DBMaker

JServer Manager User's Guide

CASEMaker®

CASEMaker Inc./Corporate Headquarters

1680 Civic Center Drive

Santa Clara, CA 95050, U.S.A.

www.casemaker.com

www.casemaker.com/support

Document No.645049-231024/DBM41-M05262003-JSEM

Publication Date: 2003-05-18

For a description of updated functions that do not appear in this manual, read the file named README.TXT after installing the CASEMaker DBMaker software.

Trademarks

CASEMaker, the CASEMaker logo, and DBMaker are registered trademarks of CASEMaker Inc. Microsoft, MS-DOS, Windows, and Windows NT are registered trademarks of Microsoft Corp. UNIX is a registered trademark of The Open Group. ANSI is a registered trademark of American National Standards Institute, Inc.

Other product names mentioned herein may be trademarks of their respective holders and are mentioned only form information purposes. SQL is an industry language and is not the property of any company or group of companies, or of any organization or group of organizations.

Notices

The software described in this manual is covered by the license agreement supplied with the software.

Contact your dealer for warranty details. Your dealer makes no representations or warranties with respect to the merchantability or fitness of this computer product for any particular purpose. Your dealer is not responsible for any damage caused to this computer product by external forces including sudden shock, excess heat, cold, or humidity, nor for any loss or damage caused by incorrect voltage or incompatible hardware and/or software.

Information in this manual has been carefully checked for reliability; however, no responsibility is assumed for inaccuracies. This manual is subject to change without notice.

Contents

1 Introduction

Welcome to JServer Manager User's guide. JServer manager is a cross-platform graphical user interface (GUI) for creating and managing databases using DBMaker—an SQL Database Management System.

This manual gives systematic instructions on how to use JServer Manager for the creation and management of multiple databases. The features of the program are explained using procedures supported by screen images containing data from the sample databases provided with the software.

This book is intended for administrators of DBMaker databases who intend to access and manage their databases from a variety of platforms and hardware configurations. Administrators using this manual may or may not be familiar with all of the features of DBMaker. It is assumed that the user does have some knowledge of working with the platform on which DBMaker is installed.

The JServer Manager User's Guide is organized according to the procedures used to create, start, stop, delete, and restore databases using the JServer Manager applet. The manual follows the structure of the interface in easy to understand steps.

1.1 Additional Resources

DBMaker provides many other user's guides and reference manuals in addition to this guide. You should consult one of the books shown below for more information on a particular subject.

For more information on designing, administering, and maintaining a DBMaker database, refer to the *Database Administrator's Guide*.

For more information on the tools and utilities provided with DBMaker, refer to the *dmSQL User's Guide*, the *JDBA Tool User's Guide*, or the *Server Manager User's Guide*.

For more information on other Java tools provided with DBMaker, refer to the *JDBA Tool User's Guide* or the *JConfiguration Tool Reference*.

For more information on the SQL language implemented by DBMaker, refer to the *SQL Command and Function Reference*.

For more information on the ESQL/C language implemented by DBMaker, refer to the *ESQL/C Programmer's Reference*.

For more information on error and warning messages, refer to the *Error and Message Reference*.

1.2 Technical Support

CASEMaker provides thirty days of complimentary email and phone support during the evaluation period. When software is registered, an additional thirty days of support will be included, thus extending the total support period for software to sixty days. However, CASEMaker will continue to provide email support for any bugs reported after the complimentary support or registered support has expired (free of charges).

Additional support is available beyond the sixty days for most products and may be purchased for twenty percent of the retail price of the product. Please contact sales@casemaker.com for more details and prices.

CASEMaker support contact information for your area (by mail, phone, or email) can be located at: www.casemaker.com/support. It is recommended that the current database of FAQ's be searched before contacting CASEMaker support staff.

Please have the following information available when phoning support for a troubleshooting enquiry or include the information with a letter or email:

Product name and version number

Registration number

Registered customer name and address

Supplier/distributor where product was purchased

Platform and computer system configuration

Specific action(s) performed before error(s) occurred

Error message and number, if any

Any additional information deemed pertinent

1.3 Document Conventions

This book uses a standard set of typographical conventions for clarity and ease of use. The NOTE, Procedure, Example, and Command Line conventions also have a second setting used with indentation.

CONVENTION	DESCRIPTION
Italics	Italics indicate placeholders for information that must be supplied, such as user and table names. The word in italics should not be typed, but be replaced by the actual name. Italics also are used to introduce new terms, and are occasionally used for emphasis.
Boldface	Boldface indicates filenames, database names, table names, column names, user names, and other database schema objects. It is also used to emphasize menu commands in procedural steps.
KEYWORDS	All keywords used by the SQL language appear in uppercase when used in normal paragraph text.
SMALL CAPS	Small capital letters indicate keys on the keyboard. A plus sign (+) between two key names indicates to hold down the first key while pressing the second. A comma (,) between two key names indicates to release the first key before pressing the second key.
NOTE	Contains important information.
➲ **Procedure**	Indicates that procedural steps or sequential items will follow. Many tasks are described using this format to provide a logical sequence of steps for the user to follow
➲ **Example**	Examples are given to clarify descriptions, and commonly include text as it will appear on the screen.
Command Line	Indicates text as it should appear on a text-delimited screen. This format is commonly used to show input and output for dmSQL commands or the content in the **dmconfig.ini** file

Figure 1-1 Document Conventions Table

2 JServer Manager Basics

The following sections briefly describe some of the features of JServer Manager that are unique to the Java applications. Please take a moment to review this chapter before reading the following chapters of this manual. Section 2.2 describes the JServer Manager workspace and the different elements of the user interface: the menu bar and the main console. Section 2.3 explains how to use the Save / select file / select path dialog boxes. These three dialog boxes operate in a similar manner and appear any time a browse button is selected or a directory path needs to be specified.

2.1 Summary of Features

JServer Manager has a wide variety of features that help you to effectively manage your database.

Creating Databases

Create single-user or multi-user databases

Link to the JConfiguration Tool for managing the **dmconfig.ini** file

Starting and Shutting Down a Database

Start and shut down databases on local computers

Link to JConfiguration Tool for managing database settings after startup

Link to Server Manager for database connection management

Deleting Databases

Delete a database

Delete file objects within the database

Backing up Databases

Perform full backups while the database is on-line or off-line

Perform full backups to disk or to tape

Perform incremental backups

Restoring Databases

Restore crashed databases from disk or from tape

Integrated User Interface

Linked to JDBA Tool and JConfiguration Tool

JServer Manager's GUI is goal oriented; systematic methods make database management easy

2.2 The JServer Manager Workspace

The JServer Manager Workspace consists of two main areas: the menu bar and the main console. Most typical database server operations can be found on the main console. The following diagram illustrates the JServer Manager Workspace

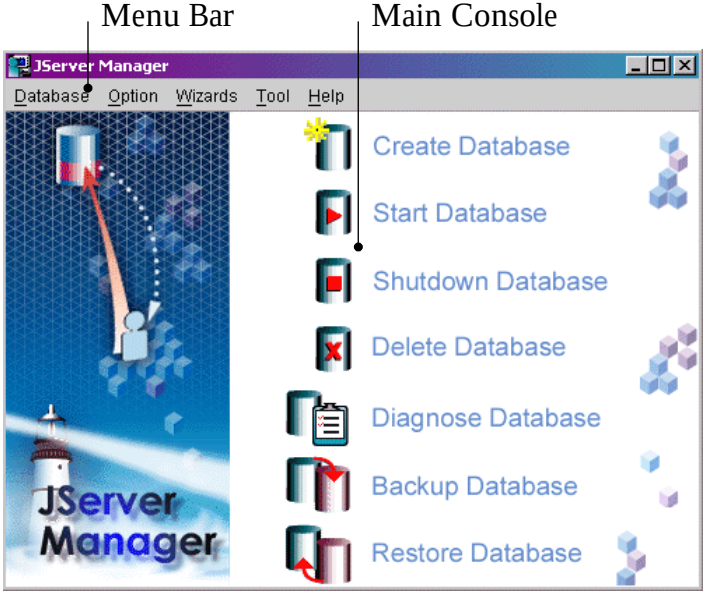


Figure 2-1 The JServer Manager Workspace

Main Console Structure

The main console contains seven icons representing the main functions of JServer Manager. Clicking on one of the icons opens a window that lets the user execute the desired function. The following chapters' headings correspond to each of these functions. The chapters use procedures to describe how to use each of these functions, and give troubleshooting tips and helpful information.

Menu Bar

The menu bar in JServer Manager has all the functions that are contained in the main console of the program. The Menu bar contains four pull down menus: Database, Option, Tool, and Help.

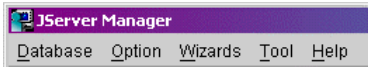


Figure 2-2 The JServer Manager Tool Bar

The Database menu gives access to the same functions available from the JServer Manager main console (See Main Console Structure).

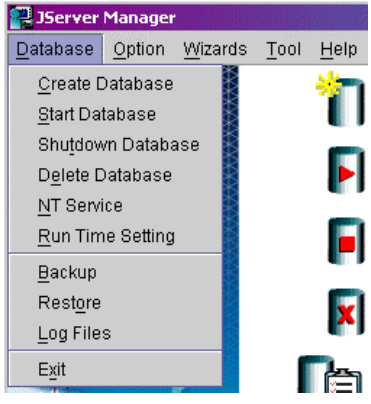


Figure 2-3 The Database Menu

The Option menu lets users change languages (currently JServer manager supports English, Traditional Chinese and Japanese).

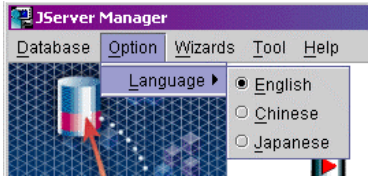


Figure 2-4 The Option Menu

The Wizards menu provides access to wizards that can help guide the user step-by-step through database creation, deletion, as well as renaming and diagnosing a database.

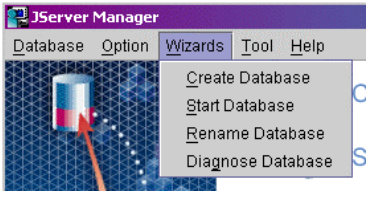


Figure 2-5 The Wizards Menu

The Tool menu provides quick access to the JDBA Tool and JConfiguration Tool (See the *JDBA Tool User's Guide* and the *JConfiguration Tool Reference* for information regarding the use of these programs), and the Help menu allows access to the help files.

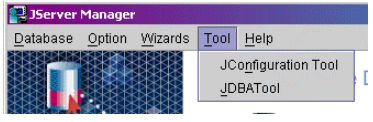
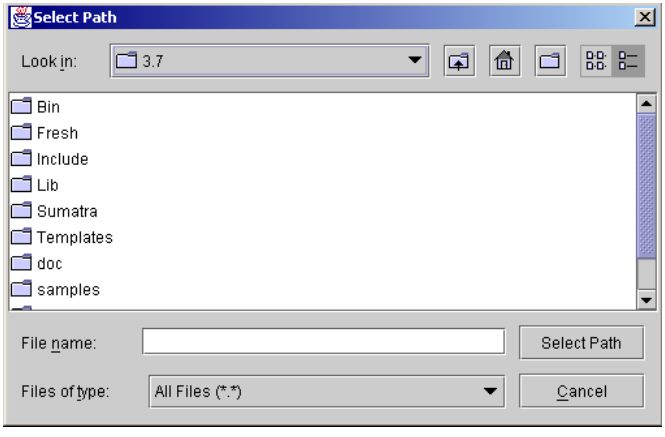


Figure 2-6 The Tool Menu

2.3 Select File / Select Path / Save dialog box

Many settings require the entry of a file name or a directory path. These may be entered manually or selected by using the browse button ... . Selecting the browse button opens the following dialog box.



The figure above shows the **Select Path** dialog box. The **Select File** and **Save** dialog boxes differ in function, but are structurally the same. Directly underneath the title bar are the **Look in** menu and five buttons:

Up One Level button: makes the current directory one level closer to the root directory.

Home button: makes the active directory the user's home directory. In Windows 98, this is the My Documents folder. In Windows NT or Windows 2000, this is the user directory.

Create New Directory button: Creates a new folder in the active directory labeled "New Folder".

The *List* and *Details* buttons are inactive in the JServer Manager Tool.

Below the menu and the buttons is the main window, which lists the files and subdirectories of the current directory. A user can browse up and down the directory tree by double clicking on a directory name (down 1 level), or clicking the *Up One Level* button. The user can select an individual file or directory from the main window box by left clicking once on it. A user can also edit any directory or file name from the main window. To edit a file name or directory name, right click on the file or directory and make changes to the name. Pressing enter saves the new directory or file name. Be sure that the name of the selected file or directory appears in the *file name* text box below.

The **File name** and **Files of type** text boxes are displayed below the main window. The file name text box should display the name of the file the user wants to select. The user can then select or cancel the action by clicking on the appropriate button to the right

3 Creating a Database

Databases can be run in a single-user or client-server mode in Windows. It is necessary to provide a port number and server address for databases that are to be started in client-server mode. A multi-user database may be started as a single-user database and vice versa. This option is provided during database creation to ensure that the port number and server address are specified if it is created as a multi-user database. Only one user may connect to a single user database.

For users that are not familiar with the settings involved in creating a database, the create database wizard has been provided. Refer to Section 3.4 for systematic instructions on use of the wizard.

Before creating a database, you should consider the following guidelines for creating database names:

Database names can be up to 32 characters long.

Database names can contain any alphanumeric characters, Chinese double byte characters, or the underscore character in any position.

Database names are not case-sensitive.

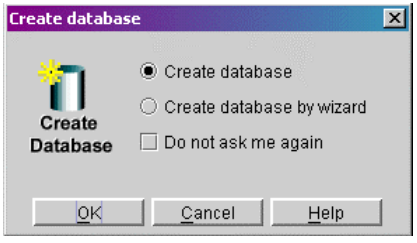
Database names must be unique among all computers that will connect to the database. Since DBMaker stores these names and other configuration details for local and remote databases in the **dmconfig.ini** configuration file, using the same name for two databases will cause a conflict. If the same database section appears twice in **dmconfig.ini**, DBMaker will use settings from the database section that appears first.

3.1 Creating a Database: Basics

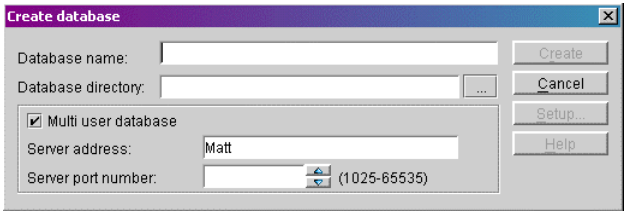
The following procedure outlines how to create a database without using the create database wizard. If you are new to DBMaker, the create database wizard can help guide you through and explain the advances database settings that may be necessary for your database. Refer to section 3.3 for more information on the create database wizard.

➲ **To create a database without using the create database wizard:**

1. Select **Create Database** from the main console or the **Database** menu. The following dialog box will appear:

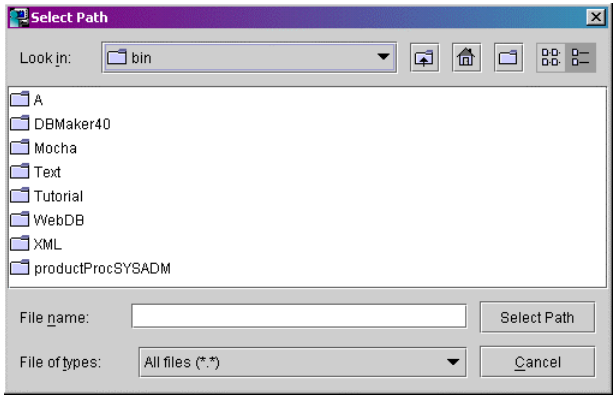


2. To use the wizard to create the database, select **Create database by wizard** and then refer to Section 3.3. Select **Create database** to avoid using the wizard, the following dialog box will appear.



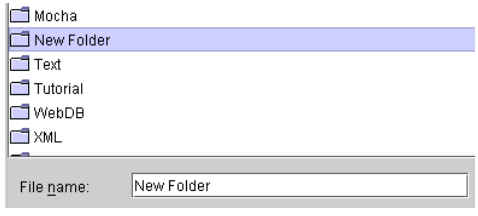
3. Type in the name for the new database in the **Database Name** field.

4. Enter the database directory into the **Database Directory** field, or select the browse button [...]. If the browse button is selected, the following dialog box will open:

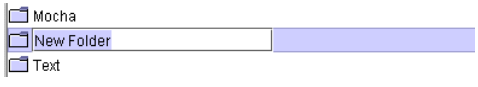


5. The default database directory path is *DBMaker\4.1\bin*. A new directory can be created for the database. To create a new database directory, first use the *Up One Level* button and/or the **Look in** menu to select the root for the database directory. Then click on the **Create New Directory** button to make a directory called **New Folder**. The new folder will appear as follows:

6. Select the folder **New Folder**, it will appear in the **File Name** text field.



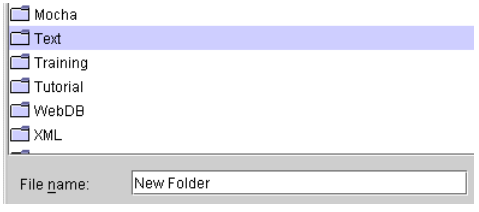
7. Right click **New Folder** again. It will appear as follows.



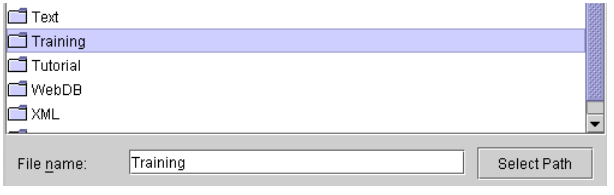
8. Selecting the directory **New Folder** again will allow the user to edit the text.

9. To change the directory name, type over the original name. Note that these changes are made directly to the operating system (use caution when changing the names of existing directories). After typing the new directory name it will appear as follows:

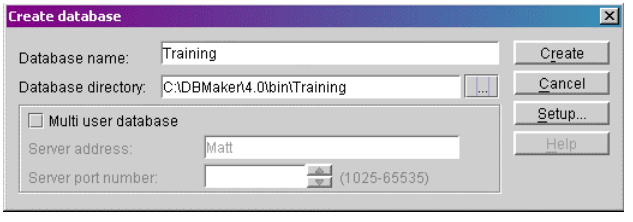
10. Press the enter key to complete the creation of the new directory. The list will reappear in alphabetical order. Note that **New Folder** still appears in the **File name** field.



11. Select the new directory name from the list, it will appear in the **File name** field.



12. To select the new directory as the database directory, click **Select Path**. The **Create Database** dialog box reappears.

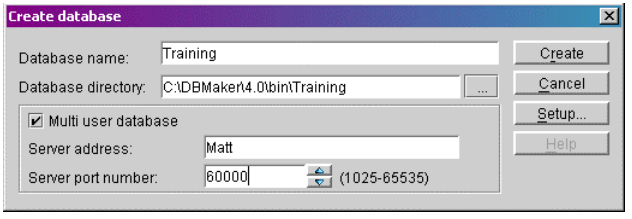


13. To start the database as a multi-user database, enable the **Multi User Database** check box. To start the database as a single user database, disable the **Multi User Database** check box.

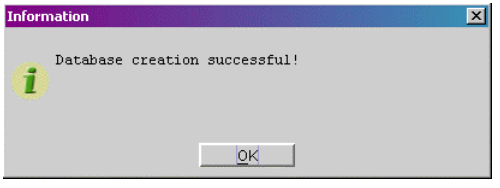
14. Enter the IP address or host name of the server in the **Server Address** field (for multi-user databases).

NOTE *If a host name is specified, ensure that Domain Name Server is set up on the server.*

15. Enter the server port number in the **Server Port Number** field. This must be an integer between 1025 and 65535(for multi-user databases).



16. Now you may choose to create the database or alter advanced database settings for the new database. To finish creating the new database, click **Create**. To change settings in the **Create Database Advanced Settings** window, select **Setup**. If no errors are encountered upon selecting **Create**, DBMaker will display the following dialog box.



NOTE *Refer to Creating a Database: Advanced Settings for information on settings that apply to database creation.*

17. Click **OK**.

3.2 Creating a Database: Advanced Settings

You can alter the following advanced settings before creating a database:

Create Database – settings that may only be applied before database creation

Storage – location and initial size of data, journal, and temporary files

Backup – frequency and destination of backup files

dmServer – network address, port number, idle time-out value, encryption

Cache and Control – Upper memory caching of data pages and system control, SQL command upper memory caching, CHAR data output padding, system File Object mapping, Cursor Behavior, and Browse Mode

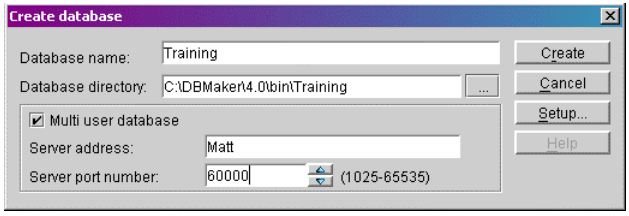
DDB Environment – Distributed database behavior

SQL Attributes – Date and time formatting for SQL statements, stored procedure settings

The tabbed pages available in this window correspond exactly with the settings in the JConfiguration Tool. To learn more about how these settings affect the creation and subsequent management of databases, refer to the *JConfiguration Tool Reference*. Database administrators should be aware of database requirements before creating the database. The language code, file name of code order, case sensitivity, and BLOB frame size are important functions of a database that can only be set before the database is created. Cache and Control, DDB Environment, and SQL attributes are discussed in *Starting a Database: Advanced Settings*..

➲ **To create a database with advanced settings:**

1. Enter the database name, directory, IP address and Port Number, if applicable, as outlined in Section 3.1



2. Click the **Setup** button. The **Create Database Advanced Settings** dialog box is displayed.

3. Enter advanced setting options (see the following procedures):

Setting Create Database Options

Setting Storage Options

Setting Backup Options

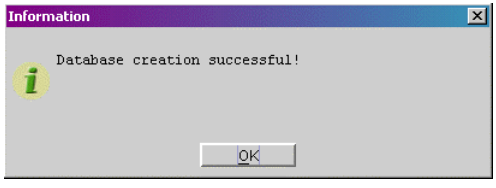
Setting dmServer Options

Setting Cache and Control Options (refer to Chapter 4, *Setting Cache and Control Options*)

Setting DDB Options (refer to Chapter 4, *Setting Distributed Database Environment Options*)

Setting SQL Attributes (refer to Chapter 4, *Setting SQL Attributes*)

4. Now the user may choose to create the database or alter advanced database settings for the new database. To finish creating the new database, click the **Create** button.



5. Click **OK**.

Setting Create Database Options

You can set the following Create Database settings. The Create Database page corresponds exactly to the Create Database page in the JConfiguration Tool. Please refer to Chapter 14 of the *JConfiguration Tool Reference* for more information.

NOTE *The settings on the Create Database page can only be set before creation of the database.*

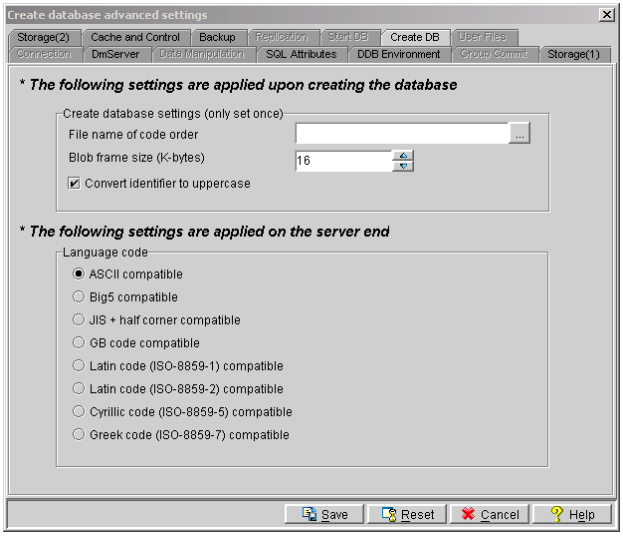


Figure 3-1 Create page of the Advanced Database Settings window

FILE NAME OF CODE ORDER

The database administrator can specify a location for the order definition file only before creating a database. The order definition file must be defined before creating a database and cannot be modified after creating a database. DBMaker uses the order definition file to assign characters a value for comparison and sorting. The default order definition is single byte characters followed by double byte characters. The order definition file is used for sorting indexes alphabetically as well as for comparison operations that use character data. See the *Database Administrators Guide* or the *JConfiguration Tool Reference* for instructions on how to create an order definition file.

BLOB Frame Size

The administrator can set the frame size for BLOB type data only before the creation of the database. A larger BLOB frame size means that more disk space is used each time a BLOB type data column is added, however, frames need to be added less frequently when BLOB type data is appended. If processor resources will be in demand, the BLOB frame size can be set to a larger size. If disk space is an issue and processor resources are not, the BLOB frame size can be decreased.

Convert Identifier to Uppercase

The database administrator can choose whether database object identifiers will be case sensitive or not only once, before the creation of the database. Identifiers do not include passwords, user-defined function names, and database names.

Language Code

The database administrator can decide which character set the database will use for CHAR type data

➲ **To set Create Database options:**

1. Click on the **Setup** button in the **Create Database** window.
2. Select the **Create Database** tab in the **Create Database Advanced Settings** window.
3. Enter a path or select the browse button  next to the **File Name of Code Order** field to indicate the location of the code order file.
4. Enter a BLOB frame size in bytes in the **BLOB Frame Size** field.
5. To make all database objects case insensitive select the **Convert Identifier to Uppercase** option.
6. To change the character set the database will use for CHAR type data, select one of the four option buttons.

   Select **ASCII compatible** to use ASCII code.

   Select **Big5 compatible** to use traditional Chinese code.

   Select **JIS + half corner compatible** to use Japanese code.

Select **GB code compatible** to use simplified Chinese code.

Select **Latin code (ISO-8859-1) compatible** to use Latin – 1 code.

Select **Latin code (ISO-8859-2) compatible** to use Latin – 2 code.

Select **Cyrillic code (ISO-8859-5) compatible** to use Cyrillic code.

Select **Greek code** (ISO-8859-7) compatible to use Greek code.

7. Click the **Save** button, or click the **Cancel** button to return to the **Create Database** dialog box.

Setting Storage Options

The database administrator can specify the storage location for different operating system files that make up the database. Storage options are set on two different pages in the Create database Advanced Settings window: Storage (1) and Storage (2). The Storage page exactly corresponds to the Storage page in the JConfiguration Tool. Please refer to the *JConfiguration Tool Reference* for more information on storage.

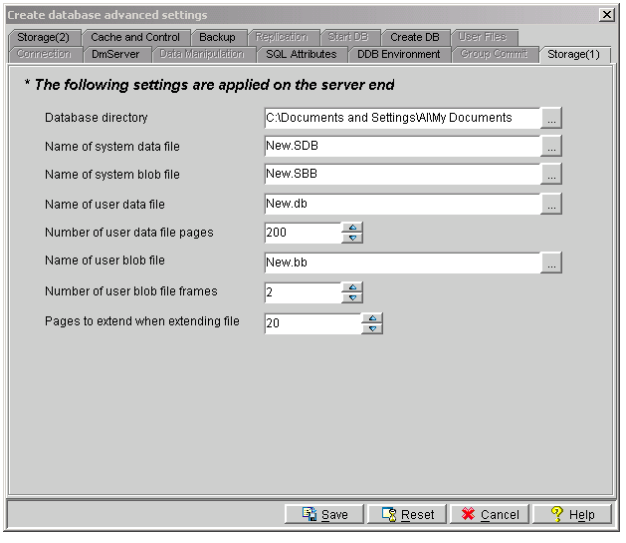


Figure 3-2 The Storage (1) page of the Advanced Database Settings window

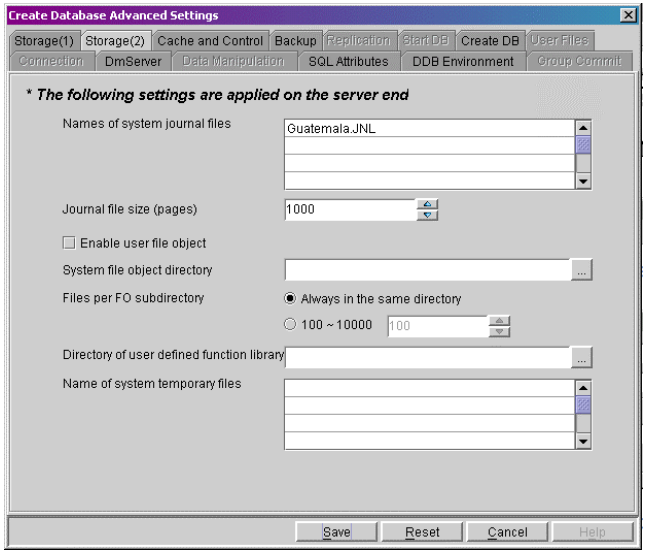


Figure 3-3The Storage (2) page of the Advanced Database Settings window

DATABASE DIRECTORY

The database administrator should have already set the database directory by following the procedure outlined in Section 3.1, however a different location may be selected for the default Database Directory at any time before database startup. The Database Directory is the default location where all operating system files that relate to the database (excluding the **dmconfig.ini** file) are stored, unless specified otherwise by the database administrator.

SYSTEM DATA FILE

The System Data File is an operating system file that contains database objects such as indexes, views, stored procedures, and synonyms. Objects under 4KB in length are stored in this file. The System Data file default path is (*Database Directory*)\(*database name*).SDB, but can be changed by the database administrator.

SYSTEM BLOB FILE

The System BLOB File is an operating system file that contains database objects such as indexes, views, stored procedures, and synonyms. Objects over 4KB in length are stored in this file. The System BLOB file default path is (*Database Directory*)\(*database name*).SBB, but can be changed by the database administrator.

USER DATA FILE

The User Data File is the operating system's file location for storing data. It contains all tables and data type records that belong to the database. The User Data file default path is (*Database Directory*)\(*database name*).DB, but can be changed by the database administrator.

NUMBER OF USER DATA FILE PAGES

The database administrator may specify how many file pages to initially allocate in the User Data File. This determines the initial operating system file size of the User Data File. Data pages are 4096 bytes long.

USER BLOB FILE

The user BLOB File is the operating system's file location for storing BLOB type data. It contains all BLOB type records that belong to the database. The User BLOB file default path is (*Database Directory*)\(*database name*).BB, but can be changed by the database administrator.

NUMBER OF USER BLOB FILE FRAMES

The database administrator may specify how many BLOB frames to initially allocate in the User BLOB File. This determines the initial operating system file size of the User BLOB File. The database administrator sets the BLOB frame size upon creation of the database. Refer to Section 3.2, Create Database: BLOB Frame Size for information on how to alter BLOB Frame Size.

PAGES TO EXTEND WHEN EXTENDING FILE

When all pages in the data file or BLOB file are full, DBMaker can automatically extend the number of pages or frames in the file to allow the database to grow. The **Number of Pages to Extend While Extending File** setting tells DBMaker how many pages or frames to add to the full file in the event that it is full. If the database administrator expects that the database will grow very quickly, then a higher number should be picked to lessen the frequency at which the file is appended. The default value is 20 pages.

JOURNAL FILES

The database administrator may specify one or more journal files for the database. If multiple journal files are created, DBMaker will store journal file entries starting with the first page of the first journal file listed, and then sequentially fill each journal file until the last file is filled. When the last journal file is filled, DBMaker will begin recording data over the old data starting with the first page of the first journal file, and so on. The size of each journal file can also be specified in the Journal File Size field.

FILE OBJECTS

The database administrator can determine if User File Objects (file objects stored outside of DBMaker's file system) can be used or not, and can specify the location of System File Objects (file objects stored within DBMaker's file system). DBMaker can automatically create subdirectories within the FO directory. Each subdirectory is filled to a threshold value with new file objects. When the threshold is reached, DBMaker creates a new FO subdirectory. The number in the **Files per FO subdirectory** field indicates this threshold value. To not create subdirectories and store all file objects in the FO directory select **Always in the same directory**.

USER DEFINED FUNCTION LIBRARY

User defined functions are compiled functions stored in a dynamic link library (DLL) that the user wants to be able to use in DBMaker. The DLLs stored in the Directory of User Defined Function Library are accessible to DBMaker and can be used in SQL Statements or ODBC applications. The default path for the Directory of User

Defined Function Library is (**DBMaker working directory**)**/shared/udf/**. Users can also enter a new path if a different one is desired. This can be achieved by typing the new path into the text field, or by clicking on the browse button (…) next to the text field. This setting corresponds to the **DB_LbDir** keyword in the **dmconfig.ini** file.

NAMES OF SYSTEM TEMPORARY FILES

System temporary files are used by DBMaker to store information about the database while the database is active. Up to eight temporary files may be specified. These files are removed when the database is shut down. If one or more filenames are written here, DBMaker will use them to store temporary information about the database. The default path for the system temporary file is (*Database Directory*)****. Users can also enter a new path if a different one is desired. This can be achieved by typing the new path into the text field, or by clicking on the browse button (…) next to the text field.

➲ **To set storage options:**

1. Click on the **Setup** button in the **Create Database** window.
2. Click the **Storage** (1) tab from the **Create Database Advanced Settings** window.
3. Enter a path into or select the browse button … next to the **Database Directory** field to indicate the location of the Database Directory
4. Enter a path into or select the browse button … next to the **Name of System Data File** field to indicate the location of the System Data File.
5. Enter a path into or select the browse button … next to the **Name of System BLOB File** field to indicate the location of the System BLOB File.
6. Enter a path into or select the browse button … next to the **Name of User Data File** field to indicate the location of the User Data File option.
7. Enter the number of pages to allocate from disk for each user data file in the **Number of User Data File Pages** field.
8. Enter a path into or select the browse button … next to the **Name of User BLOB File** field to indicate the location of the User BLOB File.
9. Enter the number of frames to allocate from disk for each user BLOB file in the **Number of User BLOB File Frames** field.

10. Enter the number of pages or frames to append each user file with when the file is full in the **Number of Pages to Extend while Extending File** field.
11. Click the **Storage** (2) tab from the **Create Database Advanced Settings** window.
12. Enter a full path or file name into the **Journal Files** field. More than one name may be entered. Typing just a file name places the journal files into the database directory.
13. Enter the number of pages to allocate from disk for each journal file in the **Journal File Size** (**pages**) field.
14. To enable the database to use external file objects, click on the **Enable User File Object** check box.
15. Enter a path into or select the browse button ... next to the **System File Object Directory** field to indicate the location of the System File Object Directory.
16. Select from **Files per FO Subdirectory**:

    To have DBMaker always store system file objects in the system file object directory, select **always in the same directory**

    To have DBMaker create subdirectories with a set number of file objects in each subdirectory, select the option button next to **100 ~10000** and enter the threshold number of file objects (between 100 and 10000) at which a subdirectory is declared full and a new one created.
17. Enter a path into or select the browse button ... next to the **Directory of User Defined Function Library** field to indicate the location of the user defined function dynamic link library
18. Enter up to eight full path nad file names into the **Name of System Temporary File** field to indicate the location of System Temporary Files.
19. Click the **Save** button.
20. Click the **Cancel** button to return to the **Create Database** window.

Setting Backup Options

The database administrator can specify how DBMaker should backup data for maximum security. The Backup page exactly corresponds to the Backup page in the

JConfiguration Tool. Please refer to the *JConfiguration Tool Reference* for more information on backup.

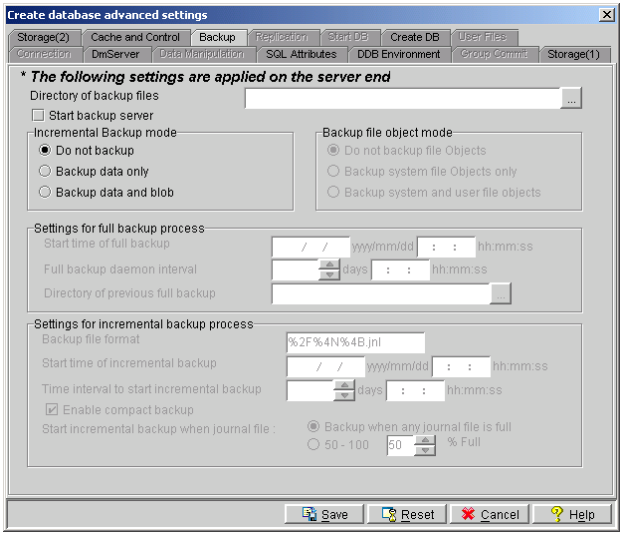


Figure 3-4 the backup page of the Create Database Advanced Settings window

DIRECTORY OF BACKUP FILES

The database administrator can specify the location for the backup files in the Directory of Backup Files field. To ensure against media failure, backup files should be stored on a disk separate from that of the current database.

START BACKUP SERVER

Starts the Backup Server to enable the use of full or incremental backup functions. The Backup Server must be started before the database is started.

INCREMENTAL BACKUP MODE

The database administrator can also select a mode for incremental backup process. Incremental backups copy only the journal files to the backup location. An incremental backup requires less time and resource to execute than a full backup, but

restoring to an incremental backup requires more time. The three modes for incremental backup are:

No Backup

Backup Data Only

Backup Data and BLOB

The modes do not affect the full backup process. After choosing a backup mode, be sure to enter a begin time and interval for the incremental backup daemon to activate it. The Backup Server must be activated for incremental backup to take place.

SETTINGS FOR INCREMENTAL BACKUP PROCESS

The settings for incremental backup process become available only when **Start Backup Server** has been enabled and the user has selected **Backup Data Only** or **Backup Data and BLOB**. Settings include backup journal file format, incremental backup start time and interval, Compact backup, and the threshold to initiate incremental backups. These settings are needed for the Incremental Backup Process to function properly

BACKUP FILE OBJECT MODE

The settings under the **Backup File Object Mode** effect how file objects are copied during the full backup process. Selecting **Do Not Backup File Objects** disables file backup during the full backup process. Selecting **Backup System File Objects Only** will result in system file objects being backed up during automatic full backups. Selecting **Backup System and User File Objects** will result in both system file objects and user file objects being copied to the backup directory during automatic full backups.

SETTINGS FOR FULL BACKUP PROCESS

These settings must be entered before attempting a full backup. Activating the Backup Server will enable the user to perform manual incremental and full backups. Full Backup is timely and resource consuming, but allows for faster restoration times. Enter a begin time and interval for the full backup daemon to activate it.

➲ **To set backup options:**

NOTE *The backup data location is specified by the* **Directory of Backup Files** *and should be on a separate disk for maximum security.*

1. Click on the **Setup** button in the **Create Database** window.
2. Click the **Backup** tab in the **Create Database Advanced Settings** window.
3. To start the backup server, select the **Start Backup Server** check box.
4. To enable full backups to be performed by the backup server,
   a) Enter a path into or select the browse button ... next to the **Directory of Backup Files** field to indicate the location of the backup directory.
   b) Indicate a date and a time in the **Start Time of Full Backup** time fields.
   c) Enter the number of days, hours, minutes, and seconds between each successive full backup in the **Full Backup Daemon Interval** days and time fields.
   d) Enter a path into or select the browse button ... next to the **Directory of Previous Full Backup** field to indicate a destination for the last full backup files.
5. To enable incremental backups to be automatically performed by the backup server,
   a) Be sure the backup server has been activated.
   b) Select a backup mode:

      To select data backups only, select the **Backup Data Only** option button.

      To backup data and BLOB files, select the **Backup Data and BLOB** option button.
   c) Enter a format for backup journal files in the **Backup File Format** field.
   d) Indicate a date and a time in the **Start Time of Incremental Backup** time fields.
   e) Enter the number of days, hours, minutes, and seconds between each successive full backup in the **Time Interval to Start Incremental Backup** time fields.
   f) To enable compact backup, click on the **Enable Compact Backup** check box

g) Incremental backups can be set to automatically execute when journal files have filled to a set percentage.

Select the **Backup when any Journal File is Full** option button to set incremental backups to execute when any journal file is filled.

Enter a value from 50 to 100 in the % **Full** field to set incremental backups to execute when any journal file is filled to the value entered.

6. Click the **Save** button.

7. Click the **Cancel** button to return to the **Create Database** window.

Setting dmServer Options

The database administrator can specify the port number when creating the database by following the procedure outlined above in section 3.1, or by selecting the **dmServer** tab in the create **Database Advanced Settings** window. In addition, the database administrator may also select server address, Network Encryption, the Idle Time-out Value, and whether server information should be saved to a log file. dmServer settings can be changed upon creation of the database or before the database is started. The **dmServer** page corresponds exactly to the **dmServer** page in the JConfiguration Tool. Please also refer to Chapter 4 of the *JConfiguration Tool Reference* for more information

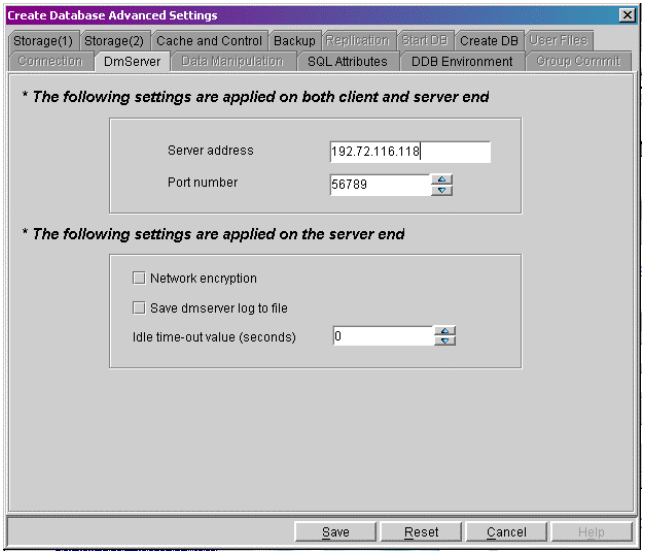


Figure 3-5 the dmServer page of the Advanced Database Settings window

SERVER ADDRESS

The database administrator can alter the server IP number or host name, if DNS (Domain Name Server) has been set up properly on the client machine. This setting must be the same for the server and all clients connecting to the database.

PORT NUMBER

Server and client machines identify a multi-user database on a TCP/IP network with the port number. It must be the same for the server an all client machines connecting to the same database. The Port Number can be any integer from 1025to 65535, and must be unique to each database on the server.

NETWORK ENCRYPTION

The database administrator can choose whether data accessed across a network is encrypted before being sent

SAVE DMSERVER LOG TO FILE

The database administrator can supervise connections and troubleshoot connection problems by enabling Save dmServer Log to file. All dmServer commands and output is saved to a log file under the database directory with the database name and the file extension .log.

IDLE TIME OUT VALUE

The database administrator can set a time limit for users connected to the database to execute a new transaction before they are automatically disconnected. Users exceeding the Idle Time out Value without showing any activity are disconnected and all resources allocated for that user are freed.

➲ **To set DmServer options:**

1. Click the **DmServer** tab in the **Create Database Advanced Settings** window. The **DmServer** page is displayed.
2. Enter the server address in the **Server Address** field.
3. Enter a port number in the **Port Number** field.
4. Select the **Network Encryption** check box to enable network encryption.
5. Select the **Save DmServer Log to File** check box to save the dmServer log.
6. Enter a time out value into the **Idle Time Out Value** check box.
7. Click the **Save** button.
8. Click the **Cancel** button to return to the **Create Database** window.

3.3 Using the Create Database Wizard

The Create Database Wizard is designed to help users define important database configuration parameters that must be set before database creation. The following chart outlines the program logic for the Create Database wizard. Refer to the separate steps in the procedure that follows the chart for a detailed description of its function.

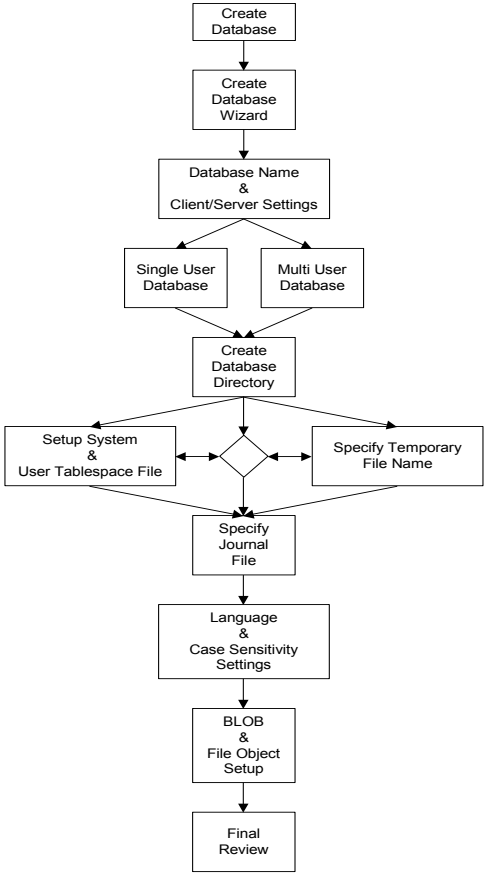
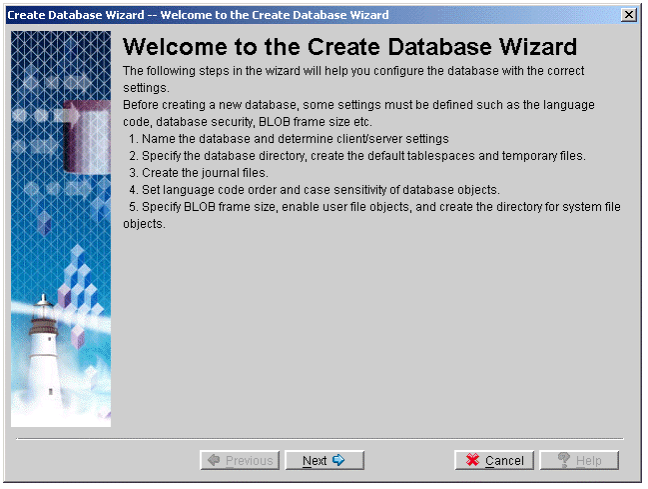


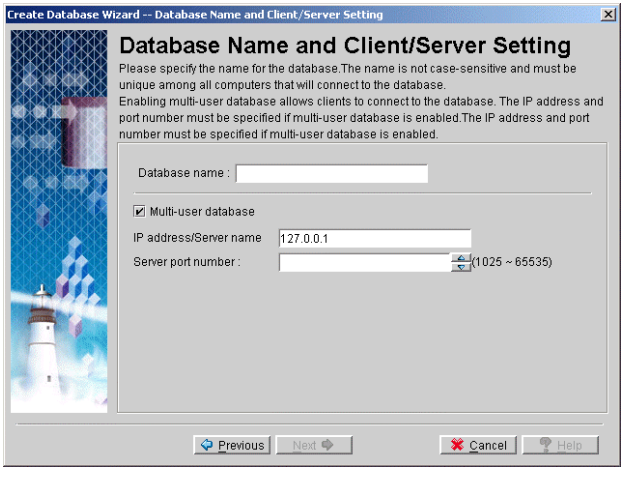
Figure 3-6 Program Logic for the Create Database Wizard

➲ **To create a database using the Create Database Wizard**

1. Select **Create Database** from the **Wizard** drop-down menu. The **Create Database Wizard** window will appear.



2. Click the **Next** button after reading the brief explanation of how the wizard will guide you through the process of creating a database. The **Database Name and Client/Server Setting** window appears.



3. Enter a name for the database in the **Database name** field.

4. The user now has the choice of creating a single user or a multi-user database.

a) To create a single user database uncheck the **Multi user database** check box. Now continue to step 7.

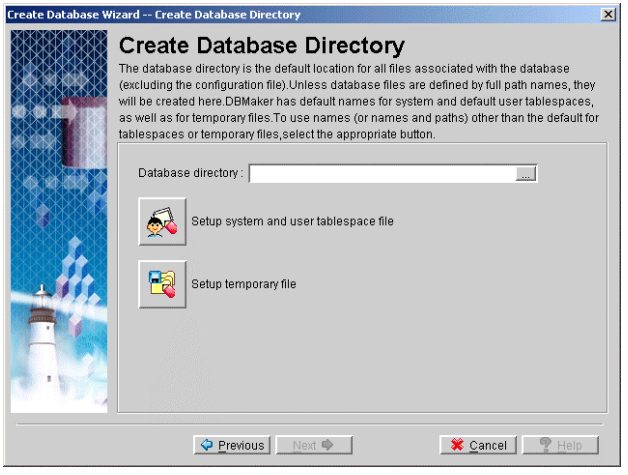
b) To create a multi-user database check the **Multi-user database** check box. Continue to step 5.

5. Enter the IP address or host name of the server in the **Server Address** field.

NOTE *If a host name is specified, ensure that Domain Name Server is set up on the server.*

6. Enter the port number of the server in the **Server Port Number** field. This must be an integer between 1025 and 65535.

7. Click the **Next** button. The **Create Database Directory** window appears.



8. Enter the database directory into the **Database Directory** field, or select the browse button ... . If the browse button is selected, the **Open Directory** dialog box will appear:

9. The default database directory path is *DBMaker\4.1\bin*. A new directory should be created for the database. To create a new database directory, click the browse button

10. In the **Select file** window, navigate to the desired root for the database directory.

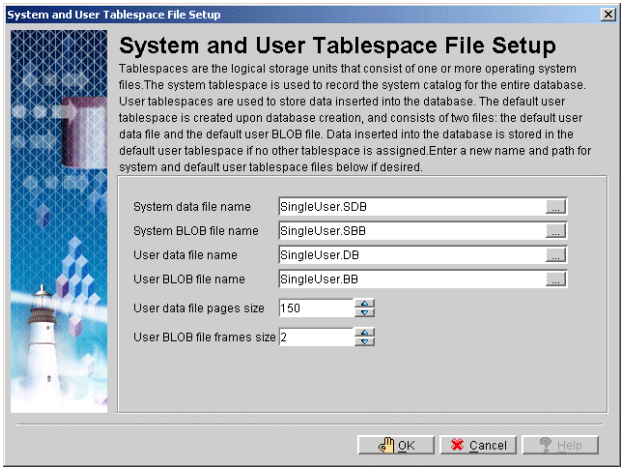
11. Click on the **Create New Directory** button to make a directory called **New Folder**.

12. Select the **New Folder** directory and enter the desired name for the database directory.

13. Select the new directory name from the list so that it appears in the **File name** field.

14. Click **OK**, the **Create Database Directory** window appears.

15. Click the **Setup system and user tablespace file** button to set the system and user tablespace files. The **System and User Tablespace File Setup** window appears.



16. Enter a path or select the browse button ... next to the **System data file name** field to indicate the location of the System Data File.

17. Enter a path or select the browse button ... next to the **System BLOB file name** field to indicate the location of the System BLOB File.

18. Enter a path or select the browse button ... next to the **User data file name** field to indicate the location of the User Data File option.

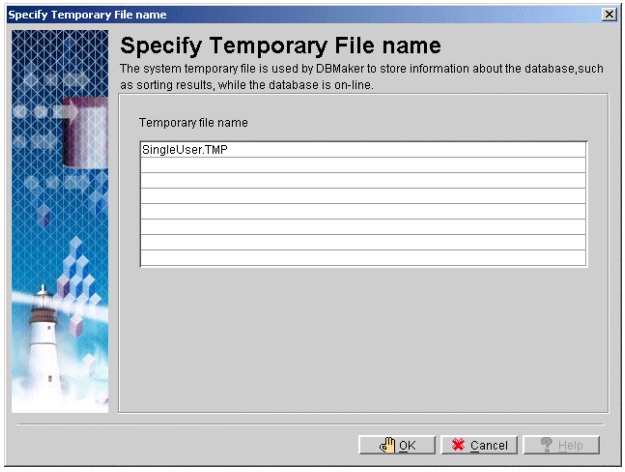
19. Enter a path or select the browse button ... next to the **User BLOB file name** field to indicate the location of the User BLOB File.

20. Enter the number of pages to allocate from disk for each user data file in the **User data file pages size** field.

21. Enter the number of frames to allocate from disk for each user BLOB file in the **User BLOB file frames size** field.

22. Click **OK**. The **Create Database Directory** window reappears.

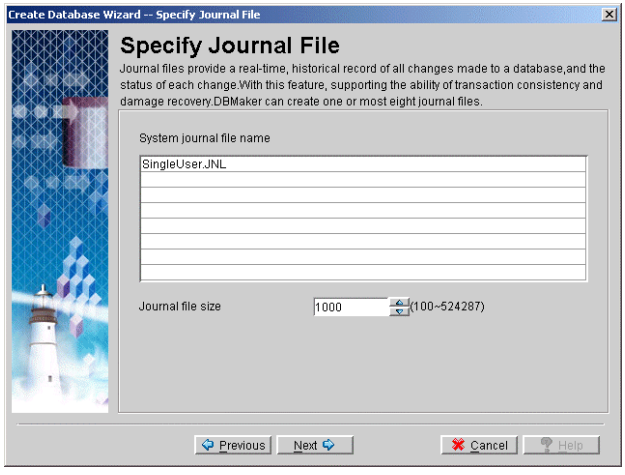
23. Click the **Setup temporary file** button to set the temporary file name. The **Specify Temporary File Name** window appears.



24. Double click in one of the fields to enter a name for the temporary system file.

25. Enter a name for the temporary system file and click **OK**. The **Create Database Directory** window reappears.

26. Click **Next**. The **Specify Journal File** window appears.

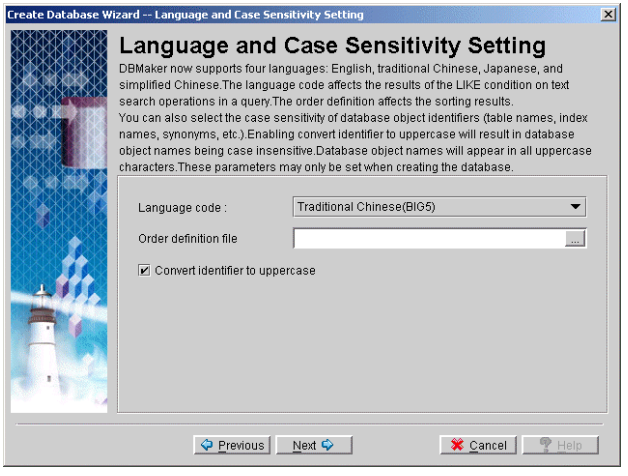


27. Double click a field to enter the journal file name.

28. Enter the journal file's name.

29. Set the journal file size. Valid values are between 100 and 524287, with the default value set at 1000.

30. Click **Next**. The **Language and Case Sensitivity Setting** window appears.



31. To change the character set the database will use for CHAR type data, select one of the four options from the **Language code** menu.

Select ASCII Compatible to use ASCII code.

Select Big5 Compatible to use traditional Chinese code.

Select JIS + Half Corner Compatible to use Japanese code.

Select GB code compatible to use simplified Chinese code.

Select **Latin code** (**ISO-8859-1**) compatible to use Latin – 1 code.

Select **Latin code** (**ISO-8859-2**) compatible to use Latin – 1 code.

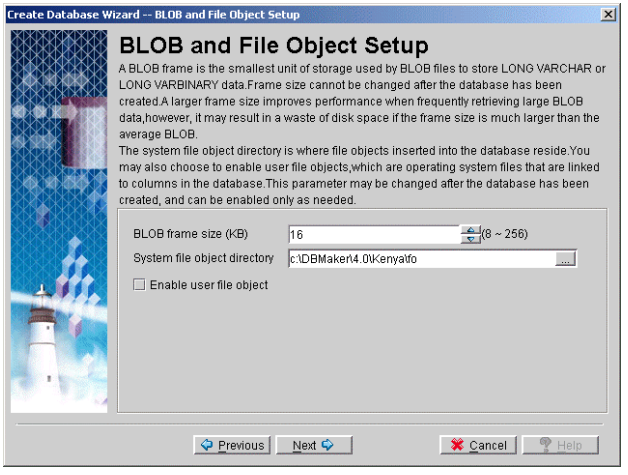
Select **Cyrillic code** (**ISO-8859-5**) compatible to use Cyrillic code.

Select or **Greek code** (**ISO-8859-7**) compatible to use Greek code.

32. Enter a path or select the browse button ... next to the **Order definition file** field to indicate the location of the order definition file.

33. To make all database objects case insensitive check the **Convert identifier to uppercase** check box.

34. Click **Next**. The **BLOB and File Object Setup** window appears.

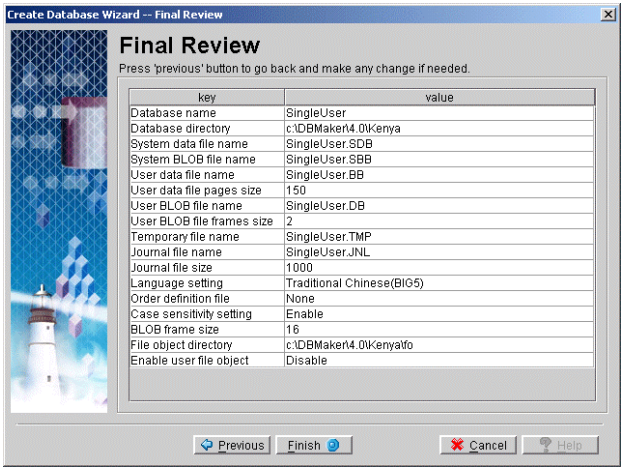


35. Enter a BLOB frame size in bytes in the **BLOB frame size (KB)** field.

36. Enter a path or select the browse button ... next to the **System file object directory** field to indicate the location of the database's system file objects.

37. Check the **Enable user file object** check box to allow user file objects.

38. Click **Next**. The **Final Review** window appears.



39. Check over the database settings. If any of the settings need to be changed, click **Previous** until reaching the appropriate window and make the changes. Then return to the **Final Review** window.

40. Click the **Finish** button when satisfied with the database settings.

4 Starting a Database

A database may be started with or without the Start Database wizard. The Start Database wizard provides an interface that explains every step of configuring a DBMaker database for startup.

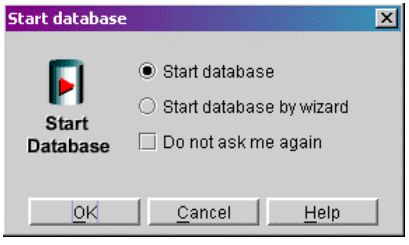
Many database performance characteristics can be manipulated by changing database configuration parameters before startup. The following sections describe how to start a database with the default DBMaker settings, with your own settings, or with the Start Database wizard.

4.1 Starting a Database: Basics

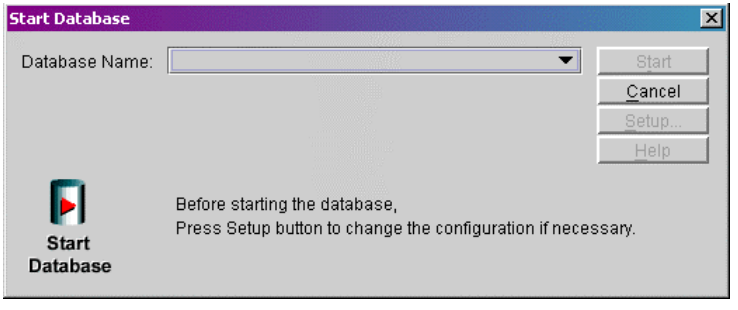
You may choose not to set any of the advanced settings if you are new to using DBMaker or your database has just been created and there is no need to adjust critical performance parameters

➲ **To start a database:**

1. To start a database, select **Start Database** from the main console. The **Start Database** window appears.

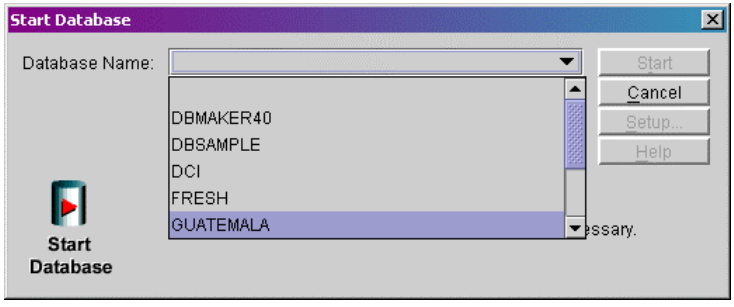


2. To use the wizard to start the database, select **Start database by wizard** and then refer to Section 4.3. Select **Start database** to avoid using the wizard, the following dialog box will appear.



NOTE *You can also select a database from the Database drop-down menu.*

3. Select a database from the **Database Name** menu.



4. Choose to start the database or alter advanced database settings for the new database. To alter advanced settings, click **Setup** and refer to section 4.2. To start the new database, click **Start**. The dmServer application will start and the following message will be displayed. If any errors occur, they will be displayed in the dmServer window.

4.2 Starting a Database: Advanced Settings

Database administrators should ensure that all necessary changes to settings are made to a database before startup. Advanced settings include:

Storage location of the database's operating system files

Upper memory caching of data pages and system control

Backup type, frequency and location

Location and frequency of table or database replication

Database startup mode

User defined file names

Network connection behavior

SQL command upper memory caching, CHAR data output padding, system File Object mapping, Cursor Behavior, and Browse Mode

Date and time formatting for SQL statements, stored procedure settings

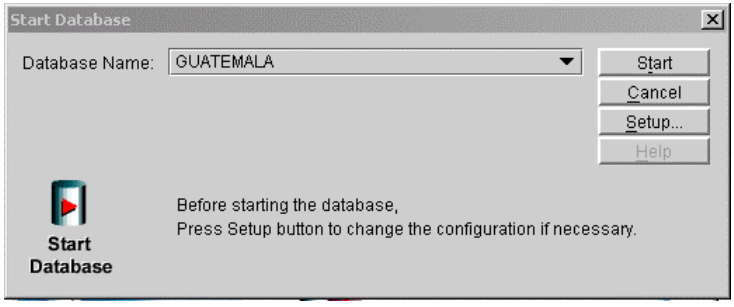
Distributed database behavior

Group commit settings

The settings in the JServer Manager correspond exactly with the settings in the JConfiguration Tool. To learn more about how these settings affect the management of databases, refer to the *JConfiguration Tool Reference*.

➲ **To start a database with advanced settings:**

1. Select **Start Database** from the main console, and select a database to start as outlined in section 4.1.



2. Click the **Setup** button. The **Start Database Advanced Settings** dialog box is displayed.

3. Enter advanced setting options (see the following procedures):

   Setting Storage Options (refer to Chapter 3, *Setting Storage Options*)

   Setting Cache and Control Options

   Setting Backup Options

   Setting Replication Options

   Setting Start Options

   Setting User File Options

   Setting dmServer Options

   Setting SQL Attributes

   Setting Distributed Database (DDB) Environment Options

   Setting Group Commit Options

4. After you have altered advanced database settings for the new database, click the **Start** button.

Setting Cache and Control Options

Cache and control settings are used to manipulate the amount of memory used by an active database, control the maximum number of users that can access the active database, and control locks. Adjusting these settings influences database performance. For more information on performance tuning, refer to the *Database Administrator's Guide.* The **Cache and Control** page exactly corresponds to the **Cache and Control** page in the JConfiguration Tool. Please refer to the *JConfiguration Tool Reference* for more information on Cache and Control.

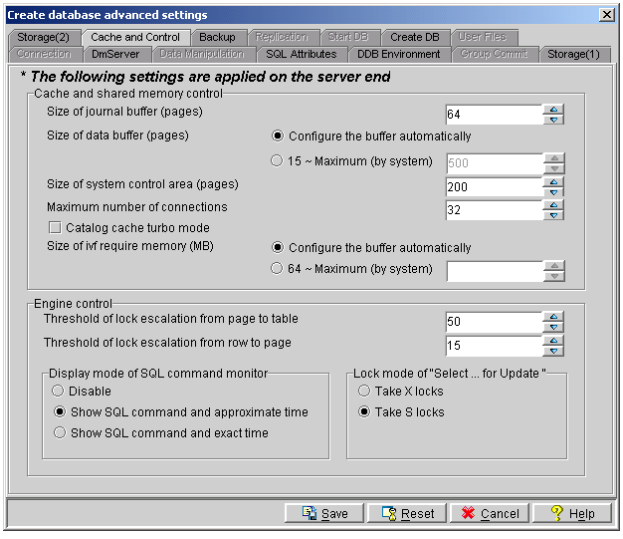


Figure 4-1 The Cache and Control page of the Start Database Advanced Settings window

This field sets the maximum amount of memory that DBMaker will reserve to cache the buffer from queries to an inverted file full text index.

SIZE OF JOURNAL BUFFER

The Journal buffer stores the most recently used journal blocks as pages. It is a component of the Database Communication and Control Area (DCCA). With enough journal buffer pages, the time required to write journal blocks to disk when data is updated and the time required to read journal blocks from disk when a transaction is rolled back is reduced.

SIZE OF DATA BUFFER

The data buffer stores the most recently used data blocks as pages. It is a component of the Database Communication and Control Area (DCCA). DBMaker uses the most recent statistics values of schema objects to determine which data pages to keep in the buffer. Increasing or decreasing the number of data pages in the data buffer has a dramatic effect on database performance.

SIZE OF SYSTEM CONTROL AREA

The System Control Area (SCA) is where concurrency control blocks are stored. It is a component of the Database Communication and Control Area (DCCA). Large transactions requiring many lock resources will need a larger SCA to function properly. If too many resources are being allocated for concurrency control, the SCA can be enlarged, or the threshold of lock escalation can be decreased.

MAXIMUM NUMBER OF CONNECTIONS

The database administrator may control the maximum number of simultaneous connections that may be made to the database.

CATALOG CACHE TURBO MODE

Enabling the **Catalog Cache Turbo Mode** extends the lifetime of the catalog cache in the SCA. The default setting is *disabled*

SIZE OF IVF REQUIRE MEMORY

This field sets the maximum amount of memory that DBMaker will reserve to cache the buffer from queries to an inverted file full text index.

THRESHOLD OF LOCK ESCALATION FROM PAGE TO TABLE

When the number of locks on pages in the same table exceeds the lock escalation threshold from page to table, DBMaker will automatically escalate the lock to a table lock. When the number of locks on rows in the same table exceeds the lock escalation threshold from row to page, DBMaker will automatically escalate the lock to a page lock. This number specifies the *Lock Escalation Threshold* for escalating a page lock to a table lock. When the number of locks on pages in the same table exceeds the lock escalation threshold, DBMaker will automatically escalate the lock to a table lock. The default value is 50 pages.

THRESHOLD OF LOCK ESCALATION FROM ROW TO PAGE

This number specifies the *Lock Escalation Threshold* for escalating a row lock to a page lock. When the number of locks on rows in the same table exceeds the lock escalation threshold, DBMaker will automatically escalate the lock to a page lock. The default value is 15 rows.

DISPLAY MODE OF SQL COMMAND MONITOR

Display Mode affects the display content of the **SQL_CMD** and **TIME_OF_SQL_CMD** columns in the **SYSUSER** system table. JDBA Tool Users can view information on all users accessing the database with the Database Monitor function. Session Information displays users currently connected to the database. The columns *Current SQL command* and *Time of Current SQL Command* display the most recent SQL transaction committed by a user and the time of execution, respectively. **Display Mode of SQL Command Monitor** sets the configuration for how these attributes are displayed. No SQL commands are shown if **Disable** is selected. The most recent SQL command executed by the user and its approximate time of execution are shown if **Show SQL command and approximate time** is selected. The most recent SQL command executed by the user and its exact time of execution are shown if **Show SQL command and exact time** is selected. Displaying the exact time of execution uses more CPU resources and slows down the database. The default mode is **Show SQL command and approximate time**.

LOCK MODE

This setting specifies the lock behavior of all "select … for update" statements at the server site. By default, DBMaker takes S locks on the result set of a "select … for update" statement. For some applications, users may want to take exclusive locks on objects that are indicated by a "select … for update". Selecting the **Take X locks** radio button denotes DBMaker will take X locks on the result set of all "select … for update" statements.

➲ **To set cache and control options:**

1. Click on the **Setup** button in the **Start Database** window. The **Start Database Advanced Settings** window appears.
2. Click the **Cache and Control** tab in the **Start Database Advanced Settings** window.
3. Enter the number of journal buffer pages to allocate from memory in the **Size of Journal Buffer** field.
4. Enter the number of data buffer pages to allocate from memory in the **Size of Data Buffer** field.
5. Enter the number of pages to allocate from memory for the System Control Area in the **Size of System Control Area** field.
6. Enter the maximum number of users that can connect to the database in the **Maximum number of Connections** field.
7. To enable catalog cache turbo mode click on the check box next to **Catalog Cache Turbo Mode**.
8. Enter the number of pages at which a page lock will escalate to a table lock in the **Threshold of lock Escalation form Page to Table** field.
9. Enter the number of rows at which a row lock will escalate to a page lock in the **Threshold of lock Escalation form Row to Page** field.
10. To change the display content of the SQL_CMD and TIME_OF_SQL_CMD columns in the SYSUSER system table, select one of the three option buttons.

    Select **Disable** to not display SQL Commands.

    Select **Show SQL command and approximate time** to display the approximate time of SQL command execution.

Select **Show SQL command and exact time** to display the exact time of SQL command execution.

11. To change the lock mode of "Select …for Update" SQL commands, select one of the two option buttons.

    Select **take X locks** to take exclusive locks on "Select …for Update" SQL commands.

    Select **take S locks** to take shared locks on "Select …for Update" SQL commands.

12. Click the **Save** button to save al changes to the **dmconfig.ini** file.

13. Click the **Cancel** button to return to the **Start Database** window, and click **Start** to start the database.

Setting Backup Options

The database administrator can specify how DBMaker should backup data for maximum security. The Backup page exactly corresponds to the Backup page in the JConfiguration Tool. Please refer to the *JConfiguration Tool Reference* for more information on backup.

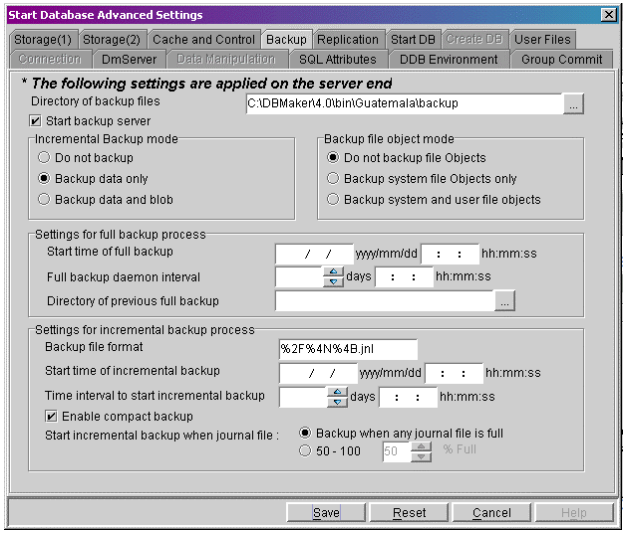


Figure 4-2 the backup page of the Create Database Advanced Settings window

DIRECTORY OF BACKUP FILES

The database administrator can specify the location for the backup files in the Directory of Backup Files field. To ensure against media failure, backup files should be stored on a disk separate from that of the current database.

START BACKUP SERVER

Starts the Backup Server to enable the use of full or incremental backup functions. The backup server must be started before the database is started.

INCREMENTAL BACKUP MODE

The database administrator can also select a mode for incremental backup process. Incremental backups copy only the journal files to the backup location. An incremental backup requires less time and resource to execute than a full backup, but restoring to an incremental backup requires more time. The three modes for incremental backup are

No Backup

Backup Data Only

Backup Data and BLOB.

The modes do not affect the full backup process. After choosing a backup mode, be sure to enter a begin time and interval for the incremental backup daemon to activate it. The Backup Server must be activated for incremental backup to take place.

Backup File Object Mode

The settings under the **Backup File Object Mode** effect how file objects are copied during the full backup process. Selecting **Do Not Backup File Objects** disables file backup during the full backup process. Selecting **Backup System File Objects Only** will result in system file objects being backed up during automatic full backups. Selecting **Backup System and User File Objects** will result in both system file objects and user file objects being copied to the backup directory during automatic full backups.

Settings for Incremental Backup Process

The settings for incremental backup process become available only when **Start Backup Server** has been enabled and the user has selected **Backup Data Only** or **Backup Data and BLOB**. Settings include backup journal file format, incremental backup start time and interval, Compact backup, and the threshold to initiate incremental backups. These settings are needed for the Incremental Backup Process to function properly.

Settings for Full Backup Process

These settings must be entered before attempting a full backup. Activating the Backup Server will enable the user to perform manual incremental and full backups. Full Backup is timely and resource consuming, but allows for faster restoration times. Enter a begin time and interval for the full backup daemon to activate it.

➲ **To set backup options:**

NOTE *The backup data location is specified by the* Directory of Backup Files *and should be on a separate disk for maximum security.*

1. Click **Setup** in the **Start Database** window. The **Start Database Advanced Settings** window appears.
2. Click the **Backup** tab in the **Start Database Advanced Settings** window.
3. To start the backup server, select the **Start Backup Server** check box.
4. To enable full backups to be performed by the backup server,
   a) Enter a path or select the browse button ... next to the **Directory of Backup Files** field to indicate the location of the backup directory.
   b) Indicate a date and a time in the **Start Time of Full Backup** time fields.
   c) Enter the number of days, hours, minutes, and seconds between each successive full backup in the **Full Backup Daemon Interval** time fields.
   d) Enter a path or select the browse button ... next to the **Directory of Previous Full Backup** field to indicate a destination for the last full backup files.
5. To select what types of file objects are backed up during the backup process:
   a) Select **Do not backup file objects** to prevent file objects from being backed up.
   b) Select **Backup system file objects only** to only back up system file objects.
   c) Select **Backup system and user file objects** to back up all file objects.
6. To enable incremental backups to be automatically performed by the backup server:
   a) Select the **Backup Only** option button to select data backups only, or Select the **Backup Data and BLOB** option button to backup data and BLOB files.
   b) Enter a format for backup journal files in the **Backup File Format** field.
   c) Indicate a date and a time in the **Begin Time of Incremental Backup** time fields.
   d) Enter the number of days, hours, minutes, and seconds between each successive full backup in the **Time Interval to Start Incremental Backup** time fields.
   e) To enable compact backup, click on the **Enable Compact Backup** check box
   f) Incremental backups can be set to automatically execute when journal files have filled to a set percentage.

Select the **Backup when any Journal File is Full** option button to set incremental backups to execute when any journal file is filled.

Enter a value from 50 to 100 in the **% Full** field to set incremental backups to execute when any journal file is filled to the value entered.

7. Click **Save**.
8. Click **Cancel** to return to the **Start Database** window, and click **Start** to start the database.

Setting Replication Options

Asynchronous Table Replication (ATR) periodically writes data from the distributor database to remote tables of the replicated database based on a schedule. The Asynchronous Table Replication Daemon must be activated before the database is started for ATR to work properly. Database replication periodically writes all changes made to the primary database onto target databases, which are essentially read only copies of the primary database located on remote servers.

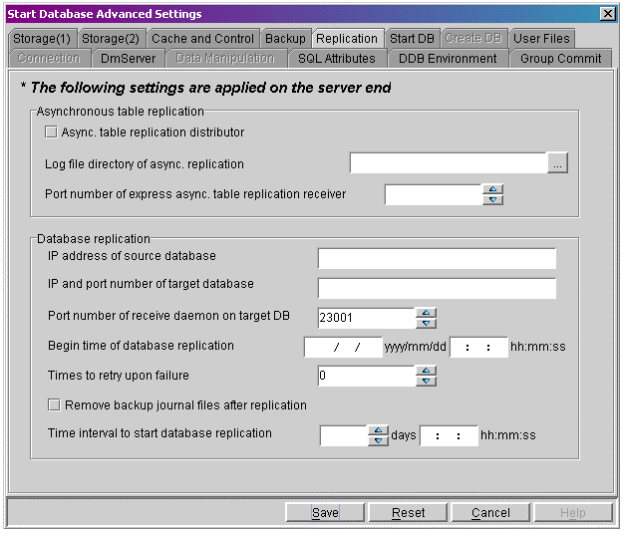


Figure 4-3 The Replication page of the Start Database Advanced Settings window

ENABLING ATR DISTRIBUTOR

The schedule of the distributor daemon is dependent on the remote (destination) database and is specified using the SQL command CREATE SCHEDULE, or using the Create Replication Schedule wizard in the JDBA Tool. Refer to the *SQL Command and Function Reference* for more information on SQL commands, or the *JDBA Tool User's Guide* for more information on using the JDBA Tool.

SETTING THE PORT NUMBER OF THE EXPRESS ATR RECEIVER

This setting is used only when an express asynchronous table replication has been created. Express ATR uses a different method from ODBC calls to update target tables. All source and destination tables that share the same tables must have the same Express ATR port number. Only databases that use the DBMaker engine can use this feature. For more information, refer to the *DBMaker Database Administrators Guide*.

SETTING THE ATR LOG FILE DIRECTORY

Asynchronous table replication error logs are associated with only the source database, and by default are located under the source database directory. Users should not manually remove the replication log files. The default path for ATR Log files is (**database name**)/(**TRPLOG**) and is automatically created by DBMaker. The total length of the ATR log file directory path must not exceed 256 characters in length.

IP ADDRESS OF SOURCE DATABASE

This field, used for database replication, specifies the port number of the RP_RECV daemon at the slave database. It must be different from the **port number** specified on the dmServer page of the slave database's **dmconfig.ini** file and the same as the port number specified by the **IP and Port Number of target database** field of the primary database. This field should be left blank on the primary database server. There is no default value.

IP AND PORT NUMBER OF TARGET DATABASES

Entering the **Port Number of Receive Daemon on Target DB** specifies the location of slave databases to the primary database. These numbers are specified on the primary

database side of the database replication, should correspond to the values in the **Server Address** and **Port Number of Receive Daemon on Target DB** fields specified in each target (slave) databases **dmconfig.ini** file. DBMaker will support up to eight slave databases for each primary database. The format should follow the following syntax:

```
address[:port number] {, address[:port number]}
```

Where the address is the IP number or host name of each slave (target) database, and the port number is the replication receive daemon port number specified in the **dmconfig.ini** file of each slave database. The default port number is 23001. The information for each slave database can be separated by a comma or blank space.

➲ **Example:**

A typical representation of IP address and port numbers of slave databases in the **dmconfig.ini** file:

```
192.168.9.222:5100, Server2:5101, Server3
```

There are three slave databases. One is 192.168.9.222 with port number 5100, another is Server2 with port number 5101, and the other is Server3 with default port number 23001.

PORT NUMBER OF RECEIVE DAEMON ON TARGET DATABASE

This field is used only if the database is a slave database. It specifies the port number of the Replication Receive Daemon at the slave database. It must be different from the Port Number set in the dmServer page that is used by the slave database and the same as the port number specified in the **Port Number of Receive Daemon on Target DB** field in the primary database. The default value is 23001.

SETTING THE REPLICATION START TIME

This value indicates at what time database replications shall begin. JConfiguration Tool automatically enters values into the first two spaces of the **yyyy** field and the first space of the **hh** field if values are entered into other fields. This setting corresponds to the **RP_BTime** keyword in the **dmconfig.ini** file. The default value is the starting time of the primary database.

TIMES TO RETRY ON FAILURE

This value specifies how many times DBMaker will try to connect to remote databases during database replication after a network failure.

REMOVE BACKUP JOURNAL FILES AFTER REPLICATION

Enabling **Remove Backup Journal Files after Replication** specifies DBMaker to delete journal backup files after sending them to remote databases during database replication.

SETTING THE REPLICATION DAEMON INTERVAL

The number in the **Time Interval to Start Database Replication** combo box specifies the time interval at which database replication occurs in days. Next to the combo box is a field for time input, which specifies the time interval in hours, minutes, and seconds. The total time interval is determined by adding the two values together, so inputting 1 into the **days** field and inputting 12:00:00 into the **hh:mm:ss** field would cause an incremental backup to be made every day and a half.

➲ **To set replication options:**

1. Click **Setup** in the **Start Database** window. The **Start Database Advanced Settings** window appears.
2. Click the **Replication** tab in the **Start Database Advanced Settings** window.
3. To enable asynchronous table replication (ATR):
   a) Enable the check box next to **Async. Table Replication Distributor**.
   b) If using express ATR, enter a value into the **Port Number of Express Async. Table Replication Receiver.**
   c) Enter a path into or select the browse button next to the **Log File Directory of Async. Replication** field to indicate a location for the ATR log file.
4. To enable database replication:
   a) Enter the **IP Address of Source Database**.
   b) If you are modifying settings for the primary (source) database, enter the **IP and Port Number of Target Database**.

c) If you are modifying settings for the target (slave) database, enter the **Port Number of Receive Daemon on Target DB**.

d) Enter a date and time into the **Begin Time of Database Replication** time fields.

e) Enter a value into the **Times to Retry upon Failure** field

f) If desired, enable **Remove Backup Journal Files after Replication**

g) Enter the number of days, hours, minutes, and seconds between each successive database replication in the **Time Interval to Start Database Replication** time fields.

5. Click **Save**.

6. Click **Cancel** to return to the **Start Database** window, and click **Start** to start the database.

Setting Start Database Options

There are six start-up modes for starting a database using JServer Manager.

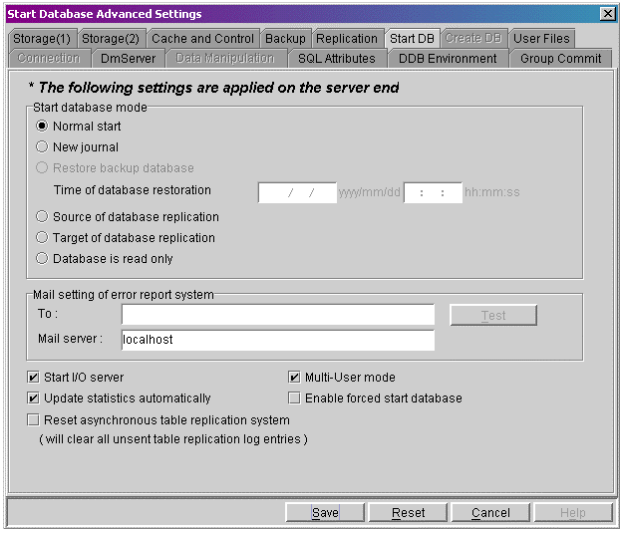


Figure 4-4 The Start Database page of the Start Database Advanced Settings window

NORMAL START MODE

This mode starts up a system normally. If the database crashed last time, DBMaker will perform crash recovery automatically to bring the database to a consistent and stable state.

NEW JOURNAL MODE

This mode starts up a system normally, but creates new journal files from names specified in the **Names of System Journal Files** on the **Storage** page. All old records will be overwritten if the previous journal file names are kept. This setting must be selected if the user wants to change the journal file size, add more journal files, or change the journal file name. It is recommended to perform an incremental or full backup before selecting this option

RESTORE BACKUP DATABASE

This mode uses the backed up database files (including the journal file) to start the database. DBMaker will use the incremental backup files to rollover the operations up to the point in time specified in the fields indicated by the **Time of Database Restoration** option. When this option is selected, Server Manager prompts the user to copy backup files to the database and then prompt the user for each incremental file that is to be used to roll the database over upon startup. If no value is specified or the date specified is later than the time of the last incremental backup, the Time of Database will be restored to its default value.

SOURCE OF DATABASE REPLICATION

This mode is used for database replication. Starting up a database with this mode makes it a primary (source) database. Database replication may be performed manually or according to a schedule, and is set using the JDBA Tool. Refer to the *JDBA Tool User's Guide* for more information on Database Replication.

TARGET OF DATABASE REPLICATION

This mode is used for database replication. Starting up a system with this mode makes it a slave database. The database will be read only and must have the proper settings

for database replication. Database replication may be performed manually or according to a schedule, and is set using the JDBA Tool. Refer to the *JDBA Tool User's Guide* for more information on Database Replication.

DATABASE IS READ-ONLY

This mode starts up a system normally, but the database is read-only, providing only read privilege to users. Starting a primary database in read-only mode prohibits users from modifying it.

MAIL SETTING OF ERROR REPORT SYSTEM

DBMaker creates an error report log of errors that occur during normal operation. These errors may not interfere in the performance of the database and may not be noticeable to the database administrator unless the error log is regularly checked. Alternatively, a destination e-mail address and SMTP server for relaying the mail may be specified and DBMaker will automatically send error messages to the recipient as soon as they occur.

START I/O SERVER

The primary purpose of the I/O server daemon is to manage data page buffers. The checkpoint daemon periodically clears the data page buffers and writes data to disk. Enabling the I/O server tells whether DBMaker should turn the I/O and checkpoint daemon on or off. The default setting is *enabled.*

MULTI USER MODE

To allow more than one user to access a database, enable Multi-user Mode. Client machines will be able to connect to the server over a network.

UPDATE STATISTICS AUTOMATICALLY

DBMaker keeps statistical information about database objects in the system tables. Periodically the database needs to be read and statistics recalculated and written to the system tables to ensure efficient database operation. Enabling **Update statistics**

automatically ensures that DBMaker automatically periodically recalculates database statistics according to an internal scheduling daemon.

ENABLING FORCED START DATABASE

This mode forces a database to start even if a serious error occurs while starting the database (such as a crash).

RESET ASYNCHRONOUS TABLE REPLICATION SYSTEM

Reset Asynchronous Table Replication System clears all unsent table replication log entries upon startup.

➲ **To set start database options:**

1. Click **Setup** in the **Start Database** window.
2. Click the **Start Database** tab in the **Start Database Advanced Settings** window.
3. Select a database startup mode by clicking on one of the following option buttons available in the **Start Database Mode** field.

   Normal Start

   New Journal

   Restore Backup Database (Enter the restoration date and time in the **Time of Database Restoration** time fields)

   Source of Database Replication

   Target of Database Replication

   Database is Read Only
4. To enable the database to have only one connection, disable the **Multi-User Mode** option.
5. To allow the database to be started even if there are errors, enable the **Enable Forced Start Database** option.
6. To clear all unsent table replication log entries upon startup enable the **Reset Asynchronous Table Replication System** option.

7. Click **Save**.
8. Click **Cancel** to return to the **Start Database** window, and click **Start** to start the database.

Setting User File Options

DBMaker allows users to create new data files or BLOB files and add them to a tablespace when the original tablespace is filled. Users specify a logical file name (without full path name) when creating a file. However, users can map this logical file name to a physical file name that is used by the operating system to access the file.

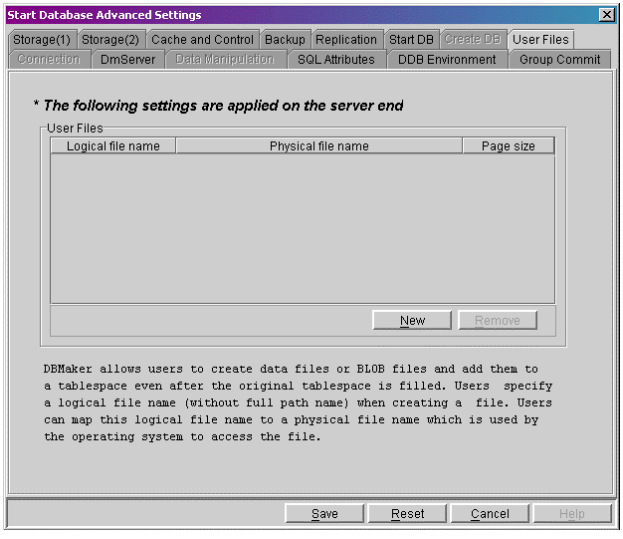
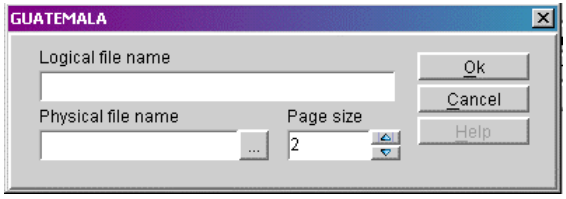


Figure 4-5 The User Files page of the Start Database Advanced Settings window

➲ **To manage files in tablespaces:**

1. Click **Setup** in the **Start Database** window.
2. Click the **User Files** tab in the **Start Database Advanced Settings** window.
3. To create a user-defined file click the **New** button on the bottom of the user files field. A dialog box appears displaying the name of the database:



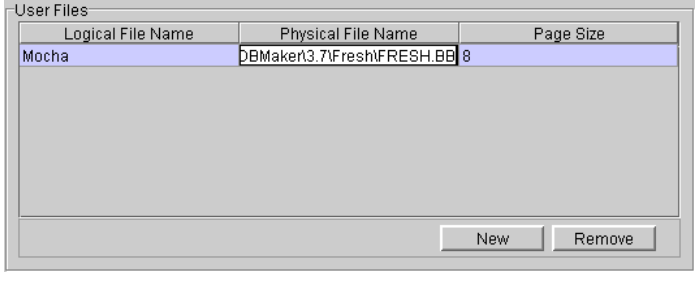
4. Enter the logical file name in the **Logical file name** field (this is the path referenced by SQL commands and by DBMaker to reference your data).

5. Enter a Physical File Name by entering a new path into the **Physical File Name** field, or by clicking on the browse button ... next to the **Physical File Name** field. This is the operating system path. The maximum number of characters for the Physical File Name path is 79.

6. Enter a Page Size into the **Page size** combo box. This indicates the number of data pages allocated for the User-Defined File. The range of values for page size is 2 to 524,287.

7. Click **OK**.

8. A new user defined file will appear in the **User Files** field. The logical file name, physical file name, and page size can be changed after the user-defined file has been created. Double clicking on the text allows the user to edit it. This is useful if data files are moved within the operating system. DBMaker will not recognize a change in the physical file name unless it is specified here.



9. To delete a user-defined file click **Remove**. DBMaker will no longer recognize the logical file name.

10. Click **Save**.

11. Click **Cancel** to return to the **Create Database** window, and click **Start** to start the database.

Setting DmServer Options

DmServer settings may be changed before a database is started. The settings correspond exactly to the **DmServer** page in the JConfiguration Tool. Please refer to Chapter 4 of the *JConfiguration Tool Reference* for more information

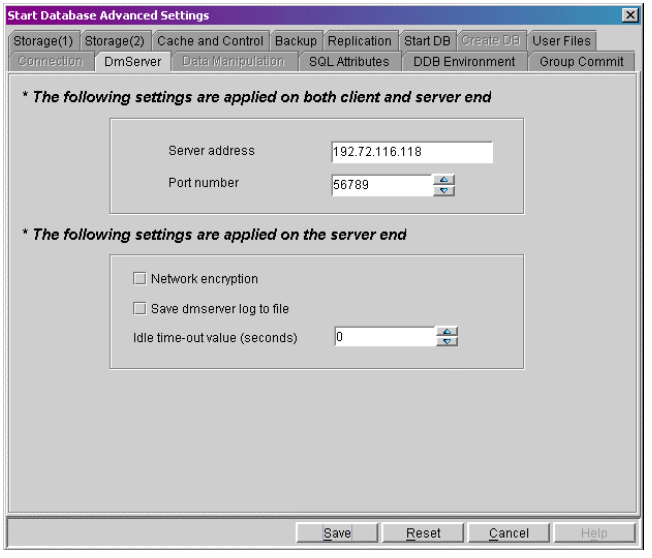


Figure 4-6 the dmServer page of the Start Database Advanced Settings window

SERVER ADDRESS

The database administrator can alter the server IP number or host name, if DNS (Domain Name System) has been configured properly over the network. This setting must be the same for the server and all clients connecting to the database.

PORT NUMBER

Server and client machines identify a multi-user database on a TCP/IP network with the port number. It must be the same for the database server and all client machines

connecting to the same database. The Port Number can be any integer from 1025to 65535, and must be unique to the database being accessed.

NETWORK ENCRYPTION

The database administrator can choose whether data accessed across a network is encrypted before being sent.

SAVE DMSERVER LOG TO FILE

The database administrator can supervise connections and troubleshoot connection problems by enabling **Save DmServer Log to file**. All DmServer commands and output is saved to a log file under the database directory with the database name and file extension .log.

IDLE TIME OUT VALUE

The database administrator can set a time limit for users connected to the database to execute a new transaction before they are automatically disconnected. Users exceeding the Idle Time out Value without showing any activity are automatically disconnected. In addition, all processor resources allocated for that user are freed.

➲ **To set DmServer options:**

1. Click the **DmServer** tab in the **Create Database Advanced Settings** window. The **DmServer** page is displayed.
2. Enter the server address in the **Server Address** field.
3. Enter a port number in the **Port Number** field.
4. Select the **Network Encryption** check box to enable network encryption.
5. Select the **Save dmServer Log to File** check box to save the dmServer log.
6. Enter a time out value into the **Idle Time Out Value** check box.
7. Click **Save**.
8. Click **Cancel** to return to the **Create Database** window, and click **Start** to start the database.

Setting SQL Attributes

SQL date and time input and output formats can be modified, as well as the location of stored procedure related files.

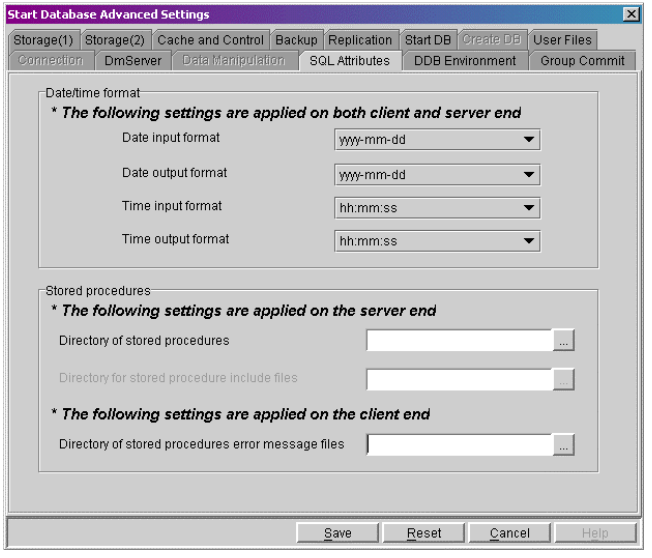


Figure 4-7 The SQL Attributes page of the Start Database Advanced Settings window

DATE / TIME FORMAT

Date and time input and output formats for SQL statements can be selected from the available formats located in the drop-down menus.

DIRECTORY OF STORED PROCEDURES

The directory shown here specifies the path stored procedure files are placed in. The stored procedure files include the dynamic linking library files and temporary files generated during stored procedure creation.

DIRECTORY OF STORED PROCEDURE ERROR MESSAGE FILES

The directory shown here specifies the path stored procedure log files are placed in. The stored procedure log files include the error log files sent from the database server while creating stored procedure and the trace log file for stored procedure execution.

DIRECTORY OF STORED PROCEDURE INCLUDE FILES

The **Directory of Stored Procedures** specifies the path stored procedure include files are placed in. It is used when the user needs extra include files for stored procedures. This keyword is only useful in non-Windows operating systems.

➲ **To set SQL Attributes:**

1. Click **Setup** in the **Start Database** window.
2. Click the **SQL Attributes** tab in the **Start Database Advanced Settings** window.
3. Select date and time input and output formats from the available drop down menus.
4. Enter a path into or select the browse button ... next to the **Directory of Stored Procedures** field to indicate a storage location for stored procedures.
5. Enter a path into or select the browse button ... next to the **Directory of Stored Procedure Error Message Files** field to indicate a location for the stored procedure error message files.
6. Enter a path into or select the browse button ... next to the **Directory of Stored Procedure** field to indicate a location for stored procedure include files.
7. Click **Save**.
8. Click **Cancel** to return to the **Create Database** window, and click **Start** to start the database.

Setting Distributed Database Environment Options

You can set DDB environment options that let your database handle queries in a distributed database environment.

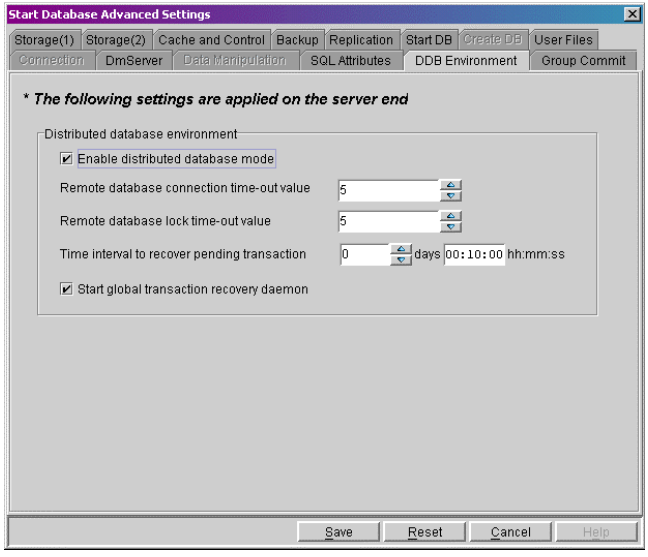


Figure 4-8 The DDB Environment page of the Start Database Advanced Settings window

ENABLE DISTRIBUTED DATABASE MODE

This setting enables the database to be used in distributed mode. It must be enabled for synchronous table replication to work. Synchronous table replication is set using the JDBA Tool. For more information on distributed data, synchronous table replication, or coordinator and participant databases, refer to the *JDBA Tool User's Guide*, or the *Database Administrator's Guide*.

REMOTE DATABASE CONNECTION TIME-OUT VALUE

The number in the combo box specifies the time in seconds that a coordinator database should wait when trying to establish a connection to a participant database.

REMOTE DATABASE LOCK TIME-OUT VALUE

The number in the combo box specifies the time in seconds that the coordinator database should wait when trying to establish a lock on the requested data in a participant database.

TIME INTERVAL TO RECOVER PENDING TRANSACTION

This value represents the time interval at which the global transaction recovery daemon is activated.

GLOBAL TRANSACTION RECOVERY DAEMON

This setting activates an automatic recovery mechanism. The mechanism checks whether a database has any problems with pending transactions, and then recovers them. This feature prevents loss of data when a network failure occurs or if there is an error at the participant database. The number in the combo box specifies the time interval in days from which the global transaction recovery daemon is activated.

➲ **To set distributed database options:**

1. Click **Setup** in the **Start Database** window.
2. Click the **DDB Environment** tab in the **Start Database Advanced Settings** window.
3. Click on the **Enable Distributed Database Mode** check box. The options below will be highlighted.
4. Enter the number of seconds to wait for a connection to be established before returning an error in the **Remote Database Connection Time-out value** field.
5. Enter the number of seconds to wait when trying to establish a lock on a remote database before returning an error in the **Remote Database Lock Time-out Value** field.
6. Enable the **Start Global Transaction Recovery Daemon** option.
7. Enter the number of days, hours, minutes, and seconds between every activation of the global transaction recovery daemon in the **Time Interval to Recover Pending Transaction** time fields.
8. Click **Save**.
9. Click **Cancel** to return to the **Create Database** window, and click **Start** to start the database.

Setting Group Commit Options

The group-commit function improves transaction processing by synchronizing journal file operations. DBMaker collects as many transactions as possible over a given interval and then commits them simultaneously to improve on-line transaction processing performance. It is suitable for large numbers of short transactions running simultaneously.

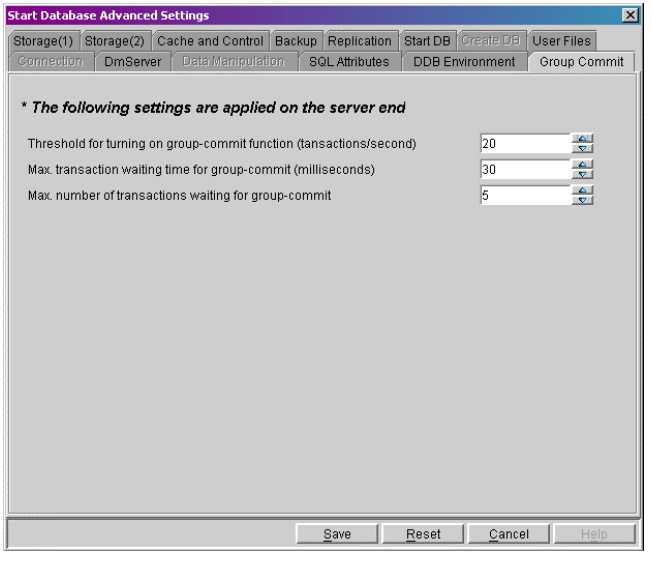


Figure 4-9 The I/O Server page of the Start Database Advanced Settings window

GROUP COMMIT THRESHOLD

The group-commit function will only become active when a certain threshold number of transactions per second is exceeded. The value in the **Threshold for turning on group-commit function** (**transactions / second**) field determines this activation threshold.

MAXIMUM TRANSACTION WAITING TIME

Each transaction will wait for a specified interval before automatically being committed, independent on the number of other transactions waiting for a group-commit. The value in the **max transaction waiting time for group-commit (milliseconds)** field indicates to the database the longest time interval any one transaction should wait. The default value is 30 milliseconds.

MAXIMUM NUMBER OF TRANSACTIONS IN WAIT STATE

Transactions waiting for a group commit will wait until the maximum transaction waiting time before the group-commit. However, if a certain number of transactions waiting for a group-commit are reached, then the group-commit will occur regardless of the waiting time. This number of transactions is specified in the **Max. number of transactions waiting for group-commit** field. The default value is five transactions.

➲ **To set Group Commit options:**

1. Click **Setup** in the **Start Database** window.
2. Click the **Group Commit** tab in the **Start Database Advanced Settings** window.
3. Enter a value in the **Threshold for turning on group-commit function (transactions / second)** field.
4. Enter a value in the **max transaction waiting time for group-commit (milliseconds)** field.
5. Enter a value in the **Max. number of transactions waiting for group-commit** field.
6. Click **Save**.
7. Click **Cancel** to return to the **Create Database** window, and click **Start** to start the database.

4.3 Using the Start Database Wizard

The Start Database Wizard is designed to help users define important database configuration parameters that must be set before a database is started. The start database wizard follows a systematic process, however, selecting different parameters will result in a different procedure.

The following chart outlines the program logic for the Start Database Wizard. Refer to the separate steps in the procedure that follow the chart for a detailed description of its function.

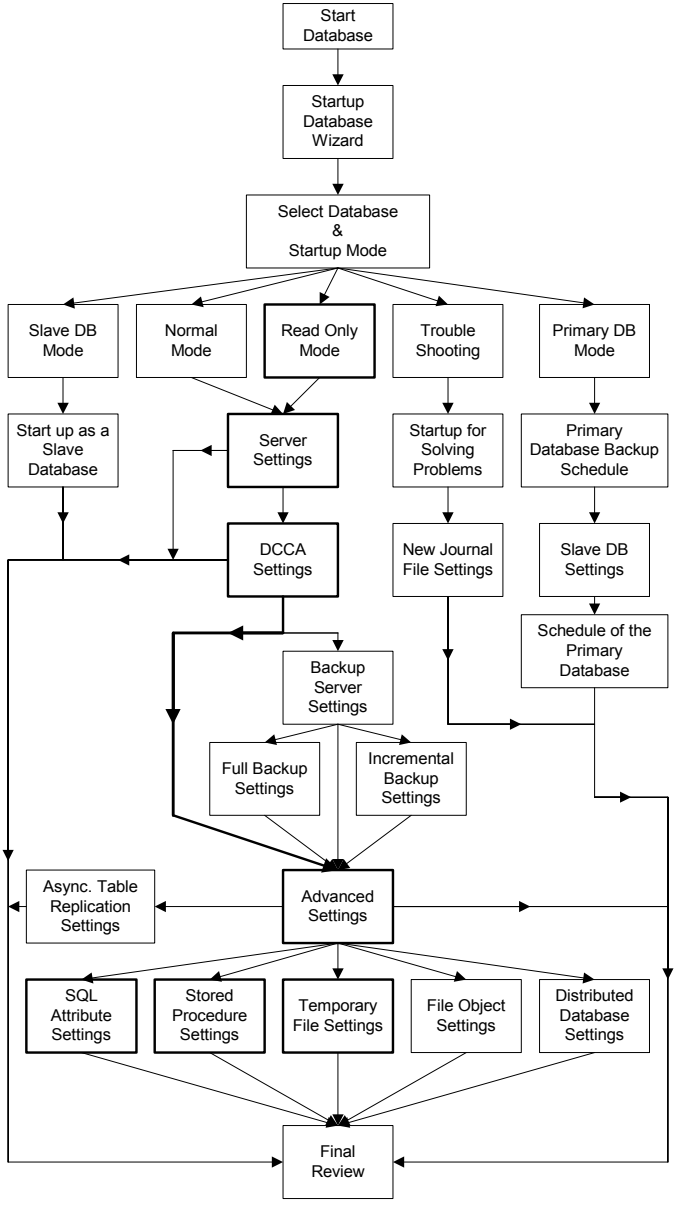
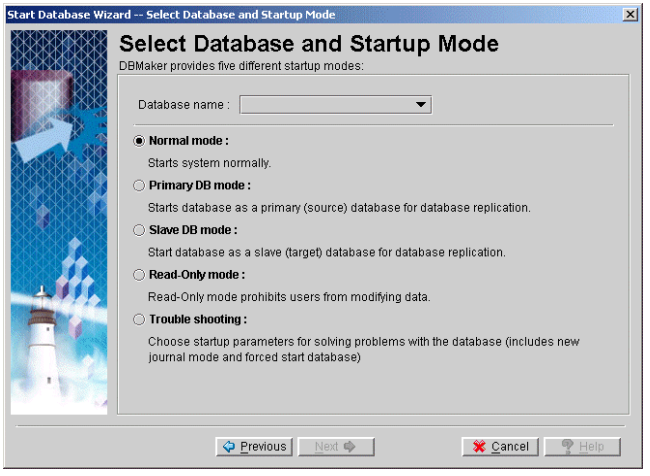


Figure 4-10 Program Logic for the Start Database Wizard

➲ **To start a database using the Start Database Wizard**

1. Select **Start Database** from the **Wizard** item in the menu bar. The **Welcome to Start Database Wizard** window appears.
2. Click the **Next** button. The **Select Database and Startup Mode** window appears. The user has 5 choices from here:



Normal Mode: Refer to Normal or Read Only Mode Startup if this option is chosen.

Primary DB Mode: Refer to Primary DB Mode Startup if this option is chosen.

Slave DB Mode: Refer to Slave DB Mode Startup if this option is chosen.

Read Only Mode: Refer to Normal or Read Only Mode Startup if this option is chosen.

Trouble Shooting: Refer to Trouble Shooting Startup if this option is chosen.

3. After choosing one of the five startup modes click the **Next** button. Refer to the following section appropriate to the startup mode chosen.

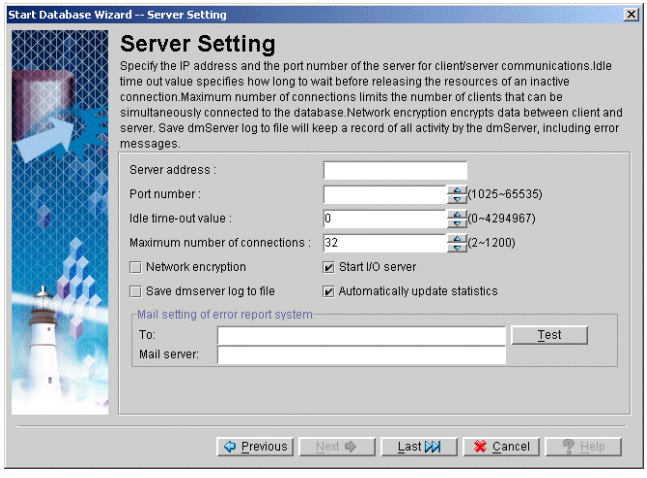
Normal or Read Only Mode Startup

Starting the database in normal mode means that the database is not the source or target for database replication, and is intended for normal use. The start database wizard allows advanced settings to be changed before database startup.

Read Only mode starts the database normally, except that no data may be modified.

➲ **To start a database in Normal or Read Only Mode**

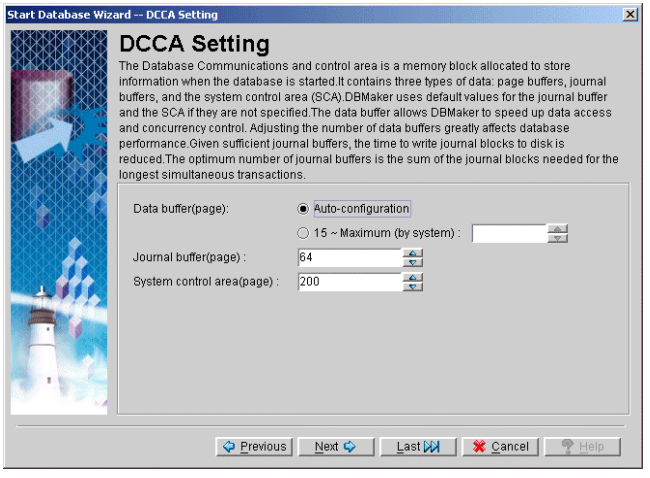
1. Start the database in **Normal mode** or **Read Only mode** from the **Select Database and Startup Mode** window.
2. The **Server Setting** window appears.



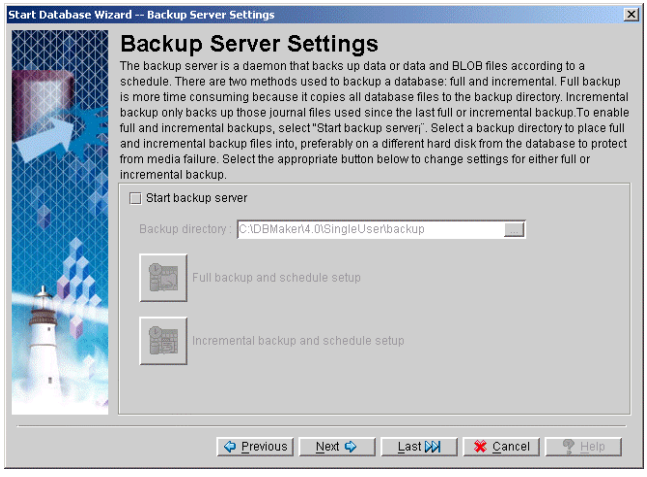
NOTE *At any point during the wizard, click* **Last** *to skip to the last step of the wizard (the* **Final Review** *window). Refer to step 41in this procedure.*

3. Enter the IP address or host name of the server in the **Server Address** field.
4. Enter the port number of the server in the **Server Port Number** field. This must be an integer between 1025 and 65535.
5. Enter the idle time out value of the server in the **Idle time-out value** field.
6. Enter a value for the maximum number of connections to the database in the **Maximum number of connections** field.

7. Check the **Network encryption** check box to allow for network encryption.
8. Check the **Save dmServer log to file** check box to save the dmServer log to file.
9. Check the **Start I/O server** check box to start the I/O server for the database.
10. Check the **Automatically update statistics** check box to have the database's statistics updated automatically.
11. **Mail setting of error report system.**
12. Click the **Next** button. The **DCCA Setting** window appears.



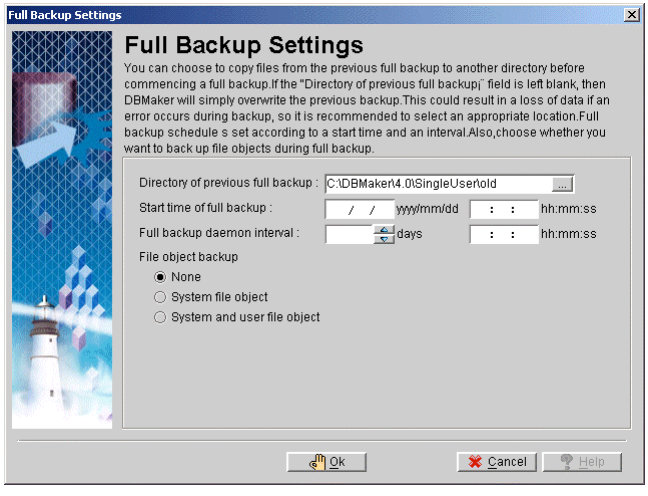
13. Enter the number of data buffer pages to allocate from memory in the **Data buffer** (**page**) field or select **Auto-configuration** to have DBMaker select the value.
14. Enter the number of journal buffer pages to allocate from memory in the **Journal buffer** (**page**) field.
15. Enter the number of pages to allocate from memory for the System Control Area in the **System control area** (**page**) field.
16. Click the **Next** button. The **Backup Server Settings** window appears.



17. To enable full and incremental backups, check the **Start backup server** check box. The **Backup directory** field, the **Full backup and schedule setup**, and the **Incremental backup and schedule setup** buttons become active.

18. Enter a path into or select the browse button ... next to the **Backup directory** field for the location of the backup data.

19. Click the **Full backup and schedule setup** button to setup the full backup settings. The **Full Backup Settings** window appears.



20. Enter a path into or select the browse button ... next to the **Directory of previous full backup** field for the location of the data from a previous backup. If this field is left blank, DBMaker will overwrite the previous backup.

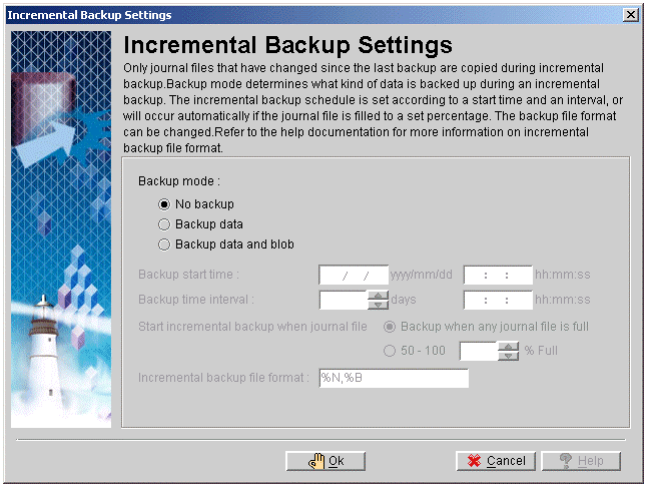
21. Enter a start time for the full backup in the **Start time of full backup** field.

22. Enter the interval for the full backup replication daemon to update in the **Full backup daemon interval** field.

23. Select whether system or user file objects are backed up.

24. Click the **OK** button. The user is returned to the **Backup Server Settings** window.

25. Click the **Incremental backup and schedule setup** button to setup the full backup settings. The **Incremental Backup Settings** window appears.



26. To enable incremental backups check the **Backup data** or the **Backup data and BLOB** check box.

27. Enter a start time for the backup in the **Backup start time** field.

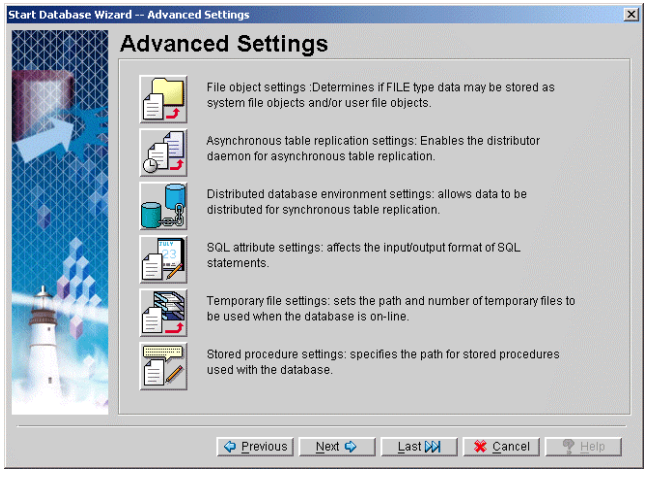
28. Enter the interval for the incremental backup replication daemon to update in the **Backup time interval** field.

29. Enter a value for the journal files to be backed up in the **Start incremental backup when journal file __% Full**, or check the **Backup when any journal file is full** check box.

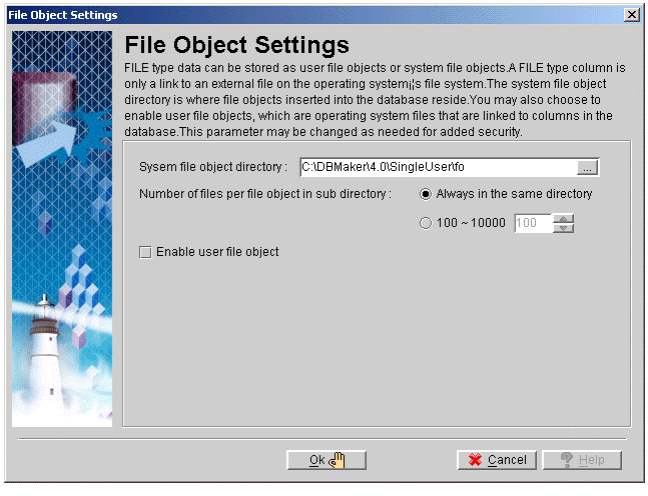
30. Enter a file format for the backup data in the **Incremental backup file format** field.

31. Click the **OK** button. The user is returned to the **Backup Server Settings** window.

32. Click the **Next** button. The **Advanced Settings** window appears.



33. To set file object options click the File object settings button. The **File Object Settings** window appears.



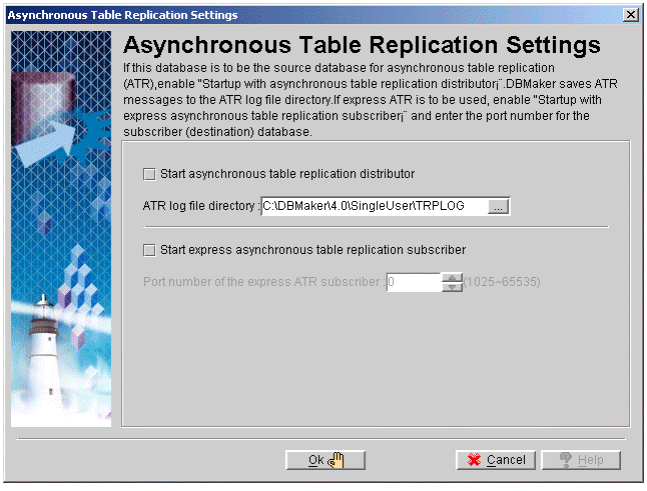
a) Enter a path into or select the browse button [...] next to the **System file object directory** field for the location of the system file objects.

b) Enter a value for the number of files per file object in the file object directory in the **Number of files per file object in sub directory** or check the **Always in the same directory** check box.

c) Check the **Enable user file object** check box to enable the use of user file objects.

d) Click the **OK** button. The user is returned to the **Advanced Settings** window.

34. To set up asynchronous table replication settings click the Asynchronous table settings button. The **Asynchronous Table Settings** window appears.



a) Check the **Start asynchronous table replication distributor** check box to start asynchronous table replication.

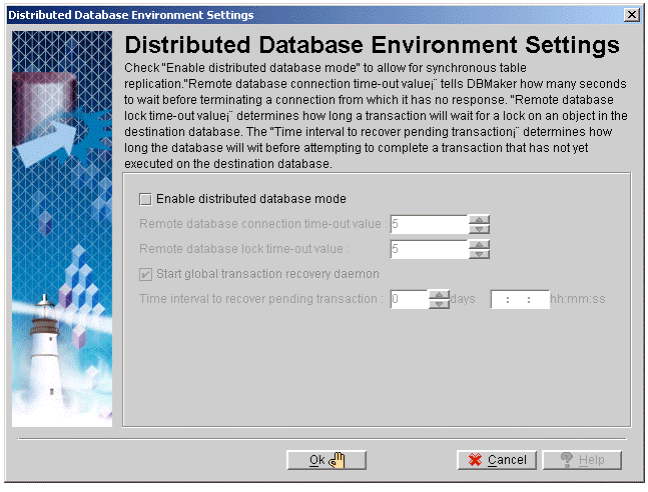
b) Enter a path into or select the browse button [...] next to the **ATR log file directory** field for the location of the ATR log files.

c) Check the **Start express asynchronous table replication subscriber** check box to start express asynchronous table replication. The **Port number of the express ATR subscriber** field becomes active.

d) Enter a port number in the **Port number of the express ATR subscriber** field.

e) Click the **OK** button. The user is returned to the **Advanced Settings** window.

35. To set the distributed database settings click the **Distributed database environment settings** button. The **Distributed database environment settings** window appears.



36. To enable DBMaker to function as a distributed database check the **Enable distributed database mode** check box. The remaining options in the window become active.

a) Set a time-out value in the Remote database connection time-out value field.

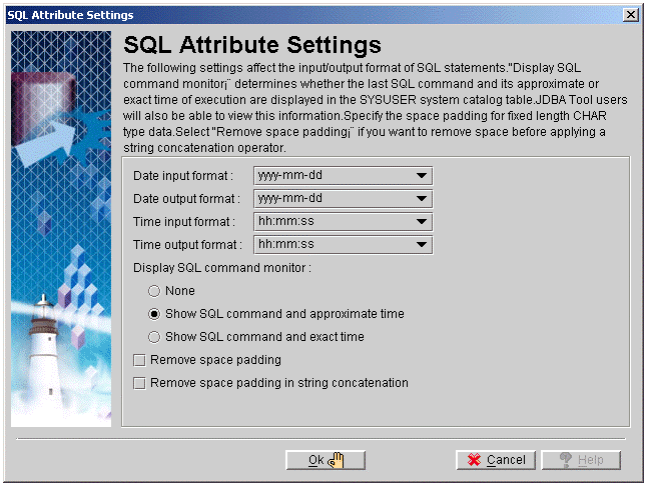
b) Set a lock time-out value in the Remote database lock time-out field.

c) Check the Start global transaction recovery daemon check box to start the daemon.

d) Set a time interval to recover pending transactions in the Time interval to recover pending transactions fields.

e) Click the OK button. The user is returned to the Advanced Settings window.

37. To set SQL attribute settings click the **SQL attribute settings** button. The **SQL Attributes Settings** window appears.



a) Select an input date format from the **Date input format** drop down list.

b) Select an output date format from the **Date output format** drop down list.

c) Select an input time format from the **Time input format** drop down list.

d) Select an output time format from the **Time output format** drop down list.

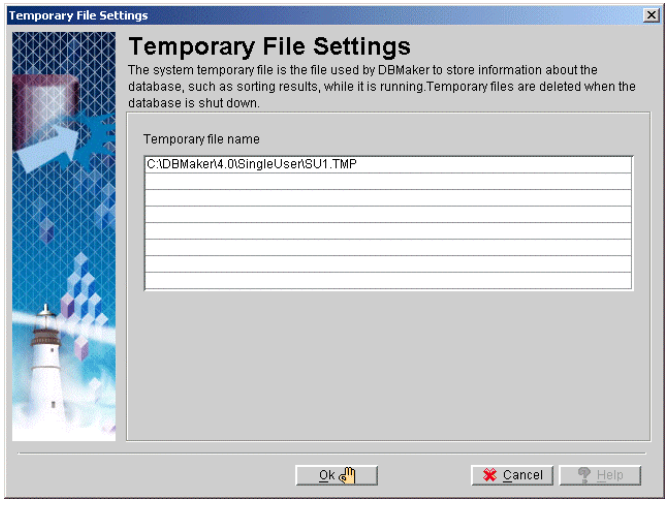
e) Select the whether the last SQL command and its time of execution are displayed in the system catalog table from the **Display SQL command monitor** check boxes.

f) Select **Remove space padding for CHAR data** to remove space padding before a CHAR type data.

g) Select **Remove space padding in string concatenation** to remove space padding before a string concatenation.

38. Click the **OK** button. The user is returned to the **Advanced Settings** window.

39. To set temporary file settings click the **Temporary file settings** button. The Temporary File Settings window appears.

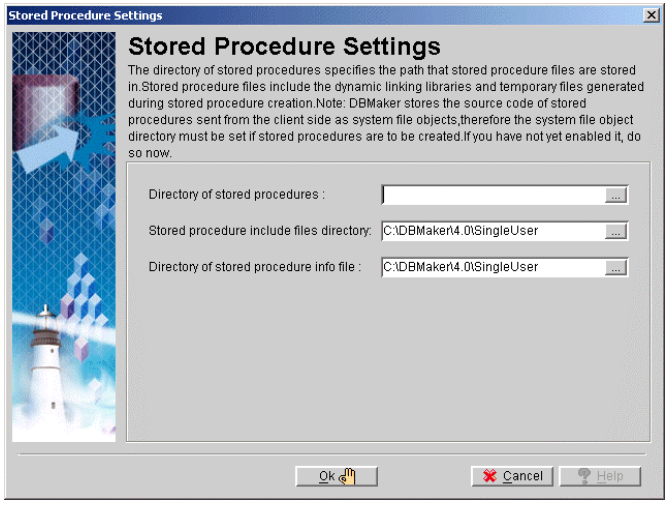


a) Double click a field to enter the temporary file name.

b) Enter the temporary file's name.

c) Click the **OK** button. The user is returned to the **Advanced Settings** window.

40. To set stored procedure options click the **Stored procedure settings** button. The **Stored Procedure Settings** window appears.



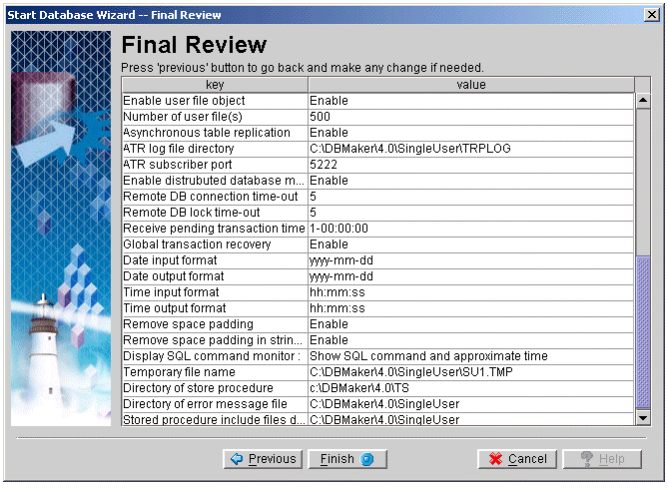
a) Enter a path into or select the browse button ... next to the **Directory of stored procedures** field for the location of stored procedures.

b) Enter a path into or select the browse button ... next to the **Stored procedures include files directory** field for the location of the ATR log files.

c) Enter a path into or select the browse button ... next to the **Directory of stored procedure info file** field for the location of information files for stored procedures.

d) Click the **OK** button. The user is returned to the **Advanced Settings** window.

41. Click the **Next** button. The **Final Review** window appears.



42. Check over the database settings. If any of the settings need to be changed click the **Previous** button to the appropriate window and make the changes. Then return to the **Final Review** window.

43. Click the **Finish** button when satisfied with the database settings. The following dialog box will appear.

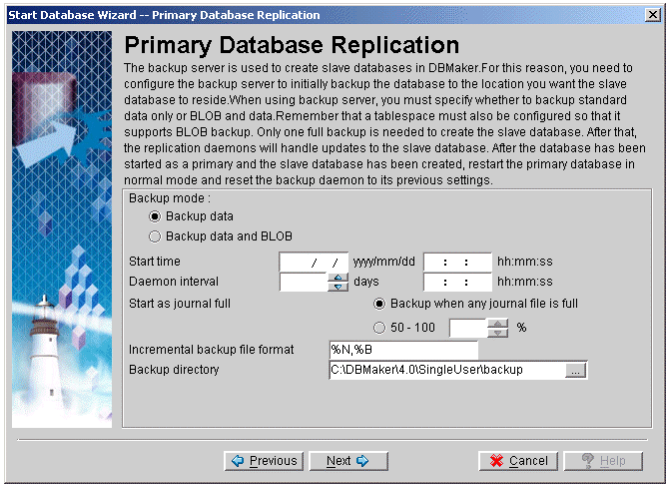
Primary DB Mode Startup

The Database Startup wizard simplifies slave database creation. A slave database is initially created from the primary database by choosing Primary DB Mode startup.

The Database Startup wizard backs the database up to another location and in the process creates the slave database. After the slave database is created, the primary database should be restarted in normal mode. The primary database is still referred to as a primary database, but it does not need to be started in Primary DB mode.

➲ **To start a database in Primary DB Mode**

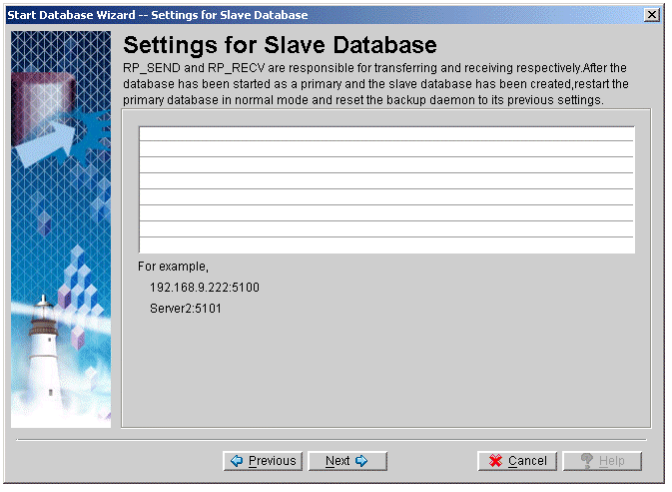
1. Start the database in **Primary DB Mode** from the **Select Database and Startup Mode** window.
2. The **Primary Database Replication** window appears. Set the backup database options.



3. Select the **Backup data** check box to backup standard data only or select the **Backup data and BLOB** check box to backup standard and BLOB data to the slave database.
4. Set the backup database's start time in the **Start time** fields.
5. Set the replication daemons' update schedule in the **Daemon interval** fields.
6. Select the **Backup when any journal file is full** check box to have the journal files backed up when any of the journal files are full, or set the percentage of a journal file being full before it is backed up.
7. Set the backup file format in the **Incremental backup file format** field.

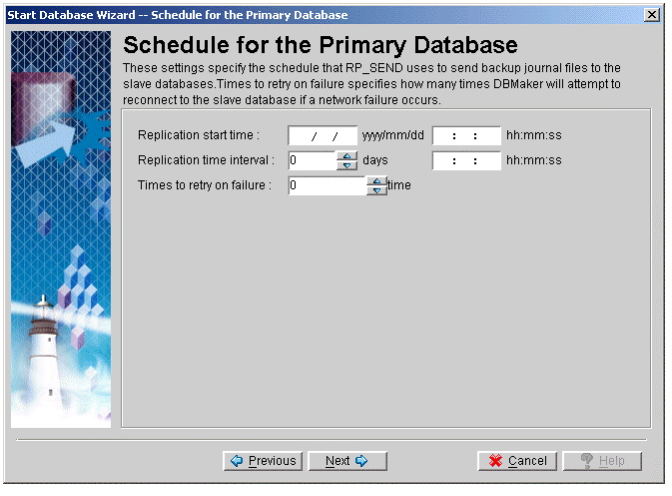
8. Enter a path into or select the browse button [...] next to the **Backup directory** field for the location of journal files sent by the backup database.

9. Click the **Next** button. The **Settings for Slave Database** window appears.



10. Set the port number and IP address for the client database.

11. Click the **Next** button. The **Schedule for Primary Database** window appears.

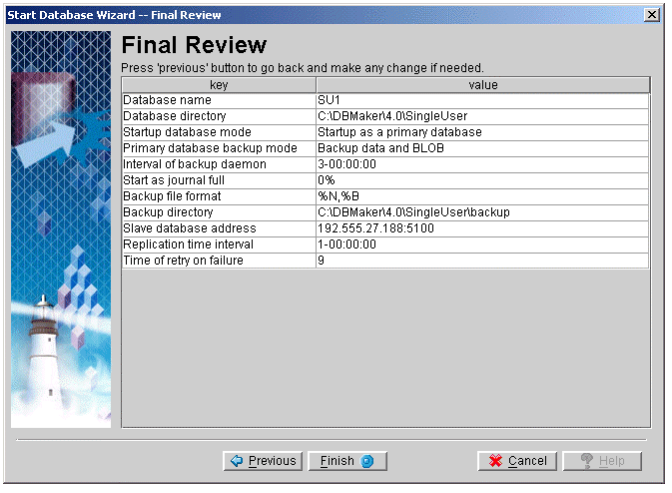


12. Enter a start time for the schedule in the **Replication start time** fields.

13. Set the time interval between replications in the **Replication time interval** fields.

14. Set the number of times that the primary database will attempt to reconnect to the slave database in the **Times to retry on failure** field.

15. Click the **Next** button. The **Final Review** window appears.



16. Check over the database settings. If any of the settings need to be changed click the **Previous** button to the appropriate window and make the changes. Then return to the **Final Review** window.

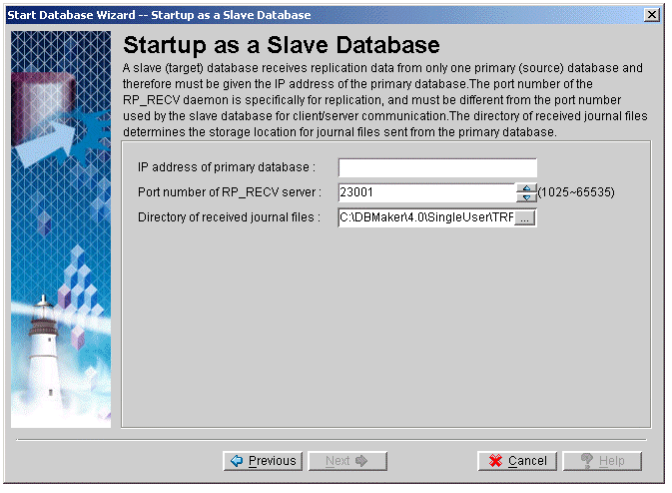
17. Click the **Finish** button when satisfied with the database settings.

Slave DB Mode Startup

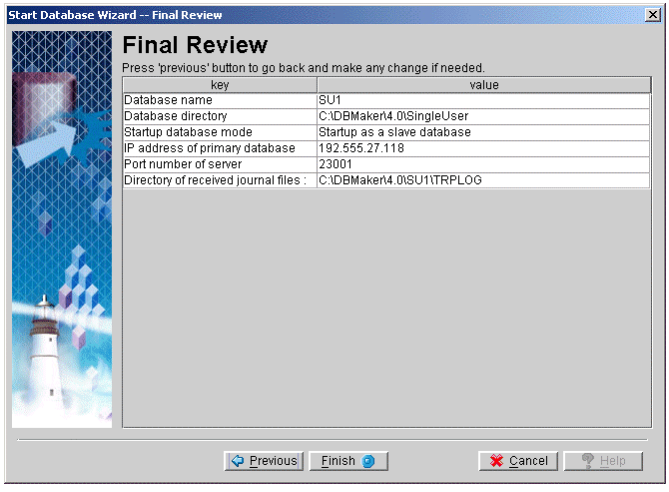
Slave DB Mode startup is used when the database is the target of database replication. A slave database must be a replica of the source database, and is initially created by backing up a database to a remote location. After the slave database has been created, it must be started as a slave database to enable it to receive updates from the primary database. Slave databases are read-only. If a database created normally is started as a slave database, it will start as a read only database and an error will occur when the database attempts to update tables from the source database.

➲ **To start a database in Slave Database Mode:**

1. Start the database in **Slave Database Mode** from the **Select Database and Startup Mode** window.
2. The **Startup as a Slave Database** window appears.



3. Enter the IP address of the server in the **IP address of primary database** field.
4. Enter the port number of the RP_RECV daemon in the **Port number of the RP_RECV server** field.
5. Enter a path into or select the browse button ... next to the **Directory of received journal files** field for the location of journal files sent by the primary database.
6. Click the **Next** button. The **Final Review** window appears.



7. Check over the database settings. If any of the settings need to be changed click the **Previous** button to the appropriate window and make the changes. Then return to the **Final Review** window.

8. Click the **Finish** button when satisfied with the database settings.

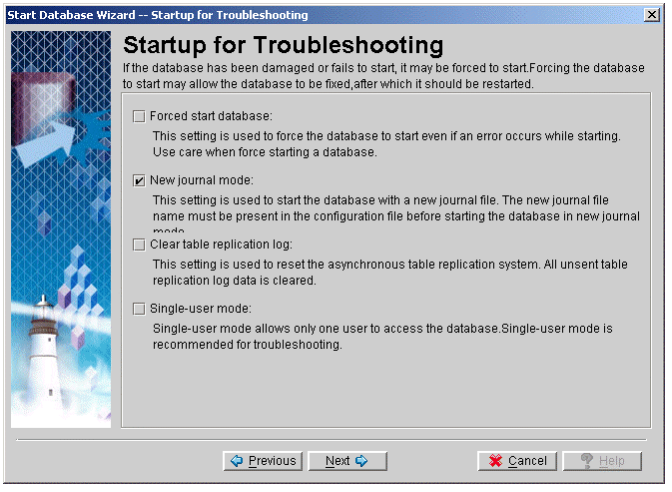
Trouble Shooting Startup

Trouble Shooting Startup is useful when you are having trouble starting a database, and can sometimes be used to diagnose and fix problems within the database

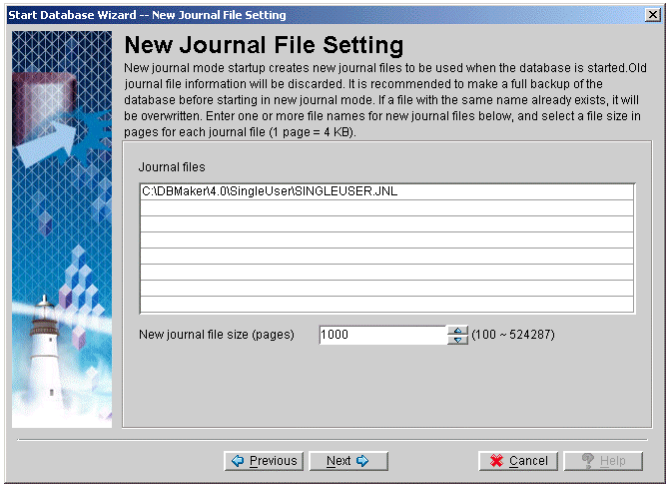
➲ **To start a database in Trouble Shooting mode**

1. Start the database in **Trouble Shooting** mode from the **Select Database and Startup Mode** window.

2. The **Startup for Trouble Shooting** window appears.



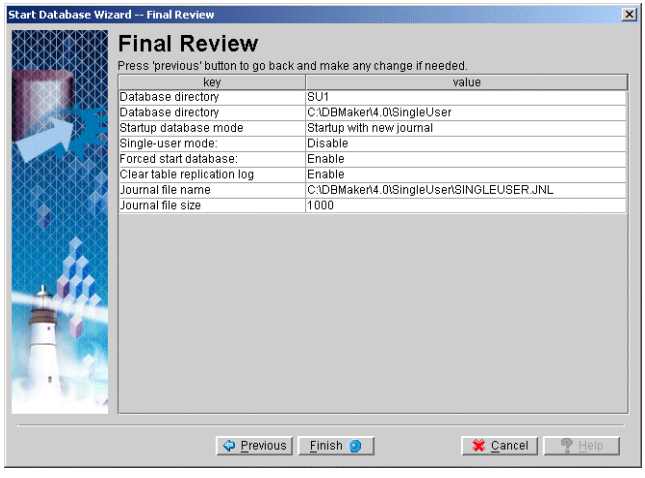
3. To force the database to startup in single user mode check the **Single user mode** check box.
4. To force start the database in normal mode, check the **Forced start database** check box.
5. Check the **Clear table replication log** check box to clear the asynchronous table replication log.
6. To start the database with a new journal file, check the **New journal mode** check box.
7. Click the **Next** button. If the **New journal mode** check box is checked, go to step 7. If the New journal mode check box is not checked, go to step 10.
8. The **New Journal File Setting** window appears.



9. Enter the file name and path of the new journal files in the **Journal files** field.

10. Enter the size of the journal files in the **New journal file size (pages)** field. Valid values are 100 to 524287.

11. Click the **Next** button. The **Final Review** window appears.



12. Check over the database settings. If any of the settings need to be changed click the **Previous** button to the appropriate window and make the changes. Then return to the **Final Review** window.

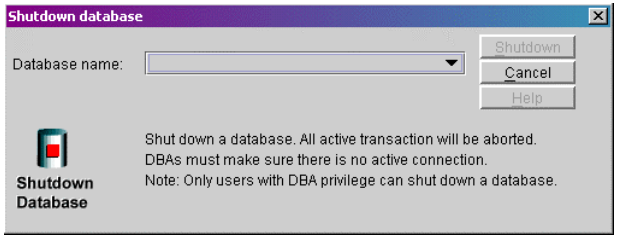
13. Click the **Finish** button when satisfied with the database settings. The following dialog box will appear.

5 Shutting Down a Database

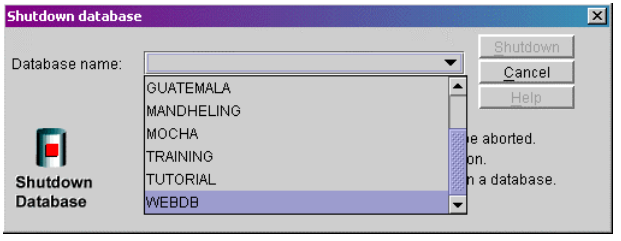
JServer Manager can be used to shut down a database. All existing connections to a database are terminated when the database is shut down. Any user connected to that database at the time of database shutdown will receive an error message. You should be sure that all users are disconnected and that there are no pending distributed transactions to the database before shutting it down. The status of pending transactions and connections may be checked using the monitor feature in the JDBA Tool. Refer to the *JDBA User's Guide* for more information.

➲ **To shut down a database**

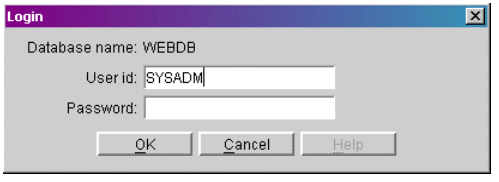
1. To shut down a database, select **Shut down database** from the main console or from the **Database** drop-down menu. The **Shutdown Database** window appears.



2. To select a database, click on the arrow on the **Database Name** field. A menu listing all databases on the server appears.



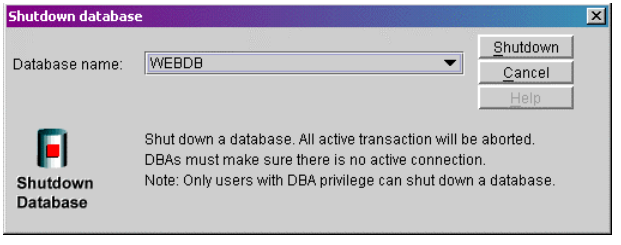
3. Select the database to be stopped from the **Database Name** menu. The **Login** dialog box appears prompting the user to log on to the database.



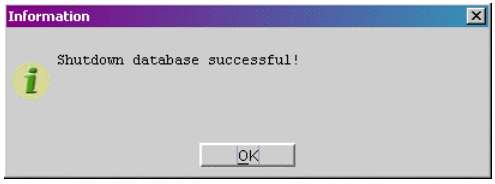
4. Enter your user ID in the **User ID** field.

NOTE *You must have DBA privilege to shut down the database.*

5. Click **OK**. The **Shut down database** window is displayed. The name of the database that is to be shut down should appear in the **Database Name** field.



6. Click the **Shutdown** button to shut down the database. All other connections to the database are terminated.



7. Click **OK**.

6 Using an NT Service

Installing an NT service allows the database server to start automatically the next time the user logs on to Windows NT. With a database installed as a service, the database administrator can log off Windows NT and still keep the database server running. Therefore, all connections between the user and client are preserved. If a database is not installed as a service in Windows NT, then all connections are terminated when logging off Windows.

To install a database as an NT service, you must log on to Windows NT with a user account that has administrator security privileges (or the Administrator account itself). Additionally, you must have SYSADM security privileges for the database.

6.1 Adding an NT Service

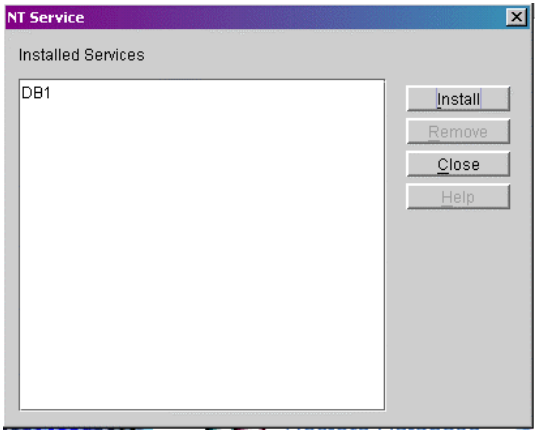
You can add new NT services for a database by selecting the database and choosing a start mode. The following start mode options re available.

Auto Started: In most cases you will select this as the start mode when installing the service. This will allow the database server to start immediately after the NT system boots up the next time you enter the system.

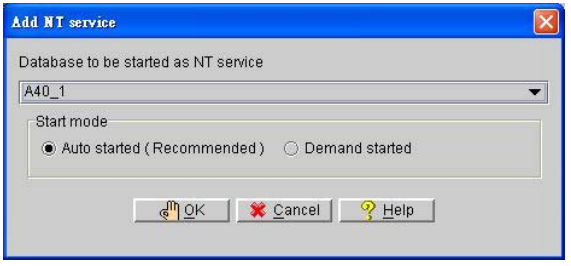
Demand Started: In some instances you might want to select this as the start mode when installing the service (for example, if the database has crashed and the database server must be force started). If demand started is selected, the user must manually start the service from within Windows NT, Windows 2000, or Windows XP. In this case, no users should connect to the server as this could potentially damage the database.

➲ **To add an NT Service to a database:**

1. Select **NT Service** from the **Database** drop down menu. The **NT Service** dialog box appears.



2. Click **Install**. The **Add NT Service** dialog box appears.



3. Select a database from the **Database to be started as NT service** menu.

4. Select the database to start an NT Service for from the **Database to be started as NT service** menu.

5. To start the database in Auto Started mode, make sure the **Auto Started** option button is selected.

6. To start the database in Demand Started mode, click the **Demand Started** option button.

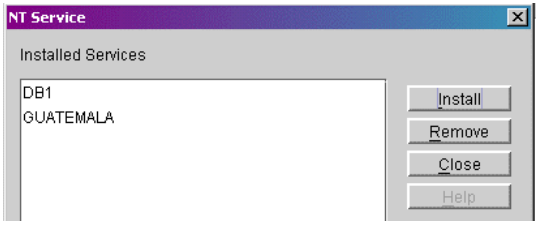
7. Click **OK**. The **NT Service** window is displayed. The database where the service is installed is displayed in the **Installed Services** field

6.2 Viewing Installed NT Services

You can view all services installed for the database. This function is useful for keeping track of the different services that DBMaker creates within Windows NT, Windows 2000, or Windows XP.

➲ **To view installed NT services:**

1. Select **NT Service** from the **Database** drop down menu. The **NT Service** dialog box appears. Installed services will appear in the **Installed Services** list.



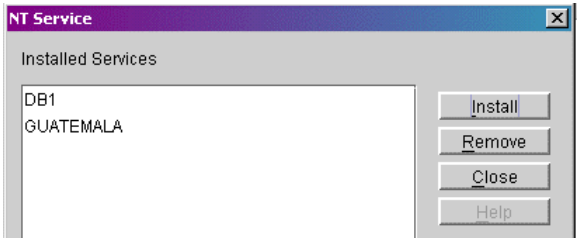
NOTE *You can also view installed NT services by selecting the Administrative Tools\Services icon in Windows 2000 or the Services icon in Windows NT*

6.3 Removing NT Services

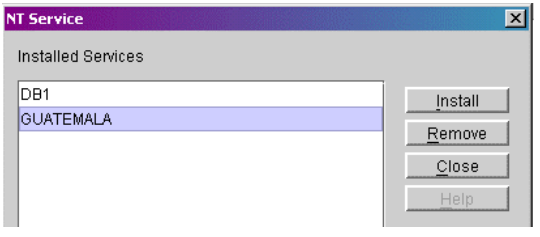
If an NT service is no longer required for a database, you can remove it from the database.

➲ **To remove NT service for a database**

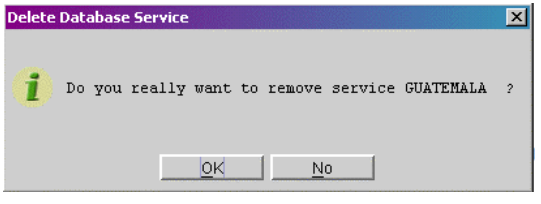
1. Select NT Service from the Database drop down menu. The NT Service window appears.



2. Select the database to be removed from NT service from the **Installed Services** list box.

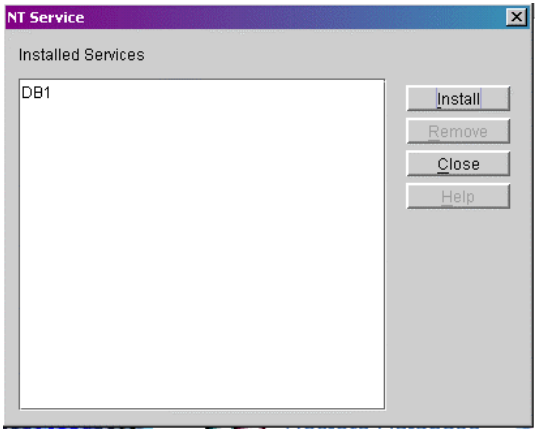


3. Click the **Remove** button. The **Delete Database Service** dialog box appears to let you confirm your choice.



4. Click **OK**. The database service is removed. Selecting **OK** will remove the database from NT service. The **NT Service** window will not show the removed database.

NOTE *If the selected database still appears on the list, close the NT Service window and reopen it. The Installed Services list will not show the removed database.*

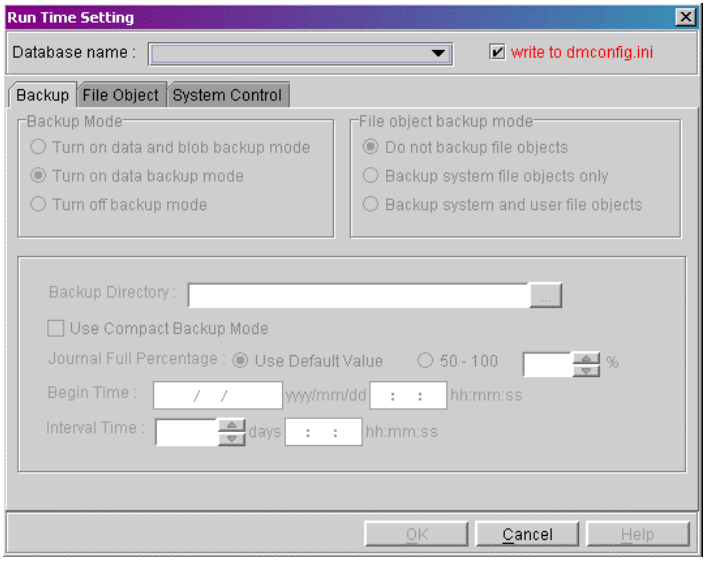


7 Altering Run Time Settings

Certain database configuration parameters may be changed while the database is on-line. These settings include backup settings (full and incremental backup frequency and location), file object settings (enable user file objects, system file object location), and system control settings.

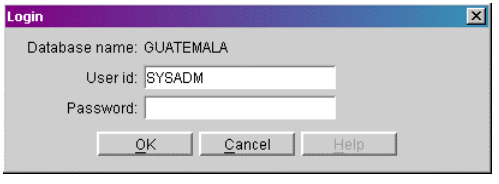
➲ **To select a database for Run Time setting changes:**

1. Select **Run Time Setting** from the database drop down menu.
2. The **Run Time Setting** window appears.

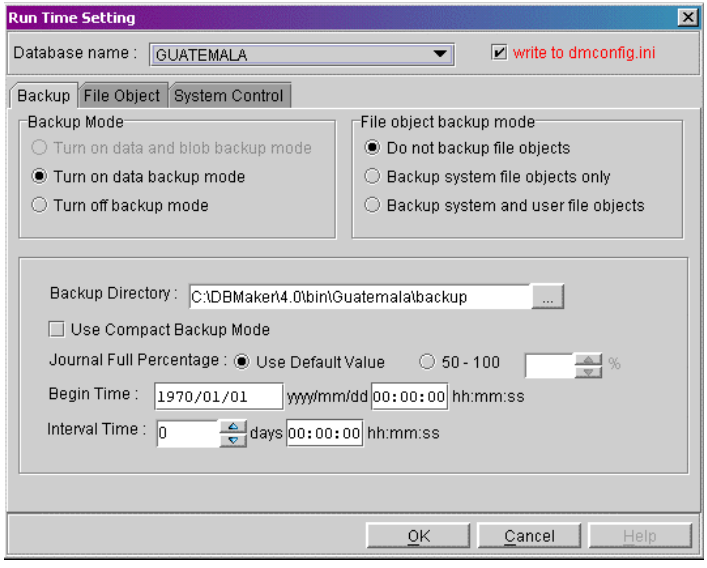


3. Select a database from the **Database Name** menu.

4. The **Login** dialog box is displayed



5. Click **OK**, the database you logged into appears in the **Database Name** field of the **Run Time Setting** dialog box.



6. To use the updated settings in the next session, make sure that the **write to dmconfig.ini** check box is enabled.

7. To allow the updated settings to apply to the current session only, clear the checkmark in the **dmconfig.ini** check box.

NOTE *The **write to dmconfig.ini** option is only available on the server side.*

8. Set the following options:

Backup Settings

File Object Settings

System Control

7.1 Changing Backup Settings

Incremental Backup settings that may be altered in the runtime include the following: file object backup mode, backup location, and deactivation of backup BLOB or deactivation of backup data. During runtime, backup modes may only be changed to a state where less data is backed up (except in the case of file objects). This may be useful, for example, if the database administrator wants to free up resources by changing the backup mode from "backup data and BLOB" to "backup data"

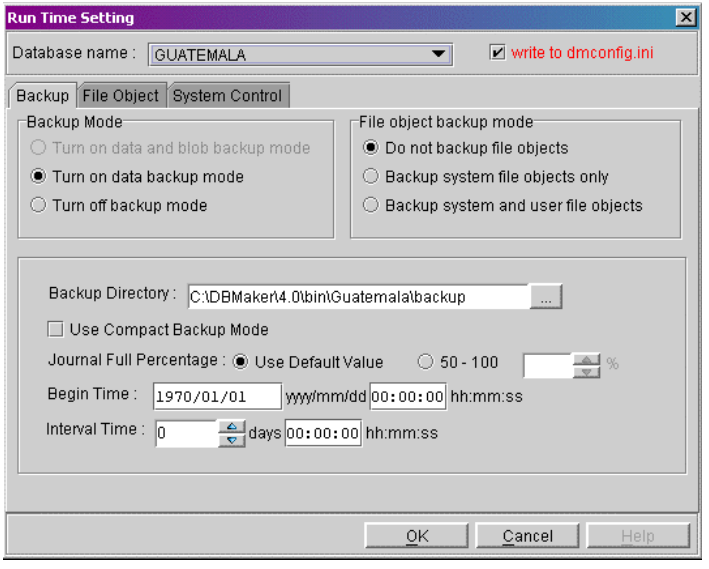


Figure 7-1 The Backup page of the Run Time Settings window

Backup Mode

You can choose to turn off the incremental backup mode, or turn off just backup BLOB mode. Run time settings only allow for deactivation of the incremental backup mode, either deactivating backup BLOB, or deactivating backup data.

To change the backup mode to Backup Data or Backup Data and BLOB you must shut down the database, start the database, select the **Setup** button, and make the changes to the **Backup** page of the **Start Database Advanced Settings** window.

The following backup modes are available.

BACKUP MODE	DESCRIPTION
Turn off backup mode	Disables the incremental backup daemon. In this mode the journal files are not backed up.
Turn on data backup mode	Sets the incremental backup daemon to backup all data journal pages.
Turn on data and BLOB backup mode is	BLOB data is written to the journal and the incremental backup daemon backs up all journal files.

File Object Backup Mode

You can choose what types of file objects to backup during incremental backups by changing the file object backup mode. The following file object backup modes are available.

FO BACKUP MODE	DESCRIPTION
Do not backup file objects	No file objects are backed up during full or incremental backups
Backup system file objects	System file objects are backed up during any full or incremental backup.
Backup system and user file objects	System and user file objects are backed up during any full or incremental backup.

Backup Directory

This is the directory where the backup server puts all full backup files and incremental backup (journal) files. You should create a backup directory on a different disk from the database files to prevent the loss of both the database and the backup files in the event of a media error. The default path for backup files is the database directory

/backup and is automatically created by DBMaker. The total length of the backup directory path must not exceed 255 characters.

Use Compact Backup Mode

User Compact Backup Mode ensures that only needed journal files are copied by the Backup Server when performing a backup. As a result, storage space is saved in the backup location. However, it also means restoring a database may take more time.

Journal Full Percentage

The user may want to allow DBMaker to create an incremental backup before the journal file is completely full. The journal trigger value specifies the percentage a journal file must fill before the Backup Server performs an on-line incremental backup. You can specify a percentage between 50 and 100 for the journal trigger value. You can combine the journal trigger value with the backup schedule to backup your database on a regular schedule.

Incremental Backup Begin Time

To set the time in which the first incremental backup will begin for the database, enter the date in the **yyyy/mm/dd** field, as well as the time in the **hh:mm:ss** field. Reenter the numbers if they appear incorrectly the first time; JConfiguration Tool automatically enters values into the first two spaces of the **yyyy** field and the first space of the **hh** field if values are entered into other fields.

Incremental Backup Interval Time

The number in the combo box specifies the time interval at which the Incremental Backup occurs in days. Next to the combo box is a field for time input, which specifies the time interval in hours, minutes, and seconds. The total time interval is determined by adding the two values together, so inputting 1 into the **days** field and inputting 12:00:00 into the **hh:mm:ss** field would cause an incremental backup to be made every day and a half.

➲ **To change backup settings:**

1. Select a backup mode from the **backup mode** box.

   To change the backup mode to backup data only, select the **Turn on data backup mode** option button.

   To turn off the backup mode, click the **Turn off backup mode** option button.

2. Select a file object backup mode from the **File object backup mode** box.

   To not backup file objects, select the **Do not backup file objects** option button.

   To backup system file objects, click the **Backup system file objects** option button.

   To backup system and user file objects, click the **Backup system and user file objects** option button.

3. Enter a path into or select the browse button ... next to the **Backup Directory** field to indicate a destination for the full and incremental backup files.

4. To enable compact backup, click on the **Use Compact Backup Mode** check box.

5. Incremental backups can be set to automatically execute when journal files have filled to a set percentage. Next to **Journal Full Percentage**:

   Select the **Use Default Value** option button to set incremental backups to execute when any journal file is completely filled.

   Enter a value from 50 to 100 in the **50 – 100 %** field to set incremental backups to execute when any journal file is filled to the value entered.

6. Indicate a date and a time at which incremental backups are to begin in the **Begin Time** fields.

7. Enter the number of days, hours, minutes, and seconds between each successive incremental backup in the **Interval_Time** time fields.

8. Select one of the other tabs at the top (to change other runtime settings) or select **OK** from the bottom of the **Run Time Settings** window

7.2 Changing File Object Settings

Selecting the **File Object** tab in the **Run Time Settings** window allows the user to enable external user file objects and change the storage location of system file objects. DBMaker can automatically create subdirectories within the FO directory. Each subdirectory is filled with new file objects up to a threshold value. When the threshold is reached, DBMaker creates a new FO subdirectory.

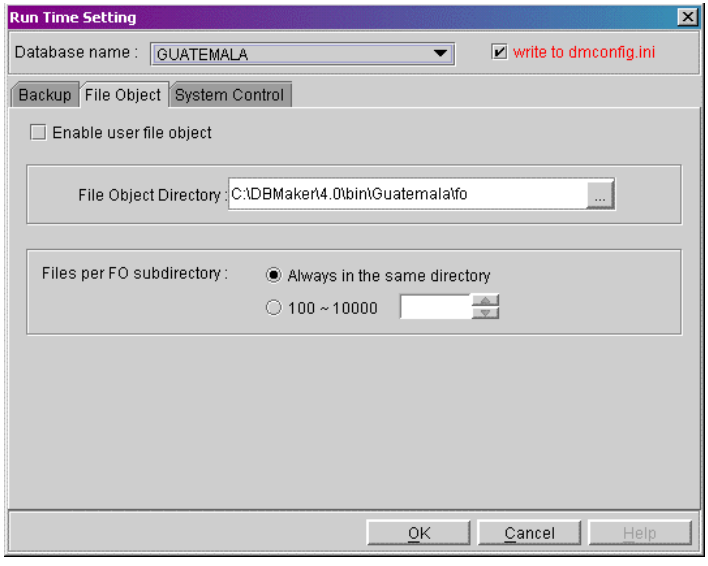


Figure 7-2 The File Object page of the Run Time Settings window

➲ **To change file object settings:**

1. Select the **File Object** tab from the **Run Time Setting** window.
2. To enable the database to use external file objects, click on the **Enable User File Object** check box.
3. Enter a path into or select the browse button ... next to the **System File Object Directory** field to indicate the location of the System File Object Directory.

4. Select from **Files per FO Subdirectory**:

   To have DBMaker always store system file objects in the system file object directory, select **always in the same directory**

   To have DBMaker create subdirectories with a set number of file objects in each subdirectory, select the option button next to **100 ~10000** and enter the threshold number of file objects (between 100 and 10000) at which a subdirectory is declared full and a new one created.

5. Select one of the other tabs at the top (to change other runtime settings) or select **OK** from the bottom of the **Run Time Settings** window.

7.3 Changing System Control Settings

The system control page allows settings to be changed that directly affect database performance. It also allows distributed mode to be enabled during the runtime and allows for the number of pages to extend a file when an autoextend tablespace is full.

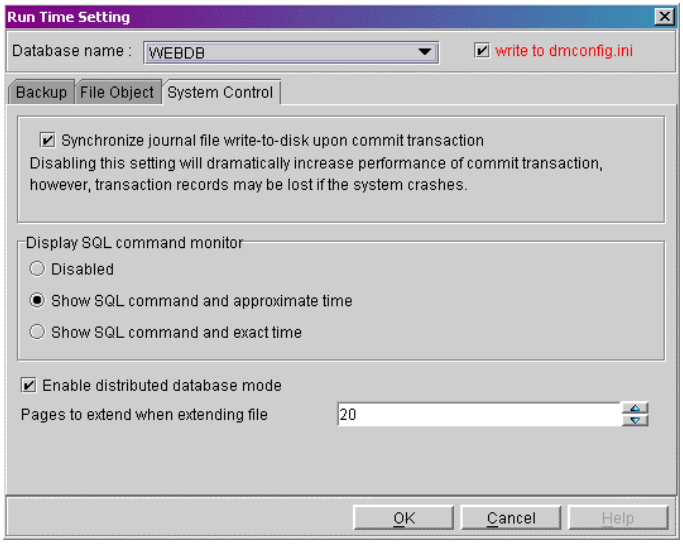


Figure 7-3The System Control page of the Run Time Settings window

Journal File Synchronization

Under normal operating conditions, when a transaction is committed, DBMaker initiates a two-step process with regards to the journal blocks involved in the transaction. The first step is to flush the relevant journal blocks into the file system of the operating system, or the operating system's cache. The second step is to force the operating system to write these journal blocks to disk, so that the transaction is preserved in the event of a system crash.

It is possible to greatly increase the performance of the commit transaction process if the second step of this process is omitted. The journal blocks in this case remain in the operating system's cache until the operating system writes them to disk. However, if the system crashes at this point, the records of any transactions that were still in the operating system's cache will be lost.

Display Mode of SQL Command Monitor

Display Mode affects the display content of the **SQL_CMD** and **TIME_OF_SQL_CMD** columns in the **SYSUSER** system table. JDBA Tool Users can view information about users accessing the database with the Database Monitor function. **Session Information** displays users currently connected to the database. The columns *Current SQL command* and *Time of Current SQL Command* display the most recent SQL transaction committed by the user and the time of execution, respectively. **Display Mode of SQL Command Monitor** sets the configuration for how these attributes are displayed. No SQL commands are shown if **Disabled** is clicked. The most recent SQL command executed by the user and its approximate time of execution are shown if **Show SQL command and approximate time** is clicked. The most recent SQL command executed by the user and its exact time of execution are shown if **Show SQL command and exact time** is clicked. Displaying the exact time of execution uses more CPU resources and slows down the database. The default mode is **Show SQL command and approximate time**

Enable Distributed Database Mode

This setting enables the database to be used in distributed mode. It must be enabled for synchronous table replication to work. Synchronous table replication is set using the JDBA Tool. For more information on distributed data, synchronous table replication, or coordinator and participant databases, refer to the *JDBA Tool User's Guide*, or the *Database Administrator's Guide*.

Pages to Extend when Extending a File

When all pages in a data file or BLOB file are full, DBMaker can automatically extend the number of pages or frames in the file to allow the database to grow. The **Number of Pages to Extend While Extending File** setting tells DBMaker how many pages or frames to add to the full file in the event that it is full. If the database administrator expects that the database will grow very quickly, then a higher number should be picked to lessen the frequency at which the file is appended. One page is equal to approximately 4KB.

➲ **To change System Settings:**

1. To disable synchronized journal block writing to disk, remove the check mark from the **Synchronize journal file write-to-disk upon commit transaction** checkbox.
2. Select a setting for the SQL command monitor:

   Select **Disabled** to not display any SQL commands

   Select **Show SQL command and approximate time** to show the most recent SQL command executed by the user and its approximate time of execution.

   Select **Show SQL command and exact time** to show the most recent SQL command executed by the user and its exact time of execution.
3. To enable distributed database mode, check the **Enable Distributed Database Mode** box.
4. Enter the number of pages to extend a file in the **Number of Pages to Extend While Extending File** field.
5. Select one of the other tabs at the top (to change other runtime settings) or select **OK** from the bottom of the **Run Time Setting** window.

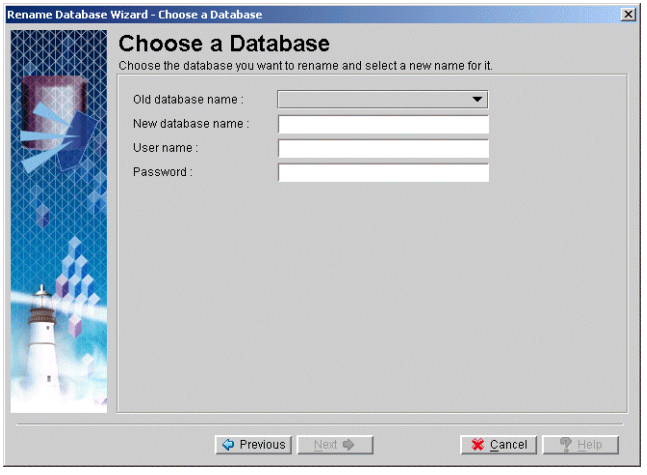
8 Renaming a Database

Databases may be renamed by using the Rename Database wizard. This easy-to-use tool simplifies the renaming of a database. When using Windows you must update the ODBC data source if you wish to connect to the database via ODBC.

➲ **To Rename a database:**

1. Click the **Wizards** option from the menu bar. A pop up menu appears.
2. Select **Rename Database** from the list. The **Welcome to the Rename Database Wizard** window appears.

3. Click the **Next** button. The **Choose a Database** window appears.



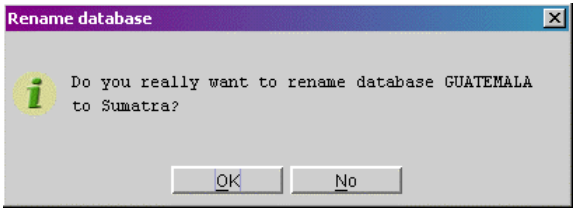
4. Select a database from the **Old database name** menu.

5. Enter a name in the **New database name** field.

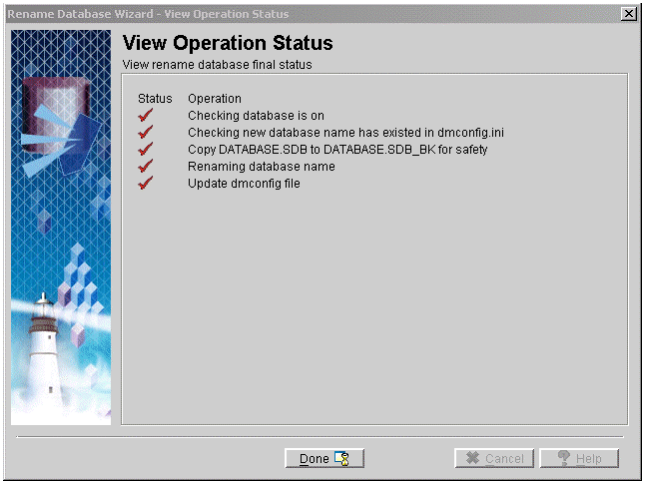
6. Enter the system administrator's password in the **User name** field. SYSADM is in the field as the default name.

7. Enter the system administrator's password.

8. Click **Next**. The **Rename database** dialog box appears.



9. Click the **OK** button. The **View Operation Status** window and the **Information** dialog box appear.



10. Click the **OK** button in the **Information** dialog box. The dialog disappears.

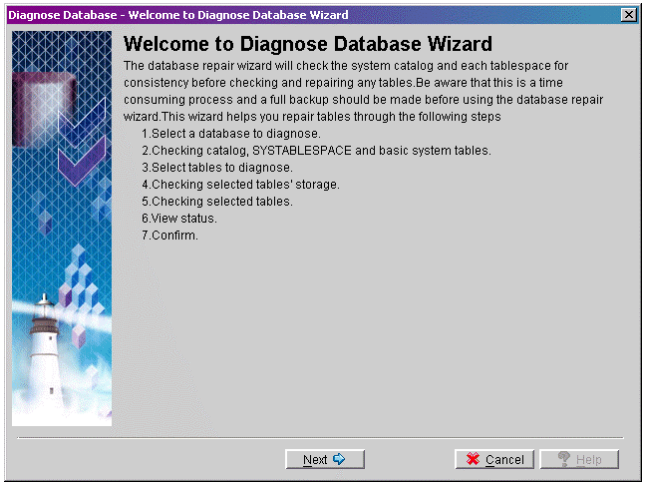
11. Click the **Done** button in the **View Operation Status** window. The user is returned to the **JServer Manager** main window.

9 Diagnosing a Database

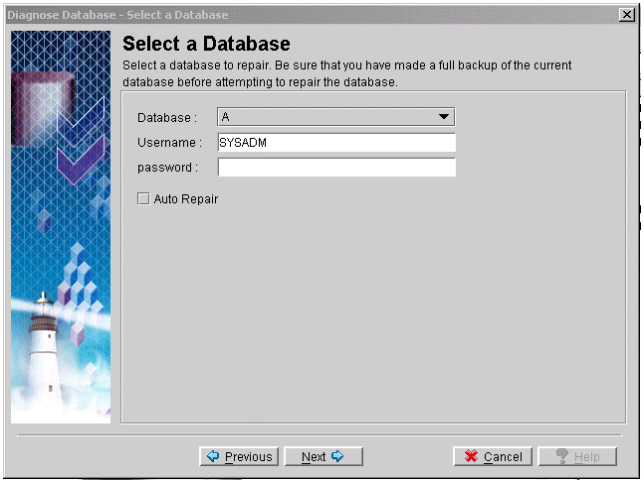
This section will deal with diagnosing and repairing a database. This will primarily be done using the **Diagnose Database Wizard**. The wizard is a very intuitive tool that will guide the user through the process of repairing a database.

➲ **To diagnose and repair a database:**

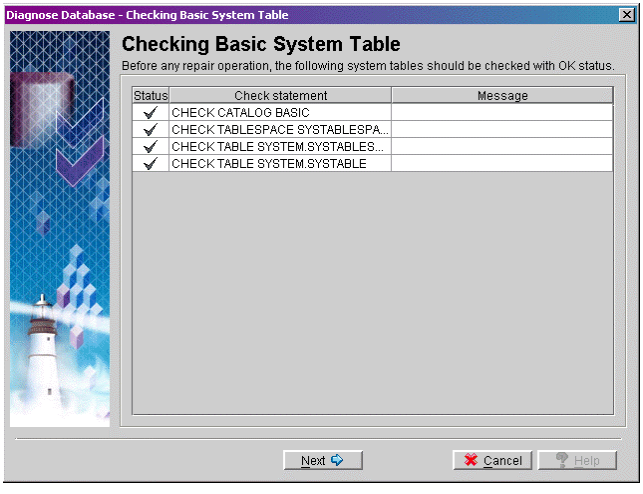
1. Select **Diagnose Database** from the main console or the **Wizards** menu. The **Welcome to the Database Repair Wizard** window appears.



2. Click the **Next** button. The **Choose a Database** window appears.



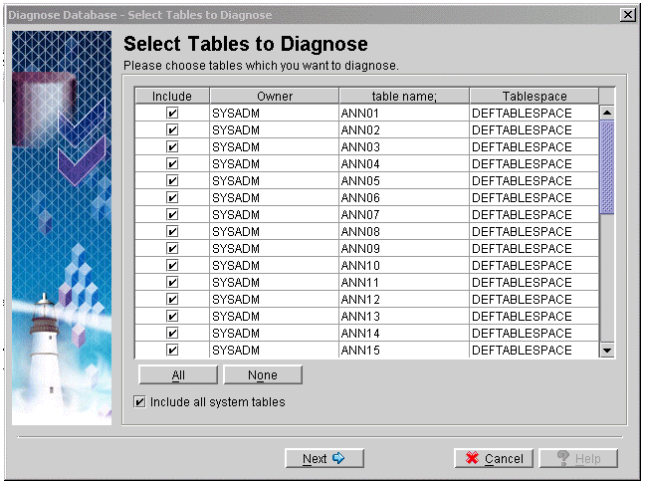
3. Select a database from the **Name** menu.
4. Enter the system administrator's name in the **User name** field.
5. Enter the system administrator's password in the **Password** field.
6. To enable automatic database repair check the **Repair automatically** check box.
7. Click the **Next** button. The **Check Basic System Table** window appears.



8. A check mark will appear in the **Status** field for each system table that is functioning normally. An X will appear in the **Status** field if there is a problem. A message will also appear in the corresponding **Message** field.

NOTE *If any check operation fails in this step, the wizard will not be able to continue.*

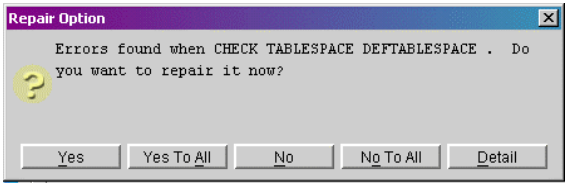
9. Click the **Next** button. The **Select Tables to Diagnose** window appears.



10. Select a table to diagnose by checking the **include** box in the appropriate row. To select all tables click the **All** button.

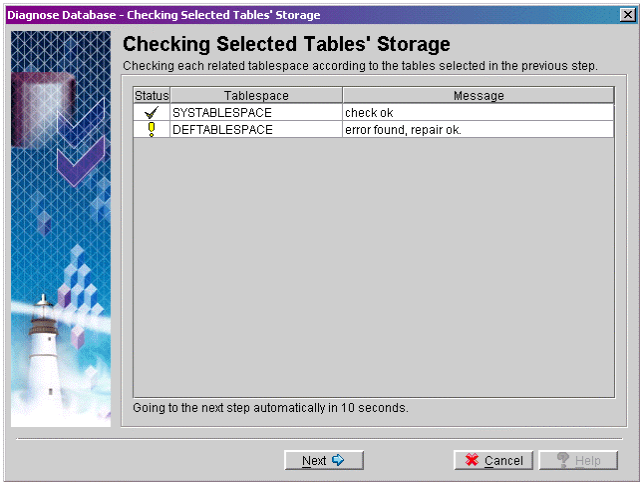
11. Click the **Next** button.

12. The **Checking Selected Tables' Storage** window will appear, displaying the tablespaces containing the selected table. If errors are found, the **Repair Option** dialog box will appear.



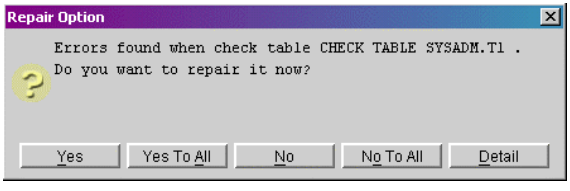
13. Select **Yes** to repair the tablespace in question. Select **Yes To All** to repair all tablespaces with errors. Select **No** to skip repairing the tablespace in question. Select **No To All** to skip repairing all tablespaces with errors. Select **Detail** to

see the error message. After all tablespaces have been repaired or skipped the **Checking Selected Tables' Storage** window will appear.

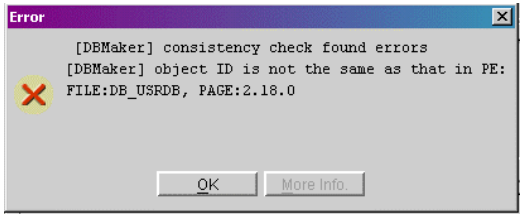


14. Click **Next** or wait 10 seconds, the wizard will automatically go to the next step.

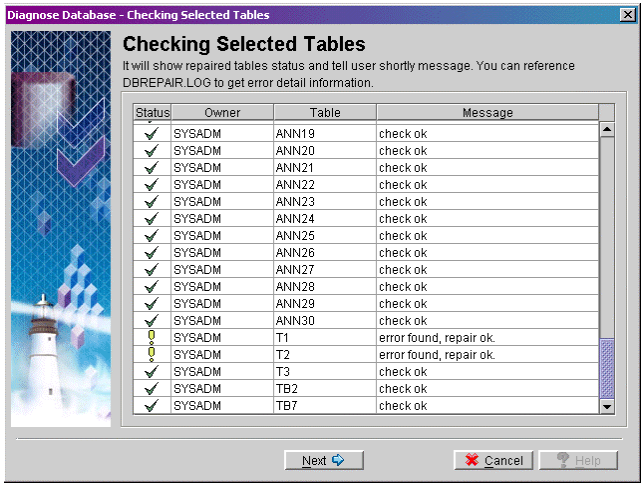
15. DBMaker will check the remaining system tables and selected user tables. If any errors are found, the **Repair Option** dialog box will appear.



16. Select **Yes** to repair the table in question. Select **Yes To All** to repair all tables with errors. Select **No** to skip repairing the table in question. Select **No To All** to skip repairing all tables with errors. Select **Detail** to see the error message. Selecting **Detail** opens the **Error** dialog box.

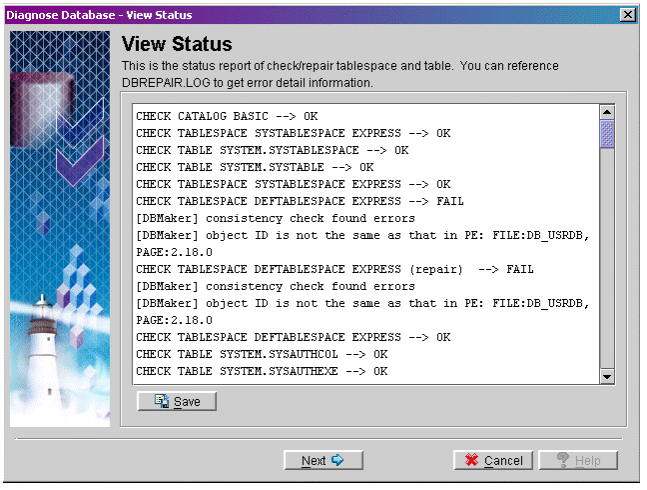


17. Click **OK** to continue checking tables. After all tables have been checked the **Checking Selected Tables** window will open



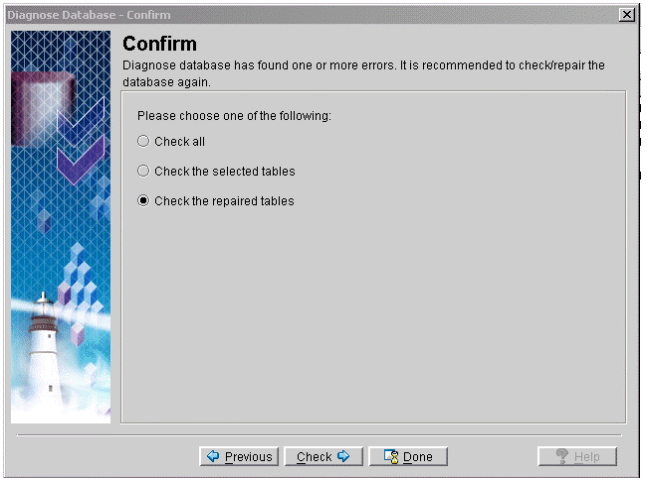
18. The **Checking Selected Tables** window displays the status of the tables checked. Click **Next** to continue. The **View Status** window will open.

NOTE *If errors were found and not repaired in a tablespace, then any tables belonging to the broken tablespace cannot be repaired.*



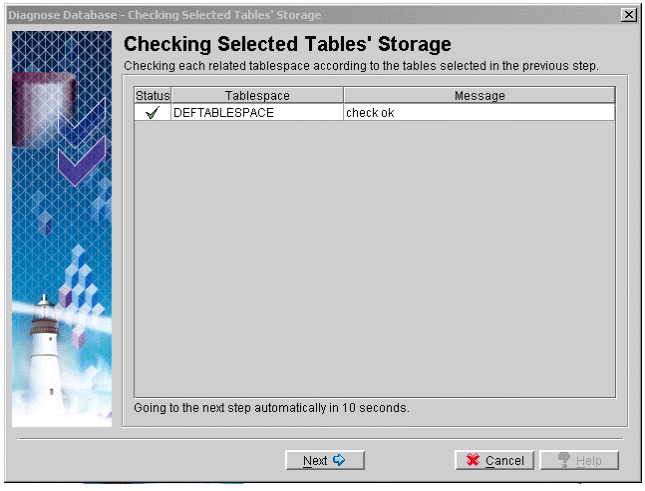
19. The **View Status** window displays a report of the error check. If any repairs have been performed, the **Next** button will appear, otherwise the **Done** button will be displayed. Click **Next** to continue, or **Done** to complete the wizard

20. The **Confirm** window will open.

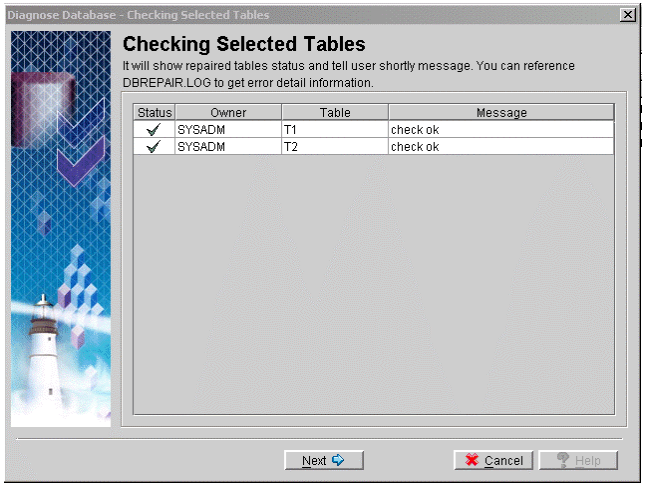


21. The **Confirm** window allows the database to be rechecked to ensure that repairs were effective. Select **Check all** to recheck the entire database. Select **Check the selected tables** to recheck those tables selected initially. Select **Check the repaired tables** to recheck those tables that errors were found in.

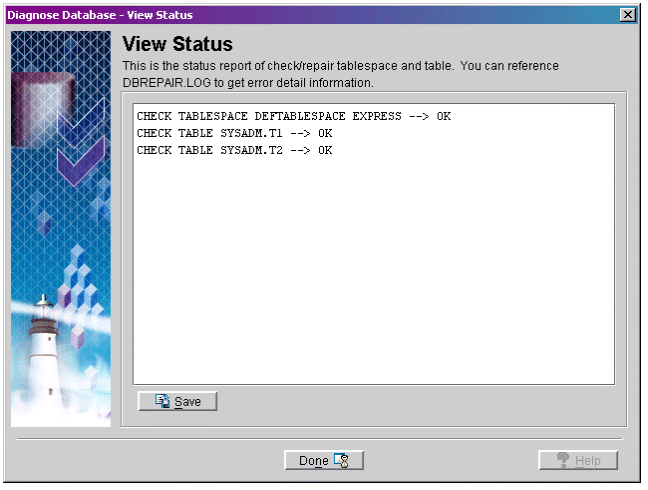
22. The Diagnose Database Wizard will return to the **Checking Selected Tables' Storage** window and will check tablespaces according to the choice made in the Confirm window.



23. Click **Next** or wait 10 seconds, the wizard will automatically go to the next step. DBMaker will recheck the selected tables.



24. Click Next. The View Status window will appear a final time.

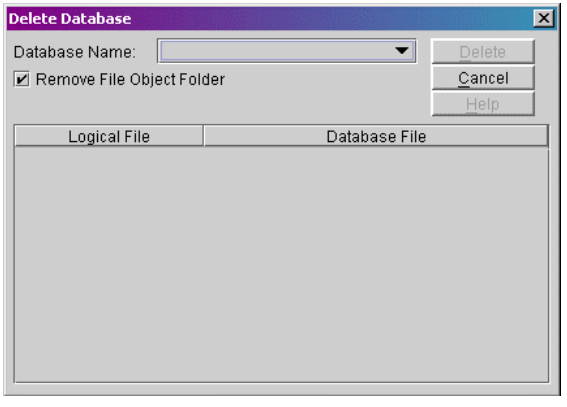


10 Deleting a Database

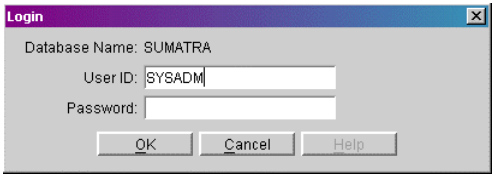
Once you no longer require a database, you can delete it using JServer Manager.

➲ **To delete a database**

1. Select **Delete Database** from the main console. The **Delete Database** window appears.

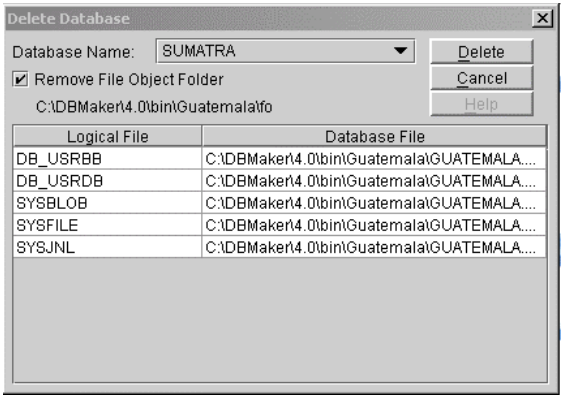


2. Select a database from the **Database Name** menu.
3. The **Login** window is displayed.

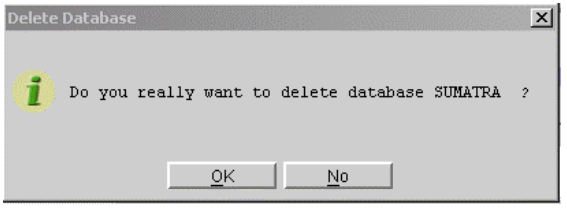


4. Enter SYSADM in the **User Name** field.

5. Click **OK**. The **Delete Database** window is displayed. All operating system files associated with the database are listed.



6. Click **Delete** to remove all of these files as well as information related to the database from the **dmconfig.ini** file. The **Delete Database** dialog box is displayed.



7. Click **OK**. The **Delete Database** dialog box will display, confirming your selection.

8. Click **OK**.

11 Backing up a Database

DBMaker allows the use of different options for backing up a database. In addition to periodic incremental or full backups made by the backup daemons, executing a backup at any time the database is on or off-line is supported.

On-line full backups increase the load on a system's processor and storage, and should only be done when client resource demands are low.

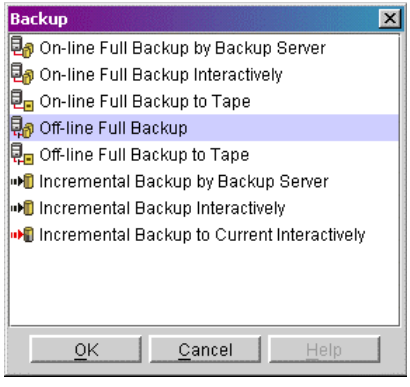


Figure 11-1 The Backup window

You can use the following backup methods.

On-Line Full Backup by Backup Server: JServer Manager automatically backs up the database to the location specified in the dmconfig.ini file. All data in the

database is copied to the backup location. The database must be started and the Backup Server activated. Clients may be connected to the database while on-line backup is being performed. File objects are also backed up when this method is used.

On-Line Full Backup Interactively: The destination of backup files may be specified. File objects are not backed up.

On-line Full Backup To Tape: All data in the database is copied to a single tape. File objects are not backed up.

Off-Line Full Backup: All data in the database is copied to the backup location. The database must not have been started before off-line full backup is performed. File objects are not backed up.

Off-line Full Backup To Tape: All data in the database is copied to a single tape. File objects are not backed up.

Incremental Backup By Backup Server: All journal blocks are copied to the incremental backup location that is specified in the dmconfig.ini file. Incremental backups can be performed while the database is on-line and clients are connected to the database.

Incremental Backup Interactively: All journal blocks are copied to an incremental backup location that may be specified at the time of the incremental backup. Incremental backups can be performed while the database is on-line and clients are connected to the database.

Incremental Backup to Current Interactively: The data in the database is backed up from the point of the most recent backup to the end of the current journal file. The advantage of performing an incremental backup to current is that you can better protect the database against crashes.

11.1 On-line Full Backup by Backup Server

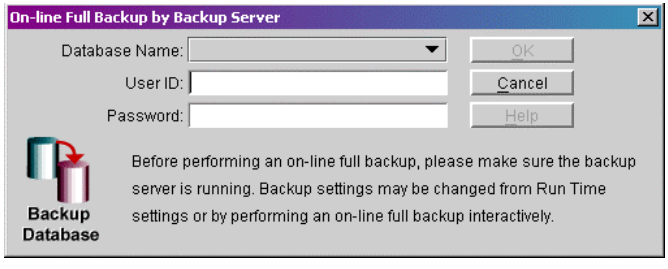
A full backup may be performed quickly and easily while the database is started using the On-line Full Backup by Backup Server. On-line full backups performed by this method are made to the location specified in the configuration file. The backup directory should be located on a disk separate from the disk the database is stored on to prevent loss of data in the case of media failure.

On-line full backup by backup server can be performed on a remote server, and may be used to perform file object backup; on-line full backup interactively is not capable of these functions

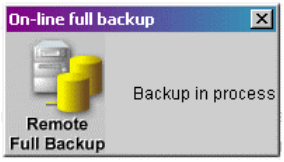
Be sure that the backup server has been activated before using this backup method. If an error message “backup server doesn’t exist” appears, shut the database down and activate the backup server when restarting. For instructions on starting the backup server, refer to sections 4.2 and 4.3.

➲ **To Perform an On-Line Full Backup by Backup Server:**

1. Select **Backup Database** from the main console or the **Database** menu.
2. Select **On-line Full Backup by Backup Server** from the **Backup** window and click **OK**. The **On-line Full Backup by Backup Server** window will open.
3. Select a database from the **Database Name** menu. Enter a user name and password (must be a user with DBA authority or higher).



4. Click **OK**. The **On-Line Full Backup** message box will appear while the backup is in process.



5. The **On-line full backup** message box is replaced by a confirmation dialog box when the backup is complete. Any old backup files will be overwritten. If a directory for previous full backup files has been specified (this must be set from the **Backup** page of the **Start Database Advanced Settings** window), then the old backup files will be copied to the directory of previous backup files.

11.2 On-line Full Backup Interactively

You can perform a backup to a database that has already been started. Performing an on-line backup involves specifying a location for the backup files. You should choose a backup directory location on a separate disk to minimize the risk of loss of data through media failure. You can also change the following incremental backup settings when making an on-line full backup.

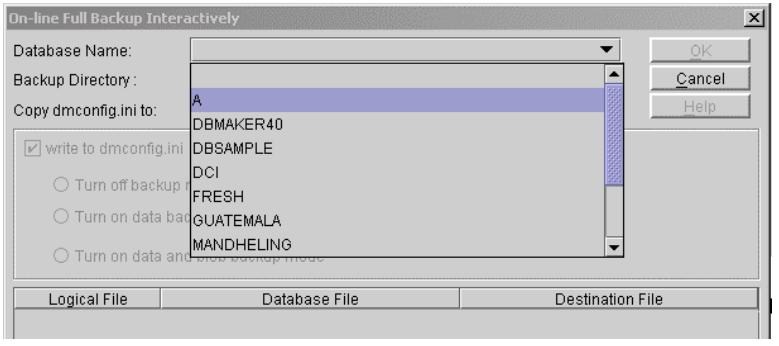
BACKUP MODE	DESCRIPTION
Turn off backup mode	Disables the incremental backup daemon. When backup mode is disabled, journal files are not backed up.
Turn on data backup mode	All data is written to the journal but the incremental backup daemon only backs up non-BLOB data in the journal files.
Turn on data and BLOB backup mode	All data is written to the journal and the incremental backup daemon backs up all journal files.
Write to dmconfig.ini	Saves changes in the incremental backup mode to the **dmconfig.ini** file. The settings will be the same the next time the database is started.

The incremental backups supplement the full backups

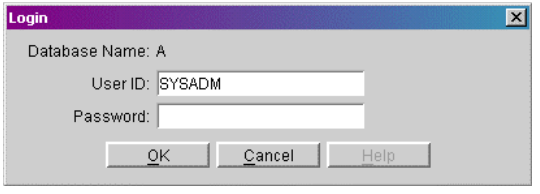
➲ **To perform an On-line Full Backup Interactively:**

1. Select **On-line Full Backup** from the **Backup** window. The **On-line Full Backup Interactively** window appears.

2. Select a database from the **Database Name** menu.



3. The **Login** dialog box is displayed.

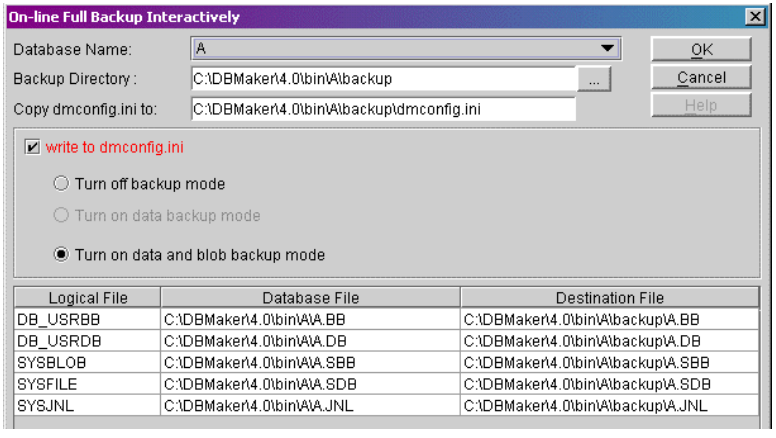


4. Enter your user ID in the **User ID** field.

NOTE *Any user with DBA security privilege may back up the database.*

5. Enter a password in the **Password** field.

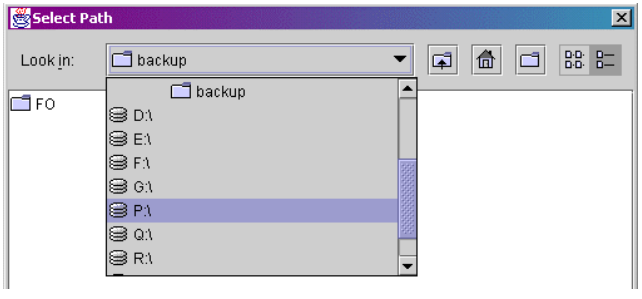
6. Click **OK**. A connection to the database is established. The **On-line Full Backup** window opens again with the list of operating system files to be backed up. The destination file location is the default backup directory specified in the configuration file, **dmconfig.ini**.



7. To select a new path for the backup directory.

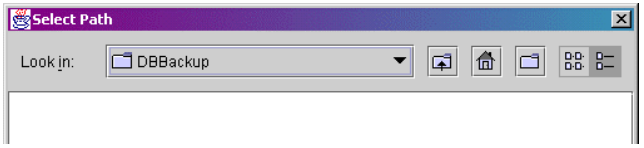
a) Click on the browse button [...]. The **Select Path** window is displayed.

b) To select a new directory or a new disk, click the *Up One Level button* or click the arrow until the desired path's root is visible.

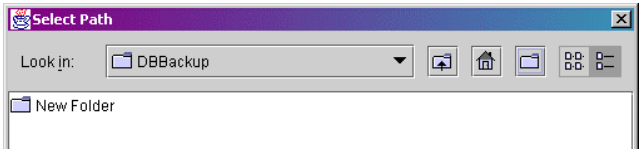


c) Select the desired disk or directory root from the **Look in** menu. A list of directories available will appear.

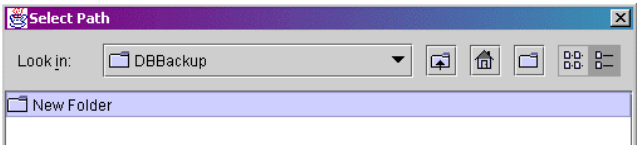
d) Select the desired path down the directory tree by double clicking on the folder icon next to the directory name. The folders in the directory are displayed.



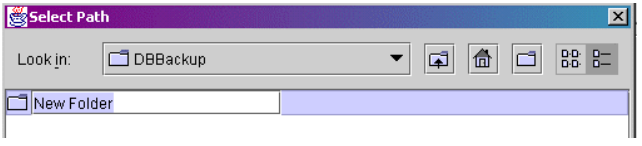
e) To create a subdirectory, click the *Create New Directory* button. A subdirectory **New Folder** appears in the directory list.



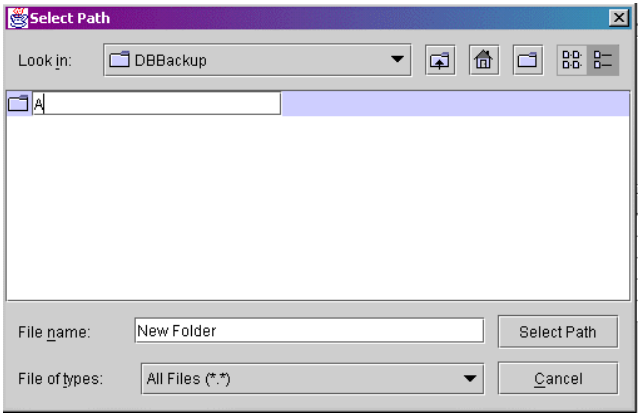
f) Select the **New Folder** button. Click the folder icon for the New Folder icon using the right mouse button.



g) Click the field again



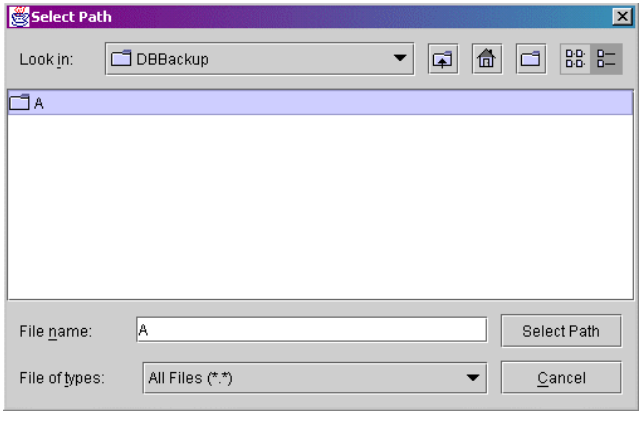
h) Enter a folder name next to the icon.



i) Press ENTER to key to save changes to the subdirectory name. Note that the directory **New Folder** still appears in the **File name** field.

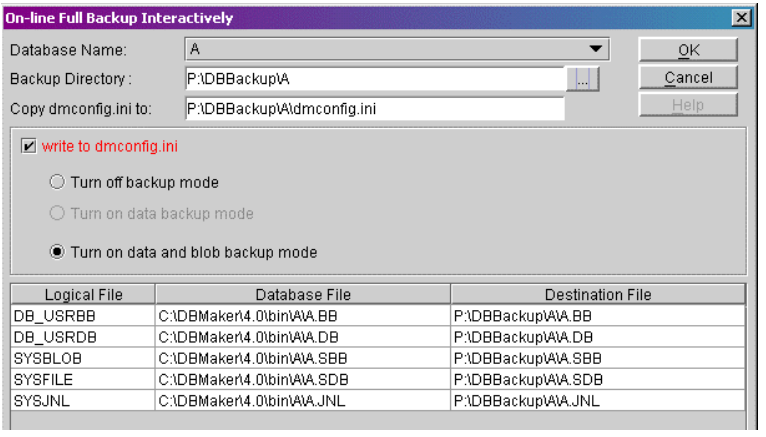
NOTE *The name should appear in the directory list in a position according to alphabetical order.*

j) Select another directory name and then reselect the new backup directory. The backup directory appears in the **File name** field.



k) Click on the **Select Path** button. The **On-line Full Backup** window shows the selected directory path in the **Backup Directory** field.

NOTE *You can also specify a location by entering it in the Backup Directory field.*



NOTE *The database administrator may specify another location for the default backup directory with the JConfiguration Tool. See the JConfiguration Tool Reference for more information.*

8. Change the incremental backup settings.

To turn on data backup mode, make sure that the **Turn on data backup mode** option button is selected.

To turn off backup mode, select the **Turn off backup mode** option button.

To turn on data and blob backup mode, select the **Turn on data and blob backup mode** option button.

9. Click **OK** to save all files to the backup directory. If files with the same name already exist in the backup directory, they will be overwritten. If a directory for previous full backup files has been specified (this must be set from the **Backup**

page of the **Start Database Advanced Settings** window), then the old backup files will be copied to the directory of previous backup file

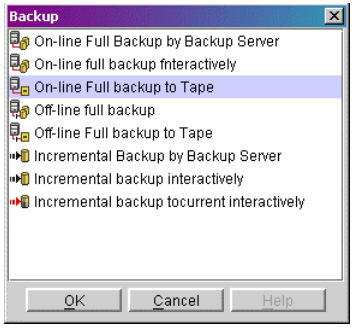
11.3 On-line Full Backup to Tape

When a database is started, you can perform an on-line full backup of your database files to a tape device. You can also change the following incremental backup settings when making an on-line full backup.

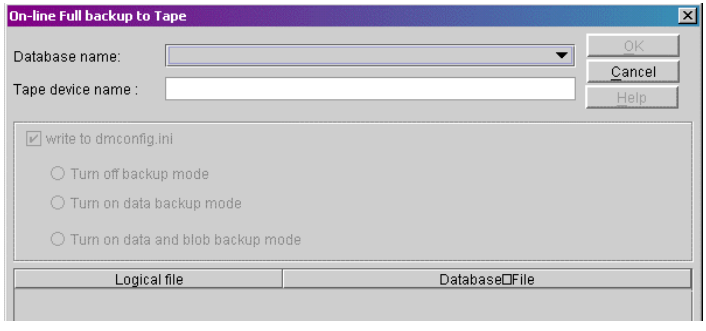
BACKUP MODE	DESCRIPTION
Turn off backup mode	Disables the incremental backup daemon. When backup mode is disabled, journal files are not backed up.
Turn on data backup mode	All data is written to the journal but the incremental backup daemon only backs up non-BLOB data in the journal files.
Turn on data and BLOB backup mode	All data is written to the journal and the incremental backup daemon backs up all journal files.
Write to dmconfig.ini	Saves changes in the incremental backup mode to the **dmconfig.ini** file. The settings will be the same the next time the database is started.

➲ **To perform a full backup to tape:**

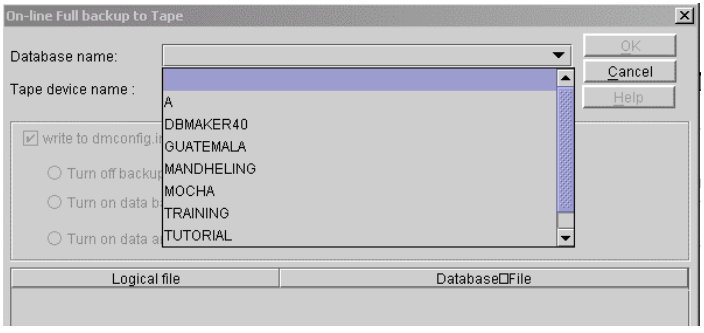
1. Select **Backup Database** from the main console. A list of different backup options is displayed.



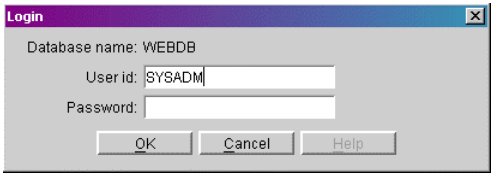
2. Select **On-line Full Backup to Tape** from the **Backup** window. The **On-line Full Backup to Tape** window opens.



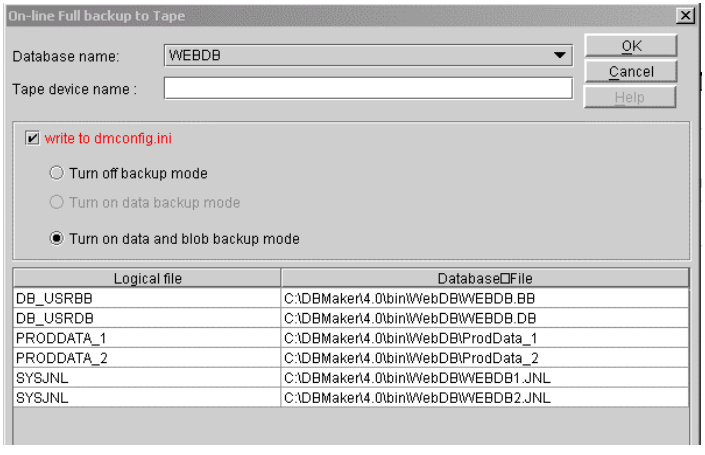
3. Select a database by clicking on the field next to **Database Name**. A drop-down list of databases available on the server will appear.



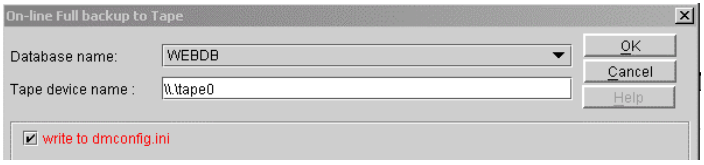
4. JServer manger will prompt you to log on to the database.



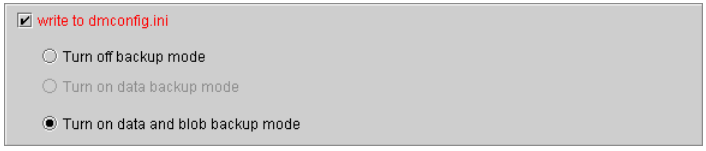
5. Enter a User ID and password into the appropriate fields.
6. Click **OK**. The On-Line Full Backup to Tape window will reopen.
7. All files to be backed up will appear in the **Database File** list.



8. Enter the path of the tape device in the **Tape Device Name** field.



9. Change the incremental backup settings:



To turn on data backup mode, make sure that the **Turn on data backup mode** option button is selected.

To turn off backup mode, select the **Turn off backup mode** option button.

To turn on data and blob backup mode, select the **Turn on data and blob backup mode** option button.

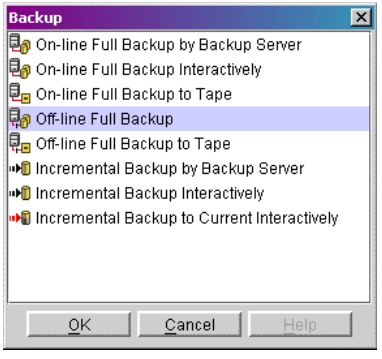
10. Click **OK**. The database will be copied to tape.

11.4 Off-line Full Backup

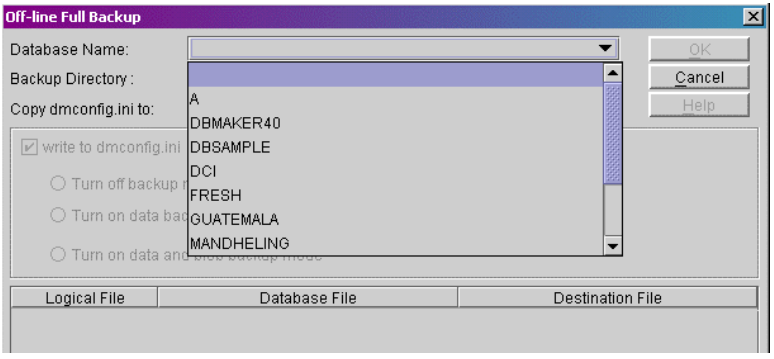
You can perform a backup to a database that is not yet started. Performing an off-line backup involves specifying a location for the backup files. You should choose a backup directory location on a separate disk to minimize the risk of loss of data through media failure.

➲ **To perform an Off-line Full Backup:**

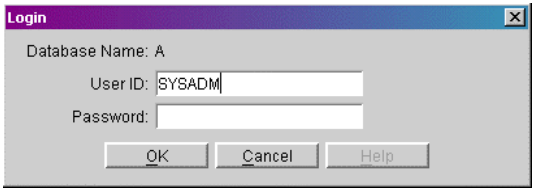
1. Select **Backup Database** from the main console. A list of different backup options is displayed.



2. Select **Off-line Full Backup** from the **Backup** window. The **Off-line Full Backup** window appears.
3. Select a database from the **Database Name** menu.



4. JServer Manager will prompt you to log onto the database.

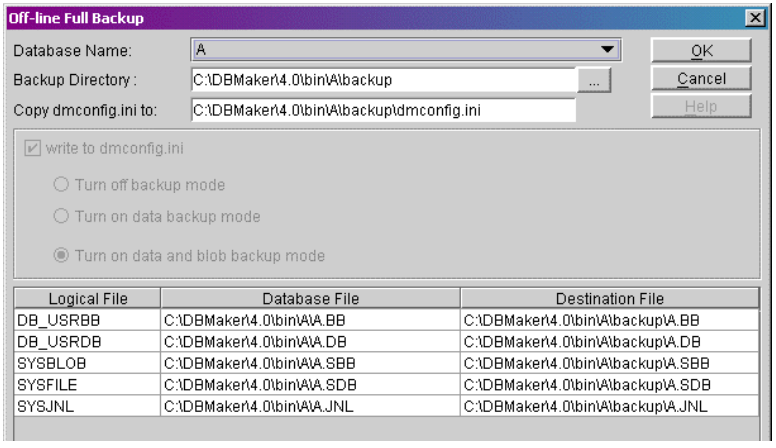


5. Enter your User-ID in the **User ID** field.

NOTE *Any user with the DBA security privilege can back up the database.*

6. Enter a password in the **Password** field.

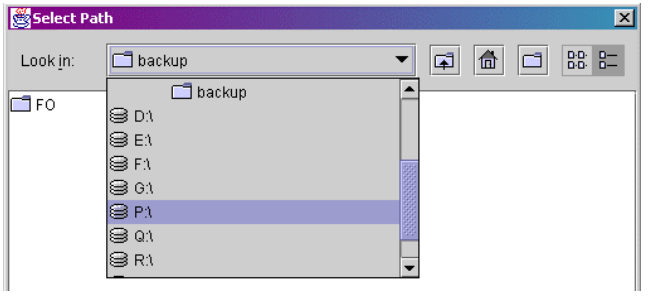
7. Click **OK**. A single user connection is established to the database. The **Off-line Full Backup** window is displayed with a list of operating system files to be backed up. The destination file location is the default backup directory specified in the configuration file, **dmconfig.ini**.



8. To select a new path for the backup directory:

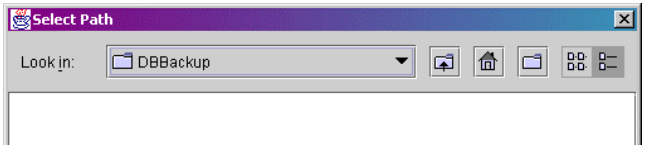
a) Click the **Browse** button [...]. The **Select Path** window is displayed.

b) To select a new directory or a new disk, click the *Up One Level button* until the desired path's root is available.

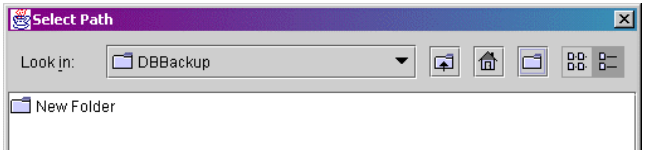


c) Select the desired disk or directory root from the **Look in** menu. A list of directories available appears.

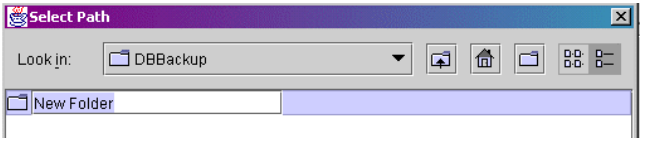
d) Select the desired path down the directory tree by double clicking a folder icon next to the directory name. The folders in the directory are displayed.



e) To create a subdirectory, click the *Create New Directory* button. A subdirectory **New Folder** appears in the directory list.

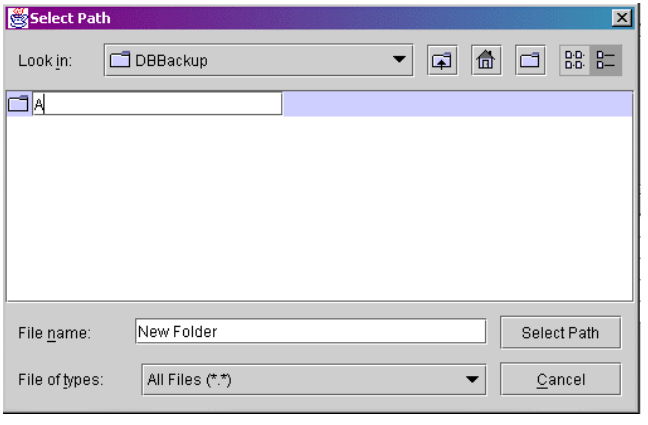


f) Select the **New Folder** item.



g) Click the folder icon for the **New Folder** item using the right mouse button.

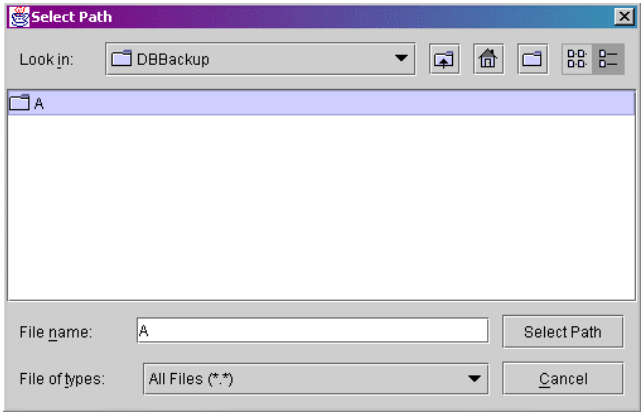
h) Enter a folder name next to the icon.



i) Press the ENTER key to save changes to the subdirectory name. Note that the directory **New Folder** still appears in the **File Name** field.

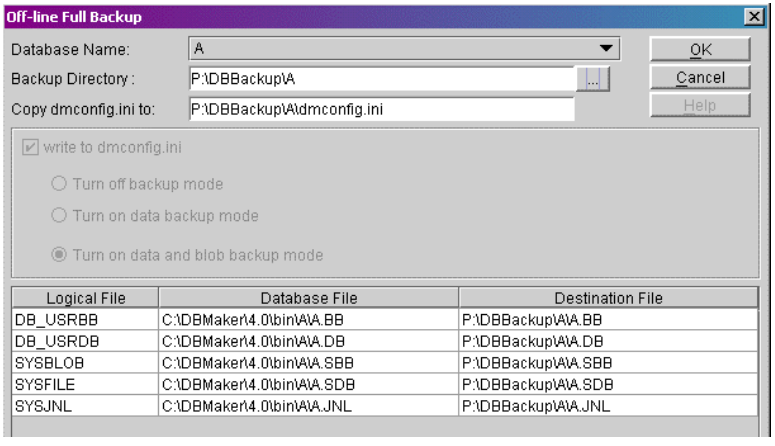
NOTE *The name should appear in the directory list in a position according to alphabetical order*

j) Select another directory name and then reselect the new backup directory. The backup directory appears in the **File Name** field



k) Click on the **Select Path** button. The **Off-line Full Backup** window shows the selected directory path in the **Backup Directory** field.

NOTE *You can also enter a new valid path in the* ***Backup Directory*** *field.*



9. Select OK to save all files in the backup directory.

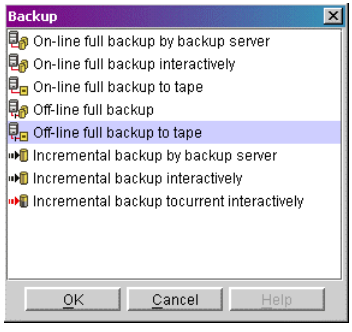
NOTE *If the files already exist in the backup directory, the Database Administrator may choose to overwrite them.*

11.5 Off-line Full Backup to Tape

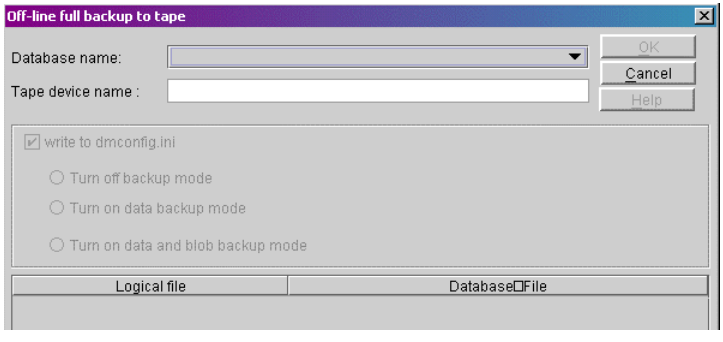
As well as backing up your database to another file location, you can back it up to tape. Performing an off-line backup involves specifying a location for the backup files. Performing backups to tape minimizes the risk of loss of data through media failure.

➲ **To perform an off-line full backup to tape:**

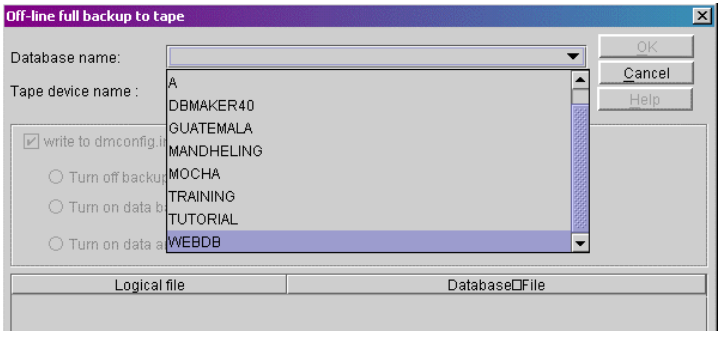
1. Select **Backup Database** from the main console. A list of different backup options is displayed.



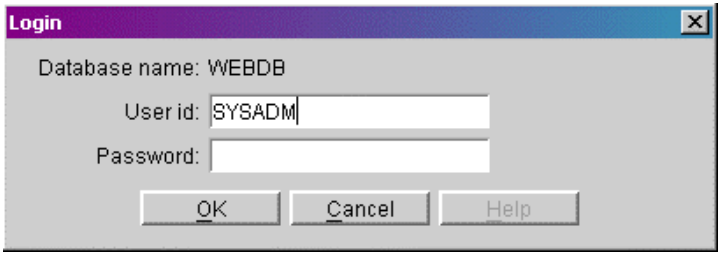
2. Select **Off-line Full Backup to Tape** from the **Backup** window. The **Off-line Full Backup to Tape** window opens.



3. Select a database from the **Database Name** menu.



4. The **Login** window is displayed.

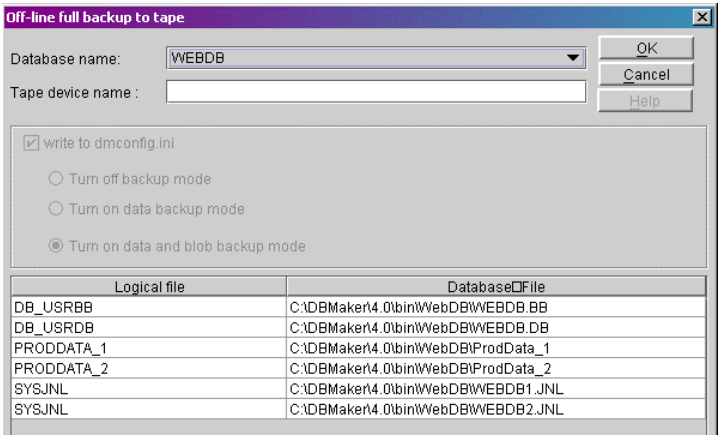


5. Enter your user ID in the **User ID** field.

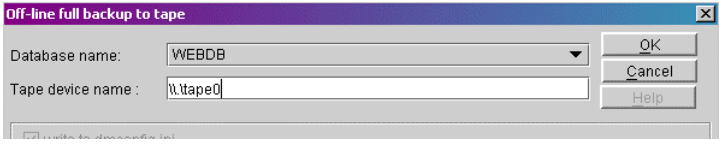
NOTE *Any user with DBA security privilege may back up the database.*

6. Enter your password in the **Password** field.

7. Click **OK**. The **Off-line Full Backup to Tape** window opens again with the list of operating system files to be backed up.



8. Enter the device name in the **Tape Device Name** field.



9. Click **OK** to execute the full backup.

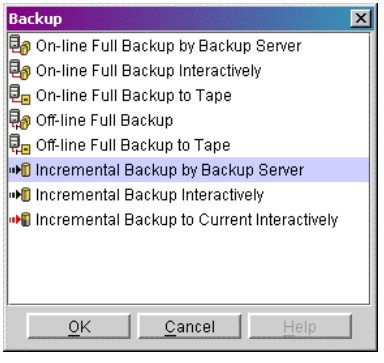
11.6 Incremental Backup by Backup Server

Incremental backups differ from full backups in that they only copy full journal files to the backup destination directory. To allow a database to recover its files, it is necessary to perform a full backup before an incremental backup.

An incremental backup may be performed quickly and easily while the database is started using the On-line Incremental Backup by Backup Server. Incremental backups performed by this method are made to the location specified in the configuration file. The backup directory should be located on a disk separate from the disk the database is stored on to prevent loss of data in the case of media failure.

➲ **To Perform an Incremental Backup by Backup Server:**

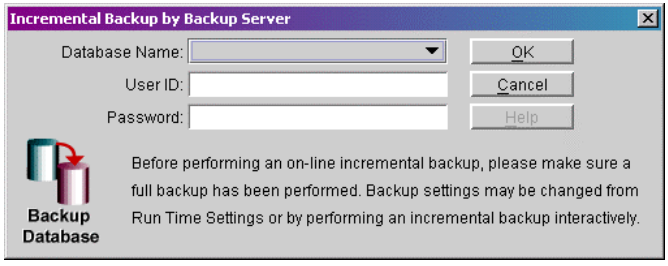
1. Select **Backup Database** from the main console or the **Database** menu. The Backup window will appear.



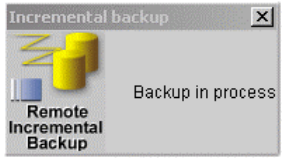
NOTE *Be sure that the backup server has been activated before using this backup method. If an error message "backup server doesn't exist" appears, shut the database down and activate the backup server when restarting. For instructions on starting the backup server, refer to sections 4.2 and 4.3.*

2. Select **Incremental Backup by Backup Server** from the **Backup** window and click **OK**. The **Incremental Backup by Backup Server** window will open.

3. Select a database from the **Database Name** menu. Enter a user name and password (must be a user with DBA authority or higher).



4. Click **OK**. The **Incremental backup** message box will appear while the backup is in process.



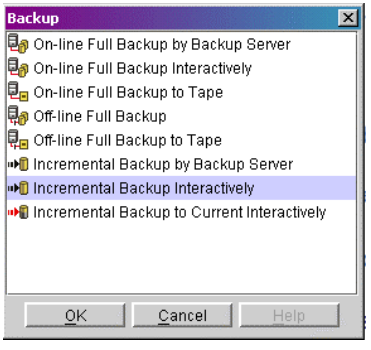
5. The **Incremental backup** message box is replaced by a confirmation dialog box when the backup is complete.

11.7 Incremental Backup

Incremental backups differ from full backups in that they only copy full journal files to the backup destination directory. To allow a database to recover its files, it is necessary to perform a full backup before an incremental backup. The incremental backup daemon can be set to automatically copy journal files when they have been filled to a set capacity. In this way, it handles all the journal files itself and makes sure that the required data is backed up. It is possible to change the destination file location for manually performed incremental backups (See below), but not recommended. Backup journal files are stored in a location indicated on the **Backup** page of the **Advanced Settings** windows, and ideally should be stored in the same directory as the full backup. If you have not started the backup server, shut the database down and restart it with this setting enabled. For more information, refer to Start Backup Server in section 4.2, or refer to the *Database Administrators Guide*.

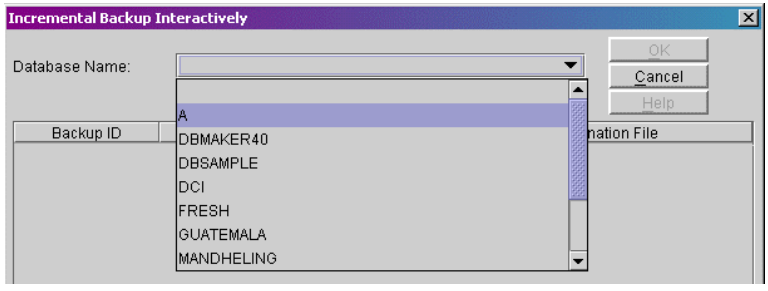
➲ **To perform an incremental backup:**

1. Select **Backup Database** from the main console. A list of different backup options is displayed.

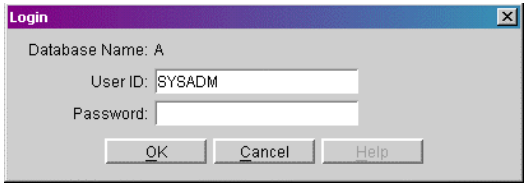


2. Select **Incremental Backup** from the **Backup** window. The **Incremental Backup** window appears.

3. Select a database from the **Database Name** menu.



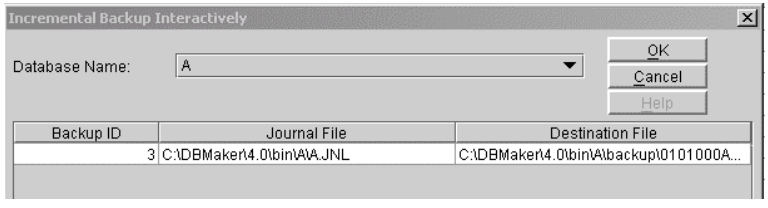
4. The **Login** window appears.



5. Enter your user ID in the **User ID** field.

6. Enter your password in the **Password** field.

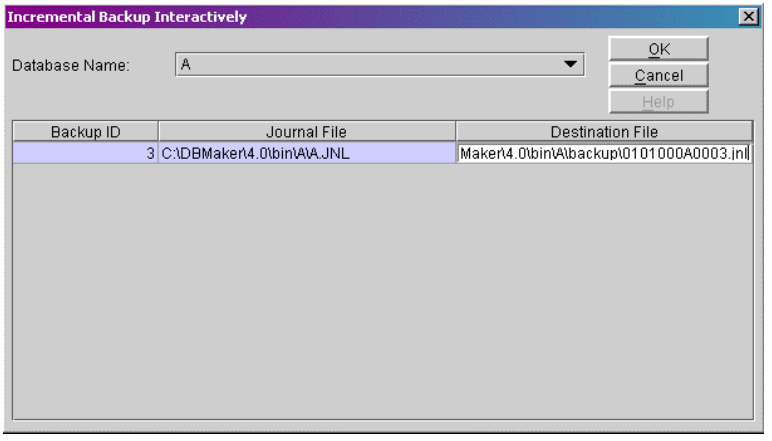
7. Click **OK**. A single-user connection is established to the database. The **Incremental Backup Interactively** window displayed displaying the journal file and destination file locations.



8. To specify a different location for the Destination File:

a) Click on the **Destination File** field.

b) Enter the full path in the **Destination File** field.



c) Press ENTER to ensure that the new destination file path is selected.

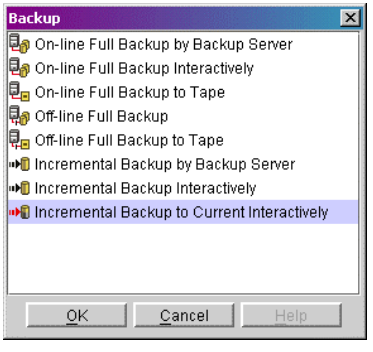
9. Click **OK** to execute the incremental backup.

11.8 Backup to Current Journal File

Incremental backup to current copies all journal files to the backup directory, including the journal file that is currently being used. Incremental backups differ from full backups in that they only copy full journal files to the backup destination directory. To allow your database to recover its files, perform a full backup before an incremental backup. The incremental backup daemon can be set to automatically copy journal files when they have been filled to a set capacity. In this way, it handles all the journal files itself and makes sure that the required data is backed up. It is possible to change the destination file location for manually performed incremental backups (See below), but not recommended. Backup journal files are stored in a location indicated on the **Backup** page of the **Advanced Settings** windows, and optimally are stored in the same directory as the full backup. For more information, refer to Start Backup Server, or refer to the Database Administrators Reference.

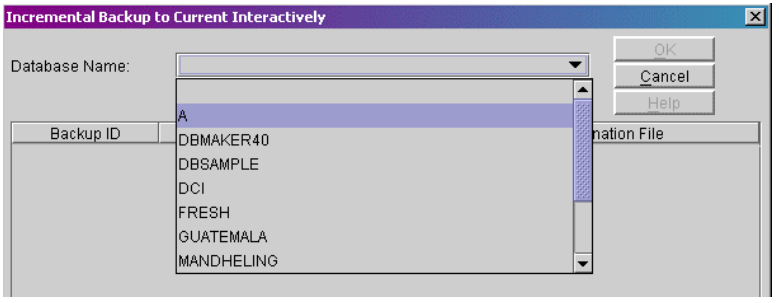
➲ **To perform an incremental backup to the current journal file:**

1. Select **Backup Database** from the main console. A list of different backup options is displayed.

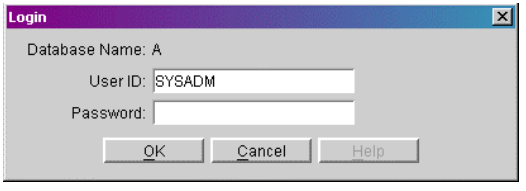


2. Select **Incremental Backup to Current Interactively** from the Backup window. The **Incremental Backup to Current Interactively** window is displayed.

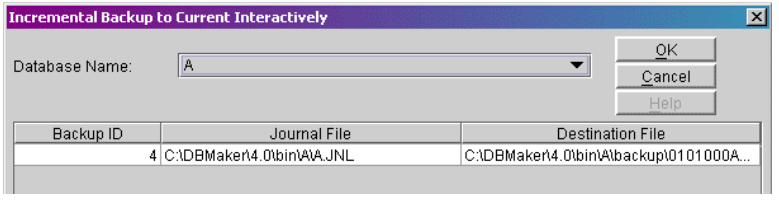
3. Select a database from the **Database Name** menu.



4. The **Login** window appears.

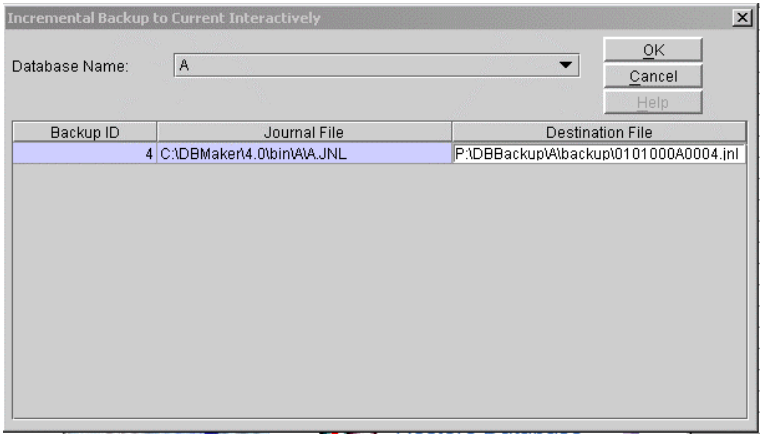


5. Enter your user ID in the **User ID** field.
6. Enter your password in the **Password** field.
7. Click **OK**. A single-user connection is established to the database. The **Incremental Backup to Current Interactively** window opens displaying the journal file and destination file locations.



8. To enter a new location of the database file from the default, choose a location.
   a) Click the **Destination File** field.

b) Enter a destination file path in the **Destination File** field.



c) Press ENTER to ensure that the new destination file path is selected.

9. Click **OK** to execute the incremental backup.

12 Managing Log Files

DBMaker keeps a set of log files that record different aspects of each database's history. In JServer Manager, you can view all log files in a database as well as save them to a location. If you wish to delete log files to manage the size of your database, you can clear them from the database.

The log file includes a date and time stamp with a character string. You can view logs of the following database functions.

Error Log (**error.log**): The error log is a record of all critical error messages returned by DBMaker that are associated with the database.

Asynchronous Table Replication Log (**atrp.log**): This log is a record of all OBDC commands that are made to target databases by the distributor daemon. The Asynchronous Table Replication log is stored in the source database and contains information about all target databases of the source database.

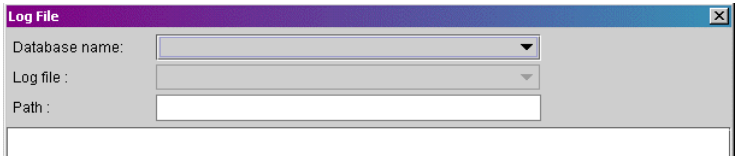
Asynchronous Table Replication Error Log (**atrperror.log**): This log file keeps a record of all errors returned by target databases upon execution of ODBC commands by the distributor daemon. The Asynchronous Table Replication Error log is stored in the source database and contains information about all target databases of the source database.

RP.LOG (**rp.log**): The replication log file keeps a record of all backup journal file updates made from the source database to slave databases. The replication log file is stored in the source database and contains information about all slave databases of the source database.

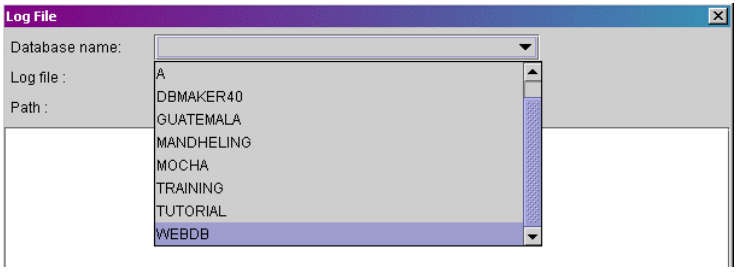
Backup History Log (**dmbackup.his**): The backup history log keeps a record of the time and date of all backup functions applied to the database. This includes full and incremental backups.

➲ **To view log files:**

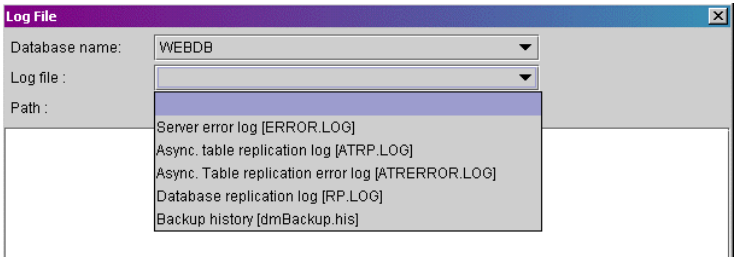
1. Select **Log File** from the **Database** drop down menu to open the **Log File** window.



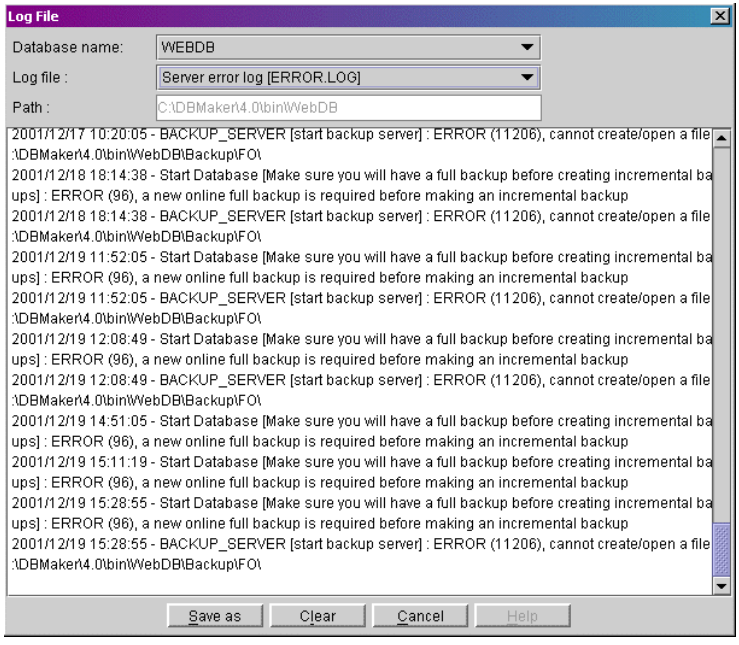
2. Select a database from the **Database Name** menu.



3. Select a log file to view from the **Log File** menu.

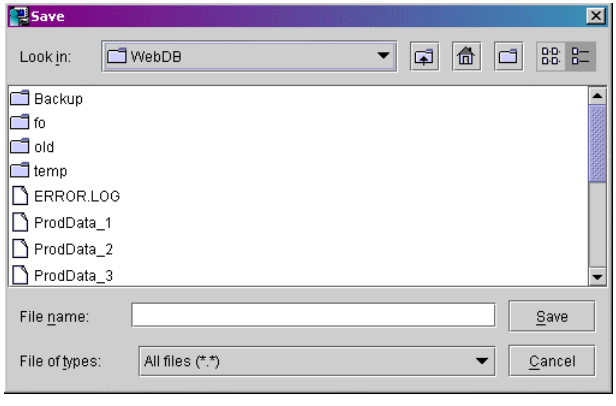


4. To browse through the records, use the scroll bar top the right. The DBA may also choose to save the log file to a different location (this does not move the file, only copies it), or clear the contents of the log file by selecting **Save** or **Clear** from the bottom of the **Log File** window.

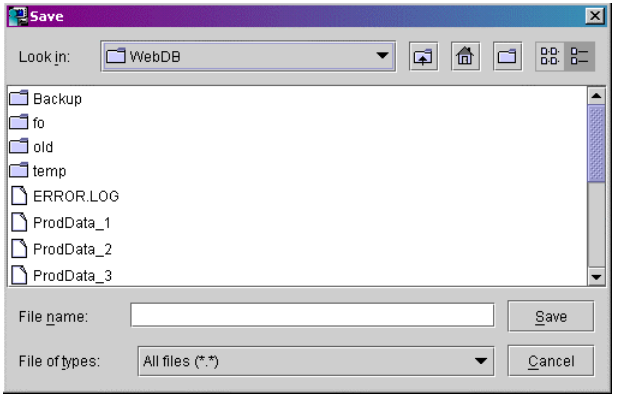


➲ **To save a log file to a different location:**

1. Select a log file.
2. To save the log file to a different location, click the **Save As** button from the bottom of the **Log File** window. The **Save** window appears.



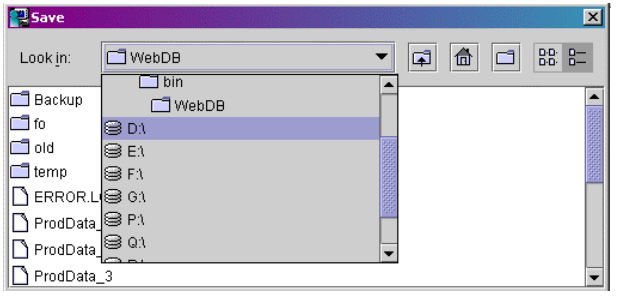
3. To preserve the file name, select the file name of the file being viewed from the list of files.



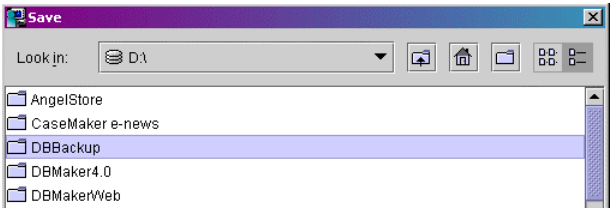
4. The database administrator can also choose to enter a new name in the **File Name** field.

5. To select a new directory or a new disk, click the *Up One Level button* button until the desired path's root is available,

6. To select another disk location for the log file, choose a location from the **Look In** menu.



7. Select the desired disk or directory root. A list of directories available appears.



8. Select the desired path down the directory tree by double clicking on the folder icon next to the directory name, a list of files contained in the directory appear.

9. To overwrite an existing file,

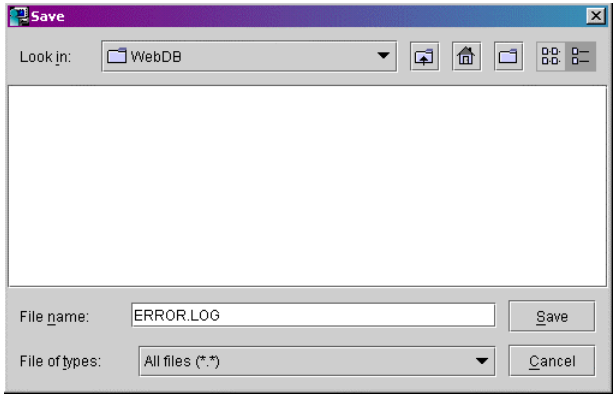
a) Select a file in the directory.

b) Click the **Save As** button. The file is overwritten,

10. To rename a log file:

a) Enter a name in the **File Name** field.

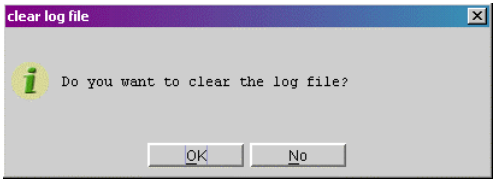
b) Select **Save** to save the log file in the current directory.



NOTE *Log files are necessary for proper database function. Saving log files to different locations or clearing necessary log files interferes with database restoration. If you want to clear your log files to manage the size of your database, you must first back up your log files.*

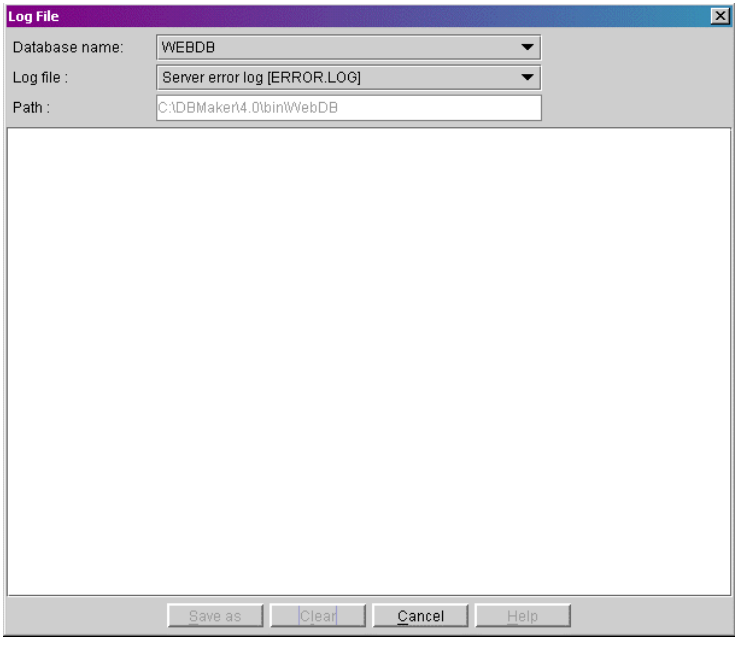
➲ **To clear the log file**

1. Select **Clear** from the bottom of the **Log File** window. JServer Manager will prompt the user to confirm this action.



NOTE *Clearing log files cannot be undone.*

2. Select **OK** to clear the Log File. The contents of the log file will be deleted.



13 Restoring a Database

The database administrator may find it necessary to restore all data from backup files if an unrecoverable error has occurred in the database. You can perform a database restoration from disk or from tape. Restoring a database from tape will recreate the database as it existed at the time of the most recent full backup. If you restore the database from disk, you can restore the database to the time of the last incremental backup.

NOTE *For more information on database restoration, refer to the Database Administrators Guide.*

The following subsections give procedures for both restoration methods.

13.1 Restoring a Database from Disk

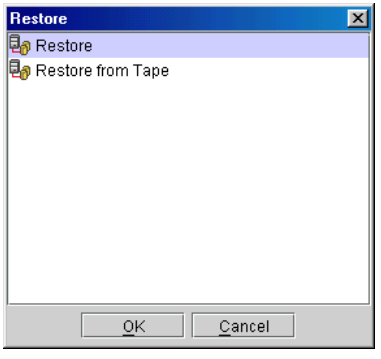
You can restore a database that has been backed up to another disk location on your computer or on the network.

NOTE *To restore a database you must have DBA privilege.*

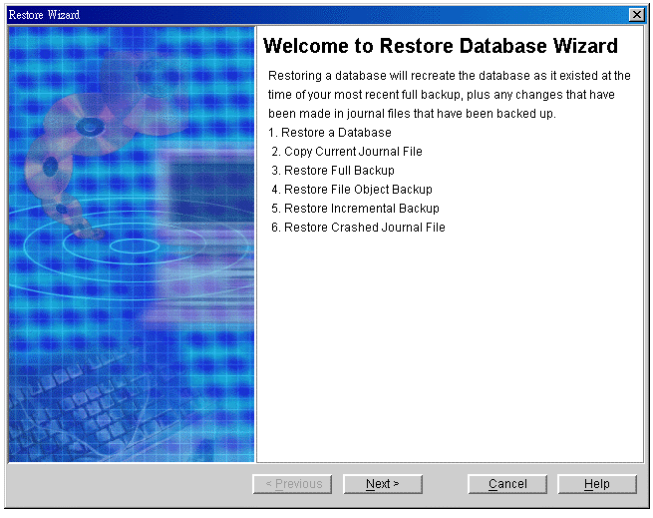
➲ **To restore a database from disk:**

1. Select **Restore Database** from the main console.

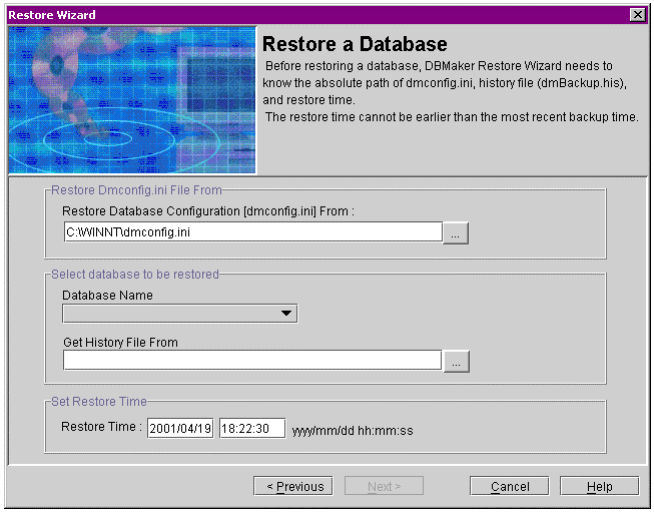
   NOTE *You can also select Restore Database from the Database menu.*



2. Make sure that **Restore** is selected from the **Restore** window.
3. Click **OK**. The **Restore Database** window is displayed.

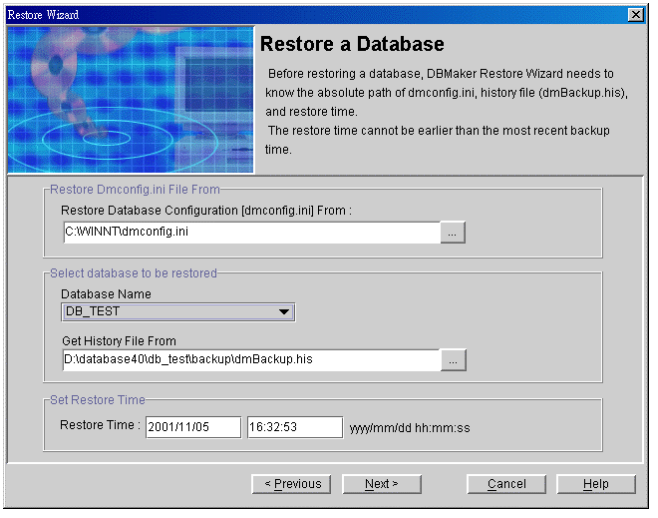


4. Click **Next**. The **Restore Wizard** window is displayed.



5. Select a database by clicking on the field under **Database Name**. A drop-down list of databases available on the server appears.

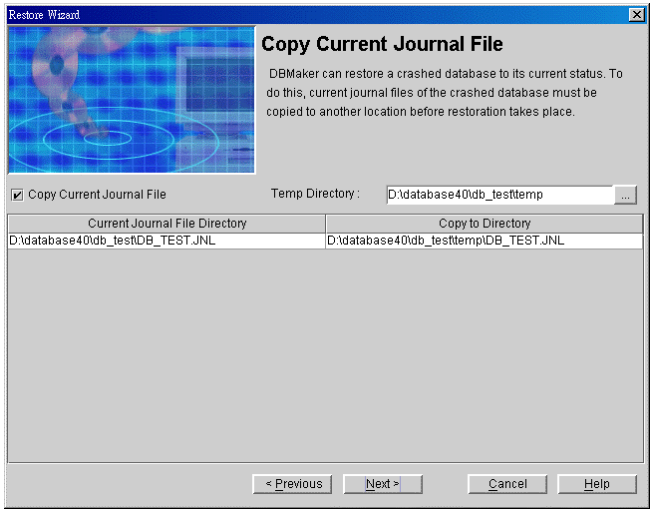
6. Select a database. The **Restore Database Configuration [dmconfig.ini] From** field shows the default location of the configuration file. If you moved the dmconfig.ini



7. If you moved the backup history file to a new location, then enter the new path or click on the browse button ... in the **Get History File From** field.

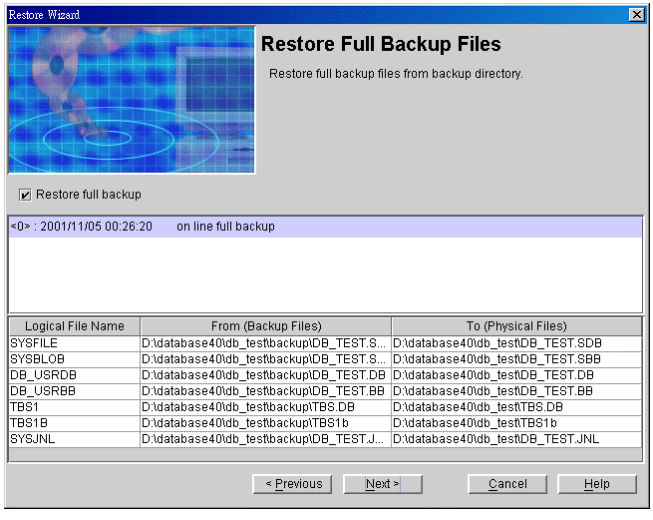
NOTE *After the database to be restored has been chosen, the location of the history log file should appear in the Get History File From field.*

8. Click Next. The **Copy Current Journal File** page is displayed. The current journal file is then copied to a temporary directory so that the database can be restored to the condition it was in just before shutdown or failure. If the current storage media is unstable, you can specify another location for storing the current journal file in the **Copy to Directory** column.



9. Make sure that the correct temporary directory location for the current journal file is displayed in the **Temp Directory** field.

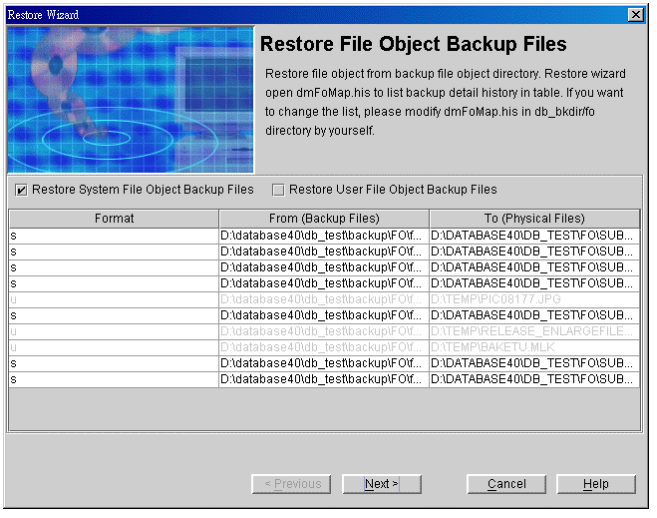
10. Select a backup that contains the files you wish to restore from the top list. The logical file names, backup files, and physical files are displayed in the bottom list.



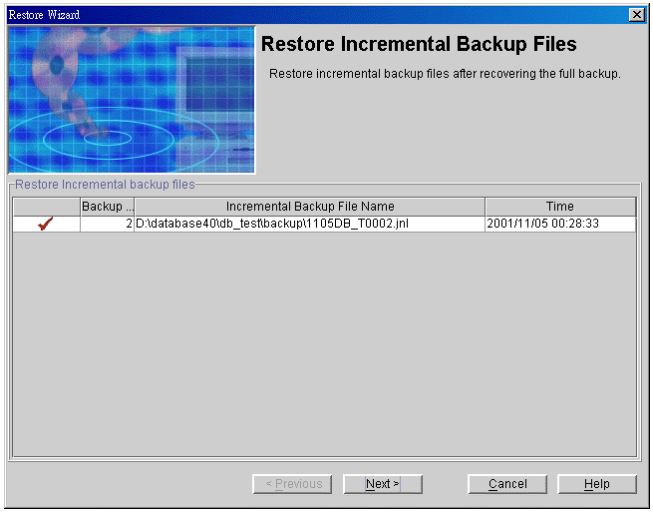
11. Click **Next**. The **Restore File Object Files** window will open.

NOTE *The **Restore File Object Files** window only opens if file objects were previously backed up. Otherwise, the **Restore Incremental Backup Files** page will open*

12. Select whether to restore system file objects or both system and user file objects by checking the appropriate box. File objects that will be restored appear black in the list; file objects that will not be restored appear gray.

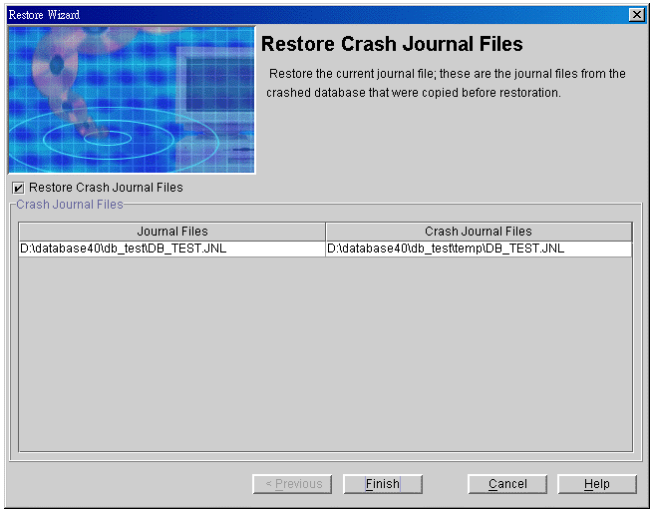


13. Click **Next**. The **Restore Incremental Backup Files** page will open.



14. The **Incremental Backup File Name** field shows all the incremental backup files. If you need to change the incremental backup file path, enter a path in the **Incremental Backup File Name** field. You can edit the file path but must not skip restoration of any of the files.

15. Click **Next**.



16. If you need to restore journal files created after the last incremental backup was made to the database, make sure that the **Restore Crash Journal Files** check box is selected. Clearing this check box will prevent you from restoring the journal files after the last incremental backup.

17. Enter the backup journal file path and file name in the **Crash Journal Files** field.

NOTE *The current journal files that were copied into a temporary directory are used to restore the database to its condition just prior to shutdown.*

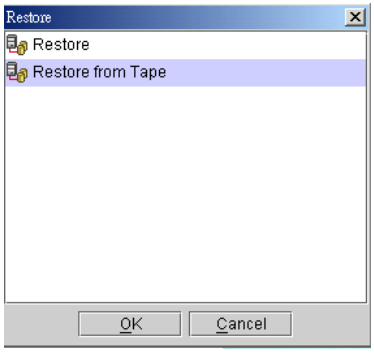
18. Click **Finish**. Restoration is complete.

13.2 Restoring a Database from Tape

You can restore a database that has been backed up to a tape device. You may choose to restore the backup history log and **dmconfig.ini** file from tape. Restoring the backup history log from tape will overwrite the current history log. All records of incremental backups made since the last full restoration to tape will be lost. It will not be possible to restore the database to a more recent status using the incremental files after the backup history log is copied from tape. To restore a database, you must have DBA authority or higher.

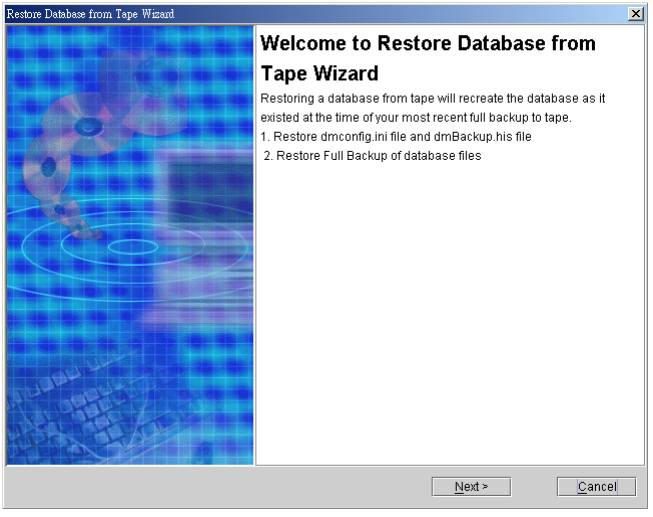
➲ **To restore a database from tape:**

1. Select **Restore Database** from the main console. The **Restore** window is opened.

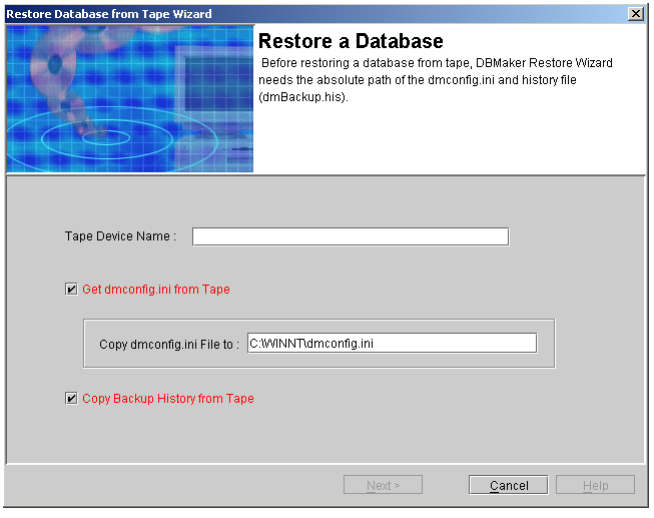


NOTE *You can also select Restore Database from the drop-down menu. The Restore window opens.*

2. Select **Restore from Tape** from the **Restore** window and click **OK**. The **Restore Database from Tape Wizard** starts.



3. Click **Next**. The **Restore a Database** page is displayed.
4. Enter the name of the tape device in the **Tape Device Name** field.
5. Insert the tape from which the database is to be restored.



6. To get the **dmconfig.ini** file from tape, make sure that the **Get dmconfig.ini from Tape** check box is selected.

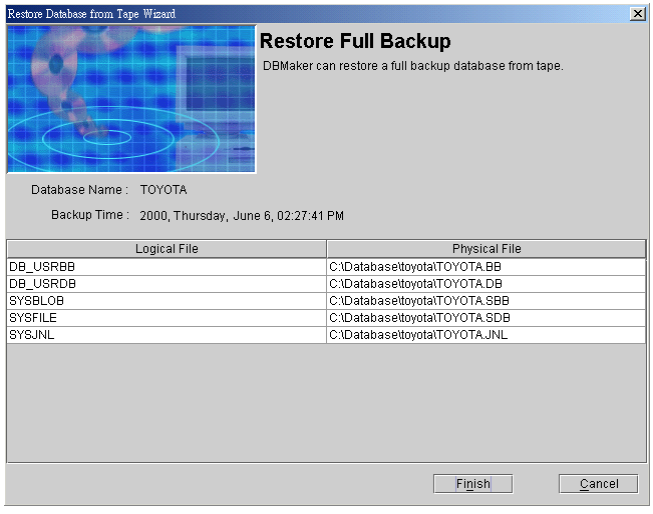
7. To get the **dmconfig.ini** file from disk, clear the check box.

8. To copy the **dmconfig.ini** file to different location, enter it in the **Copy dmconfig.ini File to** field.

9. To copy the backup history file from tape, make sure that the **Copy Backup History from Tape** check box is selected.

10. To copy the Backup History file from another source, clear the **Copy Backup History from Tape** check box.

11. Click **Next**. The name of the database and a list of the files in the database appear. The logical files should map to physical destination locations.



12. Click the **Finish** button to restore the database.

Glossary

Asynchronous Table Replication

A set of columns selected to form a projection that is replicated to the destination tables of another database. Asynchronous table replication occurs according to a schedule. Changes made to the source table are saved to a log file that is used to periodically update the destination tables.

BLOB

Acronym for Binary Large Objects. A table that consists of one column of LONGVARCHAR or LONGVARBINARY type data.

BLOB Frames

Measurement unit used to allocate disk space for BLOB type data.

Concurrency Control

A system of locks placed on objects to prevent multiple users from simultaneously manipulating the same data sets.

Coordinator Database

In a distributed database environment, the database that the client connects to. If the client accesses data from another database then that database is a participant database.

Daemon

A routine that automatically executes at a set time interval.

Data Pages

Data measurement unit for allocation of disk space.

Destination Database

The database that receives data for table replication. A database containing destination tables for a synchronous or asynchronous replication.

Destination Tables

Tables that receive replicated data from a source table. The table on the destination database that the data is replicated to.

Distributed Database Environment

A system of networked remote databases that allow for any table in the system to be accessed by a client on any of the participant (remote) databases.

Exclusive Lock (X Lock)

An access block placed on a database object that prevents other users from accessing the object

Foreign Key

A column or set of columns whose rows contain the same values as the set of columns in the primary key or unique index of another table.

Fragment

Also called a horizontal partition, a fragment is the replication of a given range of data tuples.

Journal Blocks

Internal data measurement unit (514 bytes) that DBMaker uses to manage journal data.

Journal Buffer

Upper memory where current journal blocks are stored before writing them to disk.

Journal Pages

Measurement unit for allocation of disk space.

Lock

Locks allow only one user update and delete permission on the locked object at the time the object is locked.

Page Buffer

Upper memory allocated for data pages accessed by a user.

Participant Database

In a distributed database environment, a database that is accessed by the client through a coordinator database.

Primary Database

The source database for database replication.

Primary key

A column or set of columns in a table that contain values that uniquely identify

the rows in the table.

Projection

The selected columns from a base table chosen for replication.

Publication

A data set on the source table available for the replication.

Remote Database

A database located on another server from that being accessed by the client.

Remote Tables

Tables on a database located on a server that is different from the one connected to the client.

Replication Domain

The replication fragment (horizontal partition) and projection (vertical partition) together are called a replication domain. The range of a table's data that is replicated.

Select Lock (S Lock)

An access block placed on a database object that allows other users to browse the object but not to update or delete any part of it.

Slave Database

A read-only database that receives data from a database replication

Source Database

The database that contains source tables used to replicate.

Source Table

The table on the source database that the replicated data is from.

Subscription

The data set on the destination table to receive a publication.

Synchronous Table Replication

A set of columns selected to form a projection that is replicated to the destination tables of another database. Synchronous table replication occurs simultaneously – changes made to the source table are simultaneously made to the destination table(s).

Target Database

A database that receives data from a database replication or table replication.

Index

C

D

H

I

P

R

S

T

U