation.
Unisys MCP server documentation	If you are not completely familiar with DMSII configuration, refer to your Unisys documentation.

Description of Files on the Installation Image

This topic lists the directories and files on the installation image, in alphabetical order. Databridge Host installation files (DISKINSTALL and IMAGE) are located on the root. Both 32-bit and 64-bit versions are provided for all Databridge Clients.

This directory	Contains	File description
AIX	Tar files for installing Databridge clients on the AIX platform:	
	DB_AIX64_Ora11g.tar	The 64-bit client programs for Oracle 11g

This directory	Contains	File description
	DB_AIX64_Ora12c.tar	The 64-bit client programs for Oracle 12c
	DB_AIX_Ora11g.tar	The 32-bit client programs for Oracle 11g
Console	Files for installing Databridge Client Console, in the following subdirectories:	
	AIX	dbconfig-aix.gtk.ppc64.tar.gz
		dbconsole-aix.gtk.ppc64.tar.gz
		dbconfig-aix.gtk.ppc.tar.gz
		dbconsole-aix.gtk.ppc.tar.gz
	HP-UX (Itanium)	dbconfig-hpux.gtk.ia64_32.tar.gz
		dbconsole-hpux.gtk.ia64_32.tar.gz
	Linux	dbconfig-linux.gtk.x86.tar.gz
		dbconfig-linux.gtk.x86_64.tar.gz
		dbconsole-linux.gtk.x86.tar.gz
		dbconsole-linux.gtk.x86_64.tar.gz
	SunOS	dbconfig-solaris.gtk.sparc.tar.gz
		dbconsole-solaris.gtk.sparc.tar.gz
Docs	Databridge Help documentation files in PDF format. Help for Enterprise Server and the Client Console are available from the **Help** menu of the application.	
HP-UX (Itanium)	Tar files for installing clients on Hewlett Packard ® Itanium machines running HP-UX 11i v2 or later:	
	DB_HP-UXi64_Ora11g.tar	The 64-bit client programs for Oracle 11g
	DB_HP-UXi64_Ora12c.tar	The 64-bit client programs for Oracle 12c
Linux	Tar files for installing clients on Intel® X86 series machines running various flavors of Linux including RedHat Enterprise (Release 4 or 5), SUSE or UBUNTU:	
	DB_Linux_Ora11g.tar	The 32-bit client programs for Oracle 11g
	DB_Linux64_Ora11g.tar	The 64-bit client programs for Oracle 11g
	DB_Linux64_Ora12c.tar	The 64-bit client programs for Oracle 12c
SunOS	Tar files for installing clients on Oracle SPARCstation running Solaris 10:	
	DB_SunOS_Ora11g.tar	The 32-bit client programs for Oracle 11g
	DB_SunOS64_Ora11g.tar	The 64-bit client programs for Oracle 11g
	DB_SunOS64_Ora12c.tar	The 64-bit client programs for Oracle 12c

This directory	Contains	File description
Windows32	Installation files for 32-bit versions of Enterprise Server, Databridge Client (Windows), and the Client Console.	
Windows64	Installation files for 64-bit versions of Enterprise Server, Databridge Client (Windows), and the Client Console.	

2 Installing Databridge Host

This chapter includes instructions for installing and upgrading Databridge Host software on the MCP server.

Before You Upgrade Databridge Host

Before you upgrade, please note the following:

- The SAMPLE files are overwritten by the installation. If you have modified these files, make backup copies *before* you run Databridge Installation WFL. If you changed SYMBOL/DATABRIDGE/SUPPORT, note the changes you made and make the corresponding changes to the new SYMBOL/DATABRIDGE/SUPPORT.
- You can use Accessory parameter files from earlier versions with this release. If the current version includes new options you want to use, you can add them by typing them into your existing parameter file.

Install Databridge Host

Before you install, you'll need to determine the usercode and pack that you'll use to install. For best results, we recommend that you choose a privileged usercode and the primary pack of the FAMILY substitution statement. Use the following procedure to install Databridge Host software, which includes FileXtract and DBPlus.

To install Databridge Host

1. **Sign on to the privileged usercode.** If you install to a nonprivileged usercode, all object files will be marked as restricted, and you will have to use the ODT command RESTRICT – FILE *objectcode* on every Databridge program, enter the SL commands for the DBPlus and DBLicenseSupport libraries, and then run the DBLicenseManager program to add your license key. See the instructions in “Nonprivileged Installation.”.
2. **Upload Databridge containers to the host.**

   Upload the DISKINSTALL and IMAGE files using a file transfer tool capable of binary/image transfers to the privileged usercode where Databridge is to be installed.
3. **Unwrap the Databridge installation WFL (WFL/DATABRIDGE/INSTALL).**

   Enter the following CANDE command:

   ```
   WFL UNWRAP *= AS = OUTOF DISKINSTALL
   ```

   If you want to install the Databridge software to a different pack family than primary pack of your FAMILY substitution statement (FAMILY DISK = *primarypack* OTHERWISE *secondarypack*) use the following command instead.

   ```
   START WFL/DATABRIDGE/INSTALL ("DATABRIDGE", "otherpack")
   ```

Most WFLs are copied with a SAMPLE node in the title. They will also be copied without the SAMPLE node using the ADD command so that first-time installations will have WFL files with the standard name but *existing WFL files are not replaced*. Parameter files, like DATA/SERVER/SAMPLE/CONTROL, are also copied without the SAMPLE node if the file doesn't already exist.

4 **Start the Databridge Installation WFL.**

```
START WFL/DATABRIDGE/INSTALL
```

5 **Install the license key.**

After the files are copied, the DBLicenseManager will display a screen with a legal notice followed by a screen showing licensed products.Databridge is installed with a temporary license key for all products. The first time you run DBLicenseManager you will see the expiration date for each product. Enter your 32-digit license key with hyphens separating each group of 4 digits and transmit. The list of licensed products will be refreshed based on the key you entered.

If at any time you need to add an additional license key (for example, you purchase another Databridge product) or just want to view your licensed products, you can run the DBLicenseManager program using one of the following commands from CANDE.

```
START WFL/DATABRIDGE/INSTALL ("KEY")
R DATABRIDGE/LICENSEMANAGER
```

6 **Edit WFL/DATABRIDGE/INCLUDE/SSRTITLES.**

Modify WFL/DATABRIDGE/INCLUDE/SSRTITLES such that the locations and file titles of the various compilers and system software is correct for each SSR (System Software Release) Databridge will use.

7 **For first-time installations, edit the DBServer control file (DATA/SERVER/CONTROL).** Otherwise, skip to step 8.

7a (Optional) Change `TCP/IP Port = 11367` (the default) to the port you want to use.

7b Change `SOURCE <datasourcename>` to the name of the data source that the client will use.

7c Change `Database = (<usercode>)DESCRIPTION/<databasename> ON <packname>` to reflect the usercode/location of the description file for the database to be replicated.

8 **If you are upgrading, do the following:**

8a Bring down the Databridge Server using the `AX QUIT` command.

8b Recompile your tailored support libraries by entering the following command (replacing *usercode*, *databasename*, and *pack* with your values):

```
START WFL/DATABRIDGE/COMP ("SUPPORT", "(usercode)databasename ON
pack")
```

For more information on recompiling, see the *Databridge Host Administrator's Guide*.

9 **Start the Databridge Server:**

```
START WFL/DATABRIDGE/SERVER
```

NOTE

- If your database uses guardfiles, you can include Databridge under database guardfile validation. See "Set Up Guardfiles" on page 17.
- For information about security options, see "Databridge Security" in the *Databridge Host Administrator's Guide*.

Nonprivileged Installation

If you install Databridge using a nonprivileged usercode you will need to perform the following steps after the files have been installed.

- Since each program is marked as RESTRICTEDFILE when installed using a nonprivileged usercode, use the following ODT command on every program.

  ```
  RESTRICT – FILE (usercode)OBJECT/DATABRIDGE/program ON pack
  ```

- Unrestrict the FileXtract programs.

  ```
  RESTRICT – FILE (usercode)OBJECT/FILEBRIDGE/program ON pack
  ```

- Register DBPlus as a system library using the following ODT command.

  ```
  SL DBAUDITSUPPORT = (usercode)OBJECT/DATABRIDGE/AUDITSUPPORT ON pack:
  TRUSTED,LINKCLASS=3
  ```

- Register DBLicenseSupport as a system library using the following ODT command.

  ```
  SL DBLICENSESUPPORT = (usercode)OBJECT/DATABRIDGE/LICENSESUPPORT ON pack:
  ONEONLY
  ```

- Sign on to CANDE and transmit the following command to run the DBLicenseManager program to enter the Databridge license key.

  ```
  START WFL/DATABRIDGE/INSTALL ("KEY")
  ```

After You Upgrade

Make sure to recompile any tailored support libraries. See the *Databridge Host Administrator's Guide*.

The DMSII Support library is compiled automatically after you upgrade. However, you can save time by manually compiling this library before you start replicating. For instructions, see Chapter 3, "DMSII Support," of the *Databridge Host Administrator's Guide*.

Set Up Guardfiles

If your database uses guardfiles, Databridge can be included under database guardfile validation. For validation of guardfiles to work, the relevant guardfile must specify read access (RO) for the Databridge Accessory and/or the usercode under which the Databridge Accessory is running.

In addition, the Databridge (DBEngine) must have read access to the actual database guardfile. You can accomplish this in one of the following ways:

- Run the Databridge Accessory under the usercode that owns the guardfile. For example, if the guardfile is `(PROD)GUARDIAN/FILE`, run the Accessory under the PROD usercode.
- Change the security of the guardfile to `PUBLIC IN`.
- Make sure that DBEngine is a privileged program. DBEngine is marked as a privileged program on the release media.

NOTE: Access to the guardfile itself is enforced by the MCP. Therefore, even if you make the database visible to the Accessories, you must still give DBEngine read access to the guardfile, as explained above.

Databridge Components and Guardfiles

Databridge Accessories and the Databridge Server Workers (but not the Databridge Server itself) must have read access to the database (physical or logical) they are using. No other rights are necessary. Because GenFormat and DBInfo always use the physical (versus a logical) database when searching for a quiet point, they require read access to the physical database.

Databridge DMSII Support

The Databridge DMSII Support library must have OPENINQUIRY, OPENUPDATE, and FIND access to the physical database. OPENUPDATE allows the extract routines to generate a small amount of audit when doing an OFFLINE clone.

Here's an example of a guard file fragment for a physical database. In this example, the fictional site has Databridge software installed under the (DB) usercode on the family named DBPACK and the database DESCRIPTION file is called (PROD)DESCRIPTION/BANKDB ON DMSPACK.

```
PROGRAM (DB)OBJECT/DATABRIDGE/DBINFO ON DBPACK = RO;
PROGRAM (DB)OBJECT/DATABRIDGE/GENFORMAT ON DBPACK = RO;
PROGRAM (DB)OBJECT/DATABRIDGE/DMSIISUPPORT/BANKDB/PROD ON DBPACK = RW,
        DMVERBS = (OPENINQUIRY, OPENUPDATE, FIND);
```

The `DMVERBS` list for the physical database prevents any actual updating because DELETE, STORE, etc., are not allowed.

If you are using DBTwin or the DMSII Client, the DMSII Support library on the secondary system must have updating access as well and in that case the DMVERBS list should be omitted entirely.

Accessories and Databridge Server Workers

The Accessories and Databridge Server Workers (but not the Databridge Server itself) must be listed in the guardfile that protects the database they will be using. However, if they use a logical database instead of a physical database, put these entries in the guardfile that protects the logical database.

Here's an example of guardfile fragments for a physical or logical database. (In this example, Databridge is installed under the (DB) usercode on the family named `DBPACK` and Databridge Clients run on hosts called IPADDRESS1 and IPADDRESS2.)

```
PROGRAM (DB)OBJECT/DATABRIDGE/SPAN ON DBPACK = RO;
PROGRAM (DB)OBJECT/DATABRIDGE/SNAPSHOT ON DBPACK = RO,
PROGRAM (DB)OBJECT/DATABRIDGE/LISTER ON DBPACK = RO;
PROGRAM (DB)OBJECT/DATABRIDGE/AUDITTIMER ON DBPACK = RO;
PROGRAM (DB)DBSERVER/WORKER/1 ON IPADDRESS1 = RO;
PROGRAM (DB)DBSERVER/WORKER/1 ON IPADDRESS2 = RO;
PROGRAM (DB)DBSERVER/WORKER/2 ON IPADDRESS1 = RO;
PROGRAM (DB)DBSERVER/WORKER/2 ON IPADDRESS2 = RO;
```

If you allow more than two Databridge Server Workers to run concurrently, add entries like the ones above with the appropriate number for the last node.

3 Installing Databridge Plus

How to Install Plus

Databridge Plus runs on all Unisys® MCP-hosted servers, including the CS7xxx series, Libra series, ClearPath® NX, LX, or A Series. It is installed with the Databridge Host product and uses the same license key. For installation instructions, see “Install Databridge Host” on page 15.

4 Installing Databridge on Windows

Use this procedure to install or upgrade the following Databridge products:

- Databridge Client
- Databridge Client Console
- Databridge Enterprise Server
- Databridge Cluster Aware Service for Windows Clients

Before you install

The installer gives you the option to secure Databridge Client "working directory" on page 70. The **Enable File Security** option appears after Databridge Client is successfully installed (step 5 of this procedure). When this option is selected, you can assign access rights to user IDs and groups. File security for Databridge Enterprise Server is automatically enabled during installation. For more information, see "Security Guidelines" on page 8.

If the default location of the Enterprise Server configuration directory in this release is different from what you currently use, you can specify your current configuration directory during the upgrade process (in Setup) or import your configuration files after you upgrade (in Enterprise Server). From Setup, on the **Options** tab, select your current configuration directory for the **Database Folder** box. From the **Enterprise Server** main dialog box, click **Import** to import your sources.

To install Databridge

NOTE: If you're installing on Windows Server 2012 R2 or newer, you must reboot with CORE mode disabled before installing Databridge.

---

1. Log on using a Windows account with administrator privileges.
2. To start the installer, do one of the following:
   - From the Windows64 or Windows32 folder, choose the `Setup.exe` file that matches your operating system.
   - From the download site, click the Windows download link, select a location for the installer files, and then click <Next>.
3. In the Installer wizard, accept the license agreement and click <Continue>.
4. In the tabbed window, make your selections and then click <Install Now>:

User Information tab	Personalize your install, if desired.
File Location tab	Specify a location for the Databridge program files (or accept the default). This path is saved as the INSTALLDIR string value for the Windows registry key.
Feature Selection tab	Click the Databridge component you want to install and choose **Feature will be installed on local hard drive** from the popup menu. (Items with a red "x" won't be installed.) Follow the same procedure if you choose the option to install Japanese Support (page 69) and Localization (page 69).

Options tab

Configure settings for the Databridge features you selected on the **Features Selection** tab. If you're upgrading, click **Browse** and select the existing working directory. Otherwise, use the default or specify a new path.

IMPORTANT: Do *not* specify the Program Files directory as the working directory for the Client or Enterprise Server.

The program listens for incoming connection requests at the TCP/IP port. Use the default port number unless it's already in use by another program.

Advanced tab

Specify the type of installation you want and your log file options.

Install to this PC

Installs the selected Databridge components to your computer.

Create an Administrative Install image on a server

NOTE: An administrative installation does not actually install the product. Instead, it creates an installation image that you can use to deploy Databridge users.

When you create an administrative installation image, an image of Databridge is copied to a network location for later installation to multiple workstations. This network location can be used by deployment tools to access and create packages that are deployed to workstations. Also, end users can perform installations by running `setup.exe` from this location.

Log file settings

By default an installation log file is created, but this file is deleted if the installation succeeds. (This configuration avoids accumulation of large log files after successful installations.) To save a log file for all installations, including successful ones, click **Create a log file for this installation**, and uncheck **Delete log file if install succeeds**.

The installation log file, which provides details about the installation, is saved in the user's temp directory (`%tmp%`) with a generated name that begins with atm. To open this folder, click the **Start** menu, type Run in the Search field, click **Run** to open the Run command window, type `%tmp%`, and hit enter.

5. **Client Configuration Options**. To restrict read/write access to the Client working directory in the **Client Configuration** dialog box. **Enable File Security** and then enter the user ID and groups that will have access to this directory. You can also enable Client File Security after installation by running `setfilesecurity.exe` as an Administrator. For a description of these options, see "Client Configuration Options" on page 27
6. **Updatepath Options**. If you plan to run the Client from the command line, specify whether to add the Client directory to the user path (recommended) or the system path. For a description of these options, see "Updatepath Options" on page 28.
7. If you're upgrading the Client from version 6.1 or earlier, run the Migrate program (page 23) to complete the installation. Otherwise, skip this step.
8. **Start Service**. See "Start Service Option" on page 29.
9. After installation is complete, see the "Getting Started" on page 49 section of this guide.

Run the Migrate Program on Windows

If upgrading from any version of 6.3, 6.2, or 6.1 SP3, you do not need to run migrate or dbfixup when using the service. Use the existing working directory and configuration files and simply start the service.

If you're upgrading from 6.1 (or earlier) to 6.5, use this procedure after you install the Client. (See “Installing Databridge on Windows” on page 21.) The migrate program creates a new Client working directory and files.

To run the Migrate program

1. If you are upgrading from the DBLIB SQL Server client, make sure that you first create an ODBC data source in the Windows Control Panel. Make sure that you create a SYSTEM (not a USER) data source.
2. Open a command prompt and type `migrate`. (If you didn't add the Databridge program directory to the system PATH during installation, type the full filename, including path, for migrate and enclose it in quotation marks.)

   The Migrate program creates a new global working directory for the Client and then a new working directory for each migrated data source. After updating the configuration file parameters and creating the service configuration file, the Migrate program moves user scripts and configuration files from the previous version to these directories. If you selected **Enable File Security** during installation, Databridge Client's global working directory and all files and folders will be re-created and secured.
3. Respond to the prompts:
   - If you answer `yes` and provide the working directory, the Migrate program will find your data sources and migrate them. Skip the remaining steps.
   - If you answer `no` or do not provide the path to the working directory. Complete the remaining steps.
4. Enter the name of each data source and its working directory, using the path with drive letter or a directory and name relative to the current directory.
5. Repeat step 4 for each data source. When you're done, enter an empty line (that is, `<CR>`).

   The Migrate program creates a working directory for each data source. It uses the existing configuration file to create a new binary configuration file in the config subdirectory. It removes deprecated parameters or replaces them with equivalent parameters. It moves user scripts to the scripts subdirectory. The Migrate program then starts the dbfixup program, which upgrades the client control tables and populates the dbscripts directory for each data source.

   If the `read_null_records` parameter is set to True, the Migrate program copies the Null Record files for the data sources to the `config` subdirectories. It creates a new Client service configuration file (`dbcontrol.cfg`), which includes updated data sources and any existing scheduling parameters. The Migrate program runs a dbutility `generate` command for each data sources to populate the `dbscripts` directories.

NOTE: If you need to re-run the Migrate program for any reason, first delete the global working directory created by the Migrate program.

Java JRE Installation

The Databridge Console requires the Java Runtime Environment (JRE) to run. If you don't have a Java Runtime Environment (JRE) installed, you will need to install one before you can use the console.

NOTE: You must have the correct type of JRE, one that matches the architecture of the console you install (32-bit or 64-bit). We no longer provide a JRE for Windows. If you do not have one that matches the architecture of the console (32-bit versus 64-bit), go to Oracle (http://www.oracle.com) to download the appropriate files.

Installing a JRE

You can download and install the JRE from the Oracle website.

1. Install a Java JRE from the Oracle site

   Go to the Java SE Downloads page (http://www.oracle.com/technetwork/java/javase/downloads (http://www.oracle.com/technetwork/java/javase/downloads)). Download and install the JRE or the Server JRE download (you do not need the JDK).

2. After installing the JRE, run the following command to confirm that the JRE version reported is the version you downloaded

   ```
   \<java-path>\bin\java -version
   ```

   For example:

   ```
   >"C:\Program Files\Java\jre8\bin\java" -version
   java version "1.8.0_nn"
   Java(TM) SE Runtime Environment (build 1.8.0_nn-b01)
   ```

The Working Directory

Databridge Client's global working directory (also referred to as the *service's working directory*) is required for the Client to run. The *working directory* is different from the *install directory*, where the installer copies all of its files.

The first time you install Databridge Client on UNIX, you must create a working directory. In subsequent installations, you can reuse this directory. For instructions on creating this directory, see "Install Databridge Client on UNIX" on page 33.

The first time you install Databridge Client on Windows, the installer creates the working directory for you in the location you specify. This path is saved as the WORKINGDIR string value in the Windows registry key.

Contents of the Client Working Directory

The working directory contains four subdirectories: `config`, `locks`, `logs, and scripts`.

This subdirectory	Contains
`config`	The program-generated file (`log.cfg`) that keeps track of the current log file and the binary configuration file (`dbcontrol.cfg`), which the service updates when you make changes to settings in the Client Console. If you export the service configuration file, the exported copy (`dbcontrol.ini`) is created in this subdirectory. When you add a data source, a backup copy of the service configuration file `dbcontrol.cfg` is created (`dbcontrol.bak`).
`locks`	Lock files for each data source. A lock file is created the first time a client run starts. The file is re-opened on subsequent runs and kept open for the duration of the run to prevent additional runs using the same data source. The `-u` option can no longer be used to unlock the data source in command-line runs. The locks subdirectory is created the first time you start the service (new installations) or when you run the Migrate program (upgrades).
`logs`	Log files (`*.log`) created by the service. A section in the configuration file named [Log_File] allows the user to set the criteria for switching log files. For more information, see Appendix C in the *Databridge Client Administrator's Guide*.
`scripts`	Command files launched by the service under circumstances such as, when a client run terminates or the BCNOTIFY program on the MCP sends the service a request to run a script file. This is a suitable location for saving Batch Console source files that contain the commands for the Batch Console.

Each Data Source Has a Working Directory

Each data source requires its own separate working directory within the Client working directory. This directory is created automatically when you add a new data source in the Client Console or when you upgrade existing data sources using the Migrate program. If you use the command-line client (dbutility), you must create directories for your data sources. When dbutility executes a `configure` or a `define` command, it creates the subdirectories (if they don't exist) and a binary configuration file (`dbridge.cfg`) in the `config` subdirectory.

The data source working directory contains the following subdirectories:

This subdirectory	Contains
`config`	The client configuration file (`dbridge.cfg`). For runs initiated by the Client service, this file must be binary. The command-line client can use a binary or a text file. For details, see Appendix C in the *Databridge Client Administrator's Guide*. Make sure that you delete any exported text configuration files (`dbridge.ini`) that are no longer used. These become obsolete when the configuration is updated from the Client Console. The Null record file for the data source and two program-generated files (`log.cfg` and `trace.cfg`), which keep track of the current log and trace files. **NOTE:** You can make the Client add signon information to the configuration files it creates by running the `configure` or a `define` commands with switches to specify your signon parameters. For more information, see Signon Configuration. (page 46)
`dbscripts`	Files created by the generate command, SQL scripts, bulk loader files (`*.ctl`,`*.fmt`), and command files to invoke the bulk loader. In Oracle, the control files also specify parameters for SQL*Loader.

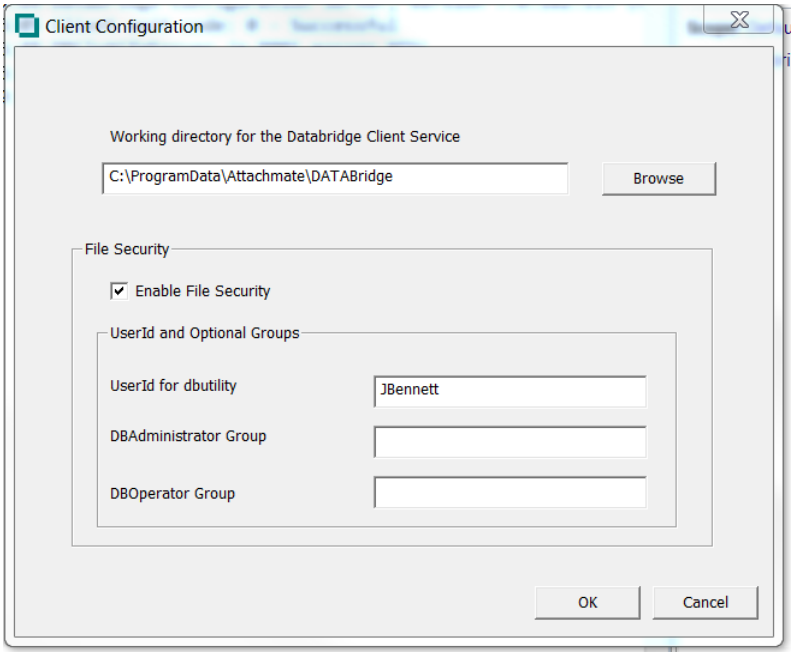
This subdirectory	Contains
`discards`	Files created by the client or the bulk loader when they encounter a data record that cannot be processed. If the Client determines that a DMSII record is bad before handing it to the bulk loader, it puts this record in the client discard file as if it was an insert operation. The data is encoded as an SQL stored procedure call using the appropriate stored procedure based on the type of update. Discard files can be executed using the database's query tool: ◆ SQL Server Management Studio query window ◆ Oracle SQL*Plus utility
`logs`	Client log files. The filename includes a user-configurable prefix followed by a date (*yyyymmdd*). If multiple log files exist for the same date, a time (*_hhmmss*) is appended to the end of the filename.
`scripts`	User script files.

The data source working directory also contains temporary files. Many of these files aren't removed by the Client and should be deleted periodically. You can safely delete them any time the Client is not running (except during the interval between a `redefine` command and a `reorg` command). Temporary files include: trace files; bcp log files; unload files; scripts that are executed by the `redefine` command; and temporary files used by the Windows client during data extraction. You should use the logmaint utility to get rid of old log files. This utility lets you specify the number of log files you want to keep at a given time.

Installation Dialogs

Client Configuration Options

After you click through the initial installation windows, accept the license agreement, and click <Install Now>, Databridge installs. When this is completed, the message “Installation completed successfully” displays. Click <Close>. The **Client Configuration Options** dialog box opens.



Working directory for the Databridge Client Service — Displays the working directory you selected in the Options tab of the installer. You can change it by entering a different path or by clicking **Browse**. This directory is required for all client operations, whether initiated from a command line or by the service.

File Security

Enable File Security — Select this check box to set the security attributes for the Client working directory and its files and folders, according to the settings specified in this dialog box.

UserID and Optional Groups — When **Enable File Security** is selected, you can specify a user or group that has read-write privileges to all directories and files created by the Client program (in addition to administrators and the built-in system account, which have these privileges by default). You can specify both individual users and groups if the user is not a member of the group. You can also specify a group that has read-only privileges to these files.

UserID for dbutility — Specify the user ID that has full read-write control of Databridge Client and its programs. dbutility commands must be run under this user ID or by a member of the group specified in the DBAdministrator Group string in the Windows registry.

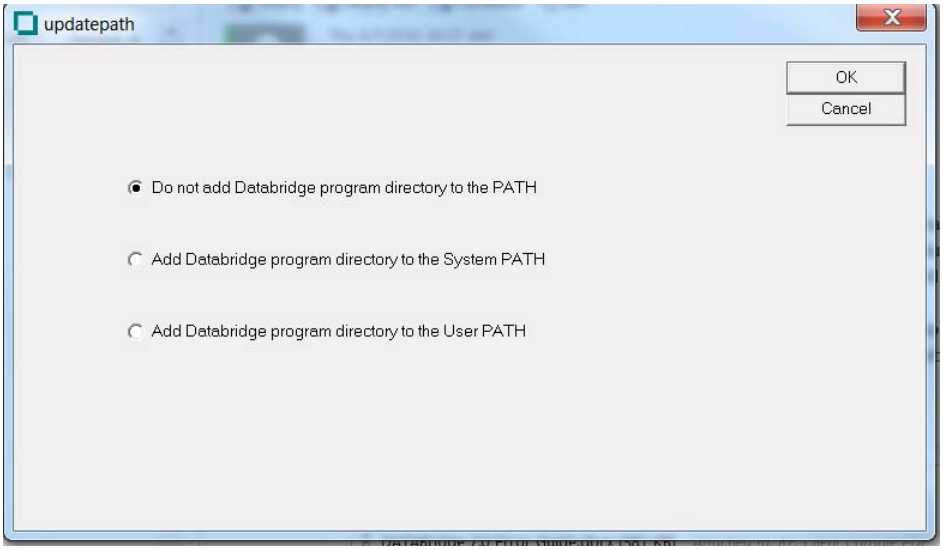
NOTE: If you reinstall Databridge, this value will be populated with the current ID, which you can clear and reenter if needed.

DBAdministrator Group	Specify a group name whose members can run the Databridge Client program and who have read-write access to files in the Client working directory.
DBOperator Group	(Optional) Specify a group name whose members can view the Databridge configuration and log files but cannot modify them. This selection is useful for groups who are not directly involved in running Databridge, but who need to monitor its operations.

Click <OK> to save your selections.

Updatepath Options

After you click <OK> in the **Client Configuration** dialog box, the Updatepath dialog box opens.



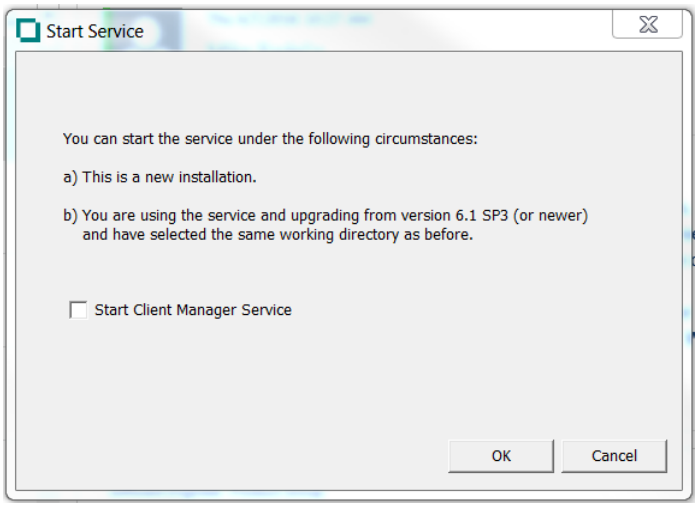
If you run the Client from a command line, use the **Updatepath** dialog box to include both the install directory and the Client directory in the system path.

NOTE: The **Updatepath** dialog box won't appear when reinstalling or repairing the installation.

This setting	Does this
Do not add Databridge program directory to the PATH	Requires that you enter the fully qualified filename (including the Databridge directory) each time you run a Databridge program from the command line. (Make sure that you enclose the path in quotation marks, as "Program Files" contains a space.) Alternatively, you can setup command files to run the Client or you can use environment variables to make this task easier. Choose this setting if you use scripts that specify the full path for the client.
Add Databridge program directory to System PATH	Makes the Databridge programs (and directory) available to anyone who uses the computer by adding the Databridge directories to the Windows Path environment variable. Avoid this option if security is a concern.
Add Databridge program directory to the User PATH	Makes the Databridge programs available only to the user who installed the program.

Start Service Option

When you've made your path selections, you can choose to start the service directly from the installer, if you meet certain criteria.



If your installation fits the criteria shown and you want to start the service, check the box and click <OK>.

Installing and Configuring Databridge Cluster

The following procedure will enable an administrator to configure and run Databridge with the service on a clustered Windows Operating system. A cluster database provides scalability and data-sharing between multiple compute nodes. This sharing is designed to offer a fault tolerant database with high availability to minimize the effects of a single failure loss of service within your storage architecture.

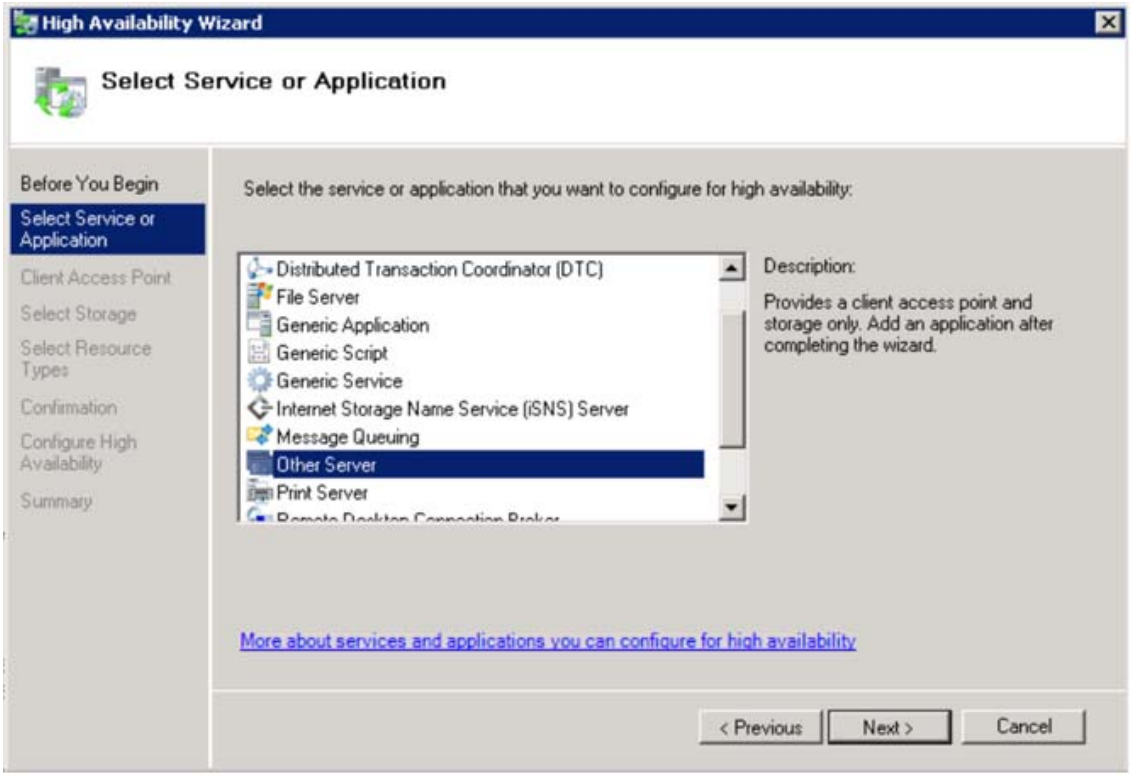
If you are attempting to install the cluster feature from an unsuccessful install of Databridge version 6.5 please read Uninstalling the Cluster section prior to your installation.

Configuring the First Node

The First Node that you will be installing the client on should be your active node. If it is not the active node you will need to move the client to the active node.

1. From the Server Manager, expand the tree view features and expand **Failover Cluster Manager** to list the available clusters. Expanding the cluster to which Databridge will be added, right-click **Services and applications** in the tree view or select **Configure a Service or Application** on the right side taskbar to launch the High Availability Wizard.
2. Click **Next** on the Before You Begin prompt.
3. In the Select Service of Application prompt select **Other Server** from the list and click **next** as seen in *Figure1-1*.

Figure 4-1 *High Availibility Wizard*



4. In the **Client Access Point** prompt enter your desired name for the cluster and configure accordingly with your IPv4 address in the **Address** field and click **next**.
5. When prompted to Select Storage select from the available disks and click **next**. This disk will be shared by Databridge nodes for configuration and logging.
6. In the Select Resources Type prompt do not select an option and click **next**.
7. Continue through the Confirmation, Configure High Availability, and Summary prompts and select **finish**.
8. Install the Databridge Client version on the **first node** by following the steps in the ***Installing Databridge on Each Node*** section.

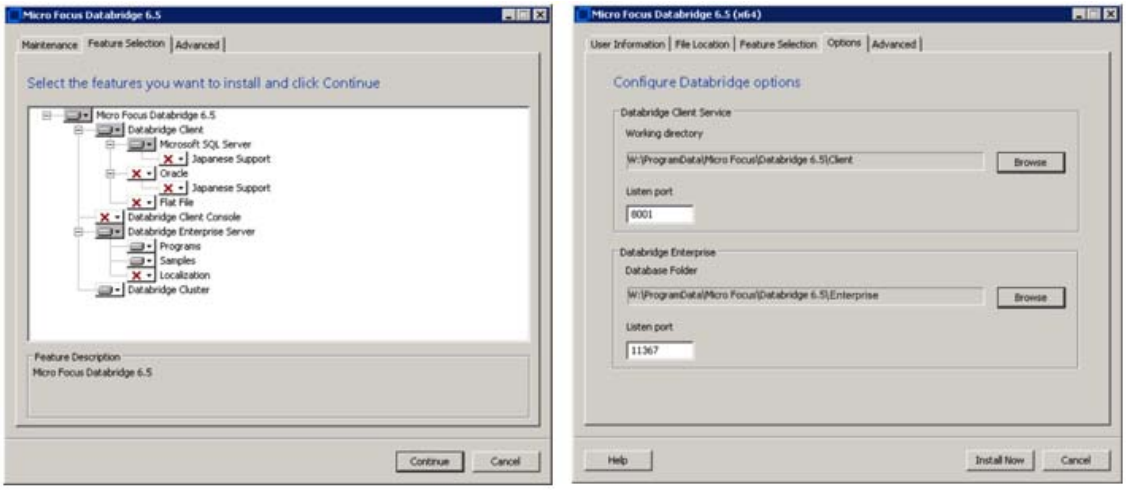
Installing Databridge on Each Node

Install the Databridge Client version 6.5 and follow the steps below. Ensure the Databridge installation is on the active node per the instructions.

1. Launch the Databridge **setup.exe** installer.
2. In the Feature Selection tab select the Databridge Enterprise Server (optional), SQL Server and Databridge Cluster to be included in the install.
3. In the **Options** tab configure your **Client Service Working Directory** and **Database Folder** from the shared drive (see *figure 1-2*). To ensure the Databridge cluster feature is functional both nodes must be configured with the same directories. The Database Folder configuration only applies if the Databridge Enterprise Server is being installed.

Example 4-1 *We have selected* ***("\ProgramData\Micro Focus\Databridge 6.5\Client")*** *for the Client Working Directory and* ***("\ProgramData\Micro Focus\Databridge 6.5\Enterprise")*** *for Databridge Enterprise Working Directory.*

Figure 4-2



4. When configuring your **Client Service Working Directory** and **Database Folder** set the ports (assigned by a network administrator) and select **Install Now**.

NOTE: Do not start the service after this step.

Configuring and Installing the Second Node

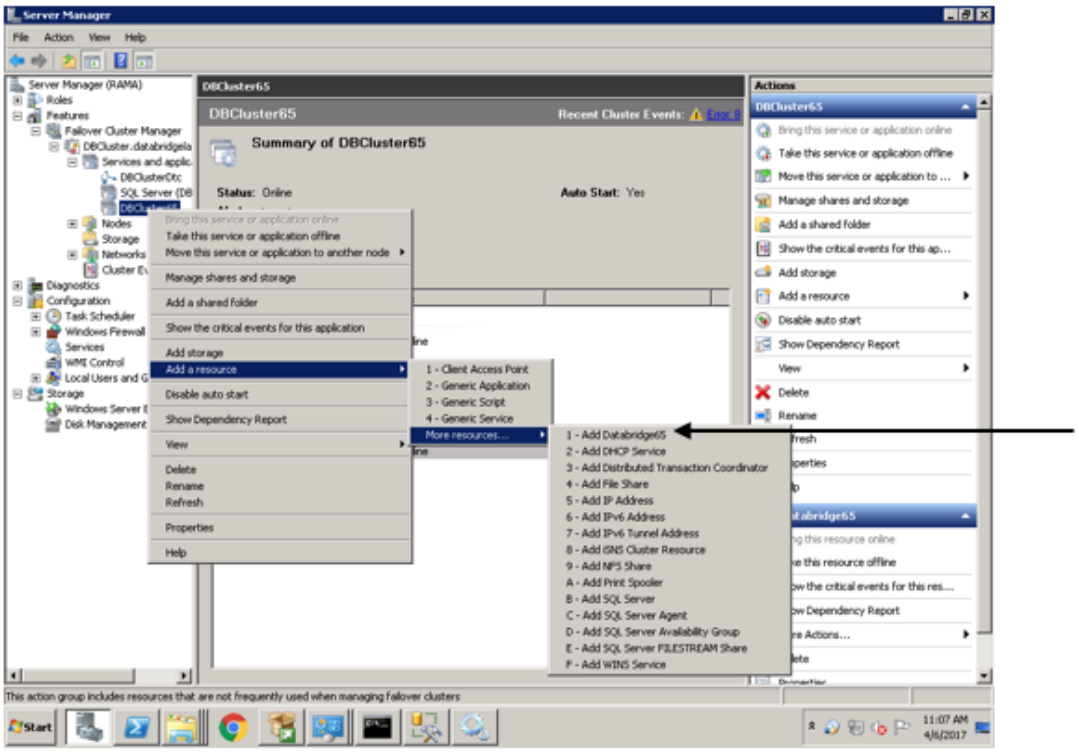
1. In the second node, launch the **Server Manager**.
2. From the Server manager right-click the newly created application under the **Service or applications** tree view from Step 4 on your first node.
3. In the right click menu select **Move this service or application to another node** and select the second node.
4. Repeat the **Installing Databridge on Each Node** steps for the second node. Be sure to configure Databridge with the same Feature Selection and Options as done for your first Node.
5. After installing and configuring the client do not start the service. Launch **Powershell** as an administrator and type the following commands:

```
import-module failoverclusters

add-clusterresourcetype Databridge65 "c:\program files\micro
focus\Databridge\6.5\clusterdll\dbcluster.dll" "Databridge Cluster Resource
Type"
```

6. Return to the Server Manager and click on **dbcluster65** located under Services and applications in the left hand tree view. **Add** dbcluster65 to **Other Resources**.

Figure 4-3



7. Right-click your created resource and select **Properties**.
8. Name the Resource added.
9. Navigate to the **Set Dependencies** tab and ensure the dependency is set on the server name and click insert.
10. Right click your created resource under **Other Resources** and select the **Bring resource on-line** option.

Uninstalling Databridge for Cluster Installation

To remove Databridge with the cluster feature, or, if your installation of Databridge with the cluster feature was unsuccessful follow the installation steps outlined to clean your workstation and re-install Databridge with the cluster feature.

1. Using Server Manager, take DBCluster offline.
2. Using Server Manager, delete DBCluster.
3. Using Powershell, `remove-clusterResourceType "Databridge65"`
4. On each node that is in **DBCluster** use Windows Explorer to navigate to Windows\cluster and delete `dbcluster.exe`.
5. In the Control Panel uninstall Databridge version 6.5.

5 Installing Databridge Client on UNIX

This section provides instructions for installing and upgrading Databridge Client, including the Client Console and daemon, on a UNIX system.

Install Databridge Client on UNIX

Use this procedure for first-time installations and upgrades on all UNIX, Linux, and related systems. In this procedure, you'll install the Client, create a Client working directory, and specify a user ID for the daemon.

If you are running Oracle Client on a non-English database, the system detects this automatically, reads the Oracle NLS parameters, and adjusts the affected client parameters accordingly.

If you are running Oracle Client on a database that uses the UTF-8 character set, the system detects this by automatically reading the Oracle NLS parameters, and adjusts the session parameters to accommodate UTF-8.

IMPORTANT: Commands, filenames, and often, passwords, are case-sensitive on UNIX systems. Type the commands as shown.

To install the Client on UNIX

1. Log on using a userid that has root privileges.
2. If a previous version of Databridge Client is running on your system, stop it.
3. Create a new install directory for Databridge 6.5 (for example, "`/opt/dbridge65`").
4. Move the appropriate tar files (Client, Client Console) to the UNIX system using binary FTP. See "Description of Files on the Installation Image" on page 12.
5. Change the directory to the install directory you created in step 3 and extract the tar file:

   ```
   tar -xvf filename
   ```

   where *filename* is the full tar filename, including location.

   The extract program creates a client subdirectory named after the database (for example, Oracle). Files are extracted to this subdirectory and to the directory you specified.
6. Next, change the owner and group for the files in the install directory to the userid (and the group to which the userid belongs) designated to run Databridge Client (that is, the USERID specified in the file globalprofile.ini). To do this, type the following command:

   ```
   chown -R dbridge:users *
   ```

   where *dbridge* is the user ID and *users* is the corresponding group.

   CAUTION: To prevent the files in the install directory from being accidentally deleted, we recommend that you leave them set to read-only (the default). Before you update the software, to prevent the extraction from failing, either remove all of the files from the install directory or make them writeable until the update is successfully installed.

   ---

7 In the editor, open the sample daemon script file (`dbdaemon`) located on the root of the install directory, update the necessary environment variables, and then save `dbdaemon` to the following location:

- On Linux and Solaris: `/etc/init.d`
- On HP-UX: `sbin/init.d`

8 Create the directory `/etc/Attachmate/DATABridge/6.5`

9 In the editor, open the `globalprofile.smp` file on the root of the install directory and make the following changes:

WARNING: This file is critical for client operations. If this file is missing or contains the wrong information, the client will not run.

9a For `INSTALLDIR`, specify the install directory you created in step 3.

9b For `WORKINGDIR`, specify the full name of the Client's working directory for 6.5. **If you're upgrading from an older version than 6.2, make sure that the working directory for 6.5 is different from the older working directory.** You create the actual directory later in this procedure.

9c For `USERID`, specify the user under which you'll run the daemon. When you start the daemon, you must log in as this user or as the root user. Clients can only be run under this userid.

9d To enable file security for the Client working directory, specify a value for `umask`. The bits specified get removed from the default file security bits. A umask of 027 (the default) indicates that the owner bits are unchanged, but the group's w and x bits are reset and all 3 bits are reset for other users. For a stronger mask, specify a value such as 077.

10 Save the file as `globalprofile.ini` to the directory `/etc/Attachmate/DATABridge/6.5`

11 Do one of the following:

- (New installations) In the home directory, use the editor to update your profile (for example, ".profile" or ".bash_profile") and include the environment variable `ORACLE_HOME`. Consult the Oracle database administrator to determine what this variable should be set to (typically, it's "/opt/oracle..."). You must also include the directory `$ORACLE_HOME/bin` in the PATH to ensure that the bulk loader `sqlldr` can be located. You may want to also include the environment variable INSTALLDIR, which points to the directory created in step 3. You can then add the Client program directory $INSTALLDIR/Oracle to the PATH, making it a lot easier to run the Client from the command line.
- (Upgrades - Optional) Update your profile so that the PATH points to the newly-installed Client.

12 For Databridge Client to find the Oracle shared libraries, add the Oracle lib (or lib32) directory to the appropriate environment variable for your system, as shown in the following table. The ORACLE_HOME environment variable is used instead of typing the complete Oracle lib directory name.

To update	Add the following to your profile
64-bit clients except on AIX 6.1 or newer	`LD_LIBRARY_PATH = $LD_LIBRARY_PATH:$ORACLE_HOME/lib`
64-bit clients on AIX 6.1 or newer	`LIBPATH = LIBPATH:$ORACLE_HOME/lib`
32-bit clients on Solaris or Linux	`LD_LIBRARY_PATH = $LD_LIBRARY_PATH:$ORACLE_HOME/lib32`
32-bit clients on AIX	`LIBPATH = LIBPATH:$ORACLE_HOME/lib32`
32-bit clients on HP-UX	`SHLIB_PATH = SHLIB_PATH:$ORACLE_HOME/lib32`

NOTE: The UNIX shell you use may require that you add a line that exports the environment variable. For instructions, consult your UNIX documentation.

13 Using the value you specified in step 9b, create a Client working directory for 6.5 that contains the three subdirectories `config`, `locks` and `logs`. (For upgrades, the Migrate program creates a new, secure working directory and subdirectories from this value. See “Run the Migrate Program on UNIX” on page 36.)

14 When installation is complete, verify that the Client directory files are correctly installed. For a list of installed files, see “Files Installed with Databridge Client and Client Console” on page 63.

15 If you run the Client from a command line, you must also update the environment variable PATH in your profile to include the directory that contains Databridge Client. If you installed the Client to the directory `/opt/dbridge65`, add “`:/opt/dbridge65/Oracle`” to the PATH. The PATH is defined as:

```
PATH=$PATH: … $ORACLE_HOME/bin:  …
```

where each specified directory is terminated by a colon, except for the last one on the line.

Individual clients use the database names as the directories names (for example, Oracle).

CAUTION: Make sure that the mknod utility’s directory (usually `/usr/sbin`) is included in the PATH. If the shell scripts used to clone data sets cannot find this utility, files will be used in place of named pipes resulting in bulk loader problems, such as broken pipes.

---

16 Do one of the following:

- If you're upgrading, proceed to “Run the Migrate Program on UNIX” on page 36 to complete the installation.
- If you're installing for the first time, install the Client Console to configure the Client parameters. See “Install the Client Console on UNIX” on page 37.
- The `"auto_reclone"` parameter has been deprecated, as this feature did not work correctly with online garbage collection reorgs. The two available options are to either stop the client when a garbage collection is detected, or, let the client **deselect** the affected data sets and **reclone** them later when all the reorgs are completed. The -y option for the process command makes it easy to **reclone** these data sets without having to do anything else.

- •The "max_wait_secs" parameter now accepts two values, the second is optional and defaults to 0. When the second parameter is non-zero, it enables code that breaks up long wait-and-retry loops, implemented by the engine alone, into smaller wait-and-retry loops in the engine. The client repeatedly issues **DBWait** remote procedure calls until the period of inactivity exceeds the value specified by the first value. This allows you to break up a one hour loop into 60 one minute loops, which ensures that the line does not go idle for long periods of time.
- The index suffixes for the Oracle client now can use the environment variable $(INDEX_NAME) which gets replaced by the actual index name when the suffix is applied. See the Databridge Client Administrator's Guide section on SQL suffixes in Appendix C: Client Configuration.
- The client implements row filtering for secondary tables derived from items (or GROUPS) that have OCCURS clauses that are not flattened. This is done by using a text file to specify the filtering criteria for such tables using a SQL like syntax.
- The client implements the flattening of unsigned NUMBER items and ALPHA items with OCCURS clauses to a CHAR of VARCHAR column instead of making each occurrence of the item into a separate column. This process is referred to as flattening the OCCURS to a string. You can have fixed format strings and CSV format strings where the delimiter can be selected.
- The DMSII time handling code now supports NUMBER(12) items that represents time in the format "HHMMSSmmmmmm", where the last 6 digits are the fractional part of the seconds.
- The SQL Server client can now handle table names that are TRANSACT-SQL reserved words by enclosing them in square brackets in all SQL statements it uses. In order to maintain backward compatibility we implemented the configuration parameter "bracket_tabnames" that enables this feature.
- The -k option was added to force the Client to drop tables rather than run cleanup scripts in a multi-source environments when recloning datasets. This is designed to be used after a DMSII reorganization that requires certain data sets to be recloned. Using the -k option for the first such clones makes the process a lot simpler, as manually dropping the tables to get them recreated with the new layouts are no longer required.

Run the Migrate Program on UNIX

Use this procedure to upgrade Databridge Client 6.1 and earlier on UNIX systems. If upgrading from any version of 6.3, 6.2, or 6.1 SP3 and are using the daemon, you do not have to run migrate or dbfixup, you can use the existing working directory and configuration files.

To run the Migrate program

1. Install Databridge Client 6.5. See "Install Databridge Client on UNIX" on page 33.
2. Make sure that the file globalprofile.ini includes values for INSTALLDIR, USERID, WORKINGDIR. If you want to enable file security for the working directory, make sure that the umask value is set. (For instructions, see "Install Databridge Client on UNIX" on page 33). The Client runs under the specified USERID in the globalprofile.ini.
3. Open a command session and type migrate. If the Client directory wasn't added to the PATH during installation, you must type the full path to migrate.

   The Migrate program first creates a new global working directory and then creates a working directory for each migrated data source. Next, it moves the existing user scripts and configuration files to this directory after updating the configuration file parameters. It also creates the daemon configuration file which contains any scheduling information defined for the individual data sources.

4 The program asks if you are upgrading from version 6.0 or 6.1 and use the daemon. If the answer is yes, enter the full filename of the global working directory. Make sure that you enter a different name for the 6.5 working directory. After a new working directory structure is created for the service, data sources from the earlier version will be migrated to subdirectories in this working directory. Skip the remaining steps.

5 When prompted, enter the name of each data source. Next, enter the location of the working directory for this data source either as an absolute path or a name relative to the current directory.

   The Migrate program creates a working directory for each data source. Existing settings are used to create a new binary configuration file in the `config` subdirectory and user scripts are moved to the `scripts` subdirectory. The Migrate program starts the dbfixup program, which upgrades the client control tables and populates the dbscripts directory for each data source. Lastly, the Migrate program applies file security to the working directory, data source directories, and all Client files.

   If the `read_null_records` parameter is set to True, the Migrate program copies the Null Record files for the data sources to the config subdirectories. A new daemon configuration file is created (`dbcontrol.cfg`), which includes updated data sources and any existing scheduling parameters. The Migrate program runs a dbutility `generate` command for each data sources to populate the dbscripts directories.

6 Repeat step 5 for each data source. When you're done, enter an empty line (that is, `<CR>`).

7 To run the Client using the daemon, install the Client Console—preferably to a different machine than the relational database to avoid using valuable system resources.

You can run the command-line program (dbutility) or the daemon after you create a script for the daemon that includes the updated ORACLE_HOME, INSTALLDIR, and WORKING_DIR parameters.

NOTE: If you need to re-run the migrate program at any point, first delete the working directory that was created by the migrate program.

Install the Client Console on UNIX

For optimal performance, install the Client Console to a workstation *connected to—not on*—the relational database, or it will use valuable system resources. You can install and run the Client Console on AIX, Linux, Solaris, and Windows operating systems. For example, you can install and run it on a Windows workstation that is connected to UNIX, or run it on both types of workstations simultaneously. See “System Requirements” on page 9.

NOTE: Use Databridge Client Console 6.5 with Databridge 6.5 software. It is incompatible with earlier Databridge versions.

To install the Client Console

1 Copy the `*.tar.gz` archives from the \Console\<UNIX> directory for the appropriate UNIX platform to the location you want.

   There are two archives; one for the dbconfig and one for dbconsole.

2 Unzip the archives:

```
gzip –d dbcon*.gz
```

3 Extract each of the files from the tarballs by running the following separate commands:

```
tar –xvf dbconsole*.tar
tar –xvf dbconfig*.tar
```

This creates two directories (dbconsole and dbconfig) in the current working directory.

4 Next, provide execute privileges to the dbconfig file and to the dbconsole file:

```
chmod +x dbconsole/dbconsole
chmod +x dbconfig/dbconfig
```

After the Client Console is successfully installed, start the service (see “Start and stop the daemon (UNIX)” on page 50), and then “Start the Client Console (UNIX)” on page 51.

If you don’t have a Java Runtime Environment (JRE) installed, you will need to install one before you can use the console. Note that you must have the correct type of JRE, one that matches the architecture of the console you install (32-bit or 64-bit). If needed, go to Oracle (http://www.oracle.com) to download the appropriate files.

6 Installing Databridge DMSII Client

This chapter explains how to install DMSII Client on a Unisys MCP-hosted mainframe. For DMSII Client to work properly, you must configure it according to the instructions provided in the *DMSII Client Administrator's Guide*

In this Chapter:

Databridge DMSII Client Installation Requirements

DMSII Client software itself takes less than 2 MB of disk space. The size of the client database depends on the size of the data sets and sets replicated from the primary system. The size of the client database audit trail depends on the number of audit files you keep on the system. For a rough estimate of the disk space required multiply the size of the primary database by the fraction of the database that will be replicated. If the whole primary database will be replicated then an equal amount of disk space will be required for the client database.

The system resource requirements are proportional to the amount of data replicated. DMSII Client uses the standard DMSII application program interface to the client database and its processor and memory usage would be similar to other application programs.

The DMALGOL and DASDL compilers must be visible to DMSII Client in order to automatically compile the client database and associated library programs. If these compilers are not available you must specify the COMPILE MANUAL option in the parameter file and compile the software using other means.

Install Databridge DMSII Client

You must install Databridge components on both the primary and secondary systems. Before you install DMSII Client on the secondary system, you must install the current version of Databridge Host software on your primary system. See "Install Databridge Host" on page 15 of this guide for Host installation instructions and system requirements.

Follow this procedure to install DMSII Client. You can install DMSII Client to the same mainframe where you installed the Databridge server software, or you can install it to a different mainframe.

Before you install, you'll need to determine the usercode and pack that you'll use to install. For best results, we recommend that you choose a privileged usercode and the primary pack of the FAMILY substitution statement.

To install Databridge DMSII Client

1. Sign on to the privileged usercode.

   If you install to a nonprivileged usercode, all object files will be marked as restricted and you will have to use the ODT command RESTRICT – FILE *objectcode* on every program.
2. Upload Databridge containers to the host.

Upload the DISKINSTALL and IMAGE files using a file transfer tool capable of binary/image transfers to the privileged usercode where DMSII Client is to be installed.

3. Unwrap the Databridge installation WFL (WFL/DATABRIDGE/INSTALL).

   Enter the following CANDE command:

   ```
   UNWRAP *= AS = OUTOF DISKINSTALL
   ```

4. Start the Databridge Installation WFL.

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

   If you want to install DMSII Client software to a different pack family than primary pack of your FAMILY substitution statement (FAMILY DISK = *primarypack* OTHERWISE *secondarypack*), use the following command instead.

   ```
   START WFL/DATABRIDGE/INSTALL ("DMSIICLIENT", "otherpack")
   ```

The WFLs are copied with a SAMPLE node in the title. They will also be copied without the SAMPLE node using the ADD command so that first-time installations will have WFL files with the standard name but *existing WFL files are not replaced*.

7 Installing Databridge Twin

In this Chapter

Installation Requirements

Databridge components must be installed on both the primary and secondary systems.

You must install the current version of Databridge Host software on your primary system before you install Databridge Twin on the secondary system. Go to “Install Databridge Host” on page 15 of this guide for Host installation instructions and system requirements.

Databridge Twin runs on all Unisys MCP-hosted mainframes. ***If you plan to use Databridge Twin and Databridge Host on separate mainframes***:

- We recommend that both mainframes use the same MCP level.
- Both mainframes must be running the same DMSII level.

Install Databridge Twin to the Secondary System

Use this procedure to install Databridge Twin to the secondary system.

CAUTION: If you run Databridge Twin and Databridge Host software on the same mainframe, the usercode from which you run Databridge Twin must be different than the usercode where Databridge Host resides.

Use this procedure to install Databridge Twin. You can install the Databridge Twin on the same mainframe where you installed the Databridge server software, or you can install it to a different mainframe.

Before you install, you'll need to determine the usercode and pack that you'll use to install. For best results, we recommend that you choose a privileged usercode and the primary pack of the FAMILY substitution statement.

To install Databridge Twin:

1. **Sign on to the privileged usercode on the secondary system.**

   If you install to a nonprivileged usercode, all object files will be marked as restricted and you will have to use the ODT command RESTRICT – FILE *objectcode* on every program.

2. **Upload Databridge containers to the host.**

   Upload the DISKINSTALL and IMAGE files using a file transfer tool capable of binary/image transfers to the privileged usercode where Databridge Twin is to be installed.

3. **Unwrap the Databridge installation WFL (WFL/DATABRIDGE/INSTALL).**

Enter the following CANDE command:

```
UNWRAP *= AS = OUTOF DISKINSTALL
```

4 **Start the Databridge Installation WFL.**

```
START WFL/DATABRIDGE/INSTALL ("TWIN")
```

If you want to install Databridge Twin to a different pack family than primary pack of your FAMILY substitution statement (FAMILY DISK = *primarypack* OTHERWISE *secondarypack*) use the following command instead.

```
START WFL/DATABRIDGE/INSTALL ("TWIN", "otherpack")
```

The WFLs are copied with a SAMPLE node in the title. They will also be copied without the SAMPLE node using the ADD command so that first-time installations will have WFL files with the standard name but *existing WFL files are not replaced*.

5 **Edit WFL/DATABRIDGE/INCLUDE/SSRTITLES.**

Modify WFL/DATABRIDGE/INCLUDE/SSRTITLES such that the locations and file titles of the various compilers and system software is correct for each SSR (System Software Release) Databridge Twin will use.

When installation is complete, go to chapter 3 of the *Databridge Twin Administrator's Guide*, “Configuring Databridge Twin” to configure Twin.

8 Setting Up a Relational Database

This chapter covers instructions for setting up a relational database to receive DMSII data.

Setting Up a SQL Server Database

Use the following procedures to set up a Microsoft SQL Server database for use with Databridge Client for SQL Server.

Databridge Client uses only the ODBC data source parameter to connect to ODBC. The ODBC data source parameter provides the database and server parameters required to run BCP.

In addition, the following notes apply to Microsoft SQL Server:

- The default schema for the Databridge user determines the ownership of tables.
- Databridge Client for SQL Server uses ODBC, which places no restrictions on the user's schema when using SQL Server 2005 or later.
- You can use primary keys with non-clustered indexes without having to edit the scripts generated by the client.
- Databridge Client for SQL Server supports the data type `bigint` and the SQL Server 2008 data types `date`, `datetime2` and `time`. To maintain backward compatibility for these data types, enable the parameters `use_bigint`, `use_date`, `use_datetime2` and `use_time` (respectively).

To set up a SQL Server database

1. If you use Integrated Windows Authentication with Databridge Client on Microsoft SQL Server 2012 and are running the service using the default account, you must add NT AUTHORITY\SYSTEM to the sysadmin Server Role. In the Microsoft SQL Server 2012 Management Studio, use the left pane to navigate to **Security > Server Roles**, right-click **sysadmin**, click **Properties**, then click the **Add** button.
2. Create an ODBC data source, unless one already exists. See the following procedure for instructions.

   IMPORTANT: When creating ODBC data sources for Databridge Client, use the Microsoft SQL Server Native Client driver. The Microsoft SQL Server driver doesn't support columns longer than 32K, multi-threaded updates, and several other important Databridge features.
3. Make sure that you have enough disk space free for the Microsoft SQL Server database.
4. If your DMSII data sets contain case-sensitive ALPHA key items, using the default Microsoft SQL Server installation may result in duplicate keys. To avoid this, **make certain the collating sequence for the client database is set to a case-sensitive or binary collation**.

Create an ODBC Data Source

Use this procedure to set up an ODBC data source. If you have not created a new SQL Server database, and are not using an existing database, you must create a new database as outlined in the previous section. This database will hold your Databridge Client control tables and the replicated DMSII tables.

To create an ODBC Data Source

1. Click **Start > Settings > Control Panel > Administrative Tools**. The **Administrative tools** dialog box opens.
2. Double-click **Data Sources (ODBC)**. The **ODBC Data Source Administrator** dialog box opens.
3. Click the **User DSN** or **System DSN** tab. User DSNs are limited to the user who created them. System DSN data sources are available to all users of the computer and are required if you use the service.
4. Click the **Add** button.
5. In the **Create New Data Source** dialog box, select the driver **SQL Native Client**, and then click **Finish**.

   CAUTION: The driver **SQL Server** doesn't support columns longer than 32K, multi-threaded updates, and several other important Databridge features.
6. In the **Create a New Data Source to SQL Server** wizard, name the data source (for example, *databasename*DS). You will enter this name as the data source parameter value in the Databridge Client Configuration file.
7. In the **Server** field, choose your local computer name, and then click **Next**.
8. To connect to this data source, choose an authentication method, and then click **Next**:

Integrated Windows Authentication	SQL Server uses the credentials of the user currently logged in to validate the login. Most installations use Integrated Windows Authentication. **NOTE:** If you select this option on Microsoft SQL Server 2012 and are running the service using the default account, you must add NT AUTHORITY\SYSTEM to the sysadmin Server Role. In the Microsoft SQL Server 2012 Management Studio, use the left pane to navigate to **Security >** Server Roles, right-click **sysadmin**, select **Properties**, and click the **Add** button.
SQL Server Authentication	Databridge connects to SQL Server using the user ID/password pair specified in the Databridge configuration file.

9. Click **Finish** to create the ODBC data source.

Setting Up an Oracle Database

The following notes apply to Oracle:

- The Oracle clients have a configuration parameter that allows you to specify a rollback segment larger than the default rollback segment. You can use a rollback segment to execute the SQL statement SET TRANSACTION USE ROLLBACK SEGMENT at the start of every transaction.
- The Oracle clients support the CLOB data type for holding DMSII ALPHA items that are too large for the VARCHAR2 data type. The parameters in PL/SQL stored procedures for these clients limit the CLOB data type to 32K (Oracle 11g). For backward compatibility, this feature is enabled via the `use_clob` parameter.
- Starting in Oracle 11g Release 2, Oracle database passwords are case-sensitive.

The Oracle open_cursors Parameter

We recommend that you add the `open_cursors` parameter to the `initSID.ora` file. Depending on the setting of the `aux_stmts` parameter, you may need to increase the value of `open_cursors` to make it slightly higher than the value of `aux_stmts`.

If `open_cursors` is not set high enough, an error occurs, indicating that the maximum number specified was exceeded. Refer to your Oracle documentation for information on setting these values.

Setting Up a User ID

If you're installing Databridge Client software for the first time, you must set up a user ID that allows the client to access the relational database.

Microsoft SQL Server

CAUTION: Do not use the `sa` user ID to run Databridge Client. The `sa` user ID typically owns tables that Databridge Client should not have access to.

If you are the Microsoft SQL Server database administrator (DBA) or if you have a user ID with DBA privileges, set up a user ID for Databridge Client that has the following:

- (Required) The ability to select from any table, including system tables. Databridge reads the data dictionary to determine if a table or procedure exists.
- (Required) The ability to create tables, stored procedures, and indexes for tables that the user ID owns.
- (Optional) Unlimited space usage. This is recommended because of the large amounts of data that can be replicated from a DMSII database.

To include rights to modify the database as needed, use the Databridge owner (db_owner) to set up the user ID. After you set up the user ID, you must configure the user ID, password, server, and database. For instructions, see “Signon Configuration” on page 46.

Oracle

CAUTION: Do not use IDs such as SYS, SYSTEM, or SYSMAN to run Databridge Client. These user IDs typically own tables that Databridge Client should not have access to.

If you are the database administrator (DBA), or if you have a user ID with DBA privileges, set up a user ID for Databridge Client that has the following.

(In Oracle 12c, the DBA role is much more restrictive and is not sufficient to run Databridge Client. In Oracle 12c, the user ID must start with C## or c##. This identifies a common user.)

- (Required) The ability to select from any table including the system table SYS.OBJ$.
- (Required) The ability to create and drop tables, stored procedures, and indexes for tables that the user ID owns.
- (Required) The ability to alter tables.
- (Required) The ability to alter the session.

- (Optional) Unlimited tablespace privileges. This is recommended because of the large amounts of data that can be replicated from a DMSII database.

NOTE: In Oracle 11g you can accomplish all of the above by granting the DBA privilege to the user. In Oracle 12c, the same applies, except you won't have select access to the above mentioned system table. You can grant select access to this system table by executing the following GRANT statement from a privileged account, such as SYSTEM.

```
GRANT SELECT ON SYS.OBJ$ TO userid
```

This allows the userid to read this system table, but does not allow the userid to update them. Some DBAs will be reluctant to grant the DBA privilege to the Databridge user, particularly on older versions of Oracle. If that is the case, you can still achieve what you need by using the appropriate GRANT statements.

For information on creating a user ID, see your Oracle documentation.

Signon Configuration

If you have installed Databridge Client for the first time, you must supply the appropriate logon parameters to your relational database. In addition, you may be required to supply a password to sign on to the Databridge Server on the host.

When you create a new data source from the Client Console, the **Hostname** and **Port** settings (in the **Add Data Source** dialog box) provide these logon parameters. If a KEY is specified for the source in the DATA/SERVER/CONTROL, provide one in the **Password** box. These settings are saved to the binary Client configuration file (`dbridge.cfg`), which automatically encodes passwords.

If you supply logon parameters via dbutility command-line options, you must type them each time you run dbutility.

NOTE: Configuration file settings override environment variables. However, command-line options override both configuration file settings and environment variables.

Signon Parameters

The following table shows signon parameters in the dbutility configuration file (`dbridge.cfg`) and their equivalent command-line options.

Configuration file parameter	Command-line option	Description
user	`-U`	Specifies the user ID for the relational database. The user ID must have read and write access to the designated database.
`password`	`-P`	Specifies the password associated with the user ID for the relational database. Note that Oracle 11g and later uses case-sensitive passwords. Configuration file only: If the password contains non-alphanumeric characters besides the underscore (_), you must enclose it in double quotation marks, as follows: `password = "a$bb%"`

Configuration file parameter	Command-line option	Description
database	-D	For Oracle, this name identifies the Oracle instance or Net8 service being accessed. If the service name contains non-alphanumeric characters, you must enclose them in double quotation marks, as follows: database = "orcl.databridge.com"
datasource	-O	For the Databridge SQL Server client, this parameter is the name that identifies the ODBC data source, as configured within the Windows Control Panel.
use_nt_authen	-W	When the ODBC data source uses Integrated Windows authentication, set this parameter to True. When the ODBC data source uses SQL Server Authentication (using userid and password), set it to False. If you set this parameter incorrectly, the client will work, but bulk loader operations will fail.

Integrated Windows Authentication for Microsoft SQL Server

The Microsoft SQL Server database provides two methods for database user authentication. SQL Server authentication uses a userid/password mechanism to authenticate user connections to SQL Server. Integrated Windows Authentication relies on the Windows operating system to authenticate user connections to SQL Server.

NOTE: Databridge Client requires a user account with administrator privileges when using Integrated Windows Authentication to access SQL Server. For details about use_nt_authen, see [signon] in Appendix C of the *Databridge Client Administrator's Guide*.

---

9 Getting Started

In this section:

Next Steps for Databridge Client

After the installation is complete, you can start using Databridge Client. If you're using Databridge Client for the first time, we recommend that you run it from the Client Console.

If you're upgrading and want to start using the Client Configurator to customize your relational database, see “Switch from User Scripts to the Client Configurator” on page 52.

If you're upgrading and plan to run the Client from a command line, use your existing Client configuration file. See Appendix C in the *Databridge Client Administrator's Guide* for information about configuration parameters.

Start and Stop the Client Manager

Start the Service (Windows)

If you installed the Client Console and checked the “Start Client Manager Service" checkbox, the Client Manager will open on its own. If not, use the following procedure to start the service in Windows.

To Start the Client Manager

1. From the **Start** menu, open your **Control Panel**, click **Administrative Tools**, then double-click **Services**.
2. In the **Services** window, find and double-click **Databridge Client Manager 6.5** in the list.

   Databridge Client Manager 6.5 Properties dialog box opens.
3. **(Optional)** On the **Recovery** tab of the dialog box, specify that the service is restarted after the first or second failures and click <Apply> to save your settings.

   NOTE: Do not set more than two failures; if the service fails more than that, a service failure is important to inform you of the problem.)
4. Click the <Start> button under “Service status”.

If the service doesn't start, the specified port may be in use. Check the service log file in `Documents and Settings\All Users\Application Data\Micro Focus\Databridge\6.5\logs`.

(The log file is saved here in Windows 7 and modern servers: `\ProgramData\Micro Focus\Databridge\6.5\logs`.)

5 Click <OK> to save and close the Properties dialog box.

6 Launch the Client Console and follow the instructions to add the service. See "Start the Client Console (Windows)" on page 51.

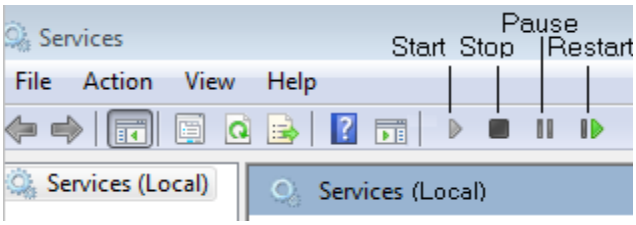
Stop the service (Windows)

Use the following procedure to stop the service in Windows.

To stop the Client Manager

1 Return to the **Services** window. (See step 1 of "Start the Service (Windows)" on page 49)

2 Click the square Stop button on the Services toolbar.



Start and stop the daemon (UNIX)

To start the daemon (UNIX)

1 Log in using the user ID you specified for the daemon in step 9c of "Install Databridge Client on UNIX" on page 33 or as a user with root privileges. If you start the daemon from the root (for example, by typing `su`), the daemon automatically assumes the specified user ID.

2 Make sure that you specified the appropriate environment variables in the dbdaemon script. If you use Oracle, add the Oracle **bin** directory to the PATH. Failing to do this causes the bulk loader scripts to not find the `sqlldr` program. Also, add the Oracle **lib** directory to the LD_LIBRARY_PATH environment variable (64-bit systems) or the program will be unable to find the shared libraries and will not start. For more information, see "Install Databridge Client on UNIX" on page 33.

3 Start the daemon by doing one of the following:

- On Linux and Solaris systems, type:

```
/etc/init.d/dbdaemon start
```

- On HP_UX systems, type:

```
/sbin/init.d
```

The daemon automatically creates the `config` and `logs` subdirectories for the data source working directory, and then places a binary configuration file (`dbcontrol.cfg`) in the `config` subdirectory. This configuration file is updated each time you add a new data source in the Client Console.

4 If you're using the Client Console, see "Start the Client Console (UNIX)" on page 51.

To stop the daemon (UNIX)

You can start or stop the daemon anytime by manually running the script with the appropriate options (start or stop).

Start the Client Console

WARNING: The platform type for the JRE and the client must match (that is they must either both be 32-bit or 64-bit). If they do not match the console will not start and the Eclipse launcher will generate the obscure message **"Java was started but returned exit code = 13"**.

Start the Client Console (Windows)

1. From the **Start** menu, choose **All Programs > Micro Focus Databridge 6.5 > Databridge Client Console 6.5**.
2. From the Client Console, click the **Help** menu, and then click **Help Contents** or **Cheat Sheets** for instructions on getting started.

Start the Client Console (UNIX)

1. From a command line, run the following command

   `cd <`*`install_path`*`>/dbconsole`

   where <*install_path*> is the location to which you installed the Client Console.
2. Next, run the following command:

   `./dbconsole &`
3. From the Client Console, click the **Help** menu, and then click **Help Contents** or **Cheat Sheets** for instructions on getting started.

The 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. It interprets a source file that contains a set of statements written in a language similar to Visual Basic. These statements can initiate a connection, perform rudimentary tests, and issue console requests, to the service. For example, by using the Batch Console in an end-of-run script that runs daily reports, you can restart the Client after the reports are generated.

To use the Batch Console, you must first create a source file for the Batch Console and place it in the `scripts` directory of the service's working directory (also referred to as the Client's global working directory). We recommend that you use a file extension that allows you to easily identify this file as a Batch Console source file (for example, .bcs). You can debug this source file by running the Batch Console from the command line, using the source filename (including directory, such as scripts*source_filename*) as the first argument of the Batch Console command.

The Batch Console always runs as a background run. Its activity is written to a log file in the current directory. The log filename uses the source filename with the extension `.log` added to it. For example, if your source filename is `sourcefile.bcs`, the log file is named `sourcefile.bcs.log`.

For more information, see Chapter 8, "Automating Client Operations with the Service" in the *Databridge Client Administrator's Guide*.

Switch from User Scripts to the Client Configurator

If you currently customize your database via user scripts but wish to start using the Client Configurator, you must first run the dbscriptfixup program for each of your data sources. This program updates the client control tables so that you no longer need the user scripts. The dbscriptfixup program also creates a new set of user scripts that are compatible with the Client Configurator for backup purposes.

NOTE: This task is not time-sensitive and can be completed at any time after installation. We recommend that you get Databridge Client 6.3 running to your satisfaction before you perform this task.

To run dbscriptfixup

1. Create a backup copy of your user scripts.
2. Open a command prompt and change the directory to the working directory of a data source.
3. Run the dbscriptfixup `verify` command (replacing *<data_source>* with the actual name):

   ```
   dbscriptfixup verify <data_source>
   ```

   The dbscriptfixup program performs a read-only review of the user scripts and reports changes you must make to the scripts.
4. Make any changes if indicated by the dbscriptfixup program and then repeat step 2 until you get no errors. You may want to remove or rename problem user scripts and deal with them later.
5. Back up the control tables for the data source to an unload file by typing the following command:

   ```
   dbutility unload <data_source> <data_source>_control_tables.sav
   ```
6. Run the dbscriptfixup upgrade command:

   ```
   dbscriptfixup upgrade <data_source>
   ```
7. Depending on the results of step 6, do one of the following:
   - If the command succeeded (exit code=0), you can now use the Client Configurator. If you want to verify the process, see the following section.
   - If the command failed (exit code=1), reload your control tables by typing the following command and then replace any user scripts you changed in step 4 to restore the configuration to its previous state.

   ```
   dbutility reload <data_source> <data_source>_control_tables.sav
   ```

Verifying the results

You can verify that the dbscriptfixup program successfully updated your control tables by executing a `redefine` command for the data source with the `-R` option. If the command finds no changes in the table layouts, you can safely use the Client Configurator, knowing that all of the changes you made via user scripts are preserved.

Start Enterprise Server

You can start Databridge Enterprise Server using the shortcuts installed on the Windows **Start** menu or from a command prompt session. When you start Enterprise Server from the **Start** menu, Enterprise Server starts the application with Administrator privileges (required on Windows Vista and Windows 2008). DBDirector will also start Enterprise Server anytime DBDirector receives a connection request from a Databridge Client.

From the Enterprise Server **Help** menu, you can find instructions for importing sources (upgrades), creating sources, and configuring the program.

Start from the Start menu

1. From the **Start** menu, click **All Programs > Micro Focus Databridge 6.5 > Databridge Enterprise > Configure Enterprise**.

   This runs Enterprise Server with elevated administrator privileges.
2. When Enterprise Server starts, the Databridge Enterprise window appears.

Start from a command session

Navigate to the directory where you installed Enterprise Server and type the following:

```
DBEnterprise.exe
```

For information about command-line options, see the *Databridge Enterprise Server Help*.

10 Installation Results

This section includes lists of files that are installed with each product.

In this chapter:

Files Installed with Databridge Host

This section describes each of the files installed with Databridge Host software. Databridge Plus is also installed with the Host software.

Databridge software is organized into the following directories:

Directory	Description
DATA/=	Parameter and data files
OBJECT/DATABRIDGE/=	Executable files
PATCH/=	Source files
SYMBOL/DATABRIDGE/=	Source files
WFL/DATABRIDGE/=	WFL job files

OBJECT/DATABRIDGE Directory

Databridge Host files are copied to the usercode designated at installation.

Filename	Description
OBJECT/DATABRIDGE/AUDITMIRROR	Object code for the Databridge AuditMirror Accessory
OBJECT/DATABRIDGE/AUDITSUPPORT	Object code for the Databridge Plus library
OBJECT/DATABRIDGE/AUDITTIMER	Object code for the Databridge AuditTimer program
OBJECT/DATABRIDGE/BCNOTIFY	Object code for the BCNOTIFY utility

Filename	Description
OBJECT/DATABRIDGE/CHANGEUSER	Object code for the Databridge ChangeUser library, which changes the usercode of a Server Accessory Worker in cases where the SOURCE specifies a USERCODE, so the Worker will run under that usercode.
OBJECT/DATABRIDGE/COBOLSUPPORT	Object code for Databridge CobolSupport library
OBJECT/DATABRIDGE/DBINFO	Object code for the Databridge Database Info utility
OBJECT/DATABRIDGE/DCKEYIN	Object code for the ODT command program
OBJECT/DATABRIDGE/ENGINE	Object code for Databridge Engine Library
OBJECT/DATABRIDGE/GENFORMAT	Object code for the Databridge GenFormat program
OBJECT/DATABRIDGE/LICENSEMANAGER	Object code for product key management program
OBJECT/DATABRIDGE/LICENSESUPPORT	Object code for the license verification library
OBJECT/DATABRIDGE/LISTER	Object code for the Databridge Lister Accessory
OBJECT/DATABRIDGE/SAMPLE/AUDITCLOSE	Object code for the sample Databridge Audit Close utility
OBJECT/DATABRIDGE/SAMPLE/COBOLGEN	Object code for the sample Databridge COBOLGen Accessory
OBJECT/DATABRIDGE/SAMPLE/DASDLGEN	Object code for the sample Databridge DASDLGen Accessory
OBJECT/DATABRIDGE/SAMPLE/EXTRACTADDRESS	Object code for the sample ExtractAddress Library
OBJECT/DATABRIDGE/SAMPLE/READDOC	Object code for the sample Databridge ReadDoc Accessory
OBJECT/DATABRIDGE/SAMPLE/REFORMAT	Object code for sample data item conversion routines
OBJECT/DATABRIDGE/SAMPLE/SQLGEN	Object code for the sample Databridge SQLGen Accessory
OBJECT/DATABRIDGE/SERVER	Object code for the Databridge Server Accessory
OBJECT/DATABRIDGE/SNAPSHOT	Object code for the Databridge Snapshot Accessory

Filename	Description
OBJECT/DATABRIDGE/SPAN	Object code for the Databridge Span Accessory
OBJECT/DATABRIDGE/SUPPORT	Object code for the Databridge Support Library

SYMBOL/DATABRIDGE Directory

Databridge Host files are copied to the usercode designated at installation.

Filename	Description
SYMBOL/DATABRIDGE/COBOLSUPPORT	Source code for providing entry points to COBOL Accessories for filtering and formatting routines in the Support Library (ALGOL file type)
SYMBOL/DATABRIDGE/DMSIISUPPORT	Source code for the DMSII Support Library
SYMBOL/DATABRIDGE/SAMPLE/AUDITCLOSE	Sample source code for closing a DMSII audit file (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/COBOLGEN	Sample source code for generating COBOL file layouts corresponding to a DMSII database (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/DASDLGEN	Sample source code for generating a DMSII DASDL source corresponding to a DMSII database (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/DATABASE	Source code for defining the sample DMSII database (DASDL file type)
SYMBOL/DATABRIDGE/SAMPLE/EXTRACTADDRESS	Sample source code for a library that will extract addresses and build VIRTUAL records (COBOL85 file type)
SYMBOL/DATABRIDGE/INTERFACE	Source code Databridge host software API (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/READDOC	Sample source code for reading audit records (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/REFORMAT	Sample source code for writing data item conversion routines in a REFORMAT library (ALGOL file type)
SYMBOL/DATABRIDGE/SAMPLE/SQLGEN	Sample source code for generating an SQL script to create tables that correspond to a DMSII database. (ALGOL file type)
SYMBOL/DATABRIDGE/SUPPORT	Source code for the Support Library (ALGOL file type)

WFL/DATABRIDGE Directory

Databridge Host files are copied to the usercode designated at installation.

Filename	Description
WFL/DATABRIDGE/INSTALL	Work flow for installing the Databridge software
WFL/DATABRIDGE/SAMPLE/AUDITCLOSE	Work flow for executing the Audit Close sample Accessory
WFL/DATABRIDGE/SAMPLE/AUDITMIRROR	Work flow that receives mirrored audit files on the secondary computer
WFL/DATABRIDGE/SAMPLE/AUDITTIMER	Work flow for executing the AuditTimer utility
WFL/DATABRIDGE/SAMPLE/BCNOTIFY	Work flow for executing BCNOTIFY
WFL/DATABRIDGE/SAMPLE/COBOLGEN	Work flow for executing the BCNOTIFY in DBNOTIFY-mode.
WFL/DATABRIDGE/SAMPLE/COPYAUDIT	Work flow that can be used in place of the DMSII COPYAUDIT WFL to postpone removal of audit files
WFL/DATABRIDGE/SAMPLE/DATABASE	Work flow for creating and populating the sample database (BANKDB) supplied with Databridge
WFL/DATABRIDGE/SAMPLE/DBINFO	Work flow for executing the Databridge Database Info utility
WFL/DATABRIDGE/SAMPLE/DBNOTIFY	Work flow for compiling a Databridge sample program, the Support Library or DMSII Support Library
WFL/DATABRIDGE/SAMPLE/INCLUDE/DBTITLE	Work flow that parses the DMSII database title for all other WFLs
WFL/DATABRIDGE/SAMPLE/LISTER	Work flow for executing the Lister accessory
WFL/DATABRIDGE/SAMPLE/NOTIFY	Work flow for notifying Databridge Server of additional audit information
WFL/DATABRIDGE/SAMPLE/READDOC	Work flow for executing the ReadDoc sample Accessory
WFL/DATABRIDGE/SAMPLE/REMOVEAUDIT	Work flow for removing audit files
WFL/DATABRIDGE/SAMPLE/SPAN	Work flow for executing Databridge Span
WFL/DATABRIDGE/SAMPLE/SQLGEN	Work flow for executing the SQLGen sample Accessory
WFL/DATABRIDGE/SAMPLE/SERVER	Work flow for executing Databridge Server
WFL/DATABRIDGE/SAMPLE/SNAPSHOT	Work flow for executing the Snapshot Accessory

DATA Directory

Databridge Host files are copied to the usercode designated at installation.

Filename	Description
DATA/AUDITMIRROR/SAMPLE/CONTROL	Sample parameter file for the AuditMirror program (SEQDATA file type)
DATA/AUDITTIMER/SAMPLE/CONTROL	Sample parameter file for the AuditTimer program (SEQDATA file type)
DATA/COBOLTODASDL/SAMPLE/BANKFILE/CONTROL	COBOLTODASDL parameter file for the BANKFILE FileXtract example.
DATA/COBOLTODASDL/SAMPLE/BANKFILE/FD	COBOL FD for the BANKFILE FileXtract example.
DATA/COBOLTODASDL/SAMPLE/LINCLOGDB/FD	COBOL FD for the LINCLOGDB FileXtract example.
DATA/COBOLTODASDL/SAMPLE/LINCLOGDB/ CONTROL	COBOLTODASDL parameter file for the LINCLOGDB FileXtract example.
DATA/DATABRIDGE/SAMPLE/LICENSE	Sample software license file (DATA file type)
DATA/ENGINE/CONTROL	Parameter file that contains the key information for running this release of Databridge, as well as other options for DBEngine operation
DATA/ENGINE/SAMPLE/CONTROL	Sample DBEngine parameter file for specifying database-specific DBEngine parameters (SEQDATA file type)
DATA/GENFORMAT/SAMPLE/CONTROL	Sample parameter file for the GenFormat program (SEQDATA file type)
DATA/GENFORMAT/SAMPLE/BANKFILESDB/CONTROL	Sample BANKFILESDB parameter file for the GenFormat program (SEQDATA file type).
DATA/GENFORMAT/SAMPLE/LINCLOGDB/CONTROL	Sample LINCLOGDB parameter file for the GenFormat program (SEQDATA file type).
DATA/GENFORMAT/SAMPLE/NAPFILESDB/CONTROL	Sample NAPFILESDB parameter file for the GenFormat program (SEQDATA file type)
DATA/GENFORMAT/SAMPLE/SYSFILESDB/CONTROL	Sample SYSFILESDB parameter file for the GenFormat program (SEQDATA file type))
DATA/GENFORMAT/SAMPLE/USERDATADB/CONTROL	Sample USERDATADB parameter file for the GenFormat program (SEQDATA file type).
DATA/LISTER/SAMPLE/CONTROL	Sample parameter file for the Lister Accessory (SEQDATA file type)
DATA/LOAD/SAMPLE/DATABASE	Data for the sample database that you can use with Databridge
DATA/SERVER/SAMPLE/CONTROL	Sample parameter file for Databridge Server (SEQDATA file type)
DATA/SERVER/SAMPLE/FILEBRIDGE/CONTROL	Sample FileXtract parameter file for the DBServer program (SEQDATA file type)

PATCH Directory

Databridge Host files are copied to the usercode designated at installation.

Filename	Description
PATCH/DATABRIDGE/COBOLFD/PARSER	Routines for parsing a COBOL FD.
PATCH/DATABRIDGE/COBOLTODASDL/PARSER	Routines for parsing a COBOLTODASDL parameter file.
PATCH/DATABRIDGE/DASDL/TWIN	A DASDL patch you must include in your database to run Databridge Twin if your restart data set does not have the required fields
PATCH/DATABRIDGE/PARSER/DEFINES	Definitions included when compiling a parser.
PATCH/DATABRIDGE/SAMPLE/SUPPORT/REFORMAT	Sample INTERNAL REFORMAT program
PATCH/DATABRIDGE/SAMPLE/SUPPORT/ ERRORHANDLER	Sample error manager routine
PATCH/DATABRIDGE/SAMPLE/SUPPORT/ FORMATADDRESS	Sample formatting routine for VIRTUAL data sets
PATCH/DATABRIDGE/SAMPLE/SUPPORT/ NAPREFORMAT	Sample INTERNAL REFORMAT program for the NAPFilesDB FileXtract database.
PATCH/DATABRIDGE/SAMPLE/SUPPORT/SHUTDOWN	Sample shutdown procedure for a Support Library
PATCH/DATABRIDGE/SAMPLE/SUPPORT/ SYSREFORMAT	Sample INTERNAL REFORMAT program for the SYSFilesDB FileXtract database.
PATCH/DATABRIDGE/SAMPLE/SUPPORT/STARTUP	Sample startup procedure for a Support Library
PATCH/DATABRIDGE/SAMPLE/SUPPORT/VIRTUAL	Sample skeleton patch to the Support Library for a VIRTUAL TRANSFORM routine

Files Installed with DMSII Client

The following files are copied to the usercode you designated when you installed DMSII Client:

File name	Description
DATA/DMSIICLIENT/SAMPLE/CONTROL	Sample parameter file for DMSII Client See “Modifying the DMSII Client Parameter File” in the “*DMSII Client Administrator’s Guide*” for an example.
OBJECT/DATABRIDGE/DMSIICLIENT	Object code for DMSII Client
OBJECT/DATABRIDGE/DMSIICLIENT/ AUTOCONNECT	Object code for the auto connect feature
PATCH/DATABRIDGE/DMSIICLIENT/SAMPLE/ DASDL	Sample patch file to specify DASDL particulars See “Modifying the DMSII Client DASDL File” in the “*DMSII Client Administrator’s Guide*” for an example.
SYMBOL/DATABRIDGE/DMSIICLIENT/LIB	Library source code used by DMSII Client when accessing a client database

File name	Description
SYMBOL/DATABRIDGE/DMSIISUPPORT	Library source code for performing the actual inquiries and updates of the client database
SYMBOL/DATABRIDGE/INTERFACE	Source code INCLUDE file for the API
WFL/DATABRIDGE/INSTALL	Work flow for installing DMSII Client
WFL/DATABRIDGE/SAMPLE/BACKUPTAILORED	Workflow for backing up tailored DMSII software, DESCRIPTION and MSUPPORT, with the update level as the last node of the file titles.
WFL/DATABRIDGE/SAMPLE/DMSIICLIENT	Work flow for executing DMSII Client
WFL/DATABRIDGE/SAMPLE/DMSIICLIENT/COMPILEDB	Work flow for compiling DMSII Client database when using the manual compile option
WFL/DATABRIDGE/SAMPLE/DMSIICLIENT/REORGDB	Work flow for compiling the DMSII Reorganization program for generating the sets for a cloned data set.

The SAMPLE work flows are also copied without the SAMPLE node if those files don't already exist.

Files Installed with Databridge Twin

The following is a list of files that are installed on the secondary system. These files are copied to the secondary system via the Databridge installation WFL:

File Name	Description
DATA/ENGINE/SAMPLE/CONTROL	**Required.** This data file contains options for running the Databridge Twin Engine.
DATA/TWIN/SAMPLE/CONTROL	**Required.** This Databridge Twin parameter file provides information to the Databridge Twin program on where and how to locate the Server SOURCE for the primary database.
OBJECT/DATABRIDGE/DBINFO	DBInfo creates reports about your DMSII database. For Databridge Twin Engine, only the DBInfo normal mode is supported because the Databridge Twin Engine does not provide access to the audit file. For instructions on DBInfo, see the *Databridge Host Administrator's Guide*.
OBJECT/DATABRIDGE/ENGINE	**Required.** This is the object file for the Databridge Twin Engine, which is a subset of the full Databridge Engine.
OBJECT/DATABRIDGE/GENFORMAT	**Optional.** Object code for the Databridge GenFormat program
OBJECT/DATABRIDGE/SUPPORT	**Required.** This is the object code for the non-tailored Support library.
OBJECT/DATABRIDGE/TWIN	**Required.** This is the object file for the Databridge Twin program, which calls the Databridge Twin Engine.

File Name	Description
PATCH/DATABRIDGE/TWIN/DASDL	**Optional.** This file is required if you use the INCLUDE statement in your DASDL instead of inserting the contents of this patch file into the DASDL. If you use an INCLUDE statement to include this file in the DASDL, this file must be on both the primary and the secondary systems.
SYMBOL/DATABRIDGE/DMSIISUPPORT	**Required.** This file is the source file for the DMSIISupport Library that will update the secondary database.
SYMBOL/DATABRIDGE/INTERFACE	**Optional**. This file supplies the API for the Databridge Engine. It is required for compiling the Support Library.
SYMBOL/DATABRIDGE/SUPPORT	**Optional.** This is the source code for the Support Library.
WFL/DATABRIDGE/INCLUDE/DBTITLE	**Required.** This WFL source file is required for running any Databridge WFL. It parses the database name you enter when you start the WFL
WFL/DATABRIDGE/INSTALL	**Required.** This WFL is required for installing Databridge Twin.
WFL/DATABRIDGE/SAMPLE/COMP	**Optional.** This WFL compiles the Support Library.
WFL/DATABRIDGE/SAMPLE/TWIN	**Required.** This is the WFL for running the Databridge Twin program.
WFL/DATABRIDGE/SAMPLE/INCLUDE/SSRTITLES	**Required.** This WFL source file is included by WFL/DATABRIDGE/COMP to specify the location of the compilers for each SSR (system software release).
WFL/DATABRIDGE/TWININITIALIZE	**Optional.** If you run this WFL, it starts WFL/DATABRIDGE/TWIN with the LOAD command. **NOTE:** This feature is deprecated, but still operates to support legacy use.

After you have verified the files, you are ready to configure the primary and secondary databases (see Configuring Databridge Twin in the *DataBridge Twin Administrator's Guide*).

To replicate the primary database, see the chapter "Replicating a Database" in the *DataBridge Twin Administrator's Guide*.

Files Installed with Databridge Enterprise Server

Use this document to verify the default installation results for Databridge Enterprise Server.

In addition to installing the Enterprise Server files, the setup program adds shortcuts to the **Start** menu from which you can configure a source and view the configuration and log files folder.

The following files are installed to `C:\Program Files\Micro Focus\Databridge\6.5\Enterprise`:

Filename	Description
DBUI.dll	The optional message and dialog text for Enterprise Server, which can be translated into other (non-English) languages

Filename	Description
DBDirector.exe	Windows service that listens for requests from Databridge Clients and then initiates the Enterprise Server application
Help folder	The Databridge Enterprise Server Help files (.html) which are viewable from the program by selecting the Help menu or by pressing the F1 key.
Support folder	Header files and DBEUser.lib for compiling user-written libraries
Support\MakeUserLib.cmd	Batch file for building user-written libraries
DBEnterprise.exe	Enterprise Server executable file
EnumDisk.exe	Enterprise Server utility that lists the MCP disks available on the server where Enterprise Server is installed
FinishMirrored.cmd	Sample batch file to run whenever an audit file is successfully mirrored to the secondary system.
LINCLOG.dll	LINC log FileXtract Reader library
Notify.cmd	Sample batch file to run when the Databridge Server accessory signals the availability of more audit files to process

Files Installed with Databridge Client and Client Console

During installation, Databridge Client 6.5 copies files to the install directory and a client-specific directory (such as SQLServer). On Windows, the install directory is located at `C:\Program Files\Micro Focus\Databridge\6.5`. In addition, the Micro Focus Installer Program installs a Setup folder in this location containing files that repair, reinstall, and remove the program.

In the following tables, the filenames represent both UNIX and Windows files. Windows extensions are provided in brackets ([ ]). The following files are copied to the install directory:

Filename	Description
bconsole[.exe]	The Batch Console automates routine Client tasks by allowing command files/shell scripts launched by the Client service to interact with the service. For more information, see the *Databridge Client Administrator's Guide*.
DATABridge_Messages.dat	A binary file that the Databridge service/daemon uses to retrieve the text associated with error status codes generated by various Databridge components.
DBClntControl[.exe]	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. The service has no effect on dbutility command-line operations.
dbctrlconfigure[.exe]	A command-line program that implements the import and export commands for the service configuration file. It converts the service configuration file (dbcontrol.cfg) from its native binary format to text format (.ini) and back using the export and import commands.

Filename	Description
logmaint[.exe]	A utility that allows you to delete all log files from the service and the individual data source directories except for the most recent. Run this utility from the working directory for the service or the data source whose log files you want to manage. The number of files to retain are specified on the command line for logmaint as optional parameters. When used for the client, it has two parameters. The first parameter is for the DBClient and dbutility log files. The second parameter is for DBClnCfgServer log files. The parameters default to 10 and can be set to a value as low as 2.
readme.txt	A text file that contains late-breaking information for the product.
setbcuserid.exe (Windows only)	A Windows program that lets you register a specific Windows userid as a trusted user. This allows this userid to run the bconsole utility without specifying a password by adding /T option to the command line. This avoids having clear-text passwords in command files that run bconsole. This is meant for runs that are initiated from places like the Windows scheduler or third-party job scheduling tools.
setfilesecurity.exe (Windows only)	A Windows program that lets you set file security for all files and directories in the Working Directory (which includes changes to the Windows registry). This program functions the same as the utility started by the installer, except that it doesn't allow you change the system or user path. You must run this program as Administrator.
showerror.exe	A utility that prints the test associated with a Databridge error number.

The following files are copied to the client-specific directory:

Filename	Description
bcp_auditor.exe (Windows only)	A program that scans the bcp output to determine if the bcp utility was successful. The bcp_auditor utility sets the exit code correctly. This utility applies only to Microsoft SQL Server databases.
DBClient[.exe]	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.
DBClntCfgServer[.exe]	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.
*dbeatran.dll *dbgaiji.smp *dbtrans_jbis8.smp *dbtrans_v24jbis8.smp	Japanese double-byte translation DLL and sample configuration files. Depending on which character set the mainframe uses, copy the appropriate dbtrans file to the config directory as `dbtrans.cfg`. Also, copy the file `dbgaiji.smp` to the config directory as `dbgaiji.cfg`.
dbfixup[.exe]	A program used for updating existing client control tables when upgrading from a previous client release.

Filename	Description
dbpwenc[.exe]	A program used to encode relational database and DBServer (or DBEnterprise Server) passwords in text configuration files and change passwords in text and binary configuration files. **NOTE:** Passwords set in the Client Console are automatically encoded in binary configuration files.
dbscriptixup[.exe]	A program that modifies the Client control tables according to changes specified by the user scripts and enables the Client Configurator to make future modifications to the table layout.
dbutility[.exe]	A program that runs Databridge Client from a command line.
getlogtail.exe	A program that extracts the tail of the current log file and writes it to the file `trace.log`.
migrate[.exe]	The upgrade program that updates configuration settings and creates the working directory structure. It copies the updated configuration files and any user scripts to the new working directory structure.
splitter[.exe]	A utility that splits large trace files into smaller files. Used primarily for sending files to Micro Focus for troubleshooting.
dumpnullrecs[.exe]	A program that displays the content of the NULL record file for a given data source.

*Installed only when the Japanese language feature is selected.

Client Console Files

Files for the Client Console are installed in the following two directories at `C:\Program Files\Micro Focus\Databridge\6.5`.

Directory	Description
DBConfig	Contains the configuration file, initialization file, and plugins subdirectory. The plugins subdirectory which contains required Eclipse plugins and source code plugins for Databridge Client Configurator.
DBConsole	Contains the configuration file, initialization file, and plugins subdirectory. The plugins subdirectory which contains required Eclipse plugins and source code plugins for Databridge Client Console.

Glossary of Terms

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). In Databridge Client, AA is the hexadecimal representation (12 character strings containing the characters 0–9 and A–F) of the AA Value on the host. Databridge Client uses the AA Values to implement unique keys for the parent structures of embedded data set records. It also uses AA Values to reference the records of data sets that do not have DMSII SETS with the NO DUPLICATES ALLOWED attribute.

AA Values are not constant. Any DMSII reorganization (record conversion, file format, or garbage collection) changes these values.

Databridge Client supports numeric AA Values that are stored as NUMBER(15) in Oracle and BIGINT in SQL Server. It also supports binary AA Values that are stored as RAW(6) in Oracle and BINARY(6) in SQL Server.

Audit Files. An audit file is created by DMSII and contains the raw format of changes made to the DMSII database by update programs. Audit file records contain the deletes, adds, and modifies that were made to the various structures. It can contain, for example, hours', days', or weeks' worth of information.

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

If you set the Databridge Engine Read Active Audit option, Databridge can access the current audit file. If you do not set Read Active Audit = true in the Databridge Engine parameter file, Databridge can access audit information up to the current audit file minus one. The audit file contains the update level at the time the audit file was created. The update level in the audit file and the update level in the DESCRIPTION file used by Databridge must match before Databridge will update a replicated database.

When an audit file is closed, DMSII creates the next one in the series. Audit files are closed for several reasons, including the following:

• An operator closes the audit file with the *mixnumber* SM AUDIT CLOSE command.

• The audit file reaches the file size set in its DASDL.

• There is an I/O error on the audit file.

• There is not enough disk space for this audit file.

• The database update level changes due to database definition changes

• The current audit file could not be found.

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 Databridge Client Manager to interact with the service.

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

client. 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.

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. Databridge Engine (also referred to as DBEngine) is the main library program on the host that reads and interprets the database DESCRIPTION file, the database CONTROL file, and the audit trail.

Databridge Server. Databridge Server is a generic term that can refer to either DBServer or Databridge Enterprise Server. The two are interchangeable as far as Databridge Client is concerned.

DBClient. A Client program that is launched by the Client Manager 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 Client Manager service as a background run.

DBServer. 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:

- Databridge Engine and Databridge Enterprise Server
- Databridge Engine and Databridge Client

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

---

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 that affects file size values (for example, AREASIZE, BLOCKSIZE, or TABLESIZE), but it does not change the layout of the records in a DMSII database.

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.

Japanese Support. Select this option to install translation for double-byte EBCDIC to Code Page 932.

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.

Localization. Select this option to install a dynamic link library (DBUI.dll), which contains the interface and message text for Enterprise Server. This file can be decompiled and translated to provide non-English text for the program interface.

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 record. A record for a data set where every data item is null.

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.

reorganization. Structural or formatting changes to records in the DMSII database, which may require parallel changes to (or re-cloning of) records in the secondary, or relational, database.

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.

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.

visible RSN. An RSN (record serial number) that is declared in the DASDL. These appear as an item in the data set and are therefore visible to the database user.

working directory. A directory structure that is required for the Client to run. This global working directory contains scripts, logs, lock files and configuration files. Each configured data source has its own working directory within the global working directory.