New Features
Mirror Activator 12.6

Document ID: DC00234-01-1260-02

Last revised: September 2006

Using resource files to create instances

A *resource file* is an ASCII text file that contains configuration information for the Mirror Replication Agent instance to be created, and the mra_admin utility.

The mra_admin utility provides two new command-line parameters that support using a resource file to create a Mirror Replication Agent instance and to validate resource files.

Syntax

mra_admin [-r *res_file* | -vr *res_file*]

Parameters

-r *res_file*

Creates a Mirror Replication Agent instance, based on the contents of the specified resource file (*res_file*).

-vr *res_file*

Validates the specified resource file (*res_file*), without creating a Mirror Replication Agent instance or making any change in the environment.

In addition to creating a Mirror Replication Agent instance, the mra_admin parameters in the resource file allow you to specify the following options:

- Create the instance user login in the primary data server, and add the instance user name to the primary database
- Stop and disable the Replication Agent thread in the primary database
- Start the new instance after it is created
- Initialize the new instance after it starts
- Record mirror log device information in the log device repository after the instance is initialized

---

Note When you *validate* a resource file with mra_admin -vr, no other action is taken, and no Mirror Replication Agent instance is created.

The following sections describe how to use the new mra_admin features:

- Creating a new resource file
- Editing a resource file
- Validating a resource file
- Creating an instance with a resource file

See the Mirror Activator *Administration Guide* for more information about Mirror Replication Agent configuration parameters.

Creating a new resource file

A resource file template named *mra.rs* is provided in the *init* subdirectory of the Mirror Replication Agent installation directory:

```
C:\sybase\MRA-12_6\init\mra.rs
```

The resource file template contains comments that describe each configuration parameter and its value.

Note Sybase® recommends that you validate each resource file *before* you create a Mirror Replication Agent instance using that resource file.

❖ **To create a resource file**

1 Copy the resource file template *mra.rs* to another file that you will edit to create the new resource file:

```
cp mra.rs pubs2.rs
```

where *pubs2.rs* is the name of the new resource file you want to create.

If you have an existing resource file, you can copy it to create a new resource file instead of copying the template.

2 Use your preferred text editor to edit the new resource file copy.

After you create a new resource file, you should validate it. See "Validating a resource file" on page 4 for more information.

Editing a resource file

The mra_admin resource file is an ASCII text file, which you can edit using any standard text editor.

Resource file contents must conform to the following:

- Configuration parameters for both the Mirror Replication Agent and the mra_admin utility must use the following format:

  ```
  param=value
  ```

  where:

  - *param* is the name of the configuration parameter.

- *value* is the value of the configuration parameter.

Note Spaces are not allowed before or after the = symbol or within the *value* string.

- Each ***param*=*value*** statement must occur on a separate line.
- If a default value exists for a configuration parameter, you can specify the default value with the string `USE_DEFAULT`:

  param=USE_DEFAULT

  where *param* is the name of the configuration parameter.
- The following mra_admin configuration parameters require a value of yes or no:
  - create_pds_username
  - disable_rat
  - start_instance
  - initialize_instance

  The yes or no value is not case-sensitive. Any string other than `[y|Y][e|E][s|S]` is interpreted as no.

Note Blank lines and lines that begin with the # symbol are ignored in the resource file.

Validating a resource file

When you invoke the mra_admin utility with the -vr option, the utility validates the specified resource file and returns information about the validation process.

The mra_admin utility validates resource files by:

- Verifying uniqueness of the Mirror Replication Agent administration port number and instance name
- Verifying access to the primary data server, Replication Server®, and RSSD, including:
  - Verifying the host name, port number, database name, user login, and password on each server

  - Verifying jConnect™ for JDBC™ stored procedures in the primary data server
  - Verifying the Replication Server database connection for the primary database
- Verifying access to mirror log devices, if specified in the resource file

If any validation fails, the mra_admin utility returns an error message and information about the failure.

You can repeat the validation process as many times as necessary. No entities are changed or created as a result of this process.

Note Sybase recommends that you validate a new resource file *before* you use it to create a Mirror Replication Agent instance.

❖ **To validate a resource file**

1 Invoke the mra_admin utility, specifying the -vr option and the name of the resource file:

```
mra_admin -vr res_file
```

where *res_file* is the name of the resource file you want to validate.

For example, if the resource file is named *pubs2.rs*, enter the following at the command prompt:

```
mra_admin -vr pubs2.rs
```

Results are returned as either:

- `Admin processing completed`

  or

- `Admin processing completed with errors`

If the validation is successful, you can skip step 2 and use the resource file to create a Mirror Replication Agent instance. See “Creating an instance with a resource file” on page 6 for more information.

If the validation encounters errors, continue to step 2.

2 Use the following procedure to correct validation errors:

  a Review the error messages to determine the cause of the failure.

  b Edit the resource file to correct the appropriate values.

c Invoke mra_admin -vr again, specifying the name of the resource file.

Repeat this step until the resource file is successfully validated.

Creating an instance with a resource file

When you invoke the mra_admin utility with the -r option, the utility first validates the specified resource file, as described in "Validating a resource file" on page 4, except:

- If the Mirror Replication Agent primary database user login does not exist in the primary data server, the utility creates it, if specified in the resource file (`create_pds_username=yes`).

  If the Mirror Replication Agent primary database user login does exist in the primary data server, and the resource file specifies that it should be created, the utility returns an error message and does not create the instance. (This error would be caught in the validation process described in "Validating a resource file" on page 4.)

- If a RepAgent thread is running or enabled in the primary database, the utility stops the RepAgent thread and disables it, if specified in the resource file (`disable_rat=yes`).

  If a RepAgent thread is not enabled or running in the primary database, and the resource file specifies that it should be stopped and disabled, the utility returns an error message and does not create the instance.

- If the resource file specifies that the new Mirror Replication Agent instance should be initialized (`initialize_instance=yes`), then:

  - The Mirror Replication Agent primary database user login must either exist in the primary data server or be created by the mra_admin utility (`create_pds_username=yes`), and
  - The RepAgent thread in the primary database must either be *not* running and disabled or be disabled by the mra_admin utility (`disable_rat=yes`).

  Otherwise, the utility returns an error message and does not create the instance.

After validating the resource file successfully, the mra_admin utility does the following:

- Creates and configures a Mirror Replication Agent instance, based on the contents of the specified resource file

- Starts the new Mirror Replication Agent instance, if specified in the resource file
- Initializes the new Mirror Replication Agent instance, if specified in the resource file

  Note When it initializes a Mirror Replication Agent instance, mra_admin quiesces the primary database. See the Mirror Activator *Administration Guide* for more information about quiescing the primary database.

- Records mirror log device information in the log device repository, if specified in the resource file

The utility also returns information about the instance created and the result.

If instance creation fails, the mra_admin utility returns an error message and information about the failure.

Note Sybase recommends that you validate a new resource file *before* you use it to create a Mirror Replication Agent instance. See "Validating a resource file" on page 4 for more information.

❖ **To create a Mirror Replication Agent instance**

- Invoke the mra_admin utility, specifying the -r option and the name of the resource file:

  ```
  mra_admin -r res_file
  ```

  where *res_file* is the name of the resource file.

  For example, if the resource file is named *pubs2.rs*, enter the following at the command prompt:

  ```
  mra_admin -r pubs2.rs
  ```

  Results are returned as either:

  - `Admin processing completed`

    or

  - `Admin processing completed with errors`

If the instance creation is successful, you can begin using the new Mirror Replication Agent instance.

If the instance creation fails, you may have to do the following:

- Drop the Mirror Replication Agent user from the primary database, and drop the login from the primary data server.
- Delete all files and subdirectories in the instance directory, and delete the instance directory in the Mirror Replication Agent installation directory.
- Edit the resource file to correct the appropriate values.

Note If the instance creation fails, use the following recovery procedure *before* you attempt to create the instance again.

❖ **To recover from instance creation errors**

1 If the resource file does *not* specify that the instance user login be created in the primary data server, skip this step and continue with step 2.

If the resource file specifies that the instance user login be created in the primary data server (that is, `create_pds_username=yes`), then:

a Check the primary database to determine if the instance user was added:

```
sp_helpuser pds_username
```

where *pds_username* is the name of the instance user in the primary database.

- If the instance user was added to the primary database, skip step 1b and go to step 1c.
- If the instance user was *not* added to the primary database, go to step 1b.

b Check the primary data server to determine if the instance login was created:

```
sp_displaylogin pds_username
```

where *pds_username* is the name of the instance login in the primary data server.

- If the instance login was created in the primary data server, continue with step 1c.
- If the instance login was *not* created in the primary data server, skip step 1c and go to step 2.

c Edit the resource file to specify that the instance user login should not be created in the primary data server:

```
create_pds_username=no
```

Note If the Mirror Replication Agent primary database user login is successfully created before the instance creation fails, you must either:

- Edit the resource file to set the value of the create_pds_username parameter to no, or
- Log in to the primary data server and drop the instance login (and drop the instance user from the primary database, if necessary).

2 Check the Mirror Replication Agent base directory on the Mirror Replication Agent host to see if a new instance directory was created. The Mirror Replication Agent base directory is:

```
%SYBASE%\MRA-12_6
```

where *%SYBASE%* is the Mirror Replication Agent installation directory.

If you do *not* find a new instance directory in the Mirror Replication Agent base directory, skip step 3 and go to step 4.

If you find a new instance directory in the Mirror Replication Agent base directory, continue with step 3.

3 To delete the new instance directory, you have two options:

- Use the mra_admin utility to delete the instance:

  ```
  mra_admin -d inst_name
  ```

  where *inst_name* is the name of the instance you want to delete.

  or

- Use operating system commands to delete all of the files and subdirectories in the new instance directory, and then delete the new instance directory.

4 Review the error messages to find the cause of the instance creation failure, and if necessary, edit the resource file to correct the appropriate values.

After editing the resource file, use mra_admin to validate the resource file:

```
mra_admin -vr res_file
```

where *res_file* is the name of the resource file.

See “Validating a resource file” on page 4 for more information.

After you complete the recovery procedure, you can retry creating the Mirror Replication Agent instance.

Log device repository

Some changes in the behavior of Mirror Replication Agent commands related to the log device repository have been implemented in ESDs since the Mirror Activator 12.6 GA release.

The following sections describe the new functionality:

- Overwriting mirror log device paths
- Retrieving mirror log device information
- Using the DBCONTEXT trace flag

Overwriting mirror log device paths

The ra_init force and ra_updatedevices commands do *not* overwrite any existing log device path in the RASD, if all of the following log device information in the RASD matches that returned by the primary data server:

- Database name
- Device ID
- Device name
- Device path

If *any* item in the preceding list does *not* match the information returned by the primary data server, ra_init force and ra_updatedevices overwrite the RASD record for that log device with the information returned by the primary data server.

Note If Mirror Replication Agent overwrites an existing log device record in the RASD, you must use ra_devicepath to specify the mirror log device path.

Retrieving mirror log device information

The ra_devicepath and ra_helpdevice commands return the following information about log devices recorded in the RASD log device repository:

- ID – the virtual device number
- Database – the name of the database the device is associated with
- Device Name – the logical name of the device
- Server Path – the path to the device (as known to the primary data server)
- Disk Mirror Path – the path to the mirror log device at the standby site

  Note If the Disk Mirror Path is returned as DEFAULT, it means that the mirror log device path recorded in the RASD is the same as the Server Path.

- Mirror Status – indicates status of the mirror log device, based on the Disk Mirror Path recorded in the RASD:
  - ACCESSIBLE – the mirror log device is accessible to the Mirror Replication Agent at the path specified.
  - NOT_VALID – the path is not valid (no mirror log device can be accessed at the path specified).
  - OPEN – the mirror log device is accessible and is currently in use by the Mirror Replication Agent.

Using the DBCONTEXT trace flag

A new trace flag named DBCONTEXT traces activity in the log device repository.

Note Trace flags are intended for use by Sybase Technical Support engineers when troubleshooting Mirror Activator.

See the Mirror Activator *Administration Guide* for more information about Mirror Replication Agent trace flags.

Adjusting Java memory

To support adjusting the amount of memory available to the JRE, all of the executable scripts (or batch files) in the Mirror Replication Agent *bin* directory refer to an environment variable named RA_JAVA_MAX_MEM.

Also, all Mirror Replication Agent instance RUN scripts created after the ESD #10 rollup is installed reference the RA_JAVA_MAX_MEM environment variable.

You can adjust the amount of memory available to the JRE by either:

- Set the value of a system variable named RA_JAVA_MAX_MEM in your environment and use the mra utility to start the Mirror Replication Agent instance, or
- Set the value of the RA_JAVA_MAX_MEM variable in the Mirror Replication Agent instance RUN script and use the RUN script to start the Mirror Replication Agent instance.

If you invoke the mra utility to start a Mirror Replication Agent instance, you can set the value of the RA_JAVA_MAX_MEM system variable in your environment to specify the amount of memory available to the JRE. Before it sets the RA_JAVA_MAX_MEM variable to a default value, the mra and mra_admin utilities check to see if it has been already set.

If you invoke the RUN script (or batch file) to start a Mirror Replication Agent instance, you can edit the instance RUN script to change the default value of RA_JAVA_MAX_MEM and specify the amount of memory available to the JRE.

Note When a Mirror Replication Agent instance is started with the RUN script, the value of the RA_JAVA_MAX_MEM variable specified in the RUN script overrides the value set elsewhere. Therefore, you can edit the RUN script to adjust the memory available to the JRE for each instance.

Updating installation directories

Some changes in the installed directories are implemented in Mirror Replication Agent 12.6 ESD #12. This section describes the Mirror Replication Agent 12.6 installed directories that are created when you install the ESD #12 rollup. The default installation directory on UNIX platforms is */opt/bin/sybase*.

When you install Mirror Replication Agent 12.6 ESD #12, the installation program creates the same subdirectories on all platforms for the new JRE directory named *JRE-1_4_2*. See the following section for more information about JRE directory names.

JRE directories

Beginning with Mirror Replication Agent 12.6 ESD #12, the JRE distributed with the Mirror Replication Agent software is installed in a directory named to indicate a more complete JRE version number. This change was implemented to prevent overwriting of the JRE required for Mirror Replication Agent during subsequent installations of other Sybase products.

With a more complete version number identified in the installed JRE directory name, the names of the JRE directories created on various platforms may be different.

Table 1 lists the name of the JRE directory created for Mirror Replication Agent 12.6 ESD #12 for each supported platform.

Table 1: Installed JRE directory names by platform

Platform	Directory name
AIX	*$SYBASE/shared-1_0/JRE-1_4_2_ca1420*
HP-UX	*$SYBASE/shared-1_0/JRE-1_4_2_04*
Linux	*$SYBASE/shared-1_0/JRE-1_4_2_08*
Microsoft Windows	*%SYBASE%\shared-1_0\JRE-1_4_2_05*
Sun Solaris	*$SYBASE/shared-1_0/JRE-1_4_2_05*

Note The JREs listed in the Mirror Replication Agent *Installation Guide* (DC20095-01-1260-03) are no longer installed by the ESD.

Setting up the Maintenance User

With Mirror Replication Agent 12.6 ESD#10, you do not need use sp_adduser to add the Maintenance User to the primary database when you set up the Maintenance User login. Instead, use sp_addlogin to create a server login account for the Maintenance User in the primary data server.

Note Versions earlier than Mirror Replication Agent 12.6 ESD #10 require adding the primary database Maintenance User login to the primary database.

See Chapter 2 in the Mirror Activator *Administration Guide* for more information about setting up the primary database Maintenance User.

Setting the Mirror Agent's default character set

By default, the Java Virtual Machine (JVM) under which a Mirror Replication Agent instance is running uses your system's default character set. The type of character data that Mirror Replication Agent can handle is determined by the character set, also known as the encoding. Unless you want to override the default character set that the JVM finds on your system, you do not need to set the character set-related environment variable.

Overriding the default character set

To override the default character set, as of ESD #12, all of the executable scripts (or batch files) in the Mirror Replication Agent *bin* directory refer to an environment variable named RA_JAVA_DFLT_CHARSET. You can set this environment variable to the character set that you want. However, the character set you specify must be the same character set that is configured on the primary database. For a list of valid Java character sets, go to the Supported Encodings on the Internationalization page under Documentation for the J2SE 1.4.2 JDK at http://java.sun.com/j2se/corejava/intl/index.jsp.

For instances created after applying ESD #12, the Mirror Replication Agent instance *RUN* scripts also refer to the RA_JAVA_DFLT_CHARSET environment variable.

Note If you are using Replication Server to replicate a number of different character sets, you must configure it for UTF8.

To override the system default character set:

- Set the value of a system variable named RA_JAVA_DFLT_CHARSET in your environment and using the ra utility to start the Mirror Replication Agent instance. By using the ra utility, allows you to override the value of the RA_JAVA_DFLT_CHARSET system variable in your environment to specify the character set,

  or

- Set the value of the RA_JAVA_DFLT_CHARSET variable in the Mirror Replication Agent instance *RUN* script and using the *RUN* script to start the Mirror Replication Agent instance. By using the instance *RUN* script (or batch file), you can edit the script to specify the default value of RA_JAVA_DFLT_CHARSET and specify the character set.

❖ **To override the system default character set for all instances**

1 Enter a character set value in the *ra* script:

   - For Windows, edit the *%SYBASE%\MRA-12_6\bin\ra.bat* file
   - For UNIX, edit the *$SYBASE/MRA-12_6/bin/ra.sh* file

     ```
     RA_JAVA_DFLT_CHARSET=charset
     ```

   where *charset* is the Java-supported encoding. For example, `ISO8859_1` or `Cp1252` for ISO-1 (also known as Latin-1), and `ISO8859_8` or `Cp1255` for Hebrew.

2 Uncomment the following lines of code:

   - For Windows:

     ```
     set RA_JAVA_DFLT_CHARSET=charset
     ```

   - For UNIX:

     ```
     RA_JAVA_DFLT_CHARSET=charset
     export RA_JAVA_DFLT_CHARSET
     ```

❖ **To override the system default character set for a Mirror Activator Agent instance**

- Enter a character set value in the *RUN* script:

- For Windows, edit the *%SYBASE%\MRA-12_6\<instance>\RUN_<instance>.bat* batch file:

  ```
  set RA_JAVA_DFLT_CHARSET=charset
  ```

- For UNIX, edit the *$SYBASE/MRA-12_6/<instance>/RUN_<instance>.sh* script:

  ```
  RA_JAVA_DFLT_CHARSET=charset
  export RA_JAVA_DFLT_CHARSET
  ```

  where *charset* is the Java-supported encoding. For example, `ISO8859_1` or `Cp1252` for ISO-1 (also known as Latin-1), and `ISO8859_8` or `Cp1255` for Hebrew.

Note In UNIX, spaces are not allowed on either side of the equal sign. For a list of valid Java character sets, refer to Supported Encodings on the Internationalization page under Documentation for the J2SE 1.4.2 JDK at http://java.sun.com/j2se/corejava/intl/index.jsp.

Setting rs_charset before starting replication

Starting with ESD #12, before you can start replication, you must set the Mirror Replication Agent configuration property rs_charset to match the character set at Replication Server. The value of the Replication Server's RS_charset is located in the Replication Server configuration file in *$SYBASE/REP-12_6/install/<instance>.cfg*.

If rs_charset does not match the Replication Server character set, then Replication Server will convert allcharacters in the LTL commands it receives from Mirror Replication Agent; this will corrupt the origin QIDs and transaction IDs.

Change in behavior of quiesce and shutdown commands

Starting with ESD #12, the quiesce and shutdown commands wait until all data in the mirror log has been read and sent to Replication Server before moving the Mirror Replication Agent to ADMIN state or shutting down.

Previously, these commands waited only until Mirror Replication Agent internal queues were processed before moving to ADMIN state or shutting down. If you need a more immediate transition of the Mirror Replication Agent, use the suspend and shutdown immediate commands.

Using encrypted columns

Starting with ESD #13, Mirror Activator supports replication of encrypted columns. Sybase Adaptive Server® version 12.5.3a and 12.5.4 allow encryption of sensitive user data at the column level. Only Adaptive Server users with special permission can access and decrypt encrypted columns.

This feature requires that you use Replication Server version 12.6 ESD #7 (bug fixed version) or later. For more information on the capabilities and limitations of replicating encrypted columns, refer to the *New Features* Replication Server version 12.6 ESD #5 documentation.