Relay Server

Contents

About this book

This book describes how to set up and use the Relay Server, which enables secure communication between mobile devices and MobiLink, Afaria, and iAnywhere Mobile Office servers through a web server.

About the SQL Anywhere documentation

The complete SQL Anywhere documentation is available in four formats:

- **DocCommentXchange** DocCommentXchange is a community for accessing and discussing SQL Anywhere documentation on the web.

  To access the documentation, go to http://dcx.sybase.com.

- **HTML Help** On Windows platforms, the HTML Help contains the complete SQL Anywhere documentation, including the books and the context-sensitive help for SQL Anywhere tools.

  To access the documentation, choose **Start** » **Programs** » **SQL Anywhere 12** » **Documentation** » **HTML Help (English)**.

- **Eclipse** On Unix platforms, the complete Help is provided in Eclipse format. To access the documentation, run *sadoc* from the *bin32* or *bin64* directory of your SQL Anywhere installation.

- **PDF** The complete set of SQL Anywhere books is provided as a set of Portable Document Format (PDF) files. You must have a PDF reader to view information.

  To access the PDF documentation on Windows operating systems, choose **Start** » **Programs** » **SQL Anywhere 12** » **Documentation** » **PDF (English)**.

  To access the PDF documentation on Unix operating systems, use a web browser to open */documentation/en/pdf/index.html* under the SQL Anywhere installation directory.

Documentation conventions

This section lists the conventions used in this documentation.

Operating systems

SQL Anywhere runs on a variety of platforms. Typically, the behavior of the software is the same on all platforms, but there are variations or limitations. These are commonly based on the underlying operating system (Windows, Unix), and seldom on the particular variant (IBM AIX, Windows Mobile) or version.

To simplify references to operating systems, the documentation groups the supported operating systems as follows:

- **Windows** The Microsoft Windows family includes platforms that are used primarily on server, desktop, and laptop computers, as well as platforms used on mobile devices. Unless otherwise specified, when the documentation refers to Windows, it refers to all supported Windows-based platforms, including Windows Mobile.

  Windows Mobile is based on the Windows CE operating system, which is also used to build a variety of platforms other than Windows Mobile. Unless otherwise specified, when the documentation refers to Windows Mobile, it refers to all supported platforms built using Windows CE.

- **Unix** Unless otherwise specified, when the documentation refers to Unix, it refers to all supported Unix-based platforms, including Linux and Mac OS X.

For the complete list of platforms supported by SQL Anywhere, see "Supported platforms" [*SQL Anywhere 12 - Introduction*].

Directory and file names

Usually references to directory and file names are similar on all supported platforms, with simple transformations between the various forms. In these cases, Windows conventions are used. Where the details are more complex, the documentation shows all relevant forms.

These are the conventions used to simplify the documentation of directory and file names:

- **Uppercase and lowercase directory names** On Windows and Unix, directory and file names may contain uppercase and lowercase letters. When directories and files are created, the file system preserves letter case.

  On Windows, references to directories and files are *not* case sensitive. Mixed case directory and file names are common, but it is common to refer to them using all lowercase letters. The SQL Anywhere installation contains directories such as *Bin32* and *Documentation*.

  On Unix, references to directories and files *are* case sensitive. Mixed case directory and file names are not common. Most use all lowercase letters. The SQL Anywhere installation contains directories such as *bin32* and *documentation*.

  The documentation uses the Windows forms of directory names. You can usually convert a mixed case directory name to lowercase for the equivalent directory name on Unix.

- **Slashes separating directory and file names** The documentation uses backslashes as the directory separator. For example, the PDF form of the documentation is found in *install-dir* *\Documentation\en\PDF* (Windows form).

  On Unix, replace the backslash with the forward slash. The PDF documentation is found in *install-dir/documentation/en/pdf*.

- **Executable files** The documentation shows executable file names using Windows conventions, with a suffix such as *.exe* or *.bat*. On Unix, executable file names have no suffix.

  For example, on Windows, the network database server is *dbsrv12.exe*. On Unix, it is *dbsrv12*.

- ***install-dir*** During the installation process, you choose where to install SQL Anywhere. The environment variable SQLANY12 is created and refers to this location. The documentation refers to this location as *install-dir*.

  For example, the documentation may refer to the file *install-dir/readme.txt*. On Windows, this is equivalent to *%SQLANY12%\readme.txt*. On Unix, this is equivalent to *$SQLANY12/readme.txt* or *${SQLANY12}/readme.txt*.

  For more information about the default location of *install-dir*, see "SQLANY12 environment variable" [*SQL Anywhere Server - Database Administration*].

- ***samples-dir*** During the installation process, you choose where to install the samples included with SQL Anywhere. The environment variable SQLANYSAMP12 is created and refers to this location. The documentation refers to this location as *samples-dir*.

  To open a Windows Explorer window in *samples-dir*, choose **Start** » **Programs** » **SQL Anywhere 12** » **Sample Applications And Projects**.

  For more information about the default location of *samples-dir*, see "SQLANYSAMP12 environment variable" [*SQL Anywhere Server - Database Administration*].

Command prompts and command shell syntax

Most operating systems provide one or more methods of entering commands and parameters using a command shell or command prompt. Windows command prompts include Command Prompt (DOS prompt) and 4NT. Unix command shells include Korn shell and bash. Each shell has features that extend its capabilities beyond simple commands. These features are driven by special characters. The special characters and features vary from one shell to another. Incorrect use of these special characters often results in syntax errors or unexpected behavior.

The documentation provides command line examples in a generic form. If these examples contain characters that the shell considers special, the command may require modification for the specific shell. The modifications are beyond the scope of this documentation, but generally, use quotes around the parameters containing those characters or use an escape character before the special characters.

These are some examples of command line syntax that may vary between platforms:

- **Parentheses and curly braces** Some command line options require a parameter that accepts detailed value specifications in a list. The list is usually enclosed with parentheses or curly braces. The documentation uses parentheses. For example:

  ```
  -x tcpip(host=127.0.0.1)
  ```

  Where parentheses cause syntax problems, substitute curly braces:

  ```
  -x tcpip{host=127.0.0.1}
  ```

  If both forms result in syntax problems, the entire parameter should be enclosed in quotes as required by the shell:

  ```
  -x "tcpip(host=127.0.0.1)"
  ```

- **Semicolons** On Unix, semicolons should be enclosed in quotes.
- **Quotes** If you must specify quotes in a parameter value, the quotes may conflict with the traditional use of quotes to enclose the parameter. For example, to specify an encryption key whose value contains double-quotes, you might have to enclose the key in quotes and then escape the embedded quote:

  ```
  -ek "my \"secret\" key"
  ```

  In many shells, the value of the key would be `my "secret" key`.
- **Environment variables** The documentation refers to setting environment variables. In Windows shells, environment variables are specified using the syntax *%ENVVAR%*. In Unix shells, environment variables are specified using the syntax *$ENVVAR* or *${ENVVAR}*.

Contacting the documentation team

We would like to receive your opinions, suggestions, and feedback on this Help.

You can leave comments directly on help topics using DocCommentXchange. DocCommentXchange (DCX) is a community for accessing and discussing SQL Anywhere documentation. Use DocCommentXchange to:

- View documentation
- Check for clarifications users have made to sections of documentation
- Provide suggestions and corrections to improve documentation for all users in future releases

Go to http://dcx.sybase.com.

Finding out more and requesting technical support

Newsgroups

If you have questions or need help, you can post messages to the Sybase iAnywhere newsgroups listed below.

When you write to one of these newsgroups, always provide details about your problem, including the build number of your version of SQL Anywhere. You can find this information by running the following command: **dbeng12 -v**.

The newsgroups are located on the *forums.sybase.com* news server.

The newsgroups include the following:

- sybase.public.sqlanywhere.general
- sybase.public.sqlanywhere.linux
- sybase.public.sqlanywhere.mobilink
- sybase.public.sqlanywhere.product_futures_discussion
- sybase.public.sqlanywhere.replication
- sybase.public.sqlanywhere.ultralite
- ianywhere.public.sqlanywhere.qanywhere

For web development issues, see http://groups.google.com/group/sql-anywhere-web-development.

Newsgroup disclaimer
iAnywhere Solutions has no obligation to provide solutions, information, or ideas on its newsgroups, nor is iAnywhere Solutions obliged to provide anything other than a systems operator to monitor the service and ensure its operation and availability.

iAnywhere Technical Advisors, and other staff, assist on the newsgroup service when they have time. They offer their help on a volunteer basis and may not be available regularly to provide solutions and information. Their ability to help is based on their workload.

Developer Centers

The **SQL Anywhere Tech Corner** gives developers easy access to product technical documentation. You can browse technical white papers, FAQs, tech notes, downloads, techcasts and more to find answers to your questions as well as solutions to many common issues. See http://www.sybase.com/developer/library/sql-anywhere-techcorner.

The following table contains a list of the developer centers available for use on the SQL Anywhere Tech Corner:

Name	URL	Description
SQL Anywhere .NET Developer Center	www.sybase.com/developer/library/sql-anywhere-techcorner/microsoft-net	Get started and get answers to specific questions regarding SQL Anywhere and .NET development.
PHP Developer Center	www.sybase.com/developer/library/sql-anywhere-techcorner/php	An introduction to using the PHP (PHP Hypertext Preprocessor) scripting language to query your SQL Anywhere database.

Name	URL	Description
SQL Anywhere Windows Mobile Developer Center	www.sybase.com/developer/library/sql-anywhere-techcorner/windows-mobile	Get started and get answers to specific questions regarding SQL Anywhere and Windows Mobile development.

Introduction to the Relay Server

The Relay Server enables secure, load-balanced communication between mobile devices and back-end servers through a web server. Supported back-end servers include MobiLink, Unwired Server, Afaria, and Mobile Office. The Relay Server provides the following:

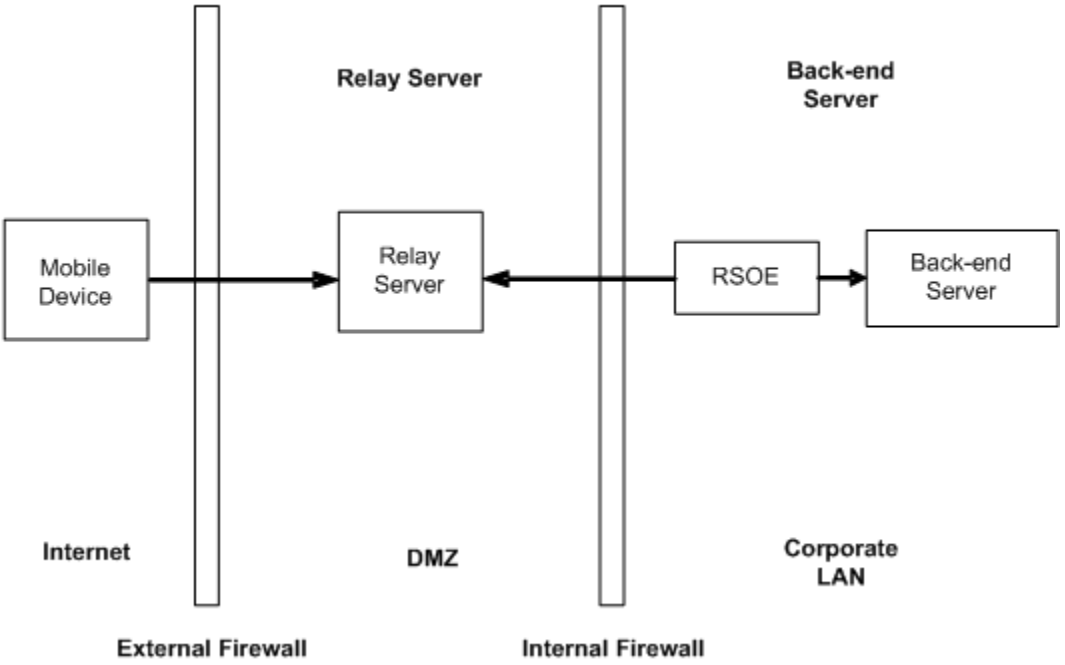
- A common communication architecture for mobile devices communicating with back-end servers.
- A mechanism to enable a load-balanced and fault-tolerant environment for back-end servers.
- A way to help communication between mobile devices and back-end servers in a way that integrates easily with existing corporate firewall configurations and policies.

Relay Server architecture

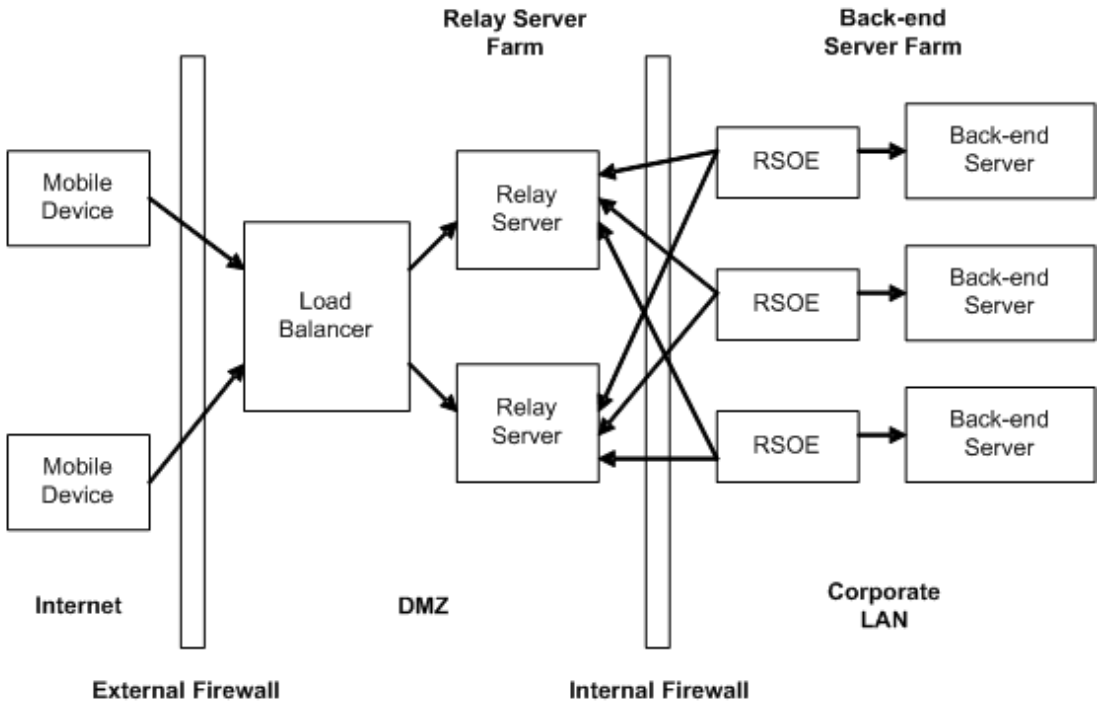
A Relay Server deployment consists of the following:

- Mobile devices running client applications and services that need to communicate with back-end servers running in a corporate LAN.
- Optional load balancer to direct requests from the mobile devices to a group of Relay Servers.
- One or more Relay Servers running in the corporate DMZ.
- At least one back-end server running in a corporate LAN that are responsible for servicing client requests.
- One Relay Server Outbound Enabler (RSOE) per back-end server. The Outbound Enabler manages all communication between a back-end server and the Relay Server farm.

The following diagram shows the Relay Server architecture with a single Relay Server.



The following diagram shows the Relay Server architecture for a more complex system with a Relay Server farm and a back-end server farm.



The Relay Server consists of a set of web extensions, a background process for maintaining state information, and a web server.

Because the Relay Server is a web extension running in a web server, all communication is performed using HTTP or HTTPS. Using HTTP easily integrates with existing corporate firewall configurations and policies. The Relay Server requires that the connection from the corporate LAN to the Relay Server be initiated from inside the corporate LAN. This provides a more secure deployment environment because it does not require inbound connections from the DMZ into the corporate LAN.

The Relay Server contains two web extensions: a client extension and a server extension. The client extension handles client requests made from applications running on mobile devices. The server extension handles requests made by the Outbound Enabler on behalf of a back-end server.

The Relay Server farm

A Relay Server farm is any number of Relay Servers with a front-end load balancer. It is possible to set up a Relay Server farm with a single Relay Server, in which case a load balancer is not required. In this case, mobile devices can connect directly to the Relay Server.

Back-end server farm

A back-end server farm is a group of homogeneous back-end servers. A client making a request through the Relay Server farm must specify the back-end server farm it is targeting.

Load balancer

The load balancer directs requests from the mobile devices to a Relay Server running in the Relay Server farm. The load balancer is not required if there is only one Relay Server.

Relay Server Outbound Enabler

The Relay Server Outbound Enabler runs on the same computer as the back-end server. Its primary function is to initiate an outbound connection to all Relay Servers in the Relay Server farm on behalf of the back-end server. There is one Outbound Enabler per back-end server. See "Outbound Enabler" on page 29.

Relay Server security

The Relay Server has built-in security features, but also relies on the security features provided by the web server. In combination with the web server, the Relay Server provides the following features for secure communications:

- Server-side certificates
- Client-side certificates
- Backend server and farm configuration
- RSOE MAC address filtering and token authentication
- Client encryption technologies (Protocol-level encryption)

Server-side certificates

Using a server-side certificate, a client communicating with the Relay Server can verify the web server that is running the Relay Server is a trusted server. The client verifies the web server's public certificate with the root certificate stored on the client. If the certificates are verified, a key exchange occurs to establish the encrypted connection.

Client-side certificates

Using a client-side certificate, the web server can verify that a client communicating with the Relay Server is a trusted client. The web server verifies the client's public certificate with its root certificate that is stored in the certificate manager on the web server machine. If the certificates are verified, a key exchange occurs to establish the encrypted connection.

Backend server and farm configuration

The *rs.config* file is used by the Relay Server to define the peer list of Relay Servers, if running in a farm environment, including the backend farm and backend server configurations. Each Relay Server in the environments needs to maintain a copy of the *rs.config* file.

The configuration of the backend farm and backend servers ensures the Relay Server only communicates with machines with which it has been configured. Any attempted communication with machines for which the Relay Server has not been configured will be refused.

The backend farm can be configured to specify the level of communication security when accepting requests from the clients and the RSOE. There is a client_security and backend_security option that allows the backend farm to specify the type of communication that can be established. This option is specified as follows:

client_security=on|off **On** indicates the client must connect using HTTPS. **Off** indicates the client must connect using HTTP. This setting is optional. If no value is specified, the client can connect using HTTP or HTTPS.

backend_security=on|off **On** indicates the RSOE must connect using HTTPS. **Off** indicates the RSOE must connect using HTTP. This setting is optional. If no value is specified, the RSOE can connect using HTTP or HTTPS.

For more information, see:

- "Relay Server configuration file" on page 23

RSOE MAC address filtering and token authentication

The RSOE establishes the connection between the backend server and the Relay Server using three phases: 1) startup phase, 2) ready phase 3) working phase. For a detailed description of the RSOE startup, see:

- Windows IIS http://www.sybase.com/detail?id=1059277
- Apache http://www.sybase.com/detail?id=1065869

In the backend server section of the *rs.config* file, each server that exists in the backend farm is configured with an ID and associated farm name. The ID corresponds to the server name. The Relay Server has the ability to verify the MAC address of the machine running the RSOE to ensure the server communicating from within the internal firewall is trusted and allowed to establish a connection with the Relay Server. The MAC property is the MAC address of the network adapter used by the RSOE. The address is specified in IEEE 802 MAC-48 format.

The backend server section also allows for the configuration of a security token that is used by the Relay Server to authenticate the backend server connection. The token must be provided upon startup of the RSOE when establishing the connection with the Relay Server.

MobiLink security

The MobiLink client uses HTTP or HTTPS to communicate with the Relay Server. With HTTPS communication, data is temporarily decrypted and re-encrypted as it is exchanged between the client and backend server. This is known as the WAP Gap. To ensure completely secure communication through the WAP Gap, it is recommended that you use the MobiLink end-to-end encryption feature to further protect data as it passes through the Relay Server. The MobiLink end-to-end encryption feature provides protocol-level encryption between the MobiLink, UltraLite, and QAnywhere clients and the MobiLink server. Both

RSA and ECC encryption types are supported. TLS security can be used in combination with end-to-end encryption. See "End-to-end encryption" [*SQL Anywhere Server - Database Administration*].

Deploying the Relay Server

The Relay Server components are installed as part of the SQL Anywhere 12 installation. The install process automatically deploys all the necessary files on the machine that is going to run the Relay Server. See the following topics for deployment information:

- "Deploying the Relay Server components to Microsoft IIS 6.0 on Windows Server 2003" on page 7
- "Deploying the Relay Server components to Microsoft IIS 7.0 or 7.5 on Windows Server 2008/ Windows Server 2008 R2" on page 10
- "Deploying the Relay Server components to Apache on Linux" on page 14

Deploying the Relay Server components to Microsoft IIS 6.0 on Windows Server 2003

The Relay Server for Windows consists of the following executables:

- *rs_client.dll*
- *rs_server.dll*
- *rs_monitor.dll*
- *rshost.exe*
- *dblgen12.dll*
- *dbsvc.exe*
- *dbfhide.exe*
- *dbtool12.dll*
- *dblib12.dll*
- *dbicu12.dll*
- *dbicudt12.dll*
- *dbsupport.exe*
- *dbghelp.dll*

For information about which versions of IIS are supported, see http://www.sybase.com/detail?id=1061806.

Setup scripts for Relay Server on IIS can be found in the *%SQLANY12%\RelayServer\IIS* directory.

To deploy the Relay Server files

1. Install the Relay Server components using the SQL Anywhere install. By default all files are installed to *%SQLANY12%* and are based on the bitness of the machine:
   - *%SQLANY12%\Bin32* and *%SQLANY12%\Bin64* are used for DLLs and executables for administration.
   - *%SQLANY12%\RelayServer\IIS\Bin32* and *%SQLANY12%\RelayServer\IIS\Bin64* are used for Relay Server specific files under the appropriate folder (for example, *Admin*, *Client*, *Monitor* or *Server*). The *Server* folder contains the *rshost.exe* and *rs.config* files.

2. Create the following directories under the Default Web Site in the Microsoft IIS Manager for use by the Relay Server:
   - *%SQLANY12%\RelayServer\IIS\BinXX\Server*
   - *%SQLANY12%\RelayServer\IIS\BinXX\Admin*
   - *%SQLANY12%\RelayServer\IIS\BinXX\Monitor*
   - *%SQLANY12%\RelayServer\IIS\BinXX\Client*

3. Create the Relay Server configuration file *rs.config* using the following guidelines.
   - The file should have four sections:
     - Options section
     - Relay server section
     - Backend farm section
     - Backend server section
   - Each section starts with a section tag, made by enclosing a keyword that identifies the section name in square brackets.
   - Add the appropriate properties to each section. A property is defined by specifying the property name on the left-hand side of an equal sign and its value on the right-hand side of the equal sign. For example, property name = value.
   - The configuration file should contain only 7-bit ASCII characters.

   See "Relay Server configuration file" on page 23.

4. Create an application pool:
   a. Start Microsoft IIS Manager Console.
   b. Right-click **Application Pools** and create a new application pool, for example RS_POOL.
   c. Edit the properties for the application pool you created.
      i. Select the **Recycling** tab and turn off all the recycling options.
      ii. Select the **Performance** tab and do the following:
         A. Turn off **Shutdown Worker Processes After Being Idle**.
         B. Set the number of worker processes to the total number of processing cores. You can further adjust this number depending on your usage and performance preferences. See the Microsoft IIS performance notes about Web garden size for more information.

5. Set the Connection timeout property of the Default Web Site to a minimum of 60 seconds. By default this value should be 120 seconds, which is sufficient.

6. Edit the properties of ias_relay_server and enable the Relay Server web extensions:
   a. Select the **Directory** tab and do the following:
      i. Set execute permissions to **Scripts And Executables**.
      ii. Click **Create** under **Application Settings**. Select the application pool you created in step 3 as the associated application pool.

b. Select the **Directory Security** tab and do the following:

i. Click **Edit** in **Authentication and Access Control**.

ii. Enable anonymous access and fill in the user name and password for an account belonging to the Administrators group.

Alternatively, you may leave the setting as the built-in user **IUSR_%computername%** and execute the following command to grant permission to access the Microsoft IIS metabase.

```
C:\Windows\Microsoft.Net\Framework\<Version>\aspnet_regiis.exe -ga
IUSR_%computername%
```

c. Under **Web Server Extensions** in the Microsoft IIS manager, add *rs_server.dll*, *rs_client.dll*, and *rs_monitor.dll* as a new Web service extension. The extension name should be ISAPI and the DLLs need to have the extension status set to Allowed.

d. Deploy the Relay Server configuration file by creating a Relay Server configuration file and copying it to the *ias_relay_server\server* directory.

7. Copy the *rs.config* file to the *ias_relay_server\server* directory.

8. Ensure optimum performance by reviewing the performance tips. See "Performance tips" on page 10.

9. Start the Relay Server State Manager as a service using a command line similar to the following:

```
dbsvc -as -s auto -t rshost -w RelayServer "%SQLANY12%\RelayServer\IIS
\BinXX\Server\rshost.exe" -q -qc -f "%SQLANY12%\RelayServer\IIS\BinXX
\Server\rs.config" -o "c:\temp\ias_relay_server.log"
```

See:

- "Starting the Relay Server State Manager as a service" on page 19
- "Relay Server State Manager command line syntax" on page 21

Note
It is recommended that you start the State Manager as a service. However, it can also be started automatically by the Relay Server. See "Relay Server State Manager" on page 19.

10. Update the Relay Server configuration for Microsoft IIS 6.0 on Windows:

a. For each computer that belongs to the Relay Server farm you are updating, copy the updated configuration file to the *%SQLANY12%\RelayServer\IIS\BinXX\Server* directory under the Relay Server web site home directory. The configuration file must be called *rs.config* if auto start is used.

b. From the *%SQLANY12%\RelayServer\IIS\BinXX\Server* directory, run the following command line to apply the configuration update:

```
rshost -u -f rs.config
```

c. Repeat the previous steps for each computer in the Relay Server farm that is being updated.

Note
After configuring the Relay Server with IIS, it is recommended that you restart the IIS server or reboot the machine.

See also

- "Relay Server State Manager" on page 19
- "File Hiding utility (dbfhide)" on page 32

Performance tips

Keep the following in mind when deploying the Relay Server to Microsoft IIS on Windows:

- The Relay Server web extension does not rely on ASP.NET. Removing the ASP.NET ISAPI filter yields better performance. The filter gets turned on by default in a standard Microsoft IIS install. To turn off the filter, do the following:

  1. Start Microsoft IIS Manager Console.
  2. Edit the properties of **Default Web Site**.
  3. Under the **ISAPI Filters** tab, remove the ASP.NET filter.

- For better performance, you can turn off the Microsoft IIS access log. To turn off the access log, do the following:

  1. Start Microsoft IIS Manager Console.
  2. Edit the properties of the *ias_relay_server* directory under **Default Web Site**.
  3. Under the **Directory** tab, clear the **Log Visits** selection.

- In a production environment, Relay Server verbosity can be set to 0 via the Relay Server configuration file. This yields better performance under high loads.

- The Relay Server does not impose restrictions on the Web garden size. One worker process may serve requests from all Outbound Enablers as well as from all the clients. However, the number of threads that can be created in the process is limited by the process heap space left available for thread creation. The thread created by Microsoft IIS has a 256k stack size. If your machine has adequate resources, experiment with a higher number of processes if you suspect you are hitting a concurrency limit when the server is loaded with thousands of concurrent requests.

Deploying the Relay Server components to Microsoft IIS 7.0 or 7.5 on Windows Server 2008/ Windows Server 2008 R2

The Relay Server for Windows consists of the following executables:

- *rs_client.dll*
- *rs_server.dll*
- *rs_monitor.dll*
- *rshost.exe*
- *dblgen12.dll*
- *dbsvc.exe*
- *dbfhide.exe*
- *dbtool12.dll*
- *dblib12.dll*
- *dbicu12.dll*
- *dbicudt12.dll*
- *dbsupport.exe*
- *dbghelp.dll*

For information about which versions of IIS are supported, see http://www.sybase.com/detail?id=1061806.

Setup scripts for Relay Server on IIS can be found in the *%SQLANY12%\RelayServer\IIS* directory.

To deploy the Relay Server files

1. Make sure that the Microsoft IIS ISAPI Extensions feature is installed.

2. Install the Relay Server components using the SQL Anywhere install. By default all files are installed to *%SQLANY12%* and are based on the bitness of the machine:
   - *%SQLANY12%\Bin32* and *%SQLANY12%\Bin64* are used for DLLs and executables for administration.
   - *%SQLANY12%\RelayServer\IIS\Bin32* and *%SQLANY12%\RelayServer\IIS\Bin64* are used for Relay Server specific files under the appropriate folder (for example, *Admin*, *Client*, *Monitor* or *Server*). The *Server* folder contains the *rshost.exe* and *rs.config* files.

3. Backup the IIS configuration file *applicationHost.config* located in the *c:\Windows\System32\inetsrv\config* folder.

4. To add an application pool for the Relay Server, edit the *applicationHost.config* file to add the following code to the **<system.applicationHost>** » **<applicationPools>** section.

```
<add name="RelayServer" queueLength="65535" autoStart="true"
managedRuntimeVersion="" managedPipelineMode="Integrated">
   <processModel identityType="LocalSystem" idleTimeout="00:00:00"
maxProcesses="20" pingingEnabled="false"
         pingInterval="00:00:30" pingResponseTime="00:01:30" />
   <recycling disallowOverlappingRotation="true">
     <periodicRestart time="00:00:00">
       <schedule>
         <clear />
       </schedule>
     </periodicRestart>
   </recycling>
   <failure rapidFailProtection="false" />
```

```
    <cpu resetInterval="00:00:00" />
</add>
```

5. To add the Relay Server application to the default site, edit the *applicationHost.config* file to add the following code to the **<system.applicationHost>** » **<applicationPools>** » **<sites>** » **<site name="Default Web Site">** section.

```
<application path="/rs" applicationPool="RelayServer">
  <virtualDirectory path="/"physicalPath="%SQLANY12%\RelayServer\IIS
\Bin32"/>
</application>
```

6. To add the Relay Server ISAPI extensions, edit the *applicationHost.config* file to add the following code to the **<system.webServer>** » **<security>** » **<isapiCgiRestriction>** section.

```
<add path="%SQLANY12%\RelayServer\IIS\Bin32\Admin\rs_admin.dll"
allowed="true" />
<add path="%SQLANY12%\RelayServer\IIS\Bin32\Client\rs_client.dll"
allowed="true" />
<add path="%SQLANY12%\RelayServer\IIS\Bin32\Monitor\rs_monitor.dll"
allowed="true" />
<add path="%SQLANY12%\RelayServer\IIS\Bin32\Server\rs_server.dll"
allowed="true" />
```

7. To add the Relay Server handlers, edit the *applicationHost.config* file to add the following code to the **<configuration>** section. For security reasons, it is recommended that the Admin extension be accessed only using HTTPS.

```
<location path="Default Web Site/rs/admin">
  <system.webServer>
    <handlers accessPolicy="Execute, Script">
    </handlers>
    <!-- For security reasons, it is recommended that the Admin extension
be accessed only using HTTPS. -->
    <security>
      <access sslFlags="Ssl" />
    </security>
  </system.webServer>
</location>
<location path="Default Web Site/rs/client">
  <system.webServer>
    <handlers accessPolicy="Execute, Script">
    </handlers>
  </system.webServer>
</location>
<location path="Default Web Site/rs/monitor">
  <system.webServer>
    <handlers accessPolicy="Execute, Script">
    </handlers>
  </system.webServer>
</location>
<location path="Default Web Site/rs/server">
  <system.webServer>
    <handlers accessPolicy="Execute, Script">
    </handlers>
  </system.webServer>
</location>
```

8. For the best performance, edit the *applicationHost.config* file to add the following code to the **<configuration>** section, according to your administration practices for IIS.


```
<location path="Default Web Site/rs">
  <system.webServer>
    <security>
      <authentication>
        <anonymousAuthentication userName="" />
      </authentication>
      <requestFiltering>
        <requestLimits maxAllowedContentLength="2147483647" />
      </requestFiltering>
    </security>
  </system.webServer>
</location>
```

Note
The Relay Server is set up for anonymous access based on these instructions. Proper security needs to be configured for IIS and the Relay Server based on the business requirements.

9. Save these changes to the *applicationHost.config* file.

10. Set the Connection timeout property of the Default Web Site to a minimum of 60 seconds. By default this value should be 120 seconds, which is sufficient.

11. Create the Relay Server configuration file *rs.config* using the following guidelines.

    - The file should have four sections:
      - Options section
      - Relay server section
      - Backend farm section
      - Backend server section
    - Each section starts with a section tag, made by enclosing a keyword that identifies the section name in square brackets.
    - Add the appropriate properties to each section. A property is defined by specifying the property name on the left-hand side of an equal sign and its value on the right-hand side of the equal sign. For example, property name = value.
    - The configuration file should contain only 7-bit ASCII characters.

    See "Relay Server configuration file" on page 23.

12. Copy the *rs.config* file to the *%SQLANY12%\RelayServer\IIS\BinXX\Server* directory.

13. Ensure optimum performance by reviewing the performance tips. See "Performance tips" on page 10.

14. Start the Relay Server State Manager as a service using a command line similar to the following:

    ```
    dbsvc -as -s auto -t rshost -w RelayServer "%SQLANY12%\RelayServer\IIS
    \BinXX\Server\rshost.exe" -q -qc -f "%SQLANY12%\RelayServer\IIS\BinXX
    \Server\rs.config" -o "c:\temp\ias_relay_server.log"
    ```

    See:

    - "Starting the Relay Server State Manager as a service" on page 19
    - "Relay Server State Manager command line syntax" on page 21

Note
It is recommended that you start the State Manager as a service. However, it can also be started automatically by the Relay Server. See "Relay Server State Manager" on page 19.

15. Update the Relay Server configuration for Microsoft IIS on Windows:
    a. For each computer that belongs to the Relay Server farm you are updating, copy the updated configuration file to the %SQLANY12%\RelayServer\IIS\BinXX\Server directory under the Relay Server web site home directory. The configuration file must be called rs.config if auto start is used.
    b. From the %SQLANY12%\RelayServer\IIS\BinXX\Server directory, run the following command line to apply the configuration update:

       ```
       rshost -u -f rs.config
       ```

    c. Repeat the previous steps for each computer in the Relay Server farm that is being updated.

Note
After configuring the Relay Server with IIS, it is recommended that you restart the IIS server or reboot the machine.

See also

- "Relay Server State Manager" on page 19
- "File Hiding utility (dbfhide)" on page 32

Deploying the Relay Server components to Apache on Linux

On Linux, the Relay Server files are installed to the */opt/sqlanywhere12* as part of the SQL Anywhere installation. See the following topics for deployment information.

For information about which versions of Apache on Linux are supported, see http://www.sybase.com/detail?id=1061806.

To deploy the Relay Server files

1. Copy the following executables and shared objects from the to the */opt/sqlanywhere12* directory to the *<apache-install-dir>\modules* directory:
   - *mod_rs_ap_client.so*
   - *mod_rs_ap_server.so*
   - *rshost*
   - *dblgen12.res*
   - *libdbtasks12.so*
   - *libdbicudt12.so*
   - *libdbicu12_r.so*
   - *libdblib12_r.so*
   - *dbsupport*
   - *dbfhide*
   - *libdblib12.so*
   - *mod_rs_ap_monitor.so*
   - *mod_rs_ap_admin.so*
2. Create the Relay Server configuration file *rs.config*. See "Relay Server configuration file" on page 23.
3. Copy *rs.config* into the *<apache-install-dir>\modules* directory. The server module expects the *rshost* executable to be in the same directory where you copied the *rs.config* file.
4. Edit the Relay Server configuration file *rs.config* using the following guidelines.
   - The file should have four sections:
     - Relay server section
     - Backend farm section
     - Backend server section
     - Options section
   - Each section starts with a section tag, made by enclosing a keyword that identifies the section name in square brackets.
   - Add the appropriate properties to each section. A property is defined by specifying the property name on the left-hand side of an equal sign and its value on the right-hand side of the equal sign. For example, property name = value.
   - The configuration file should contain only 7-bit ASCII characters.

   See "Relay Server configuration file" on page 23.
5. Set the PATH and LD_LIBRARY_PATH environment variables to include the Apache *<apache-install-dir>\modules* directory.
6. Edit the Apache *conf/httpd.conf* file.
   a. Add the following lines to load the Relay Server client and server modules:

```
LoadModule iarelayserver_client_module modules/mod_rs_ap_client.so
LoadModule iarelayserver_server_module modules/mod_rs_ap_server.so
```

> **Note**
> All modules are invoked using different URLs and all modules explicitly look for the string *iarelayserver* in the URL path. That part of the URL need not change.

b. Add the following line to load the SQL Anywhere Monitor support module:

```
LoadModule iarelayserver_monitor_module modules/mod_rs_ap_monitor.so
```

c. Add the following line to load the Remote Administration support module:

```
LoadModule iarelayserver_admin_module modules/mod_rs_ap_admin.so
```

d. Add the following line to create a <*locationMatch*> section for the client module:

```
<LocationMatch /cli/iarelayserver/* >
    SetHandler iarelayserver-client-handler
</LocationMatch>
```

e. Add the following line to create a <*location*> section for the server module:

```
<Location /srv/iarelayserver/* >
    SetHandler iarelayserver-server-handler
    RSConfigFile "/<apache-install>/modules/rs.config"
</Location>
```

> **Note**
> You must specify an `RSConfigFile` directive which specifies the location of the Relay Server configuration file, *rs.config*. The *rs.config* file must reside in the same directory where the `rshost` executable is deployed.

f. Add the following line to create a <*location*> section for the SQL Anywhere Monitor module:

```
 <Location  /mon/iarelayserver/* >
    SetHandler iarelayserver-monitor-handler
</Location>
```

g. Add the following line to create a <*location*> section for the Remote Administration module:

```
<Location /admin/iarelayserver/* >
  SetHandler iarelayserver-admin-handler
</Location>
```

h. If the TimeOut directive is set, ensure it is set to at least 60 seconds.

7. On Linux, if any of the following environment variables are set globally when Apache spawns a process, then there is nothing further needed for the configuration of Apache: $TMP, $TMPDIR or $TEMP.

   If any of the above environment variables are not set globally, or if you want the default Relay Server log file to go in a specific temporary directory (for example, when the State Manager is started automatically but without customizations), then edit the file */<apache-dir>/bin/envvars* to set and then export TMP.

   For example, to edit $TMP in the envvars file, do the following:

```
set TMP="/tmp"
export TMP
```


This sets the environment variable in the shell that Apache creates before it spawns its processes.

Note
The Apache user process must have write permissions to the specified *tmp* directory.

8. If you want to update the Relay Server configuration while it is started:
   a. Copy the updated configuration file to the <*apache-install-dir*>*modules* directory under the Apache install directory. The configuration file must be called *rs.config* if auto start is used.
   b. From the /<*Apache-install*>/*modules* directory, run the following command line to apply the configuration update:

      ```
      rshost -u -f rs.config
      ```

   c. If the Relay Server is set up as a farm with more than one server, repeat the previous steps for each computer in the Relay Server farm.

See also

- "Relay Server State Manager" on page 19
- "File Hiding utility (dbfhide)" on page 32

Relay Server State Manager

The Relay Server State Manager is a process that is responsible for maintaining Relay Server state information across client requests and Outbound Enabler sessions. The State Manager is also responsible for managing the log file used by the Relay Server. The State Manager can either be started automatically by the Relay Server or started as a service.

The default log file name is *ias_relay_server_host.log*. On Windows, this file is located in the directory specified by the TEMP environment variable. On Linux, the file is located in the directory specified by the TMP, TEMP, or TMPDIR environment variables. If none of those variables are set, a log file is created in the */tmp* directory.

Note
The Apache user process must have write permissions to the specified *tmp* directory.

On a graceful shutdown, the State Manager renames the log file to a file of the form *<yymmdd><nn>.log* where *<yymmdd>* represents the date on which the log file was renamed and *<nn>* is the sequential version number of the log file for that day.

Starting the State Manager as a service is the recommended method. Note that starting the State Manager manually on a command line is not supported.

It is possible to specify the options that are used by the Relay Server to start the State Manager. To change the options, set the **start** property in the options section of the Relay Server configuration file. For example:

```
[options]
start = "rshost -o c:\temp\myrshost.log"
```

Note that you must specify the name of the Relay Server State Manager executable (`rshost`) before the options.

Starting the Relay Server State Manager as a service

The State Manager can be started as a service by using the Service utility (dbsvc). The start property in the options section of the Relay Server configuration file should be set to **no**. See "Options section" on page 26.

The Service utility is used to create, modify and delete services. For a full listing of usage information, run *dbsvc* without any options.

To set up an auto-started State Manager service named RelayServer on Windows

```
dbsvc -as -s auto -t rshost -w RelayServer "%SQLANY12%\RelayServer\IIS\BinXX
\Server\rshost.exe" -q -qc -f "%SQLANY12%\RelayServer\IIS\BinXX\Server
\rs.config" -o "c:\temp\ias_relay_server.log"
```


To set up an auto-started State Manager service named RelayServer on Unix

```
dbsvc -y -a <apache-user> -t rshost -w RelayServer -q -qc -f /<your-director>/
rs.config -os 100K -ot /tmp/rs.log
```

Remarks

The syntax of dbsvc on Windows is different than Unix. In Unix, you do not specify the full path of the executable as the first parameter after -w switch argument.

Use full paths only.

In Unix, use a user account (if possibly the same) so that Apache-user processes can attach to the State Manager shared memory and be able to read it and write to it.

To start the service

```
dbsvc.exe -u rs
```

To stop the service

```
dbsvc.exe -x rs
```

To uninstall the service

```
dbsvc.exe -d rs
```

Starting the Relay Server State Manager automatically

The State Manager process is started automatically when the first Outbound Enabler connects to the Relay Server. This is the default behavior when the start property in the options section of the Relay Server configuration file is not specified or is explicitly specified as auto. The default log file location is *%temp%\ias_relay_server_host.log*. See "Options section" on page 26.

Starting the Relay Server State Manager automatically with customized options

When auto start is desired but you want to override some default behavior such as verbosity level or log file location, you can use the start property in the options section of the Relay Server configuration file to explicitly specify your State Manager command line. The -f option cannot be used in this case and the configuration file must be named *rs.config* and be placed in the same directory as the server extension. See "Relay Server State Manager command line syntax" on page 21.

Note
If you are using IIS, do not specify a log file location under the wwwroot directory. Microsoft IIS does not allow a worker process to create a file under the published tree.

Relay Server State Manager command line syntax

rshost [*option*]+

Parameters

Options The following options can be used to configure the State Manager. They are all optional.

rshost options	Description
-f *filename*	Indicates the name of the Relay Server configuration file.
-o *filename*	Indicates the name of the file to use for logging.
-os *size*	Controls the size of the log file and provides additional information in the log file banner. When -os is specified, the old log is renamed using the *<yymmdd><nn>.olg* format. The log banner is rewritten to the new active log file, with the addition of the machine name, processor architecture, build target and operating system information.
-oq	Prevents a popup window if there is a startup error.
-q	Runs in a minimized window.
-qc	Closes the window on completion.
-u	Updates the configuration of a running Relay Server.
-ua	Archives the log file to *<yymmdd><nn>.log* and truncates the file.

Relay Server configuration file

A Relay Server configuration file is used to define both a Relay Server farm and the back-end server farms connecting to the Relay Server farm. The Relay Server configuration file is divided into sections:

- "Relay Server section" on page 23
- "Backend farm section" on page 24
- "Backend server section" on page 25
- "Options section" on page 26

Each section starts with a section tag. A section tag is formed by enclosing a keyword that identifies the section name in square brackets. For example, [relay_server] denotes the start of the Relay Server section.

The section tag is followed by several lines defining various properties related to the section being defined. A property is defined by specifying the property name on the left-hand side of an equal sign and its value on the right-hand side of the equal sign. For example, `property name = value`. All section and property names are case insensitive. Comments are marked with pound sign (#) character at the beginning of a line.

The configuration file should contain only 7-bit ASCII characters. The sections can be specified in any order.

Relay Server configuration files can be created, imported and deployed using the Relay Server plug-in for Sybase Central. See "Relay Server plug-in for Sybase Central" on page 37.

Relay Server section

The Relay Server section is used to define a single Relay Server, so there must be a Relay Server section for each Relay Server in the farm. This section is identified by the relay_server keyword.

Relay Server section properties

The following properties can be specified in a Relay Server section:

- **enable** Specifies whether this Relay Server is to be included in the Relay Server farm. Possible values are:
  - **Yes** Indicates that this Relay Server is to be included in the Relay Server farm.
  - **No** Indicates that this Relay Server should not be included in the Relay Server farm.

  The default is Yes. This property is optional.
- **host** The hostname or IP address that should be used by the Outbound Enabler to make a direct connection to the Relay Server.
- **http_port** The HTTP port that should be used by the Outbound Enabler to make a direct connection to the Relay Server. A value of **0** or **off** disables HTTP connections. By default, this property is enabled and set to 80.

- **0 or off** Disable HTTP access from Outbound Enabler.
- **1 to 65535** Enable HTTP at the specified port.

- **https_port** The HTTPS port that should be used by the Outbound Enabler to make a direct connection to the Relay Server. A value of **0** or **off** disables HTTPS connections. By default, this property is enabled and set to 443.
  - **0 or off** Disable HTTPS access from Outbound Enabler.
  - **1 to 65535** Enable HTTPS at the specified port.
- **description** Enter a custom description to a maximum of 2048 characters. This property is optional.

Backend farm section

The backend farm section specifies the properties of a back-end server farm. A back-end server farm is a group of homogeneous back-end servers. A client making a request through the Relay Server farm must specify the back-end server farm it is targeting. There is one backend farm section for each back-end server farm.

This section is identified by the backend_farm keyword.

Backend farm section properties

The following properties can be specified in a backend farm section:

- **active_cookie** Specifies whether or not a cookie is set to retain client-server affinity.
  - **yes** To maintain client-server session affinity, the Relay Server injects a standard HTTP set-cookie command with a proprietary cookie name in the response.
  - **no** An active cookie is not set. Use this option when the backend farm is serving a sessionless browser application. For example, when the backend farm is providing a sessionless SQL Anywhere web service.

  For best results, set this control as follows:

Backend server type	active_cookie setting	active_header setting
MobiLink	no	yes
SQL Anywhere	no	no

- **active_header** Specifies whether or not a header is set to maintain client-server session affinity.
  - **yes** To maintain client-server session affinity, the Relay Server injects a proprietary header in the response in case intermediaries tamper with the active_cookie.

  - **no** A proprietary header is not set. Setting this option cuts down on traffic volume if the backend farm is serving only browser applications, or if the active_cookie is working well for all the clients of this backend farm.

- **backend_security** Specifies the level of security required of an Outbound Enabler in the back-end server farm to connect to the Relay Server farm. The possible values are:

  - **on** Indicates that all connections from the back-end farm must by made using HTTPS.
  - **off** Indicates that all connections from the back-end farm must be made using HTTP.

  This property is optional. If no value is specified, either HTTP or HTTPS can be used to connect.

- **client_security** Specifies the level of security the back-end server farm requires of its clients. The possible values are:

  - **on** Indicates that clients must connect using HTTPS.
  - **off** Indicates that clients must connect using HTTP.

  This property is optional. If no value is specified, clients can connect using either HTTP or HTTPS.

- **description** Enter a custom description to a maximum of 2048 characters. This property is optional.
- **enable** Specifies whether to allow connections from this back-end server farm. Possible values are:

  - **Yes** Allow connections from this back-end server farm.
  - **No** Disallow connections from this back-end server farm.

  The default is Yes. This property is optional.

- **id** The name assigned to the back-end server farm, to a maximum of 2048 characters.
- **verbosity** You can set verbosity to the following levels:

  - **0** Log errors only. Use this logging level for deployment. This is the default.
  - **1** Request level logging. All HTTP requests are written to the log file.

  Errors are displayed regardless of the log level specified, and warnings are displayed only if the log level is greater than 0.

Backend server section

The backend server section defines a back-end server connection. It specifies the information that is used by the Outbound Enabler when it connects to the Relay Server farm on behalf of a back-end server. There is a backend server section for each Outbound Enabler connecting to the Relay Server farm. The backend server section also assigns a back-end server to a back-end server farm.

This section is identified by the backend_server keyword.

Backend server section properties

The following properties can be specified in a backend server section:

- **description** Enter a custom description to a maximum of 2048 characters. This property is optional.
- **enable** Specifies whether to allow connections from this back-end server. Possible values are:
  - **Yes** Allows connections from this back-end server.
  - **No** Disallows connections from this back-end server.

  The default is Yes. This property is optional.
- **farm** The name of the back-end server farm that this back-end server belongs to.
- **id** The name assigned to the back-end server connection, to a maximum of 2048 characters.
- **MAC** The MAC address of the network adapter used by the Outbound Enabler to communicate with the Relay Server. The address is specified using the IEEE 802 MAC-48 format. To get the MAC address in the correct format, look in the Relay Server Outbound Enabler console or log. This property is optional. If it is not specified, MAC address checking does not occur.
- **token** A security token that is used by the Relay Server to authenticate the back-end server connection, to a maximum of 2048 characters. This property is optional.
- **verbosity** You can set verbosity to the following levels:
  - **0** Log errors only. Use this logging level for deployment. This is the default.
  - **1** Request level logging. All HTTP requests are written to the log file.

  Errors are displayed regardless of the log level specified, and warnings are displayed only if the log level is greater than 0.

Options section

The options section is used to specify properties that apply to each Relay Server in the farm. Only one options section is allowed.

This section is identified by the options keyword.

Options section properties

The following properties can be specified in an options section:

- **start** The method used to start the State Manager. The possible values are:

- **auto** The State Manager is started automatically using the State Manager command line defaults.
- **no** The State Manager is started externally as a Windows service.
- **full path** Specify the full path to the State Manager executable (*rshost*).

The default is auto. This property is optional.

- **shared_mem** Specifies the maximum amount of shared memory that the Relay Server uses for state tracking. The default is 10 megabytes. This property is optional.
- **verbosity** You can set verbosity to the following levels:
  - **0** Log errors only. Use this logging level for deployment. This is the default.
  - **1** Request level logging. All HTTP requests are written to the log file.

  Errors are displayed regardless of the log level specified, and warnings are displayed only if the log level is greater than 0.

Relay Server configuration file format

This is the basic format of a Relay Server configuration file:

```
#
# Options
#
[options]
# List of Relay Server properties that apply to all Relay Servers
option = value

#
# Define a Relay Server section, one for each
# Relay Server in the Relay Server farm
#
[relay_server]
# List of properties for the Relay Server
property = value

#
# Define a backend server farm section, one for each backend
# server farm
#
[backend_farm]
# List of properties for a backend server farm
property = value

#
# Define a backend server section, one for each
# Outbound Enabler connecting to the Relay Server farm
#
[backend_server]
# List of properties for the backend server connection
property = value
```

Outbound Enabler

The Outbound Enabler runs on the same computer as the back-end server. Its purpose is to:

- Open an outbound connection from the computer running in the corporate LAN to the Relay Server farm running in the DMZ.
- Forward client requests received from the Relay Server to the back-end server and forward back-end server responses back to the client via the Relay Server.

When the Outbound Enabler starts, it makes an HTTP request to retrieve the list of Relay Servers running in the farm. This is done using the server URL that maps to the web server extension component of the Relay Server. The server URL can map directly to a Relay Server or it can map to a load balancer. If the server URL maps to a load balancer, the load balancer forwards the request to one of the Relay Servers running in the farm. The Relay Server that receives the request from the Outbound Enabler returns the connection information for all Relay Servers in the farm. The Outbound Enabler then creates two outbound connections, called channels, to each Relay Server returned. One channel, called the up channel, is created using an HTTP request with an essentially infinite response. The response is a continuous stream of client requests from the Relay Server to the Outbound Enabler. The second channel, called the down channel, is created using an HTTP request with an essentially infinite content length. The request is formed by a continuous stream of server responses to client requests.

When the Outbound Enabler receives a client request on the up channel from one of the Relay Servers it has connected to, it forwards it to the back-end server that the Outbound Enabler is servicing. Once a response is received from the back-end server, it gets forwarded to the Relay Server from which it received the corresponding request using the down channel.

Outbound Enabler syntax

rsoe [*option*]+

rsoe @{ *filename* | *environment-variable* } ...

Parameters

Options The following options can be used with the Outbound Enabler. The -cr option is required, all the others are optional.

rsoe options	Description
@data	Reads options from the specified environment variable or configuration file. If you want to protect passwords or other information in the configuration file, you can use the File Hiding utility to obfuscate the contents of the configuration file. See "File Hiding utility (dbfhide)" on page 32.

rsoe options	Description
-cr *"connection-string"*	Specifies the Relay Server connection string. The format of the Relay Server connection string is a semicolon separated list of name-value pairs. The name-value pairs consist of the following: • **host** IP address or hostname of the Relay Server. The default is localhost. • **port** The port the Relay Server is listening on. This is required. • **url_suffix** URL path to the server extension of the Relay Server. By default, the rsoe requires the url_suffix to be specified. • **https** 0 - HTTP (default) 1 - HTTPS For **https=1**, the following options can also be specified: • **tls_type** RSA • **certificate_name** Common name field of the certificate. • **certificate_company** Organization name field of the certificate. • **certificate_unit** Organization unit field of the certificate. • **trusted_certificates** File containing a list of trusted root certificates.
-cs *"connection-string"*	Sets the host and port used to connect to the back-end server. The default is `"host=localhost;port=80"`. To enable periodic backend server status requests, add the status_url parameter to -cs. The status_url parameter is specified in the format `status_url=/<your-status-url>`. The following example shows how to specify status_url with -cs. `-cs "host=localhost;port=80;status_url=/getstatus/"` Use the -d option to specify the frequency of the backend server status requests.
-d *seconds*	Sets the frequency of the backend server liveness ping and backend server status request. The default is 5 seconds.

rsoe options	Description
-dl	Use this option to display log messages in the Relay Server Outbound Enabler console. By default, log messages are not displayed for verbosity levels 1 and 2.
-f *farm*	Specifies the name of the farm that the back-end server belongs to.
-id *id*	Specifies the name assigned to the back-end server.
-o	Logs output messages to a file.
-oq	Prevents the appearance of the error window when a startup error occurs.
-os	Sets the maximum size of the message log files. The minimum size limit is 10 KB.
-ot	Truncates the log file and logs messages to it.
-q	Run with a minimized window on startup.
-qc	Shuts down the window on completion.
-s	Stops the Outbound Enabler.
-t *token*	Sets the security token to be passed to the Relay Server.
-uc	Starts the rsoe in shell mode. This is the default. Applies to Unix and Mac OS X. You should only specify one of -uc, -ui, -um, or -ux. When you specify -uc, this starts the rsoe in the same manner as previous releases of the software.
-ud	Instructs the rsoe to run as a daemon. This option applies to Unix platforms only.
-ui	Starts the rsoe in shell mode if a usable display is not available. This option is for Linux with X window server support. When -ui is specified, the server attempts to find a usable display. If it cannot find one, for example because the X window server isn't running, then the rsoe starts in shell mode.

rsoe options	Description
-ux	For Linux, opens the rsoe messages window where messages are displayed. When -ux is specified, the rsoe must be able to find a usable display. If it cannot find one, for example because the DISPLAY environment variable is not set or because the X window server is not running, the rsoe fails to start. To run the rsoe messages window in quiet mode, use -q. On Windows, the rsoe messages window appears automatically.
-v *level*	Set the verbosity level to use for logging. The *level* can be **0**, **1**, or **2**: ● **0** Log errors only. Use this logging level for deployment. ● **1** Session level logging. This is a higher level view of a synchronization session. ● **2** Request level logging. Provides a more detailed view of HTTP requests within a synchronization session. Levels 1 and 2 are only written to the log file and are not displayed. To have all log messages displayed, use the -dl switch.

File Hiding utility (dbfhide)

The File Hiding utility (dbfhide) uses simple encryption to obfuscate the contents of configuration files and initialization files.

Syntax

dbfhide *original-configuration-file encrypted-configuration-file*

Option	Description
original-configuration-file	Specifies the name of the original file.
encrypted-configuration-file	Specifies a name for the new obfuscated file.

The Relay Server and Outbound Enabler detect that a configuration file has been obfuscated using dbfhide and process it.

This utility does not accept the @data parameter to read in options from a configuration file.

Integrated Outbound Enabler

By using the OE protocol for the -x option for mlsrv12, you can use an integrated Outbound Enabler instead of the stand-alone Outbound Enabler invoked with the **rsoe** command. Using the integrated Outbound Enabler has the following advantages:

- Reduced use of system resources, especially sockets.
- Provides a single, integrated log file. Lines printed to the MobiLink server log from the integrated Outbound Enabler will have the prefix **<OE>**.
- Deployment is simplified.
- Liveness checks between the Outbound Enabler and the MobiLink server are eliminated.

For more information about how to use the integrated Outbound Enabler, see "-x mlsrv12 option" [*MobiLink - Server Administration*].

Deployment considerations

The following considerations should be noted when using the Outbound Enabler:

- **Outbound Enabler as a service** The Outbound Enabler may also be set up and maintained as a service using the Service utility. See "Outbound Enabler as a service" on page 33.
- **Authentication** You cannot use simple or digest authentication. The *rsoe.exe* does not support simple or digest authentication with web servers, regardless of the web server type or operating system.

Outbound Enabler as a service

The Outbound Enabler can be started as a service by using the Service utility (dbsvc). The Service utility is used to create, modify and delete services. For a full listing of usage information, run *dbsvc* without any options.

To set up an auto-started RSOE service named oes (Outbound Enabler service) on Windows

```
dbsvc -as -s auto -t rsoe -w oes "%SQLANY12%\BinXX\rsoe.exe"
-cr "host=relayserver.sybase.com;port=80 " -cs "host=localhost;port=80 " -f
FarmName -id ServerName -t token
```

To set up an auto-started RSOE service named oes (Outbound Enabler service) on Unix

```
dbsvc -y -a <some-user-account> -t rsoe -w oes @/<full-dir-path>/oe.config
```

Remarks

The syntax of dbsvc on Windows is different than Unix. In Unix, you do not specify the full path of the executable as the first parameter after -w switch argument.

Use full paths only.

On Unix, specify the Outbound Enabler parameters in a command file only. Do not use command line switches in the setup dbsvc command.

To start the service

```
dbsvc.exe -u oes
```

To stop the service

```
dbsvc.exe -x oes
```

To uninstall the service

```
dbsvc.exe -d oes
```

See also

- "SQL Anywhere web services high availability and scale-out solutions" [*SQL Anywhere Server - Database Administration*]

Updating a Relay Server farm configuration

A Relay Server farm configuration is defined by the contents of the Relay Server configuration file. Each Relay Server in a Relay Server farm shares the same Relay Server configuration file, so when you update a Relay Server farm configuration you must update the Relay Server configuration file at each Relay Server in the farm. Updates include any of the following:

- Adding a new Relay Server to the Relay Server farm.
- Creating a new backend server farm and allowing it access to the Relay Server farm.
- Adding a new backend server to an existing backend server farm.
- Changing the properties of a Relay Server, backend server farm, or a backend server.
- Changing options.

One way to update a Relay Server configuration is to shutdown all Relay Servers, replace the Relay Server configuration file with the updated version, and restart all the Relay Servers. However, shutting down and restarting the Relay Servers means that users of the Relay Server may incur a service interruption.

The preferred method of updating a Relay Server configuration is to use the Relay Server State Manager to update the configuration while a Relay Server farm is running without interrupting service.

Updating a Relay Server configuration is done by launching a new instance of the Relay Server State Manager using the following command line format:

```
rshost -u -f <filename>
```

The -u option instructs the Relay Server State Manager to perform an update operation. The -f option specifies the name of the configuration file containing the updated configuration. See “Relay Server State Manager” on page 19.

Below is an overview of the steps required to update a Relay Server farm configuration:

1. Make your changes to the master copy of the Relay Server configuration file.
2. On each computer running an instance of a Relay Server that belongs to the Relay Server farm being updated, do the following:
   a. Replace the old configuration file with the updated configuration file.
   b. Run the Relay Server State Manager with the updated configuration file.

Updating a Relay Server configuration for Microsoft IIS on Windows

To update a Relay Server configuration for Microsoft IIS on Windows

1. For each computer that belongs to the Relay Server farm you are updating, copy the updated configuration file to the *%SQLANY12%\RelayServer\IIS\BinXX\Server* directory under the Relay Server web site home directory. The configuration file must be called *rs.config* if auto start is used.
2. From the *%SQLANY12%\RelayServer\IIS\BinXX\Server* directory, run the following command line to apply the configuration update:

   ```
   rshost -u -f rs.config
   ```

3. Repeat the previous steps for each computer in the Relay Server farm that is being updated.

Updating a Relay Server configuration for Apache on Linux

To update a Relay Server configuration for Apache on Linux

1. Copy the updated configuration file to the */modules* directory under the Apache install directory. The configuration file must be called *rs.config* if auto start is used.
2. From the */<Apache-install>/modules* directory, run the following command line to apply the configuration update:

   ```
   rshost -u -f rs.config
   ```

3. Repeat the previous steps for each computer in the Relay Server farm that is being updated.

Relay Server plug-in for Sybase Central

The Relay Server plug-in for Sybase Central provides an easy way to work with the Relay Server. Use the Relay Server plug-in to do the following:

- Create, import, and deploy Relay Server configuration files.
- View Relay Server configuration file properties.
- Add Relay Servers, Relay Server farms, backend servers and backend server farms.
- View and edit Relay Servers, Relay Server farms, backend servers and backend server farms.

Working with Relay Server configuration files (Sybase Central)

You can use Sybase Central to work with Relay Server configuration files. From Sybase Central you can:

- Create a Relay Server configuration file.
- Open a Relay Server configuration file.
- Import a Relay Server configuration file.
- Deploy a Relay Server configuration file.

To create a Relay Server configuration file

1. In the **Folders** view of Sybase Central, right-click **Relay Server 12** and choose **New** » **Configuration File**.
2. Browse to the directory where you want the configuration file saved on the machine running Sybase Central. This is not the same as the deployment location.
3. In the **File Name** field, type the name of the configuration file. Normally this would be *rs.config*.
4. Ensure the *.config* extension is selected in the **Save as type** field.
5. Click **Save**. A Relay Server farm is automatically created, to which you can add the necessary Relay Servers and backend servers.

To open a Relay Server configuration file

1. In the **Folders** view of Sybase Central, right-click **Relay Server 12** and choose **Open Configuration File**.
2. Browse to the directory where the configuration file is located, select the file and click **Open**.

To import a Relay Server configuration file

1. In the **Folders** view of Sybase Central, right-click **Relay Server 12** and choose **Import Configuration File**.
2. Enter the URL for the existing Relay Server.
3. If the Relay Server requires authentication, enter the **User Name** and **Password** and click **OK**.

> **Note**
> If the Relay Server requires HTTPS communication, the root certificate for the server needs to be stocked in the Java Key and Certificate Management Utility. This can be accomplished using the Java Keytool utility. Sybase Central accesses the Java Key and Certificate Management Utility when the root certificate is required for communication.

To deploy a Relay Server configuration file

1. In the **Folders** view, right-click the Relay Server configuration file you want to deploy and choose **Deploy**.
2. Enter the URL for the Relay Server.
3. If the Relay Server requires authentication, enter the **User Name** and **Password** and click **OK**.

> **Note**
> If the Relay Server requires HTTPS communication, the root certificate for the server needs to be stocked in the Java Key and Certificate Management Utility. This can be accomplished using the Java Keytool utility. Sybase Central accesses the Java Key and Certificate Management Utility when the root certificate is required for communication.

4. The **Server List** page shows existing Relay Servers. To deploy the configuration file to one or more of the Relay Servers, select the server(s) from the list and click **Add**.

   To remove a Relay Server from the list, select the server(s) and click **Remove**.

Managing Relay Servers and Relay Server farms (Sybase Central)

You can use Sybase Central to manage Relay Servers and Relay Server farms. From Sybase Central you can:

- Add Relay Servers to a Relay Server farm.
- View or edit Relay Server properties.
- View or edit Relay Server farm properties.

To add Relay Servers to a farm

1. In the **Folders** pane under the configuration file you want to work with, right-click the Relay Server farm you want to add Relay Servers to and choose **New** » **Relay Server**.

2. Ensure **Enable this Relay Server** is selected.

3. Enter the path information for the **Host** you want to connect to. Click **Ping** if you want to check that a connection can be established to the specified host.

4. Select the communication protocol to use. This can be **HTTP** or **HTTPS**.

5. Specify the port(s) to be used for the selected protocol(s).

6. If desired, type a description of the Relay Server in the **Description** field.

7. Click **Apply** if you want to continue adding Relay Servers or **OK** to add the Relay Server and close the **Create Relay Server** window.

To view or edit Relay Server properties

1. In the **Folders** pane, click the Relay Server farm that contains the Relay Server you want to work with. The Relay Servers in that farm are listed in the right pane.

2. Right-click the Relay Server you want to edit or view, and choose **Properties**.

3. Makes the necessary changes to the Relay Server properties and click **Apply** or **OK**.

To view or edit Relay Server farm properties

1. In the **Folders** pane, right-click the Relay Server farm you want to work with and choose **Properties**.

2. Makes the necessary changes to the Relay Server farm properties and click **Apply** or **OK**.

Managing backend servers and backend server farms

You can use Sybase Central to manage backend servers and backend server farms. From Sybase Central you can:

- Create a backend server farm and add backend servers to it.
- View or edit backend server properties.
- View or edit backend server farm properties.

To create a backend server farm

1. In the left pane, right-click the Relay Server configuration file you want to work with, and choose **New** » **Backend Server Farm**.

2. Ensure **Enable this backend server farm** is selected.

3. Type the name associated with the new backend server farm.

4. Choose a protocol for clients to use to connect to the backend server farm.

5. Choose a protocol for the Relay Server Outbound Enabler (rsoe) to use to connect to the backend server farm.

6. For client-server affinity, choose the type of server you are using. For a MobiLink HTTP server with standalone Outbound Enabler or a MobiLink server with an embedded Outbound Enabler, select **MobiLink**. For a typical SQL Anywhere web service select **SQL Anywhere**. Advanced custom settings are available by selecting **Custom** as the server type. Once the **Custom** server type is selected, you have full control over the following affinity settings:
   a. Check the **Active_cookie** option if you want the Relay Server to use a standard HTTP set-cookie command to retain client-server affinity.
   b. Check the **Active_header** option if you want the Relay Server to use a proprietary header to retain client-server affinity.

7. If desired, type a description of the backend server farm in the **Description** field.

8. Click **Apply** if you want to continue adding backend server farms or **OK** to add the backend server farm and close the **Create Backend Server Farm** window.

To view or edit backend server farm properties

1. In the **Folders** pane, right-click the backend server farm you want to work with and choose **Properties**.

2. Makes the necessary changes to the backend server farm properties and click **Apply** or **OK**.

To add servers to a backend server farm

1. In the **Folders** pane, right-click the backend server farm you want to work with and choose **New** » **Backend Server**.

2. Ensure **Enable this backend server** is selected.

3. Type the name associated with the new backend server.

4. To enforce MAC address checking, click to select the **Enforce MAC Address Checking** checkbox.

5. If you selected MAC address checking, enter the RSOE MAC address using the IEEE 802 MAC-48 format. To get the MAC address in the correct format, look in the Relay Server Outbound Enabler console or log. Multiple MAC addresses separated by an exclamation mark (!) are reported by the Outbound Enabler if multiple adapters are currently active on your backend server machine. Select the most permanant one for the Relay Server to check against. The `ipconfig /all` command on Windows provides a detailed listing of your network adapters together with associated MAC addresses.

6. Specify the security token that is used by the Relay Server to authenticate the back-end server connection. You can use a maximum of 2048 characters.

7. If desired, type a description of the backend server in the **Description** field.

8. Click **Apply** if you want to continue adding backend servers or **OK** to add the backend server and close the **Create Backend Server** window.

To view or edit backend server properties

1. In the **Folders** pane, click the backend server farm that contains the backend server you want to work with. The backend servers in that farm are listed in the right pane.
2. Right-click the backend server you want to edit or view, and choose **Properties**.
3. Makes the necessary changes to the backend server properties and click **Apply** or **OK**.

Monitoring your Relay Server

This section describes how to use the SQL Anywhere Monitor to monitor your Relay Server farms.

SQL Anywhere Monitor for Relay Server

The SQL Anywhere Monitor, also referred to as the Monitor, is a browser-based administration tool that provides you with information about the health and availability of SQL Anywhere databases, MobiLink servers, MobiLink server farms, and Relay Server farms.

This section describes how to use the Monitor to collect metrics about *Relay Server farms*. For information about using the Monitor with:

- SQL Anywhere databases, see "SQL Anywhere Monitor" [*SQL Anywhere Server - Database Administration*].
- MobiLink servers and MobiLink server farms, see "SQL Anywhere Monitor for MobiLink" [*MobiLink - Server Administration*].

The Monitor provides the following functionality:

- **Constant data collection** Unlike many of the other administration tools available with SQL Anywhere, the Monitor collects metrics all the time, even when you are not logged in to the browser. The Monitor collects metrics until you shut it down.
- **Email alert notification** As the metrics are collected, the Monitor examines the metrics and can send email alerts when it detects conditions that indicate something is wrong with a resource.
- **Browser-based interface** At any time, you can log in to the Monitor using a browser to review alerts and metrics that have been collected.
- **Monitor multiple databases, MobiLink servers, MobiLink server farms, and Relay Server farms** From one tool, you can simultaneously monitor multiple resources running on the same or different computers.
- **Minimal performance impact** The Monitor can be used routinely in development and production environments because monitoring does not degrade performance.

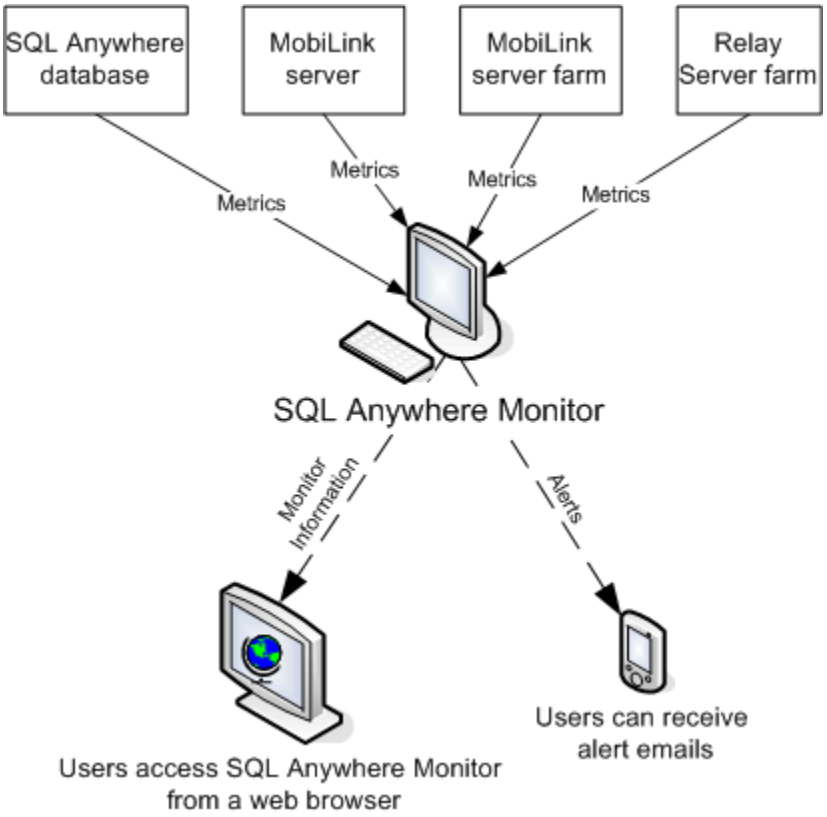
The Monitor is designed to help any type of user, whether they are a DBA or not, who is responsible for such tasks as:

- Ensuring that a Relay Server farm is connected to the network.
- Ensuring that there is enough disk space or memory available for a Relay Server farm.
- Ensuring that a Relay Server is ready to relay traffic.
- Reviewing the traffic volume, throughput, and failure rate of a Relay Server farm.

See also

Monitor architecture

The Monitor collects metrics and performance data from SQL Anywhere databases, MobiLink servers, MobiLink server farms, and Relay Server farms running on other computers, while a separate computer accesses the Monitor via a browser.



The Monitor is a Flash-based application that is served to a web browser via the SQL Anywhere built-in HTTP server. You interact with the Monitor through its browser interface.

There are two editions of the Monitor:

- **SQL Anywhere Monitor Developer Edition** The Monitor Developer Edition is intended for development and testing use. It is installed by default with SQL Anywhere and it uses the installed SQL Anywhere on the back-end.
- **SQL Anywhere Monitor Production Edition** This Production Edition is intended for deployment and production use. It is installed separately, runs as a service, and includes a fully-contained SQL

Anywhere installation. This edition is only available for certain editions of SQL Anywhere. See "Installing the SQL Anywhere Monitor in a production environment" on page 86.

Requirements

For the computer where the Monitor is installed

- To run the SQL Anywhere Monitor on a Windows system that has the Windows Firewall enabled, you must add a port exception for port 4950.
- The SQL Anywhere Monitor reserves 384 MB of virtual address space when you start the Monitor. When you start the Monitor on Linux, you must ensure that you have least this amount available.
- To monitor resources that are secured using ECC encryption or FIPS-certified encryption, you need a separate license. See "Securing the Monitor" on page 88.
- The Monitor can run on the same computer as the resources it is monitoring, but it is recommended, particularly in production environments, that you run the Monitor on a different computer to minimize the impact on the Relay Server farm or other applications.
- It is recommended in production environments that you run the Monitor Production Edition. See "Installing the SQL Anywhere Monitor in a production environment" on page 86.
- When running the Monitor Developer Edition, you must have SQL Anywhere 12.0.0 installed. The Monitor Developer Edition uses the installed SQL Anywhere on the back-end.

For the computer accessing the Monitor

- Install the latest version of Adobe Flash Player that is available for your operating system. The Monitor is backwards compatible with version 10 of Adobe Flash Player. To determine the correct version, see http://www.adobe.com/products/flashplayer/systemreqs/.
- Enable JavaScript in your browser.
- Ensure that the computer where you are using a browser to access the Monitor is connected to the network where the Monitor is installed.

  You can access the Monitor from the same computer as it is running on, but it is recommended, particularly in production environments, that you access the Monitor from a different computer.

Limitations

- You can use the Monitor to collect metrics about the following versions of SQL Anywhere databases, MobiLink servers, MobiLink server farms, and Relay Server farms:
  - SQL Anywhere 9.0.2, 10.0.0, 10.0.1, 11.0.0, 11.0.1, and 12.0.0
  - MobiLink 11.0.0 with at least the first EBF applied, 11.0.1, and 12.0.0
  - Relay Server 12.0.0

- You can only run one edition of the Monitor on a computer at a time.
- On Linux, the Monitor Developer Edition can only be run by the user who installed it.
- On Linux, only the user with administrator privilege can install and run the Monitor Production Edition.

See also

- "Monitor quick start" on page 46

Monitor quick start

The following steps are required to set up Relay Server farm monitoring:

1. Start the Relay Server farm that you want to monitor (if it is not already running).
2. Install the Monitor on a computer that is always connected to your network.

   The Monitor can run on the same computer as the resources it is monitoring, but it is recommended, particularly in production environments, that you run the Monitor on a different computer to minimize the impact on the Relay Server Farm or other applications.
3. Start the Monitor and open it in your browser. See "Start the Monitor" on page 54.
4. Log in to the Monitor as an administrator. The default user is an administrator user with the name **admin** and the password **admin**.

   Note
   You must be logged in to the Monitor as an administrator to perform the following tasks. See "Monitor users" on page 75.
5. As an administrator, in the left navigation menu, choose **Tools** » **Administration** and add as a resource to be monitored. See "Add resources" on page 66.
6. As an administrator, add new users and change the password for the admin user. See "Create Monitor users" on page 76.
7. As an administrator, configure alert thresholds for the resources. See "Alerts" on page 79.
8. As an administrator, configure the backup and maintenance schedule. See "Back up the Monitor" on page 85.
9. Click **Close** to exit **Administration**.
10. Click **Overview**. In the **Resource List** widget, you will see the resources that are being monitored. See "Overview dashboard" on page 59.

Tutorial: Using the Monitor

Use this tutorial to set up monitoring of a Relay Server farm. This tutorial uses the Monitor Developer Edition.

Lesson 1: Log in to the Monitor as the default administrator

To start and log in to the Monitor

1. Start the Monitor. The following steps assume that the Monitor is not currently running in the background.

   To start the Monitor Developer Edition (Windows): Choose **Start** » **Programs** » **SQL Anywhere 12** » **Administration Tools** » **SQL Anywhere Monitor**.

   To start the Monitor Developer Edition (Linux): Run the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory. Run the following:

   ```
   samonitor.sh launch
   ```

   The Monitor starts collecting metrics and a browser opens the default URL where you can log in to the Monitor: *http://localhost:4950*.

   Note
   If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running.

2. Log in to the Monitor as the default *administrator* user.

   In the **User Name** field, type **admin**, and in the **Password** field, type **admin**.

   Note
   You must be logged in to the Monitor as an administrator to perform the following lessons in the tutorial. Read-only and operator users do not have permission to perform all the tasks.

   To check your Monitor user type

   1. Log in to the Monitor.
   2. Choose **Tools** » **User Settings** and review the **User Type** setting.

   See "Monitor users" on page 75.

By default, the Monitor opens the **Overview** dashboard that contains two widgets:

1. The **Resource List** widget lists the resources that are being monitored. When you first open the Monitor, it is only monitoring itself via the default resource, named **SQL Anywhere Monitor**. You cannot modify this resource, nor can you stop monitoring it.

2. The **Alerts List** widget lists any alerts from the monitored resources.

See also

- "Lesson 2: Set up the Monitor to monitor a Relay Server farm" on page 48
- "Start the Monitor" on page 54

Lesson 2: Set up the Monitor to monitor a Relay Server farm

In this lesson, you start a Relay Server Farm, and then add it as a resource to be monitored.

To add a resource to the Monitor

1. Start the Relay Server Farm.
2. Log in to the Monitor as an administrator. See "Lesson 1: Log in to the Monitor as the default administrator" on page 47.
3. Choose **Tools** » **Administration**.
4. Select **Resources**, and then click **Add**.
5. Select **Relay Server Farm**, and then click **Next**.
6. In the **Name** field, type a name for the Relay Server Farm resource, and then click **Next**.
7. In the **Host** field, type the hostname or IP address of the computer on which the Relay Server is running, and then, click **Next**.

   This term is required. See "Host connection parameter" [*SQL Anywhere Server - Database Administration*].
8. Click **Create**.
9. The new resource is created and monitoring starts.

10. Click **OK**.

11. Click **Close**.

12. Choose **Overview** » **Resource List**. Click the resource to create and open a dashboard for the resource.

See also

- To collect metrics from a SQL Anywhere database, see "Lesson 2: Set up the Monitor to monitor a database" [*SQL Anywhere Server - Database Administration*].
- To collect metrics from a MobiLink server or MobiLink server farm, see "Lesson 2: Set up the Monitor to monitor a MobiLink server" [*MobiLink - Server Administration*].

Lesson 3: Test an alert

In this lesson, you intentionally trigger an alert so you can practice handling alerts.

To view and resolve an alert

1. Trigger an alert by shutting down the resource.

2. Log in to the Monitor as an administrator. See "Lesson 1: Log in to the Monitor as the default administrator" on page 47.

3. Choose **Dashboards** » **Overview**.

   In the **Resource List**, the **Status** for the Relay Server Farm resource changes to a red circle with a white x through it. This icon indicates that the resource is unavailable.

   In the **Alerts** widget, an **Availability Alert** appears and its status is **Active**.

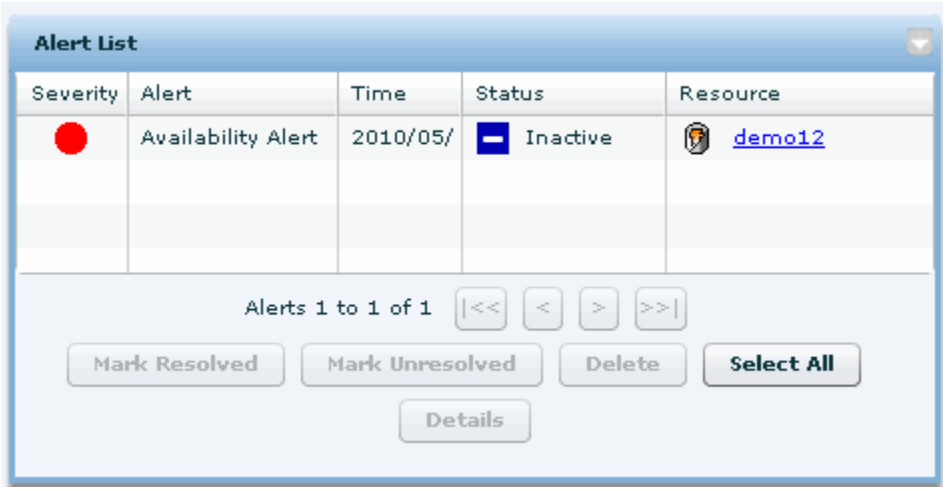
   It can take a few seconds for these changes in state and status to occur. By default, the Monitor collects information from the resource every 30 seconds. See "Collection intervals" on page 71.

4. In the **Alerts** dashboard, click the **Availability Alert**, and then click **Details** to read the description.

5. Click **OK** to close the alert.

6. Restart the Relay Server Farm.

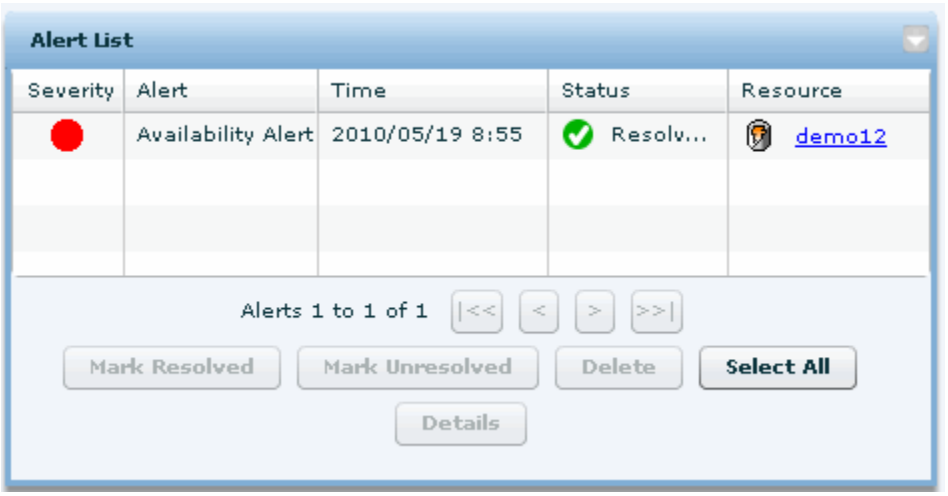
   In the Monitor, in the **Resource List**, the **Status** for the Relay Server Farm resource changes to a yellow triangle. This icon indicates that the resource is being monitored and it has an alert. In the **Alerts List**, the **State** of the **Availability Alert** changes to **Inactive**. An inactive alert indicates that the issue that triggered the alert is no longer present, but the alert has not been resolved or deleted.



7. Resolve the alert by selecting the alert and clicking **Mark Resolved**.

> **Note**
> Only administrators and operators can resolve and delete alerts. See "Monitor users" on page 75.

Now in the **Resource List**, the **Status** for the Relay Server Farm resource is blank. No icon in the **Status** column indicates that the resource is being monitored and it has no alerts. In the **Alerts List**, the **State** of the **Availability Alert** changes to **Resolved by *admin*** where *admin* is your Monitor user name.



8. Delete the alert by selecting the alert and clicking **Delete**.

When prompted, click **Yes** to confirm the deletion.

Lesson 4: Set up the Monitor to send emails when alerts occur

When an alert occurs, it is always listed in the **Alerts List** widget on the dashboard for the particular resource. See "Lesson 3: Test an alert" on page 49.

In this lesson, you set up the Monitor to send you an email whenever an alert occurs.

To set up email notification

1. Log in to the Monitor as an administrator. See "Lesson 1: Log in to the Monitor as the default administrator" on page 47.

2. Add a user that can receive emails.
   a. Choose **Tools** » **Administration**.
   b. Choose **Users**, and then click Add.
   c. In the User Name field, type **JoeSmith**.
   d. In the **Password** and the **Confirm Password** fields, type **password**.
   e. In the **Email** field, enter a valid email address.
   f. For the **User Type**, select **Operator**.

      An operator can receive alerts via email and can resolve and delete alerts. This user can access most of the Monitor widgets but it cannot not access the **Administration** window.
   g. Click **Next**.
   h. When prompted to choose the resources you are interested in, click **Check All**.
   i. Click **Save**.

      The new user is created and you are returned to the **Administration** window.

3. Configure email alert notification.

   Note
   You must be logged in to the Monitor as an administrator to perform the following tasks. Only administrators can configure the Monitor to send emails. See "Monitor users" on page 75.

   a. In **Administration** window, select **Configuration** and click **Edit**.
   b. On the **Alert Notifications** tab, select **Send Alert Notifications By Email**.
   c. Configure other settings as required. See "Enable the Monitor to send alert emails" on page 83.
   d. Test that you have properly configured email notification.

      Click **Send Test Email**.
   e. When prompted, enter an email address to send the test email to and click **OK**.

      A test email is sent to the email address specified.
   f. Click **Save**.
   g. Click **Close**.

When an alert occurs, an email is sent to the specified user with information about the alert.

Lesson 5: Add a new dashboard and widgets

Dashboards are specific to each user. Any user can add, edit or delete their own dashboards. In addition, any user can add, edit, or delete the widgets that exist in their dashboards. By default, when a dashboard is created, it is populated with default widgets. In this lesson, you add a dashboard to the Monitor, and then add widgets.

To add a new dashboard

1. Log in to the Monitor.
2. Choose **Dashboards** » **Add new**.
3. Select **A Dashboard To Monitor The Following Resource**, and choose the resource.
4. In the **Dashboard Name** field, type **user_joe**.
5. In the **Number Of Columns** field, type **2**.
6. Click **OK**.

   A new dashboard appears, with the 4 following widgets: **Alert List**, **Key Performance Metrics**, **Server Info**, and **Connections**.

The following procedure describes how to create a new metrics widget that displays graphs, instead of the default spark lines.

To add a metrics display widget with graphs

1. In the upper right corner of the **user_joe** dashboard, click **Customize**, and then click **Add Widget**.
2. Select **Metrics**, and then click **Next**.
3. In the **What Do You Want To Name This Widget?** field, type **Metrics display**.
4. In the **Which Resource Are You Interested In?** field, choose the Relay Server Farm resource.
5. In the **What Kind Of Display Do You Want To See?** field, choose **Graph**.
6. For **Which metrics do you want to see?**, select **CPU Usage**, **Memory Metrics** » **Cache Size**, and **Queries Processed**.
7. Click **Create**.

   A widget called **Metrics display** appears in the dashboard.

   To maximize the size of a widget, in the widget pane, click the dropdown menu arrow, and choose **Maximize**.

To view details on the graph, position your cursor above specific points on the graph.

See also

- "Dashboards" on page 61
- "Widgets" on page 62

Lesson 6: Cleanup

In this lesson, you remove the Relay Server resource, which deletes the collected metrics and stops data collection. In a production environment when you want to continue monitoring your Relay Server farm, you leave both the Relay Server farm and the Monitor running.

Note
You must be logged in to the Monitor as an administrator to perform the following tasks. Only administrators can remove resources. You cannot delete the **SQL Anywhere Monitor** resource. See "Monitor users" on page 75.

To stop monitoring a resource

1. Log in to the Monitor as an administrator.
2. Remove the Relay Server Farm resource.
   a. Choose **Tools** » **Administration**.
   b. Select **Resources**.
   c. Select the Relay Server Farm resource, and click **Stop**.
   d. Click **Remove**.
   e. Click **Yes** to confirm that you want to remove the resource.
   f. Click **Close**.
3. Log out of the Monitor.

   Click **Logout**.
4. Close the browser window where you are viewing the Monitor.
5. Exit the Monitor and stop monitoring.
   - On Windows, in the system tray, right-click the SQL Anywhere Monitor icon and choose **Exit SQL Anywhere Monitor**.
   - On Linux, run the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory:

     ```
     samonitor.sh stop
     ```

   The Monitor stops collecting metrics for all resources.

6. Shut down the Relay Server Farm.

Start the Monitor

When you start the Monitor, it connects to the resources and collects metrics for *all* resources in the Monitor.

The Monitor is designed to run silently in the background. You interact with the Monitor through its browser interface. It is recommended that you leave the Monitor running continuously in the background to collect the metrics.

Note
You can only run one edition of the Monitor on a computer at a time.

To start the Monitor Developer Edition (Windows)

1. On the computer where the Monitor is installed, choose **Start** » **Programs** » **SQL Anywhere 12** » **Administration Tools** » **SQL Anywhere Monitor**.

   The Monitor connects to the resources and begins collecting metrics for all resources in the Monitor, and:

   - A browser opens the default URL for logging in to the Monitor: *http://localhost:4950*.

     Note
     If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running.

     See "Log in remotely to the Monitor" on page 58.

   - The SQL Anywhere Monitor icon appears in the system tray.

2. Log in.

   In the browser, enter your user name and password for the Monitor. The user name and password for the Monitor are case sensitive. There are three different types of users: administrators, operators, and read-only users. Each type of user has different permissions. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

   You can select **Remember Me On This Computer** to have your session persist for two weeks or until you log out. If **Remember Me On This Computer** is cleared, your session expires when you close the browser or log out.

To start the Monitor Developer Edition (Linux)

1. On the computer where the Monitor is installed, start the Monitor.

   Run the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory:

   ```
   samonitor.sh launch
   ```


The Monitor connects to the resources and begins collecting metrics for all resources in the Monitor, and a browser opens the default URL for logging in to the Monitor: *http://localhost:4950*.

Note
If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

2. Log in.

   In the browser, enter your user name and password for the Monitor. The user name and password for the Monitor are case sensitive. There are three different types of users: administrators, operators, and read-only users. Each type of user has different permissions. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

   You can select **Remember Me On This Computer** to have your session persist for two weeks or until you log out. If **Remember Me On This Computer** is cleared, your session expires when you close the browser or log out.

To start the Monitor Production Edition (Windows)

1. On Windows, by default, the Monitor Production Edition runs automatically as a service when installed. Because it runs as a service, this Monitor also starts automatically when the computer starts. If you stop the Monitor, you can restart the Monitor service by doing one of the following steps:

   - On the computer where the Monitor is installed, choose **Start** » **Programs** » **SQL Anywhere 12 Monitor** » **SQL Anywhere Monitor**.
   - Run the *samonitor* batch file from the *bin32* or *bin64* directory in the Monitor installation directory. This batch file starts the Monitor service:

     ```
     samonitor.bat start
     ```

   The Monitor starts collecting metrics.

2. Browse to the URL for logging in to the Monitor. The default URL is *http://localhost:4950*.

Note
If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

3. Log in.

   When prompted, enter your user name and password for the Monitor. The user name and password for the Monitor are case sensitive. There are three different types of users: administrators, operators, and read-only users. Each type of user has different permissions. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

You can select **Remember Me On This Computer** to have your session persist for two weeks or until you log out. If **Remember Me On This Computer** is cleared, your session expires when you close the browser or log out.

To start the Monitor Production Edition (Linux)

1. On Linux, by default, the Monitor Production Edition runs automatically as a service when installed. Because it runs as a service, this Monitor also starts automatically when the computer starts. If you stop the Monitor, you can restart the Monitor service by running the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory. This script starts the Monitor service:

   ```
   samonitor.sh launch
   ```

   The Monitor starts collecting metrics and the browser opens.

2. Browse to the URL for logging in to the Monitor. The default URL is *http://localhost:4950*.

   Note
   If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

   The Monitor connects to the resources and begins collecting metrics for all resources in the Monitor, and a browser opens the default URL for logging in to the Monitor: *http://localhost:4950*.

   Note
   If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

3. Log in.

   When prompted, enter your user name and password for the Monitor. The user name and password for the Monitor are case sensitive. There are three different types of users: administrators, operators, and read-only users. Each type of user has different permissions. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

   You can select **Remember Me On This Computer** to have your session persist for two weeks or until you log out. If **Remember Me On This Computer** is cleared, your session expires when you close the browser or log out.

See also

- "Stop the Monitor" on page 57
- "To stop the Monitor Production Edition" on page 57
- "Log out from the Monitor" on page 58
- "Overview dashboard" on page 59

Stop the Monitor

Stopping the Monitor stops the collection of metrics for all resources.

Caution
In most cases, it is recommended that you leave the Monitor running, but close the browser. Closing the browser does not stop the collection of metrics.

To stop monitoring a specific Relay Server farm see "Stop monitoring resources" on page 68.

To stop the Monitor Developer Edition (Windows)

- In the system tray, right-click the SQL Anywhere Monitor icon and choose **Exit SQL Anywhere Monitor**.

  The Monitor stops collecting metrics for all resources.

  To restart the Monitor, see .

To stop the Monitor Developer Edition (Linux)

- Run the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory:

  ```
  samonitor.sh stop
  ```

  The Monitor stops collecting metrics for all resources.

  To restart the Monitor, see .

To stop the Monitor Production Edition

By default, the Monitor Production Edition runs automatically as a service when installed on Windows and Linux. To stop the Monitor and the collection of all resource metrics, you must stop the service.

To stop the Monitor service (Windows)

- To stop the Monitor service, run *samonitor.bat* from the *bin32* directory in the Monitor installation directory:

  ```
  samonitor.bat stop
  ```

  This batch file stops the Monitor service. The Monitor stops collecting metrics for all resources.

  To restart the Monitor service, see "Start the Monitor" on page 54.

To stop the Monitor service (Linux)

- To stop the Monitor service, run the *samonitor.sh* script from the *bin32* or *bin64* directory in the Monitor installation directory:

```
samonitor.sh stop
```

This script stops the Monitor service. The Monitor stops collecting metrics for all resources.

To restart the Monitor service, see "Start the Monitor" on page 54.

See also

- "Start the Monitor" on page 54
- "Log in remotely to the Monitor" on page 58
- "Log out from the Monitor" on page 58
- "Overview dashboard" on page 59

Log in remotely to the Monitor

The computer that you are using to log in to the Monitor must be connected to the network where the Monitor is running.

To log in to the Monitor

1. On the computer where the Monitor is installed, start the Monitor. See "Start the Monitor" on page 54.
2. On the computer that is accessing the Monitor, browse to the default URL for logging in to the Monitor: **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. For example, *http://samplehost:4950*.
3. When prompted, enter your user name and password for the Monitor. The user name and password for the Monitor are case sensitive. See "Monitor users" on page 75.

See also

- "Start the Monitor" on page 54
- "Stop the Monitor" on page 57
- "Log out from the Monitor" on page 58
- "Overview dashboard" on page 59

Log out from the Monitor

Logging out from the Monitor has *no* effect on the collection of metrics. If you want to stop collecting metrics, then:

- Stop monitoring the resource. See "Stop monitoring resources" on page 68.
- Exit the Monitor. See "Stop the Monitor" on page 57.

To log out from the Monitor

- Click **Logout** in the upper right corner.

  If you select **Remember Me On This Computer** when you log in to the Monitor, then closing the browser does not log you out of the Monitor.

See also

- "Start the Monitor" on page 54
- "Stop the Monitor" on page 57
- "Log in remotely to the Monitor" on page 58
- "Overview dashboard" on page 59

Overview dashboard

The **Overview** dashboard provides an overview of the health and availability of the resources (for example, Relay Server farms) being monitored.

By default, the **Overview** dashboard contains the **Resource List** widget and the **Alert List** widget.

1: Resource List widget

The **Resource List** widget contains a table that lists the resources being monitored, as well as an indication about the overall health of each resource. The table always contains the default resource, named **SQL Anywhere Monitor**, which reports on the health of the Monitor itself. You cannot modify the **SQL Anywhere Monitor** resource, nor can you stop monitoring it.

To view detailed information about a resource

- Click the resource name in the **Resource List** widget.

  The dashboard for the selected resource opens. See "Dashboards" on page 61.

In the **Resource List** widget, the **Status** column provides information about the connections between the Monitor and its resources. The **Status** column indicates whether the resource requires someone to perform an action on it.

A resource has one of the following statuses:

Icon	Status	Description
No icon present	**Healthy**	There are no unresolved alerts for the resources.
	**Alerts Present**	There are one or more unresolved alerts for the resource.
	**Unavaila-ble**	The resource is unavailable. For example, the resource is down.
	**Monitoring Stopped**	The resource is not being monitored because of a blackout or because a user manually stopped monitoring the resource.

2: Alert List widget

The **Alert List** widget contains the alerts for the monitored Relay Server farms.

To view detailed information about an alert

- Select the alert in the **Alert List**, and then click **Details**.

  A window opens showing the details of the alert.

An alert has one of the following statuses:

Icon	Status	Description
	Active	Active alerts are alerts where the alert condition still applies. No one has resolved the alert.
	Inactive	An inactive alert indicates that the issue that triggered the alert is no longer present, but the alert has not been resolved or deleted.
	**Resolved**	An administrator or operator has marked the alert resolved.

An alert has one of the following levels of severity:

Icon	Severity	Description
	High severity	High severity alerts indicate problems that require a user's immediate attention. For example, when a resource exceeds the low disk space threshold, a high severity alert is issued.
	Medium severity	Medium severity alerts indicate problems that require a user's attention as the problems could escalate. For example, when a resource exceeds the CPU usage threshold, a medium severity alert is issued.
	Low severity	Low severity alerts indicate problems. For example, when a resource has a failed connection, a low severity alert is issued.

See "Alerts" on page 79.

Dashboards

Dashboards are specific to each user. Any user can add, edit, or delete their dashboards.

To add a new dashboard

1. Log in to the Monitor.
2. In the **Dashboards** pane, click **Add New**.
3. Follow the instructions in the window to add a dashboard.
4. Click **OK**.

   The Monitor creates and opens the new dashboard. The name of the dashboard also appears under the **Dashboards** list on the left navigation menu.

To edit or delete a dashboard

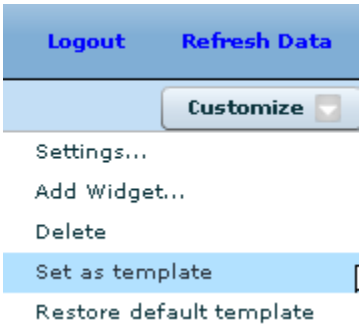
1. Open the dashboard.
2. In the upper right corner of the dashboard, click **Customize**, and then choose either:
   - **Settings** Edits the dashboard's settings.
   - **Delete** Deletes the dashboard.

Dashboard templates

By default, when a dashboard is created, it is populated with default widgets. You can change the default widgets that appear with a new dashboard by configuring the Monitor to use a specified dashboard's widgets and layout. See "Widgets" on page 62.

To create a dashboard template

1. Open the dashboard that you want to use as the template.
2. In the upper right corner of the dashboard, click **Customize**, and then click **Set As Template**.



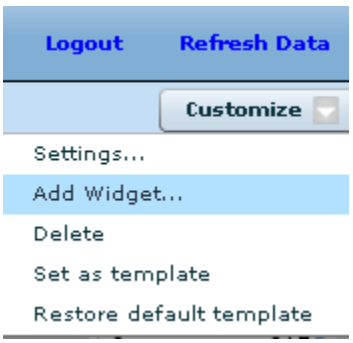
Now when you add a new dashboard, it contains the same widgets and layout as the template dashboard.

Widgets

Any user can add, edit, or delete the widgets that exist in their dashboards. By default, when a dashboard is created, it is populated with default widgets.

To add widgets

1. Open the dashboard.
2. In the upper right corner of the dashboard, click **Customize**, and then click **Add Widget**.



3. Specify a name for the widget in the **What Do You Want To Name This Widget** field.
4. Specify the resource in the **Which Resource Are You Interested In** field.
5. Choose one of the following display types for the **What Kind Of Display Do You Want** field.

- **Table display** The table display provides a general outline of the relative values. This display type provides sparklines—simple graphs that are good for showing trends and variations. The table display is the default display type.
- **Graph display** The graph display is more detailed and is useful for determining exact values at specific times. For example, you notice an unusual peak in a sparkline. To find out more information, such as when a peak occurred, add a widget that uses the graph display to display one or two metrics.

6. Follow the instructions in the **Add Widget** window to add a widget.

   For information about a specific metric, see "List of metrics" on page 73.

7. Click **Create**.

To edit or delete a widget

- In the widget pane, click the dropdown arrow in the upper right corner of the widget, and then choose either:
  - **Settings** Edits the widget's settings.
  - **Delete** Deletes the widget.

See also

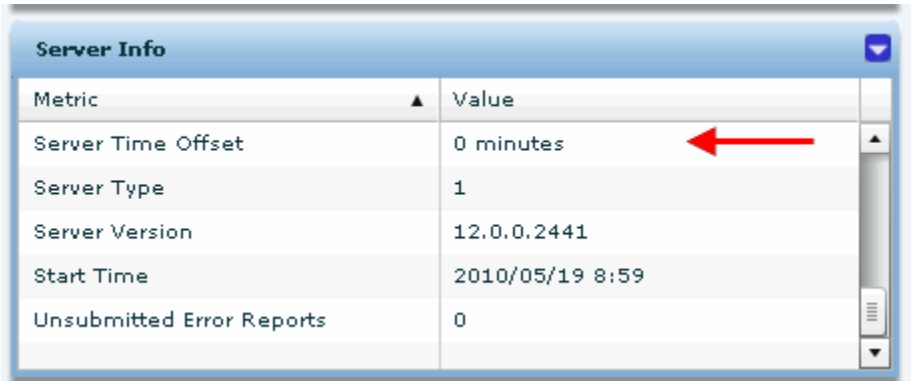
- "Dashboards" on page 61
- "Overview dashboard" on page 59

Understanding how time is displayed

All times displayed in the Monitor are based on the 24-hour clock and are local to the time on the computer that the Monitor is running on.

To find the time difference between the Monitor and your browser

1. Log in to the Monitor.
2. Open the dashboard for the resource.
3. In the **Server Info** widget, find the **Server Time Offset** metric.



The **Server Time Offset** records the time difference between the time on the computer that the Monitor is running and the time on the computer that you are using to view the Monitor data.

See also

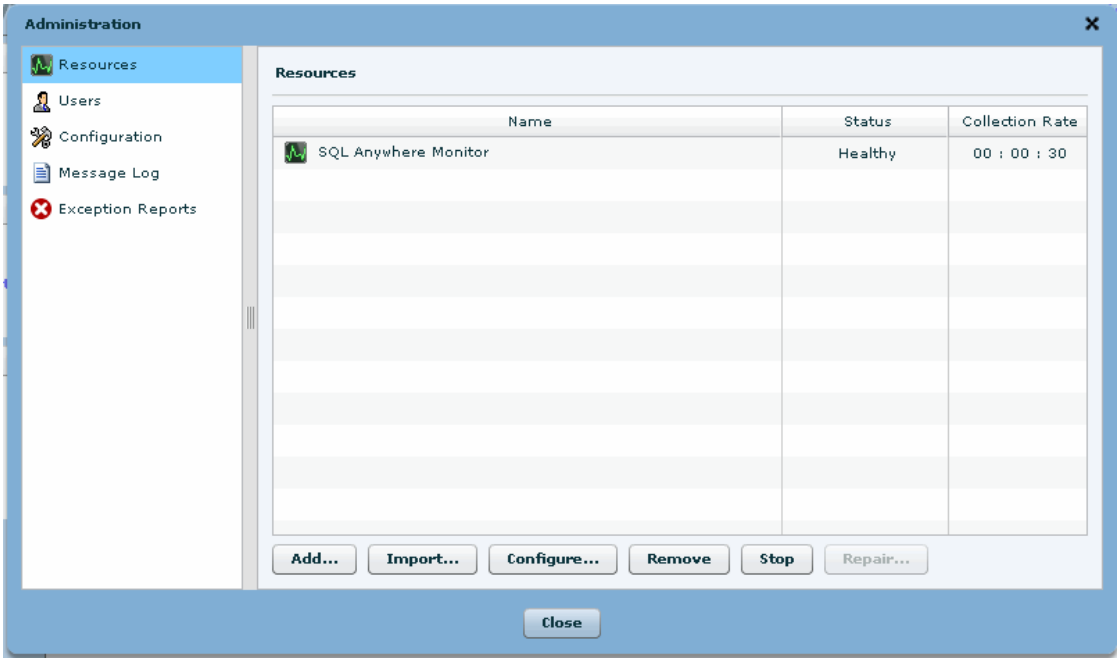
- "Metrics" on page 71

Administration window

Note
Only administrators can access the **Administration** window. For information about administrators, see "Monitor users" on page 75.

As an administrator, you can select the resources (for example, Relay Server farms) that you want to monitor, and:

- Add, edit, and delete resources. See "Resources" on page 66.
- Add and edit users. See "Monitor users" on page 75.
- Configure the Monitor. See "Back up the Monitor" on page 85.
- View the Message Log. See "Message Log" on page 65.
- View the Exception Reports. See "Exception Reports" on page 65.



See also

- "Metrics" on page 71

Message Log

The **Message Log** contains informational messages from and about the SQL Anywhere Monitor regarding its operation. Messages are displayed in a table with the most recent message at the top.

To view the message log

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Message Log**.

Exception Reports

When the SQL Anywhere Monitor experiences a fatal error or a crash occurs, an exceptions report is created about what was happening at the time of the problem.

To view the exception reports

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Exception Reports**.

Resources

A **resource** is a SQL Anywhere database, a MobiLink server, a MobiLink Server Farm, or a Relay Server farm. As an administrator, you add resources to the Monitor, and then you start monitoring them.

Monitoring of a resource starts:

- Automatically when the Monitor starts. When you start the Monitor, by default, it connects to the resources and collects metrics for *all* resources in the Monitor.
- Automatically when an administrator adds a resource. After a resource is added, the Monitor attempts to connect to the resource and starts monitoring it. See "Add resources" on page 66.
- Automatically at the end of a blackout period. The Monitor automatically attempts to connect to the resource and resume monitoring.
- When an administrator opens the **Administration** window, clicks **Resources**, selects a resource from the list, and clicks **Start**.

SQL Anywhere Monitor resource

The default resource, named **SQL Anywhere Monitor**, reports on the health of the Monitor itself. This default resource is useful for monitoring the computer that the Monitor is running on and sending alerts when the Monitor is experiencing problems. You cannot modify this resource, nor can you stop monitoring it.

See also

- "Resource List widget" on page 59

Add resources

To monitor a Relay Server farm, an administrator must first add the resource to the Monitor. By default, after the resource is added, monitoring starts.

As an administrator, you can add a resource one at a time or you can add multiple resources via an Import file. See "Add multiple resources" on page 67.

To add a Relay Server farm resource to monitor

1. Log in to the Monitor as an administrator.
2. Click **Administration**.
3. Click **Resources**.
4. Click **Add**.
5. Follow the instructions in the **Add Resource** window to add a resource to monitor a Relay Server farm. In the **Host** field, specify the hostname or IP address of the computer on which the Relay Server farm is running. You can use the name localhost to represent the current computer. This term is required. See "Host connection parameter" [*SQL Anywhere Server - Database Administration*].
6. Click **Create**.
7. Click **OK**.
8. Click **Close**.

   The resource appears in the **Overview** dashboard **Resource List**.
9. Optional: Add a dashboard for the resource. By default, when a resource is added, a dashboard is not.
   a. Open the **Overview** dashboard.

      Click **Dashboards** » **Overview**.
   b. In the **Resource List**, click the new resource.

      The Monitor creates and opens a new dashboard for the resource. The name of the new dashboard appears in the **Dashboards** section of the sidebar.

      See "Dashboards" on page 61.

See also

- "Add multiple resources" on page 67
- "Start the Monitor" on page 54

Add multiple resources

As an administrator, you can add multiple resources to the Monitor by creating a comma separated values (CSV) file, and then importing the file.

To add multiple resources to the Monitor

1. Create a CSV file.

   Each line in the CSV file contains information about a single resource. Each comma-separated term within a line describes an attribute of the resource. The order of the terms is important. The following table describes the terms and their order:

- **CSV file format for Relay Server farm resources**

2. Log in as an administrator to the Monitor.
3. Click **Administration**.
4. Click **Resources**.
5. Click **Import**.
6. Locate the import file and click **Open**.

   The resources from the import file are added to the Monitor and monitoring starts.
7. Click **Close**.
8. Click **Close**.
9. The imported resource is added to the **Resource List** in the **Overview** dashboard.

Stop monitoring resources

You stop monitoring resources when you do not want the Monitor to collect metrics from a Relay Server farm. For example, you want to stop monitoring when you know that the resource will be unavailable; otherwise, you receive alerts until the resource is available. Except for the default Monitor resource, you can stop monitoring any resource at any time.

When you stop monitoring a resource, the Monitor:

- Stops collecting metrics for the resource.
- Stops issuing alerts for the resource.

There are two ways to stop monitoring a resource:

- **Schedule a regular, repeating, blackout period** This method is a good choice when the following conditions apply:
  - You must repeatedly stop monitoring the Relay Server farm. For example, you perform regular maintenance at the end of each month.
  - You know in advance how long the Relay Server farm is unavailable. For example, you know that your regular maintenance takes four hours.
  - You need monitoring to automatically restart. When a blackout completes, the Monitor attempts to reconnect to the resource and to continue collecting data.

  To use this method, you create blackouts to make the Monitor stop monitoring at specified times. See "Automatically stop monitoring resources using blackouts" on page 69.

- **Manually stop the monitoring** This method is a good choice when the following conditions are met:

  - You need to stop monitoring for infrequent or one-time tasks. For example, you need to stop monitoring because the computer that the resource is running on needs to be taken off-line for special maintenance.

  - You are available to restart the monitoring afterward. When a resource has been stopped manually, the Monitor waits for you to restart the monitoring.

  To use this method, see "Manually stop monitoring resources" on page 69.

If you want to permanently stop monitoring a resource, you can remove it from the Monitor. See "Remove resources" on page 70.

Manually stop monitoring resources

The following procedure describes how to manually stop a resource. For information about what happens when you stop a resource, see "Stop monitoring resources" on page 68.

To manually stop a resource

1. Log in as an administrator to the Monitor.
2. Choose **Tools** » **Administration**.
3. Click **Resources**.
4. Select the resource, and then click **Stop**.
5. Click **Close**.

See also

- "Add resources" on page 66
- "Automatically stop monitoring resources using blackouts" on page 69

Automatically stop monitoring resources using blackouts

Blackouts are times when you do not want the Monitor to collect metrics. When a blackout completes, the Monitor attempts to reconnect to the resources and to continue collecting data.

The following procedure describes how to stop a resource using blackouts. For information about what happens when you stop a resource and about when you should use blackouts, see "Stop monitoring resources" on page 68.

Blackouts occur in the local time of the resource. See "Understanding how time is displayed" on page 63.

To configure the blackout time

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Resources**.
4. Select the resource, and then click **Configure**.
5. Click **Blackouts**.
6. Click **New**.
7. In the **New Blackout Period** window, specify the date and time (24 hour clock) for the blackout.

   The time is local to the computer where the resource Relay Server farm resides.
8. Click **Save**.
9. Click **Save**.
10. Click **OK**.
11. Click **Close**.

See also

- "Add resources" on page 66
- "Manually stop monitoring resources" on page 69

Remove resources

You should only remove resources when you are certain that you don't need to monitor them; for example, if the Relay Server farm is no longer being used.

Removing a resource causes the Monitor to:

- Permanently stop monitoring the resource.
- Discard the metrics collected for the resource.

Only administrators can remove resources. You cannot delete the **SQL Anywhere Monitor** resource.

To remove a resource

1. Log in to the Monitor as an administrator.
2. Click **Administration**.
3. Click **Resources**.

4. Select the resource.
5. Click **Remove**.
6. Click **Yes**.
7. Click **Close**.

See also

- "Stop monitoring resources" on page 68

Metrics

The Monitor collects and stores metrics from Relay Server farms, including, but not limited to:

- Whether the resource is running.
- Whether the computer that the resource is running on is running properly and is connected to the network.
- Whether the resource is listening and processing requests.
- The average time it takes for the back-end server to process an HTTP request.

Collection intervals

Metrics displayed in the Monitor are only as precise as the collection interval. As an administrator, you can change the rate at which metrics are collected by the Monitor. By default, metrics are collected every 30 seconds.

To set the collection interval for a resource

1. Log in to the Monitor as an administrator.
2. Click **Resources**.
3. Select the resource, and then click **Configure**.
4. Click **Collection Interval**.
5. Specify the interval at which metrics are collected (hours:minutes:seconds). The collection interval must be at least 10 seconds long.
6. Click **Save**.
7. Click **OK**.
8. Click **Close**.

See also

- "Refresh metrics" on page 72

Export metrics

You can export metrics that have a graph or table associated with them, to an XML file. For example, most of the metrics in the **Key Performance Metrics** widget can be exported.

To export metrics

1. Log in to the Monitor.
2. Open a dashboard.
3. On a widget that displays the metrics, click the dropdown menu arrow, and then click **Export**.
4. Follow the instructions in the **Export Metrics** window to export metrics.
5. Click **Export**.
6. When prompted, specify a file name.

   An XML file is created containing the specified metrics.

Refresh metrics

By default, the Monitor display is automatically refreshed every minute. You can change the refresh the interval by clicking **Tools** » **User Settings**. This setting is independent of the collection interval rate for a resource, which specifies how often the Monitor collects metrics from the resource being monitored. See "Collection intervals" on page 71.

To set the refresh rate

1. Choose **Tools** » **User Settings**.
2. Change the settings as required. The default is one minute.
3. Click **OK**.

When you click **Refresh Data** the Monitor retrieves and displays the latest metrics.

To refresh metrics

- Click **Refresh Data**.

When you press F5, the Monitor reloads the browser, retrieves the metrics that the Monitor has collected to date, and displays the metrics.

To reload the Monitor

- Press F5.

List of metrics

List of metrics for Relay Server Farm resources

Metric Name	Description
Active Requests	Shows total number of concurrent requests
Average Back End Server Processing Time	Measures the average time it takes for the backend server to process an HTTP request.
Average Request Completion Time	Measures the average time after a response is sent to the client that the Relay Server waits for a backend connection to close.
Average Request Time	Measures the average time the Relay Server waits to receive the entire HTTP request from the client.
Average Request-Response Cycle Time	Measures the average time the Relay Server takes to process a request and receive a response.
Average Response Time	Measures the average time it takes the Relay Server to receive the entire HTTP response from the backend server.
Channel Timeout	The maximum latency for the Relay Server to detect a blackout of a backend service.
CPU Usage	Measures the average percentage of CPU time used by the Relay Server.
Down Channel Bytes	Shows the total number of bytes coming down from the outbound enabler to the backend server extension during the last collection period.
Error Count	Shows the total number of errors that occurred during the last collection period. This count includes the error count of the children.

Metric Name	Description
Farm Status	Shows current status of the Relay Server Farm: • **0** The Relay Server Farm is enabled and it can provide service normally. • **1** The Relay Server Farm is disabled so that the outbound enabler cannot connect and provide service. The clients also cannot access this backend farm via Relay Server. • **2** The Relay Server Farm is enabled, but some of the outbound enabler(s) are not connected to all Relay Servers in the Relay Server Farm. Quality of service can be affected. • **3** The Relay Server Farm is enabled but none of the backend server is connected and so no client will be able to access this backend farm.
Free Disk Space For Log File	Shows the disk space available for the Relay Server Farm log file.
Host	Shows the name of the computer running the Relay Server Farm.
Operating System Version	Shows the operating system on which the Relay Server software is running, including build numbers and service packs.
Peak Active Requests	Shows the maximum number of concurrent requests since the Relay Server started.
Peak Shared Memory Usage	Shows the maximum usage of shared memory of the Relay Server since the Relay Server started.
Processor Architecture	Shows type of processor on which the Relay Server farm is running.
Relay Server Start Time	Shows the start up time for the Relay Server. For backend servers, shows the time the outbound enabler is connected to the backend server extension.
Relay Server Status	Shows the status of the Relay Server.
Relay Server Version	Shows the version of the Relay Server software being run.
Request Bytes	Shows the total number of bytes from HTTP requests coming up from the client to the client extension.
Requests Completed	Shows the total number of finished request-response cycles.

Metric Name	Description
Response Bytes	Shows the total number of bytes in HTTP responses coming down from the client extension to the client.
Sessions Created	Shows the total number of new HTTP sessions created.
Shared Memory Limit	Shows the total shared memory available to the Relay Server.
Unavailable Since	Shows the time since the resource became unavailable.
Up Channel Bytes	Shows the total number of bytes going up from the backend server extension to the outbound enabler.
Web Server Version	Shows the version of the Web Server software being run.

Monitor users

You must log in to the Monitor with a user name and password. The user name and password for logging in to the Monitor are case sensitive.

Default user

By default, when you first start the Monitor, it has one administrator user, named **admin**, with the password **admin**. By default, this user has full permissions. It is recommended that you change the default administrator password to restrict access to the Monitor. See "Edit Monitor users" on page 78.

The Monitor supports three types of users:

User type	Description
Read-only user	Has read-only access to monitor resources. Read-only users can view the metrics, but cannot access the **Administration** window. When you create a user from the log-in screen, this user is a Read-only user. A user name and password are required.
Operator	Has read-only access to monitor resources and can receive alerts. These users can view the metrics, can receive email alerts, and can resolve and delete alerts. However, operators cannot access the **Administration** window. A user name and password are required.
Administrator	Has the same access as an operator, and can also configure resources and add users. Administrators can also access the **Administration** window. The default user, **admin**, is an administrator. A user name and password are required.

All users, once they have logged in, can check their user type and change their settings. However, only an administrator can change a user's type. See "Edit Monitor users" on page 78.

To check your Monitor user type

1. Log in to the Monitor.
2. Choose **Tools** » **User Settings**.
3. Review the **User Type** setting in the window.

See also

- "Administration window" on page 64

Create Monitor users

By default, from the log-in screen, anyone can create their own read-only user and have read-only access to the Monitor. However, administrators can change this default so that only administrators can create users; see "Only allow administrators the ability to create users" on page 79.

To create a read-only user from the log-in screen

1. Click **Create New User**.

   Note
   If the **Create New User** link is not available, then the administrator has changed the default behavior so that only administrators can create new users. Contact your Monitor administrator to create a new user.

2. Fill in the information for the new user.

   An email address is only required if you want to receive email alerts from the Monitor. Contact your Monitor administrator to change your user type to operator or administrator, and sign you up to receive email alerts. See "Send alert emails" on page 83.

When logged in, administrators can create any type of user, including other administrators.

To create new users when logged-in as an Administrator

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Users**.
4. Click **New**.

5. Fill in the information for the new user. An email address is only required for users who should receive email alerts from the Monitor.
6. Select a language from the **Preferred Language Type** dropdown menu. The specified language sets the language used by the Monitor, including the language used in alerts.
7. Select a user type. Each type has different permissions. See "Monitor users" on page 75.
8. Click **Next** to create the resource dashboards for this user.
9. Click **Save**.
10. Click **Close**.
11. If you created an operator or an administrator user, this user can receive alert notifications by email. See "Send alert emails" on page 83.

See also

- "Edit Monitor users" on page 78
- "Only allow administrators the ability to create users" on page 79

Associate Monitor users with resources

You must associate an operator or administrator user with a resource if you want the user to receive email alerts about the associated resource.

To associate an operator or administrator with a resource

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Ensure that the operator or administrator has an email address specified in their user account.

   Click **Users** and verify that the user has an email address specified.
4. Click **Resources**, select a resource from the list, and then click **Configure**.
5. Click **Operators**.
6. Select a user from the **Available Operators** list and then click **Add**.

   The user name appears in the **Selected Operators** list.
7. Click **Save**.
8. Click **OK**.
9. Click **Close**.

10. Ensure that the Monitor is set up to send alert notifications by email. See "Enable the Monitor to send alert emails" on page 83.

See also

- "Send alert emails" on page 83

Edit Monitor users

All users, once they have logged in, can change their user settings by clicking **Tools** » **User Settings**. However, only an administrator can change a user's type.

To edit an existing Monitor user

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Users**.
4. Select the user to edit, and then click **Edit**.
5. Change the settings for the user as required. An email address is only required for users who should receive email alerts from the Monitor.
6. Click **Save**.
7. Click **Close**.
8. If you created an operator or an administrator user, this user can receive alert notifications by email. See "Send alert emails" on page 83.

See also

- "Create Monitor users" on page 76
- "Delete Monitor users" on page 78

Delete Monitor users

Deleting a user removes the user from the Monitor and disassociates the user from any resource.

To delete an existing Monitor user

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click the **Users**.
4. Select the user, and then click **Delete**.

5. Click **Yes** to delete the selected user.

   The user is deleted from the Monitor.

6. Click **Close**.

See also

- "Create Monitor users" on page 76
- "Edit Monitor users" on page 78

Only allow administrators the ability to create users

By default, from the log-in screen, anyone can create their own user name and password and have read-only access to the Monitor. However, as an administrator, you turn off this feature so that only administrators can create users.

To restrict user creation to administrators

1. Log in to the Monitor as an administrator.
2. Chose **Tools** » **Administration**.
3. Click **Configuration**.
4. Click **Edit**.
5. Click **Options**.
6. Clear the **Allow Anyone Read-only Access To The SQL Anywhere Monitor** option.
7. Click **Save**.
8. Click **Close**.

See also

- "Create Monitor users" on page 76
- "Edit Monitor users" on page 78

Alerts

An **alert** is a condition or state of interest about a resource that should be brought to an administrator's or operator's attention. Alerts are detected by the Monitor based on metrics that are collected. They are not detected at being monitored.

There are predefined alerts for conditions such as low disk space, failed login attempts, and high memory usage. You can change the default threshold values by editing the resource. See "Specify alert thresholds" on page 81.

When an alert condition is met, the alert is listed in the **Alert List** widget for the specified resource. See "Alerts List widget" on page 60.

By default, alerts appear in the **Alert List** widgets and they include information about the cause of the problem, and provide advice for resolving the problem. In the **Resource List** the resource's status changes to reflect the existence and severity of the alert. See "Overview dashboard" on page 59.

You can configure the Monitor to send an email to operators and administrators when an alert occurs. See "Send alert emails" on page 83.

View alerts

The Monitor keeps only the most recent 50 alerts in the alert list.

> **Note**
> Any logged-in user can view alerts; however, only operators and administrators can resolve and delete alerts.

To view an alert

1. Choose **Alerts** » **Today**.
2. Select an alert from the list, and then click **Details**.
3. Click **OK**.

See also

- "Resolve alerts" on page 80
- "Delete alerts" on page 81
- "Send alert emails" on page 83

Resolve alerts

As an operator or an administrator, you can mark an alert as resolved after the issue that triggered the alert has been addressed.

Resolving an alert causes the Monitor to change the alert **Status** to **Resolved by *user-name***, but leaves the alert in the alert list. If you want to remove the alert from the list, you must delete it. See "Delete alerts" on page 81.

To resolve an alert

1. Log in to the Monitor as an administrator or operator user.
2. Click **Overview**.
3. In the **Alerts List** widget, select an alert from the list and click **Mark Resolved** to resolve the selected alert.

The value in the **Status** column changes to **Resolved by *your-user-name***.

If this alert was the resource's only unresolved alert, the resource's status in the **Resource List** widget changes to Healthy (no icon is present).

See also

- "Overview dashboard" on page 59
- "Delete alerts" on page 81
- "Send alert emails" on page 83
- "View alerts" on page 80
- "Alerts" on page 79

Delete alerts

As an operator or an administrator, you can delete any alert from an **Alerts List**. You can delete alerts, regardless of their statuses.

To delete alerts

1. Log in to the Monitor as an administrator or operator user.
2. Click **Overview**.
3. In the **Alerts List** widget, select an alert from the list and click **Delete**.

   The alert is removed from the alerts list.

See also

- "Resolve alerts" on page 80
- "Send alert emails" on page 83
- "View alerts" on page 80
- "Alerts" on page 79

Specify alert thresholds

As an administrator you can configure when alerts should be issued. But, you cannot configure what metrics the Monitor collects nor can you configure the default resource, the SQL Anywhere Monitor.

To configure when alerts should be issued

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Select **Resources**, and then select a resource from the list.
4. Click **Configure**.

5. Click **Alert Thresholds**. Edit the thresholds. For definitions of the alerts, see "Alert thresholds" on page 82.
6. Configure the other settings as required.
7. Click **Save**.
8. Click **OK**.
9. Click **Close**.

See also

- "Metrics" on page 71
- "Alert thresholds" on page 82

Suppress alerts for unsubmitted error reports from resources

As an administrator, you can configure whether the Monitor sends out alerts when resources have unsubmitted error reports. By default, the Monitor does not send these alerts. For information about error reports and about how to submit them, see "Error reporting in SQL Anywhere" [*SQL Anywhere Server - Database Administration*].

To suppress alerts for unsubmitted error reports

1. Log in to the Monitor as an administrator.
2. Choose **Tools** » **Administration**.
3. Click **Configuration**.
4. Click **Edit**.
5. Click **Options**.
6. Select **Suppress unsubmitted error report alerts form resources**.
7. Click **Save**.
8. Click **Close**.

Alert thresholds

As an administrator, you can configure the thresholds that are used to trigger alerts. See "Specify alert thresholds" on page 81.

The following list describes the alerts and their default thresholds.

Alert thresholds for Relay Server farm resources

- **Alert When Relay Server CPU Usage Exceeds The Given Threshold For The Given Number Of Seconds** The **Threshold** default is 80 percent. The **Seconds** default is 30.
- **Alert When A Back-End Farm Has Only Partial Connectivity For Longer Than The Given Number of Seconds** The default is 30.
- **Alert When Relay Server Shared Memory Usage Exceeds The Given Threshold For The Given Number of Seconds** The **Threshold** default is 80. The **Seconds** default is 30.
- **Alert Whenever A Back-End Farm Is Unavailable For Longer Than 2 Minutes** This option is cleared by default.
- **Alert Whenever a Fatal Error Occurs Within the Relay Server Farm** This option is selected by default.
- **Suppress Alerts For The Same Condition That Occur Within Minutes** This option prevents you from receiving duplicate alerts within a specified time. The default is 30 minutes.

Send alert emails

As an administrator, you can configure the Monitor to send an email to specified operators and administrators when an alert occurs for specified resources.

To have the Monitor send alert notifications by email

1. Log in to the Monitor as an Administrator.
2. Create an administrator or operator with an email address. See "Create Monitor users" on page 76.
3. Associate the administrator or operator with a resource. See "Associate Monitor users with resources" on page 77.
4. Enable the Monitor to send emails. See "Enable the Monitor to send alert emails" on page 83.

Enable the Monitor to send alert emails

As an administrator, you can configure the Monitor to send emails to operators and administrators when alerts occur. The Monitor supports the SMTP and MAPI protocols for sending emails.

To enable the Monitor to send alert notifications by email

1. Log in to the Monitor as an Administrator.
2. Choose **Tools** » **Administration**.
3. Click **Configuration**.

4. Click **Edit**.
5. Click **Alert Notification**.
6. Select **Send Alert Notifications By Email**.
7. Choose either SMTP or MAPI for the **Which Protocol Do You Want To Use To Send Alerts By Email?** field.
8. Configure the other settings as required.
   - **MAPI**
     - **User Name** Type the user name for the MAPI server.
     - **Password** Type the password for the MAPI server.
   - **SMTP**
     - **Server** Specify which SMTP server to use. Type the server name or the IP address for the SMTP server. For example, *SMTP.yourcompany.com*.
     - **Port** Specify the port number to connect to on the SMTP server. The default is 25.
     - **Sender Name** Specify an alias for the sender's email address. For example, *JoeSmith*.
     - **Sender Address** Specify the email address of the sender. For example, *jsmith@emailaddress.com*.
     - **This SMTP Server Requires Authentication** Select this option if your SMTP server requires authentication.
       - **User Name** Specify the user name to provide to SMTP servers requiring authentication.
       - **Password** Specify the password to provide to SMTP servers requiring authentication.
9. Click **Send Test Email**.
10. When prompted, enter an email address to send the test email to and click **OK**.

    A test email is sent to the email address specified.
11. Click **Save**.
12. Click **Close**.

    When an alert occurs, the Monitor sends alert emails to operators and administrators who have specified email addresses in their user accounts. These users receive emails for the resources that they are associated with. See "Create Monitor users" on page 76 and "Associate Monitor users with resources" on page 77.

See also

- "Resolve alerts" on page 80
- "Delete alerts" on page 81
- "View alerts" on page 80

Back up the Monitor

By default, the Monitor performs maintenance on the metrics once a day at midnight. Maintenance affects metrics, not alerts.

As an administrator, you can:

- Schedule the Monitor to back up metrics.
- Control the amount of disk space that the Monitor uses.
- Perform maintenance on demand.

To back up metrics and control the amount of disk space used to store metrics

1. Log in to the Monitor as an Administrator.
2. Click **Administration**.
3. Select **Configuration**, and then click **Edit**.
4. Click **Maintenance**.
5. Specify a time (24-hour clock) when the Monitor should perform maintenance. By default, it performs maintenance at midnight. The time is local to the computer where the Monitor is running. See "Understanding how time is displayed" [*SQL Anywhere Server - Database Administration*].
6. Specify a directory where the Monitor should save the backed up data. The directory must exist on the computer where the Monitor is running.
7. Customize the **Data Reduction** settings:
   - **Reduce Metrics To A Representative Daily Value For Metrics Older Than** When you select this option, an average is taken for all numeric metrics that are older than the specified number of days, and then the numeric metrics are deleted. Non-numeric metrics are not deleted.
   - **Delete Values Older Than** When you select this option, all metrics that are older than the specified length of time are deleted.
   - **Delete Old Metrics When The Total Disk Space Used By The SQL Anywhere Monitor Becomes Greater Than (MB)** When you select this option, you specify the maximum amount of space that can be used to store the metrics. When the amount of disk space used reaches or exceeds the amount specified, the Monitor deletes metrics, starting with the oldest metrics. Metrics are deleted until enough free space exists to store new metrics.
8. Click **Perform Maintenance Now** to run the backup immediately. When prompted, click **OK**.
9. Click **Save**.
10. Click **Close**.

The following procedure describes how to replace the Monitor database with a backup copy.

To restore a backup copy of the Monitor database

1. Stop the Monitor, if it is running.
2. From your backup directory, copy the *samonitor.db* database file and *samonitor.log* log file.
3. Paste these files into the directory where the current Monitor database and log file are located. When prompted, overwrite the existing files.

   The default locations of the version 12.0.0 Monitor database files are listed in the following table:

Operating system	Monitor directory
Windows XP (installed with SQL Anywhere)	`C:\Documents and Settings\All Users\Documents\SQL Anywhere 12\Monitor\samonitor.db`
Windows XP (installed on a separate computer)	`C:\Documents and Settings\All Users\Documents\SQL Anywhere 12 Monitor\samonitor.db`
Windows Vista (installed with SQL Anywhere)	`C:\Users\Public\Documents\SQL Anywhere 12\Monitor \samonitor.db`
Windows Vista (installed on a separate computer)	`C:\Users\Public\Documents\SQL Anywhere 12 Monitor \samonitor.db`
Linux (installed with SQL Anywhere)	`/opt/sqlanywhere12/samonitor.db`
Linux (installed on a separate computer)	`/opt/sqlanywhere12/samonitor.db`

4. Re-start the Monitor.

Installing the SQL Anywhere Monitor in a production environment

It is recommended, particularly in production environments, that you:

1. Install and run the Monitor Production Edition.

   Advantages to running the Monitor Production Edition include:

   - The Monitor runs in the background as a service.
   - The Monitor starts automatically when the computer starts.

- Upgrades and updates of SQL Anywhere do not overwrite or affect the Monitor Production Edition. In contrast, upgrades and updates of SQL Anywhere can affect the Monitor Developer Edition as it uses the installed SQL Anywhere on the back-end.

2. Install the Monitor on a computer that is different from the computer where the resources are running.

   Advantages to running the Monitor on a separate computer include:

   - The impact on the Relay Server farm or other applications is minimized.
   - Monitoring is not affected if something happens to the computer where the resources are installed.

These instructions explain how to install the SQL Anywhere Monitor Production Edition.

To install the Monitor Production Edition (Windows)

1. Run the *setup.exe* file from the *Monitor* directory on your installation media, and follow the instructions provided.

   On Windows, the Monitor service is started by the installation..

2. Open the default URL for logging in to the Monitor: *http://localhost:4950*.

   Note
   If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

3. Log in.

   When prompted, enter your user name and password for the Monitor. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

To install the Monitor Production Edition (Linux)

1. As the root user, run the *setup.tar* file from the *Monitor* directory on your installation media, and follow the instructions provided.

   Note
   On Linux, the Monitor Production Edition can only be run by the root user.

2. On Linux, by default, the Monitor Production Edition automatically starts the Monitor service.

3. Open the default URL for logging in to the Monitor: *http://localhost:4950*.

Note
If you are accessing the Monitor over a network, browse to **http://***computer-name***:4950**, where *computer-name* is the name of the computer the Monitor is running. See "Log in remotely to the Monitor" on page 58.

4. Log in.

   When prompted, enter your user name and password for the Monitor. The default user is an administrator with the name **admin** and the password **admin.** See "Monitor users" on page 75.

Upgrading the Monitor and migrating resources and metrics

Caution
Uninstalling the Monitor removes the application, as well as the resources and collected metrics.

If you want to preserve your current Monitor resources and metrics, you must:

1. Install a new version of the Monitor.
2. Migrate the resources and metrics.
3. Uninstall the older version of the Monitor.

See "Upgrading the SQL Anywhere Monitor and migrating resources and metrics" [*SQL Anywhere 12 - Changes and Upgrading*].

Securing the Monitor

You can secure communications between both the Monitor and your browser, and between the Monitor and resources it monitors.

Securing communications between the Monitor and your browser using transport-layer security (TLS)

You can use transport-layer security (TLS) to secure communication between the Monitor and your browser. The Monitor runs a web server that supports HTTPS connections using SSL version 3.0 and TLS version 1.0.

To set up TLS security for the Monitor

1. Obtain digital certificates from a certificate authority or create self-signed certificates with the Certificate Creation utility (createcert). See "Certificate Creation utility (createcert)" [*SQL Anywhere Server - Database Administration*].
2. Alter the Monitor start line string to use the certificates.

3. Configure your browser to accept your new certificates, if required.

For more information, see http://www.sybase.com/detail?id=1063938.

Securing connections between the Monitor and your resources

You can use ECC or FIPS to encrypt communications between the Monitor and the resources it monitors. ECC encryption and FIPS-certified encryption require a separate license.

Separately licensed component required
ECC encryption and FIPS-certified encryption require a separate license. All strong encryption technologies are subject to export regulations.

See "Separately licensed components" [*SQL Anywhere 12 - Introduction*].

Troubleshooting the Monitor

In addition to the recommendations listed below, Administrators can use the Message Log and Exception Reports features to troubleshoot the Monitor. See "Message Log" on page 65 and "Exception Reports" on page 65.

Problem	Recommendation
When you press F5 to refresh the browser window, you are required to log in to the Monitor.	Enable JavaScript in your browser.
You receive a network communication error when you try to log in to the Monitor.	Start the Monitor. See "Start the Monitor" on page 54.
After upgrading to the latest version of Adobe Flash Player you continue to receive instructions to upgrade Adobe Flash Player.	Verify that the installed version Adobe Flash Player is supported by your operating system. The Monitor is backwards compatible with version 10 of Adobe Flash Player. To determine the correct version, see http://www.adobe.com/products/flashplayer/systemreqs/.
The Monitor is unable to start monitoring a SQL Anywhere database resource.	Verify that the resource's password verification functions and login procedures allow the user sa_monitor_user to connect to the resource.

Problem	Recommendation
You are not receiving any alert emails.	Verify that the Monitor is properly configured to send emails and send a test email. See "Enable the Monitor to send alert emails" on page 83. Verify that the alert emails from the Monitor are not being blocked by a virus scanner. See "xp_startsmtp system procedure" [*SQL Anywhere Server - SQL Reference*].
The number of unscheduled requests reported by the Monitor appears to be less than the actual number of unscheduled requests.	When collecting metrics about the number of unscheduled requests, the Monitor executes query on the resource. This query could be an unscheduled request. Unscheduled queries are processed sequentially as they arrive. Therefore, if there are unscheduled requests when the Monitor attempts to execute its query, then this query must wait for the existing unscheduled requests to complete before it can execute. As a result, when the Monitor collects the number of unscheduled requests, this number does not include the unscheduled requests that existed between the time when the Monitor issued its query and the query executed.
You are not receiving alerts when the database disk space surpasses the specified threshold.	Between Monitor collection intervals, it is possible for a database to exceed the specified disk space alert threshold and the amount of space available. In such a case, the database would stop responding before the Monitor could collect the disk usage metrics and issue an alert. If your database grows quickly, set the disk space alert threshold to a higher number so that you can receive an alert before the database runs out of space. See "Alert thresholds" on page 82.
You can't see the **Administration** window when you are logged into the Monitor.	You must be logged in to the Monitor as an administrator to have access to the **Administration** window. See "Monitor users" on page 75.

Problem	Recommendation
You uninstalled the Monitor before migrating your data and now you have lost your resources and metric data.	Uninstalling the Monitor removes the application, as well as the resources and collected metrics. When upgrading, you need to install the new version of the Monitor, migrate your data, and then uninstall the old version. However, if you regularly backed up your Monitor, you can use the Migrate utility to migrate the data from the backed up files to the new version of the Monitor. See "Back up the Monitor" on page 85.
When monitoring a database that is part of database mirroring system, you receive errors about not being able to modify a read-only database.	In the **Administration** window, select the mirrored database resource and click **Edit**. In the **Other** field, type **NODE=PRIMARY**. See "NodeType (NODE) connection parameter" [*SQL Anywhere Server - Database Administration*].

Sybase Hosted Relay Service

The Sybase Hosted Relay Service is a farm of Relay Servers hosted by Sybase. It is intended to ease the development of mobile applications that use MobiLink data synchronization and to simplify the evaluation process for developers, especially where data is sent using public wireless networks. You do not need to ask your IT department to install anything or open any holes in your corporate firewall. All communication between MobiLink and the hosting service uses HTTP(S) via an outbound connection initiated by MobiLink.

The Sybase Hosted Relay Service is not intended for production deployments. Before deploying your production application, you must first install the Relay Server in your own corporate infrastructure.

Using the Sybase Hosted Relay Service

The following sections describe how to perform some basic tasks.

Subscribing to the Sybase Hosted Relay Service

To use the Sybase Hosted Relay Service you must first subscribe to it.

To subscribe to the Sybase Hosted Relay Service

1. Go to http://relayserver.sybase.com/account. This takes you to the Sybase Hosted Relay Service home page.

2. Create an account by clicking **Register**.

3. You are asked to specify a **Subscription ID** (choose one that is unique to your organization) and **Password**, provide contact information for your self and your organization, and agree to the **Hosted Relay Service Terms of Service**. Click **Submit**.

   Once you have successfully registered, an email is sent to you confirming your registration.

Logging in to the Sybase Hosted Relay Service

To log in to the Sybase Hosted Relay Service

1. Log in to your newly created account by clicking **Log In**.

2. Enter the **Subscription ID** and **Password** you entered during the registration process. Once logged in, you are taken to the **Account Information** page. The account information page allows you to modify subscriber information and specify the back-end server farm(s) that will be accessing this service.

Adding a server farm

To add a server farm

1. Click to select the type of farm you want to add. Choose from **Add New MobiLink Farm**, **Add New Afaria Farm**, and **Add New iAnywhere Mobile Office Farm**.
2. Enter a unique **Farm Name** to describe the server farm.
3. Provide a unique name for each server in the farm. You can specify a maximum of two servers.
4. Click **Create Farm**. A confirmation is displayed if the farm was successfully added.
5. Click **Configuration Instructions** to learn more about using the service. The instructions are based on the information you provided.
6. Click **Log Out** when you are done.

Using MobiLink with the Relay Server

The following sections provides information about using the Relay Server with MobiLink.

For information about which operating systems and browsers are supported for the Relay Server, see http://www.sybase.com/detail?id=1002288.

For information about deploying the Outbound Enabler, see "Deploying the MobiLink server" [*MobiLink - Server Administration*].

Connecting a client to the Relay Server farm

Once a Relay Server farm has been properly configured, a client connects to the Relay Server farm using the following URL:

http://<*Relay Server client extension URL*>/<*farmname*>

Options

Option	Description
<*Relay Server client extension URL*>	For Microsoft IIS on Windows, <*domain name*><*relayserver.sybase.com*>/*ias_relay_server/client/rs_client.dll* For Apache on Linux, <*domain name*><*relayserver.sybase.com*>/*cli/iarelayserver* Use *relayserver.sybase.com* as the <*domain name*> if you are using the publicly available Sybase Hosted Relay Service. For information about subscribing to the service as well as instructions for setting up your backend servers, see "Using the Sybase Hosted Relay Service" on page 93.
<*farmname*>	Identifies the back-end farm (a group of back-end servers) that Relay Server forwards the client request to.

SQL Anywhere MobiLink client connection example

A SQL Anywhere MobiLink client should specify the following options to connect to server farm **F1**:

```
-e "ctp=http;
     adr='host=relayserver.sybase.com;
        url_suffix=url_suffix=/rs/client/rs_client.dll/F1'"
```

For HTTPS, change `http` to `https`.

UltraLite/UltraLiteJ MobiLink client connection example

An UltraLite/UltraLiteJ MobiLink client should set the following properties in the ULSyncParms class to connect to server farm **F1**:

- Set the stream type to HTTPS.
- Set the stream parameters to the following:

```
"host=relayserver.sybase.com;url_suffix=/rs/client/rs_client.dll/F1"
```

QAnywhere client connection example

A QAnywhere client should specify the following options to connect to server farm **F1**:

```
-x "http(host=relayserver.sybase.com;url_suffix=/rs/client/rs_client.dll/F1"
```

Sample scenario

Suppose company ABC has developed a mobile application and now wants to set up the deployment run-time to service the mobile application. Initially, the mobile deployment consists of 10000 devices and grows in the future. The customer therefore wants a fault tolerant and load-balanced environment that is able to handle the load today and be easily extended to handle more mobile deployments in the future. Based on the data synchronization characteristics of the mobile application, the customer has determined that the following configuration is needed:

- 2 MobiLink servers
- 2 Relay Servers
- 1 load balancer

Since the company uses Microsoft IIS as its web server, the Microsoft IIS version of the Relay Server is used.

Notes

- Each Relay Server is deployed on its own computer. Two computers, with host names **rs1.abc.com** and **rs2.abc.com** are used.
- Each MobiLink server is deployed on its own computer. The two MobiLink servers are assigned names **ml1** and **ml2** and belong to the back-end server farm called abc.mobilink.
- The load balancer is addressable using the host name **www.abc.com**.
- For maximum security, HTTPS is used by all clients and Outbound Enablers connecting to the Relay Servers. It is assumed that all web servers are equipped with a certificate from a well known Certificate Authority (CA), and the back-end server computers all have the corresponding trusted root certificates in their standard certificate store.

To set up the Relay Server farm

1. The first step is to create the Relay Server configuration file.

   The filename containing the configuration must be called ***rs.config*** . For this particular scenario, the following configuration file is used:

```
#
# Options
#
[options]
verbosity = 1

#
# Define the Relay Server farm
#
[relay_server]
host = rs1.abc.com

[relay_server]
host = rs2.abc.com

#
# Define the MobiLink backend server farm
#
[backend_farm]
id = abc.mobilink
client_security = on
backend_security = on

#
# List MobiLink servers that are connecting to the Relay Server farm
#
[backend_server]
farm = abc.mobilink
id = ml1
token = mltoken1

[backend_server]
farm = abc.mobilink
id = ml2
token=mltoken2
```

2. Deploy the configuration file *rs.config* along with the Relay Server components to the two computers that are running the Relay Server.

3. Start MobiLink server on the two computers that are running the MobiLink servers using the Integrated Outbound Enabler.

   On the computer running MobiLink server with id ml1:

```
mlsrv12 -x oe<config=oe1.txt> -zs ml1 <other ML options>
```

   where oe1.txt = `-f abc.mobilink -id ml1 -t mltoken1 -cr "host=www.abc.com;port=443;https=1"`

   On the computer running MobiLink server with id ml2:

```
mlsrv12 -x oe<config=oe2.txt> -zs ml2 <other ML options>
```

where oe2.txt = `-f abc.mobilink -id ml2 -t mltoken2 -cr "host=www.abc.com;port=443;https=1"`

See "-x mlsrv12 option" [*MobiLink - Server Administration*].

4. Once all servers and Outbound Enablers are running, MobiLink clients are able to connect to the farm using the following connection information:
   - **HTTPS** protocol
   - **host** www.abc.com
   - **url_suffix** /rs/client/rs_client.dll/abc.mobilink

Index

G

H

I

L

M

N

O

P

R

S

T

U

W