

1. SQL Monitor 2 documentation	3
1.1 About SQL Monitor	4
1.2 Requirements	5
1.2.1 Supported platforms and hardware guidelines	7
1.3 Installing	10
1.3.1 Installing SQL Monitor components	12
1.3.2 Using SQL Monitor with IIS	15
1.3.3 Adding servers to monitor	17
1.3.4 Troubleshooting installing	19
1.3.4.1 Error messages - connecting to the Data Repository	20
1.3.4.2 Error messages - creating a Data Repository database account	21
1.4 Licensing	22
1.4.1 Activating and deactivating	23
1.4.2 Troubleshooting licensing	26
1.5 Upgrading	27
1.5.1 How to upgrade	28
1.6 Using SQL Monitor for the first time	30
1.7 Working with overviews	33
1.7.1 Global Overview	34
1.7.2 Cluster overview	35
1.7.3 Host machine overview	37
1.7.4 SQL Server instance overview	39
1.7.5 Database overview	42
1.7.6 Rewinding time with Back in Time mode	43
1.8 Working with alerts	45
1.8.1 List of alerts	46
1.8.2 Using the Alert Inbox	57
1.8.3 Viewing details of an alert	66
1.8.4 Filtering alerts	72
1.8.5 Configuring alerts	77
1.9 Working with analysis data	83
1.10 Configuring SQL Monitor	86
1.10.1 Configuring monitored servers	87
1.10.1.1 Monitoring status explained	91
1.10.1.2 Reconnecting automatically after monitoring stopped	94
1.10.2 Setting up email notification	95
1.10.3 Configuring Profiler trace	98
1.10.4 Organizing monitored servers into groups	100
1.10.5 Purging SQL Monitor data	103
1.10.6 Changing the Base Monitor connection	105
1.10.7 Setting maintenance windows	108
1.11 Managing security and user roles	112
1.11.1 Resetting your SQL Monitor password	113
1.12 Worked examples	114
1.12.1 Example configuration using virtualization software	115
1.13 Troubleshooting	117
1.13.1 Error messages	118
1.13.1.1 SQL Monitor browser error: Cannot display webpage; connection timed out; webpage not available; could not locate remote server	119
1.13.1.2 SQL Monitor connection error: Cannot generate SSPI context	120
1.13.2 Issues with IIS, WMI and ASP.NET	121
1.13.2.1 Issues with IIS and ASP.NET	122
1.13.2.2 Moving from SQL Monitor Web server to IIS	124
1.13.2.3 WMI leaks memory on Windows 7 and Windows Server 2008 R2	125
1.13.3 Unexpected behavior	126
1.13.3.1 Global Overview doesn't display any details in your browser	127
1.13.3.2 Issues caused by clock skew	128
1.13.3.3 tempdb size increases when clearing purge backlog	129
1.13.4 Technical questions	130
1.13.4.1 Accessing SQL Monitor through a firewall	131
1.13.4.2 How do I test data collection methods?	132
1.13.4.3 Moving the Data Repository database	138
1.13.4.4 Viewing component information and log files	139
1.14 Release notes and other versions	141
1.14.1 SQL Monitor 2.3 release notes	142
1.14.2 SQL Monitor 2.2 release notes	143
1.14.3 SQL Monitor 2.1 release notes	144
1.14.4 SQL Monitor 2.0 release notes	145
1.14.5 SQL Response 1.3 release notes	146
1.14.6 SQL Response 1.2 release notes	147
1.14.7 SQL Response 1.1 release notes	148
1.14.8 SQL Response 1.0 release notes	149

1.15 Learn more about database administration	150
2. Navigating in SQL Monitor	151

SQL Monitor 2 documentation

SQL Monitor 4.1 and earlier versions contain a security vulnerability. This is fixed in SQL Monitor 3.10 and SQL Monitor 4.2.

For more information, see [this page](#).

About SQL Monitor

SQL Monitor is a web-based monitoring and alerting tool for SQL Servers.

It displays real-time data about the current performance of all host machines, SQL Server instances, and databases that you choose to monitor, and raises alerts when problems occur.

For more information, see the [SQL Monitor product page](#).

Quick links

[Release notes for SQL Monitor 2.3](#)

[Using SQL Monitor for the first time](#)

[Learn more about database administration](#)

[Troubleshooting](#)

About SQL Monitor

SQL Monitor is a web-based monitoring and alerting tool for SQL Servers. It displays real-time data about the current performance of all host machines, SQL Server instances, and databases that you choose to monitor, and raises alerts when problems occur. Use SQL Monitor to:

- [Identify servers that require attention](#) using the Global Overview
- [Drill down](#) to view the current health and performance of any server or database
- [View detailed alerts](#) for a wide range of issues such as job failures, deadlocks, overdue backups and fragmented indexes
- [Configure each type of alert](#) for each server, job, database etc., to suit your environment; and set up email notifications
- [Analyze performance metrics](#) over time to determine performance trends and identify abnormal performance
- [Create custom metrics and alerts](#) specific to your environment
- ["Rewind time"](#) to review the activity on your servers when a problem occurred in the past
- [Run your own reports](#) against the Data Repository SQL Server database that contains all collected data

Quick start guide

To start using SQL Monitor, you need to:

1. Check prerequisites and access permissions. See [Requirements](#).
2. Install SQL Monitor components - Base Monitor, Web Server and Data Repository. See [Installing](#).
3. Create an Administrator password. See [Creating the Administrator role](#).
4. Choose the servers you want to monitor. See [Adding servers to monitor](#).
5. Activate server licenses. See [Licensing](#).
6. Configure email settings for alerts. See [Setting up email notification](#).

Worked examples

To learn about using SQL Monitor, see:

- [Example configuration using virtualization software](#)

Requirements

.NET runtime

The Web Server computer and Base Monitor computer both require **.NET 3.5 SP1**.

If you don't have .NET 3.5 SP1 installed, [download it here](#).

Account permissions required

The following are the *minimum permissions* required to run SQL Monitor and monitor your servers.

To test these permissions, see [How do I test data collection methods?](#)

If you want to access SQL Monitor through a firewall, additional permissions are required. See [Accessing SQL Monitor through a firewall](#)

SQL Monitor Web service account

- The account should have **Log on as service** rights.
- The account should have **Full Control** over the folder `C:\Documents and Settings\All Users\Application Data\Red Gate\SQL Monitor 3`. For Vista and Windows 7: `C:\ProgramData\Red Gate\SQL Monitor 3`.
- The account should have **Full Control** over the folder `C:\Documents and Settings\All Users\Application Data\Red Gate\Logs\SQL Monitor 3` or equivalent location.

The SQL Monitor Web Service is not installed if you use IIS as your Web Server.

SQL Monitor Base Monitor service account

- The account should have **Log on as service** rights.
- The account should have **Full Control** over the folder `C:\Documents and Settings\All Users\Application Data\Red Gate\Logs\SQL Monitor 3`. For Vista and Windows 7: `C:\ProgramData\Red Gate\Logs\SQL Monitor 3`.
- The login should be a member of the **db_owner** database role on the Data Repository database (called RedGateMonitor by default).

Monitoring host Windows machines

The account should be an administrator on the machine.

Monitoring SQL Server instances

The account used to monitor your SQL Server instance should have the following permissions:

For SQL Server 2005 and SQL Server 2008:

- member of the **db_datareader** role on the msdb system database.
- member of **SQLAgentReader** role on the msdb system database.
- member of the **db_ddladmin** database role on all databases (needed to run **DBCC SHOWCONTIG** required by the Fragmented index alert).
- **VIEW ANY DEFINITION** server permission.
- **ALTER TRACE** server permission (if you want to enable trace data).
- **VIEW SERVER STATE** and **VIEW DATABASE STATE** database permissions on all databases.
- **sysadmin** role required for Integrity check overdue alerts and to allow SQL Monitor to turn on the deadlock trace flag (this flag is required for Deadlock alerts to be raised; you can turn on the flag manually if you don't want to enable sysadmin permissions).

For SQL Server 2000:

If you want SQL Monitor to be able to collect trace data (trace data can optionally be displayed as part of some alerts), then the account must be a

member of the **sysadmin** server role.

If you do not want SQL Monitor to collect trace data, then the account should have the following permissions:

- member of the **db_datareader** database role on the msdb system database.
- member of the **db_datareader** database role on the master database.
- member of the **db_ddladmin** database role on all databases (needed to run **DBCC SHOWCONTIG** required by the Fragmented index alert).

The sysadmin fixed role is a superset of these permissions, and can also be used, but is not explicitly required except for trace collecting.

Supported platforms and hardware guidelines

Supported platforms

Host Windows machines

SQL Monitor monitors both the host Windows machine as well as SQL Server instances. You can monitor the following host Windows machines:

- Windows 2000
- Windows XP
- Windows Server 2003
- Windows Server 2008
- Windows Vista
- Windows 7

SQL Server versions

You can monitor the following versions of SQL Server:

- SQL Server 2000 SP4
- SQL Server 2005
- SQL Server 2008
- SQL Server 2012 (tested against CTPs available at the time of release)

SQL Monitor Web Server

Supported operating systems for the Web Server computer if you don't have an IIS server (SQL Monitor will install its own standalone Web Server):

- Windows XP
- Windows Server 2003
- Windows Server 2008
- Windows Vista
- Windows 7

If you want to add SQL Monitor to an existing IIS server, the following versions are supported:

- IIS 6
- IIS 7 or later

Supported browsers for the web application

- Firefox 3 or later
- Chrome 2 or later
- Safari 5 or later
- Opera 10 or later
- IE 7 or later

If you experience slow performance using IE, try switching to an alternative browser.

SQL Monitor Base Monitor service

Supported operating systems for the Base Monitor computer:

- Windows XP (32 bit) SP3 or later
- Windows XP (64 bit) SP2 or later
- Windows Server 2003 SP2 or later
- Windows Server 2008 or later
- Windows Vista
- Windows 7

32-bit SQL Server instances running on 64-bit Windows machines are not supported by SQL Monitor, and some performance counter objects are not available. For more information, see this MSDN article on [64-bit Support](#).

SQL Monitor Data Repository

The SQL Monitor Data Repository requires an MS SQL Server 2005 or 2008 database to store the data it collects from your monitored servers.

Support for clusters

SQL Monitor supports Microsoft cluster servers. Other proprietary clustering server systems may not behave as expected and are not supported.

Hardware requirements and performance guidelines

The information in this section is based on in-house testing at Red Gate, and is intended to be indicative only. In your environment, the performance may vary.

Specifications for Base Monitor machine

The Base Monitor machine runs the service that monitors the machines and instances you monitor in SQL Monitor.

- **Processor:** 3 GHz dual core should be sufficient to monitor 10 servers (a server is defined as a host machine plus single SQL Server instance).
- **Physical memory:** 2 GB RAM should be sufficient to run SQL Monitor as well as Windows OS and other small applications. The SQL Monitor process itself should use approximately 300 to 400 MB, depending on the number of servers being monitored.

Monitoring more servers will require a more powerful machine; in our testing, a 64-bit dual quad-core processor machine with 8 GB of RAM could monitor 30+ machines quite comfortably.

Growth in size of Data Repository database

The Data Repository is the SQL Server database that stores all data collected by SQL Monitor for all monitored machines and instances.

- Expect the database to use up to 150 MB per server (host machine plus single SQL Server instance) per day.
- 10 servers over 7 days will therefore increase the size of your database by approximately 10 GB.

If your monitored servers host a large number of objects and databases, the Data Repository will use significantly more storage space, potentially up to a maximum of 450 MB per day.

Creating the Data Repository database

Before installation, we recommend creating the Data Repository database manually using a SQL Server management tool with settings appropriate to your environment. This allows you to estimate the eventual size of your Data Repository based on the guidelines above, and set fixed autogrowth and transaction log size relative to your database size. It should help prevent autogrowths, which can negatively affect performance and contribute to physical file fragmentation. For more information, see:

[Considerations for the autogrow and autoshrink settings in SQL Server](#)

[Recover from a full transaction log in a SQL Server database](#)

You can allow SQL Monitor to create the Data Repository for you using the model database on your SQL Server system as a template. If the model database defaults are insufficient for the estimated eventual size of your database, consider creating the database manually instead. For more information about the model database defaults, see [model Database](#).

The database is created by SQL Monitor using the SIMPLE recovery model. If you change to FULL recovery model, the database growth will be much greater.

Network bandwidth impact on monitored servers

Total network traffic (inbound plus outbound) : 10 KB/sec per server.

A server is defined as a host machine plus a single SQL Server instance. For multiple instances, the impact will be slightly greater per additional instance. For servers with a large number of objects (for example, 200 databases and 20,000 tables), the total network traffic may be up to 20

KB/sec.

Running SQL Monitor on a virtual machine

You can run SQL Monitor without problems on a virtual machine, but you should ensure that your VM host can deliver the required resources (CPU and RAM) described above. If your physical machine hosts multiple VMs, for example, then this will limit the resources available to SQL Monitor, which may cause slow performance.

If you locate the Data Repository database on a SQL Server instance running on a VM, then we recommend that you use a physical disk or partition rather than a virtual disk for the database data files. For more information about using mapped hardware in Hyper-V, see [http://msdn.microsoft.com/en-us/library/cc768529\(v=bts.10\).aspx](http://msdn.microsoft.com/en-us/library/cc768529(v=bts.10).aspx).

Installing

SQL Monitor is a web application that runs in your browser. It does not require anything to be installed on the SQL Servers you want to monitor.

Installation overview

SQL Monitor comprises three main components that need to be installed:

- **Web Server**
The Web Server delivers all the pages for the SQL Monitor web interface
- **Base Monitor service**
The Base Monitor is a Windows service that continuously monitors your SQL Servers
- **Data Repository database**
The Data Repository is an MS SQL Server database that stores all the data collected by the Base Monitor service



Click



in the installation wizard to read more about a SQL Monitor component or installation option.

For information about installing on an IIS server, see [Using SQL Monitor with IIS](#).

If you want to access SQL Monitor through a firewall, see [Accessing SQL Monitor through a firewall](#).

Where to install components

The Web Server and Base Monitor service can be installed on the same computer, or on different computers on your network.

The SQL Server database for the Data Repository can be hosted on any SQL Server instance, but the Base Monitor computer will need access to this database.

Installing the Web Server and Base Monitor service on the same computer that's hosting the SQL Server instances you want to monitor is not recommended.

- The **Base Monitor service** computer needs to be able to connect to the following:
 - the SQL Servers you want to monitor, for collecting data about their performance
 - the Data Repository database, for storing the collected data
- The **Web Server** should be accessible by any PC or device on which you want to view the SQL Monitor client, and must be able to connect to the Base Monitor.
- The **Data Repository** requires a SQL Server 2005 or 2008 database.

The Base Monitor and Web Server computers should always be switched on.

Account credentials required for installing

When installing and setting up SQL Monitor you need to provide credentials for the following:

- connecting the Base Monitor service to the Data Repository (if you use a Windows account, this account will also run the Base Monitor service)
- connecting to the machine (the physical or virtual server) hosting the SQL Server you want to monitor
- connecting to each SQL Server instance you want to monitor

What's next?

Install the SQL Monitor components. For more information, see [Installing SQL Monitor components](#).

Installing SQL Monitor components

Installing the SQL Monitor Web Server

1. Run the SQL Monitor installer on the computer that you want to host the Web Server.
If you choose to install only the Web Server, the installation wizard will finish once the Web Server is installed; otherwise the wizard will continue with options for installing the Base Monitor and Data Repository.
2. Choose whether to install the SQL Monitor Web Server or use an existing IIS Web server.
If your computer is **already in use** as an IIS Web Server, you can choose to configure IIS to add SQL Monitor. See [Using SQL Monitor with IIS](#).
If this is not the case, you will need to install the SQL Monitor Web Server. The SQL Monitor Web Server is a self-contained XSP Web Server that runs using the .NET 3.5 SP1 runtime.
3. Select a TCP port for incoming connections to the Web Server. Use the default port of 8080 unless it is already in use.
4. If you are installing just the Web Server, click **Finish** to close the installation wizard and automatically open SQL Monitor in your browser. You can then download the Base Monitor installer to another computer and install the Base Monitor and Data Repository.

SQL Monitor relies on the JavaScript engine of your web browser to interpret and execute JavaScript correctly. The engines used by certain browsers may not perform as well as others, and in turn this may affect the performance of SQL Monitor. Should you experience performance issues, try switching to one of the other supported browsers listed in [Requirements](#).

We recommend that you copy the SQL Monitor URL link displayed on the final page of the wizard.

If you are installing both components, the wizard will continue automatically with options for installing the Base Monitor service and Data Repository database.

Installing the SQL Monitor Base Monitor service

1. Specify where to install the files that will run the service, and provide a writable folder for the configuration file.
The configuration file stores the location of the Data Repository and the connection string for accessing the database. The configuration file requires a writeable location so that it can be updated by SQL Monitor if you move the Data Repository database to a different SQL Server instance.
2. Select a TCP port to use for communicating with the Web Server and the Data Repository.
Use the default port of 7399 unless it is already in use.

You now need to create a SQL Server database to use for the Data Repository, and ensure the Base Monitor service can connect to it.

Setting up the Data Repository

The SQL Monitor Data Repository requires an MS SQL Server 2005 or 2008 database to store all the data it collects from your monitored servers.

We recommend that you estimate the eventual size of your database using the guidelines described in [Supported platforms and hardware guidelines](#), and use a SQL Server management tool to create a database with settings appropriate to your environment. You can then follow the instructions below on using an existing database.

If you are installing the Data Repository on a virtual machine, make sure that the SQL Server database is installed on a physical disk mapped to the virtual machine. Storing the Data Repository database on a virtual disk will affect performance and is not recommended.

Using an existing database for the Data Repository

If you have followed our recommendations and created a new database, it should be completely empty, i.e., it should not contain any tables.

You can also select a database created during installation of a previous version of SQL Monitor version 2.

To specify your Data Repository database:

1. Click **Set Up Data Repository** and then select **Use an existing SQL Server database**.
2. Click **Next**, then select the SQL Server instance.
3. In the **Database** box, select the database you want to use; databases on the SQL Server should be automatically detected, and appear in the list.
You can type an instance name or IP address directly into the Server box if the instance name does not appear in the drop-down list.
4. Click **Next**. SQL Monitor checks the database. Once the database connection has been confirmed, click **Close**.

To make sure CPU isn't wasted on monitoring files that don't need monitoring, we recommend that you configure your antivirus settings to ignore SQL Monitor database .mdf and .ldf files.

To create a new database for the Data Repository

To create the Data Repository, you need an account with **Create Database** permissions on the specified SQL Server. This account is used **ONLY** to create the database that SQL Monitor uses as its Data Repository. Once the Data Repository has been created, the credentials you enter here are not used by SQL Monitor.

1. Click **Set up Data Repository** and then select **Create a new SQL Server database**.
2. Click **Next**, then enter the SQL Server instance name.
You can type an instance name or IP address directly into the **SQL Server** box if the instance name does not appear in the drop-down list.
3. By default, SQL Monitor will create a database called `RedGateMonitor`. To use a different name, in the **Database** box, type the name you want.
4. Under **Create database using**, choose one of the following:
 - **Use current credentials** to use the account that you are currently logged into.
 - **Specify an account**. If you select Windows authentication, SQL Monitor assumes this account is in the current domain. To use a different domain account, enter credentials in the format `username@domain-name` or `domain-name\username`. Enter the password.
Note: If you specify a different Windows account, and this account fails, SQL Monitor will automatically attempt to create the database using your current credentials.
5. Click **Create Now** to create the database. Once database creation has been confirmed, click **Close**. If you go back a page in the wizard, this will not undo the creation of the database.

If you don't have a login for the SQL Server with the correct permissions, use the `sp_addsrvrolemember` stored procedure to assign a login to the **dbcreator** role. [More information about sp_addsrvrolemember](#).


If you see an "Invalid sequential schema version" or "Database is not empty...." error when creating the database, see [Error messages: creating a Data Repository database](#).

If the error is related to permissions, make sure you have **Create Database** permissions on the specified SQL Server.

If you have antivirus software installed on your server, we recommend that you configure it to stop monitoring the folders where your SQL Monitor database .mdf and .ldf are stored; this can free up valuable CPU.

Connecting to the Data Repository database

The Base Monitor service requires credentials to connect to the database you just created in order to store and retrieve collected data. This account must have **administrator privileges** (db_owner database role) on the database.

Connect to Data Repository database using: 

Authentication:

User name: Domain: acme-corp

Password:

[Edit connection properties...](#)

Windows authentication

If you connect to the database using Windows authentication, the Windows account you select will also be used to run the SQL Monitor Base Monitor service. If the account does not have permissions to run a Windows service, then it will be granted these permissions.

The user name defaults to your current domain. You can change the domain in the **User name** box using either of the following formats:

- mydomain\username
- username@mydomain

If this account does not have **Log on as Service** permissions, SQL Monitor will automatically attempt to grant them to the account.

If you select SQL Server authentication, the Base Monitor service will run as Local Service.

SQL Server authentication

You can connect to the database using SQL Server authentication. If you provide SQL Server login credentials, the Base Monitor service will run under the [Local Service account](#).

If you encounter an error message when connecting to the database, see [Account credentials required](#).

Error and feature usage reporting

SQL Monitor can send data back to Red Gate about the features you use and any application errors you encounter. This helps us to improve SQL Monitor for future releases.

Data is anonymized before we receive it, and no confidential, sensitive or commercial information is sent to us.

If you are happy to report errors, turn on the **Send error reports** check box, and optionally enter your email address. We will only contact you in the event of an error for which we require further information, to help us eliminate bugs.

Summary

1. Review all your installation options on the Summary page. If you want to change anything, use the **Back** button to edit the required page.

Once your database for the Data Repository has been created, it will not be deleted if you go back and set up a different Data Repository.

2. Click **Install** to start installing SQL Monitor using the selected settings.
3. When installation has completed, click **Finish** to launch SQL Monitor.

What next?

To start using SQL Monitor, you first need to create a password to use when logging in to the SQL Monitor pages. See [Creating the Administrator role](#).

Using SQL Monitor with IIS

SQL Monitor does not install IIS. If you do not have an existing IIS Web Server, you will need to install the SQL Monitor Web Server. See the [Installing SQL Monitor components](#).

ASP.NET

SQL Monitor is an ASP.NET application. Some default installation of IIS do not enable ASP.NET; check that your IIS server has ASP.NET enabled before installing SQL Monitor.

For more information about ASP.NET troubleshooting, see [Issues with IIS and ASP.NET](#).

For IIS 7

Before you install, make sure your account permissions have **Full Control** over the folders described in [SQL Monitor Web Service account](#).

1. Run the SQL Monitor installer on your IIS server, and select **Use existing IIS Web Server**.
2. Select a destination folder for the SQL Monitor website.
All the files for the SQL Monitor website will be extracted to this folder.
3. After completing installation, use IIS Manager to add the SQL Monitor website:
 - a. Select **Add web site** and choose a name for the website.
 - b. Enter the physical path for the website folder - this is the location of the website folder specified during installation of SQL Monitor.
 - c. Specify an available port number and apply any other options, such as binding or certification that you need.
 - d. Click **Browse** to test that the website is working; you should see the SQL Monitor front page.

A new application pool is automatically created for the SQL Monitor website. If you want to add SQL Monitor to an existing application pool, edit the **Advanced properties** for the site to change its application pool.

[Read more about setting up your IIS website \(Microsoft KB article\)](#)

For IIS 6

Before you install, make sure your account permissions have **Full Control** over the folders described in [SQL Monitor Web Service account](#).

1. Run the SQL Monitor installer on your IIS server, and select **Use existing Web Server**.
2. Select a destination folder for the SQL Monitor website.
All the files for the SQL Monitor website will be extracted to this folder.
3. After completing installation, use IIS Manager to add the SQL Monitor website:
 - a. Right-click **Web Sites** and click **New > Web site**.
 - b. In the Web Site Creation Wizard, type the name of the website (for example "SQL Monitor") in the **Description** box.
 - c. Enter the IP address of your IIS server and specify an available port (80 is the default port).
 - d. Leave the Host header box empty, unless you have a specific requirement for a host header.
 - e. In the Web Site Home Directory page, copy the path you specified in the SQL Monitor installer into the **Path** box (by default, this is: C:\Program Files\SQL Monitor 2\Website)
 - f. Ensure that **Allow anonymous access to this Web site** is checked. SQL Monitor requires anonymous access to the website. For more information about website authentication, please read the following Windows KB article for more information: <http://support.microsoft.com/kb/324274> (How to configure IIS Web site authentication in Windows Server 2003).
 - g. Leave the default permissions on the Web Site Access Permissions page.
4. SQL Monitor is an ASP.NET MVC application that uses URLs without extensions, so you will need to use wildcard mapping:
 - a. Right-click the Monitor website you just created and click **Properties**.
 - b. In the Properties dialog, go to the Home Directory page and click **Configuration**. (If you have set SQL Monitor to be a virtual directory beneath an existing website, then go to the Virtual Directory page).
 - c. Under Wildcard application maps, click **Insert**.
 - d. In the Add/Edit Application Extension Mapping dialog, type the path to the aspnet.isapi.dll file in the **Executable** box. This should be mapped to C:\windows\microsoft.net\framework\v2.0.50727\aspnet_isapi.dll or, for 64 bit Windows, to C:\windows\Microsoft.NET\Framework64\v2.0.50727\aspnet_isapi.dll.
 - e. Ensure that Verify that file exists is **not** checked.
 - f. Reload the SQL Monitor page to check that it displays correctly.

Moving from SQL Monitor Web Server to IIS

If you already have SQL Monitor installed and are using the default XSP web server, you can later move hosting to IIS by doing the following:

1. Ensure that there is an IIS Web Server installed on the machine, and that ASP.NET is enabled.
2. Run the SQL Monitor installer and select **Use existing Web Server**.

3. On the Set up Web Server page, browse to and select the existing XSP web server file path as the install location for the IIS web server files. By default, these are located in:
C:\Program Files\Red Gate\SQL Monitor 2\Web
4. Complete the installation by following the remaining instructions for IIS 7 or IIS 6 above.
5. Stop the XSP server. To do this, go to Services (in XP, **Start > All Programs > Administrative Tools > Services**) and stop **SQL Monitor Web Service**.

Adding servers to monitor



Monitoring

Monitored servers

Add servers to monitor; edit connection credentials; remove or suspend monitored servers; configure maintenance windows.

Trace


Enable or disable Profiler trace on selected servers.

Groups

Organize your monitored servers into groups.

Once you have installed SQL Monitor and created the Administrator login password, choose the SQL Servers you want to monitor.

1. Go to the **Configuration** tab and under **Monitoring**, select **Monitored servers**.
2. Click the **Add SQL Server to monitor** button. The Add SQL Server panel is expanded:

3. Enter the fully qualified name of the instance you want to monitor in the **SQL Server** box, including any subdomains. You can type the first few characters of a server name to display matching suggestions. To add a cluster, enter the fully qualified machine name of the management node, or any node in the cluster. Click  for more information about accepted formats.
4. You will need to provide two sets of credentials:
 - to connect to the host Windows machine on which the instance is running
 - to connect to the SQL Server instance

By default SQL Monitor uses the Base Monitor Service account for both sets of credentials. This is the Windows account you specified during installation to connect to the Data Repository (the SQL Server database that stores all collected data). You can specify a different account to use to connect to the host Windows machine and to the SQL Server instance.

For more information, see [Monitoring SQL Server instances](#).

5. To set more advanced properties, for example, to connect to a SQL on a different port, click **Edit properties** to display the Connection properties box:

Connection properties ✕

Settings for connecting to the SQL Server:

Network protocol: ▼

Port number:

Network packet size: bytes

Connection time-out: seconds

Execution time-out: seconds

Security: Encrypt connection

[Restore defaults](#)

Enter the properties you want to change for this instance, then click **OK**. If you change the connection properties, they are identified as **Custom**:

Password:

Connection: Custom [Edit properties](#)

The connection properties are persisted until you move to a different page in SQL Monitor. If you add more servers, they will use the custom connection properties.

- When you click **Add**, the host machine and instance(s) are both added to the list of monitored servers below the **Add SQL Server to monitor** panel. Once added, server names cannot be edited. If you have typed the name incorrectly, the server will be added with a status of "Connection failed (Unreachable)". You will need to remove the server, and then add it again.

Server	Credentials	Status
<input type="checkbox"/> testnetwork	(Base monitor account)	✓ Monitoring Connected
<input type="checkbox"/> (local)	(Same as Windows host)	✓ Monitoring Connected

Once connection has been established and a data collection event has been successful, the status is displayed as Monitoring (Connected). If SQL Monitor encounters a problem and a different status is displayed, see [Monitoring status explained](#) for more information.

Editing the credentials or properties of monitored servers

See [Configuring monitored servers](#).

Troubleshooting installing

- Error messages - connecting to the Data Repository
- Error messages - creating a Data Repository database account

Error messages - connecting to the Data Repository

The table below lists the error messages that may be displayed if SQL Monitor cannot connect to the Data Repository.

Message	Possible cause
Login failed	<ul style="list-style-type: none">• Windows user name or domain name incorrect.• Windows password is incorrect.• Account does not have permissions to access the Data Repository database.• Permissions cannot be escalated to Log on as service (required for the account to run the Base Monitor service).
Login failed for user <username>	SQL Server user name or password incorrect.
Login failed for user <username>. Reason: the password of the account must be changed	Password policy has been enforced for this SQL Server login. The password must be changed.
The SELECT permission was denied on the object...	SQL Server account has only minimum access permissions to this instance. Ensure that the Windows login or database user associated with this SQL Server login is not a member of db_denydatareader group.

Error messages - creating a Data Repository database account

The table below lists the error messages that may be displayed if SQL Monitor cannot create a database account for the Data Repository.

These messages may relate to either the authentication details you have entered, or to your current Windows login (i.e., the login on the PC running the installer).

Message	Possible cause
Login failed for user <username>	User name or password is incorrect, or account does not have permissions to log in to the specified SQL Server.
CREATE DATABASE permission denied in database 'MASTER'	Account does not have CREATE DATABASE permissions on the SQL Server. dbcreator role permissions are required.
A network-related or instance-specific error occurred while establishing a connection to SQL Server. The server was not found or was not accessible	The SQL Server instance cannot be located or connected to on the network; check the following: <ul style="list-style-type: none"> • The name of the SQL Server is entered correctly. • The database engine is configured to accept remote connections. • SQL Server service is running on the specified machine. • Your firewall is configured to allow connections to this server (check the port number and instance). • The Network protocol is enabled/supported on the SQL Server (click Edit connection properties to change the network protocol). • Ping the server to ensure that there is network connection to the host server.
Login failed. The login is from an untrusted domain and cannot be used with Windows authentication	SQL Server has rejected the credentials because the SQL Server is on a different domain from the account: <ul style="list-style-type: none"> • Ensure that the account is a domain account. • Verify that you are logged in to the correct domain and that SQL Server is set up to allow this login.
A connection was successfully established with the server, but then an error occurred during the pre-login handshake	The SQL Server you selected does not support encrypted connections: Click Edit connection properties and ensure that the Encrypt connection option is turned off.

Licensing

SQL Monitor requires a license for each monitored host machine (including virtual machines), regardless of how many SQL Server instances are running on the machine. If you are managing clusters, a separate license is required for each node in the cluster.

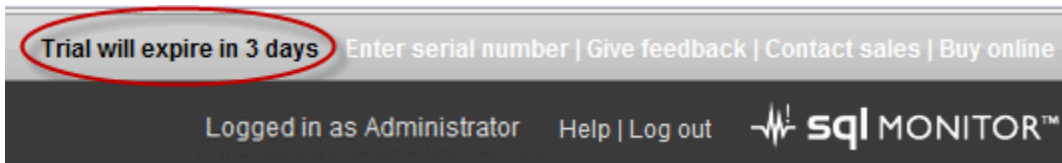
Finding your serial number

When you purchase SQL Monitor, the Sales team will send you an email containing the serial number(s) to activate your licenses. To access your serial numbers at any time, login to the [Red Gate website](#) and follow the instructions.

Using the trial version of SQL Monitor

How long before my trial expires?

If you're currently using an evaluation version of SQL Monitor, the number of days remaining before your trial expires is displayed in the top right corner of the page:



What happens when my trial period expires?

Under special circumstances, Red Gate Sales may be able to extend your trial period. The trial license can only be extended once: if your extended trial has expired, contact [Red Gate support](#).

Once your trial has expired, SQL Monitor stops collecting data for your monitored servers and all unlicensed machines are removed from the Overview pages. Their status is changed to Unlicensed on the Monitored servers page. Click [Contact sales](#) to send an email to Red Gate, or [Buy online](#) to access the website.

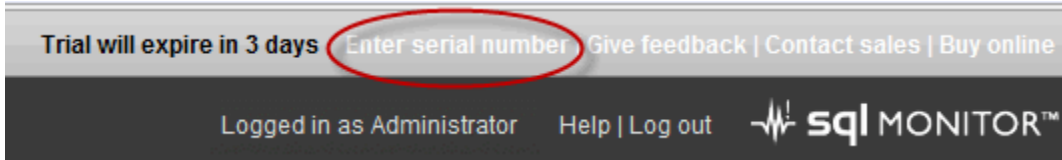
Activating and deactivating

Once you've activated your serial numbers, SQL Monitor allocates licenses to the servers you're monitoring automatically.

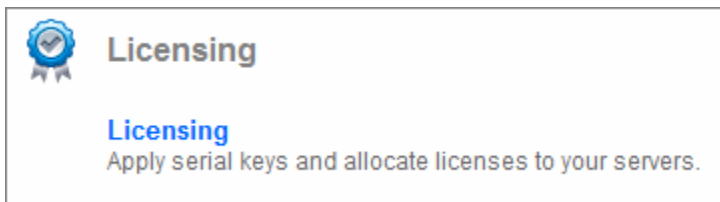
Licensing your servers

Go to the Licensing page by doing one of the following:

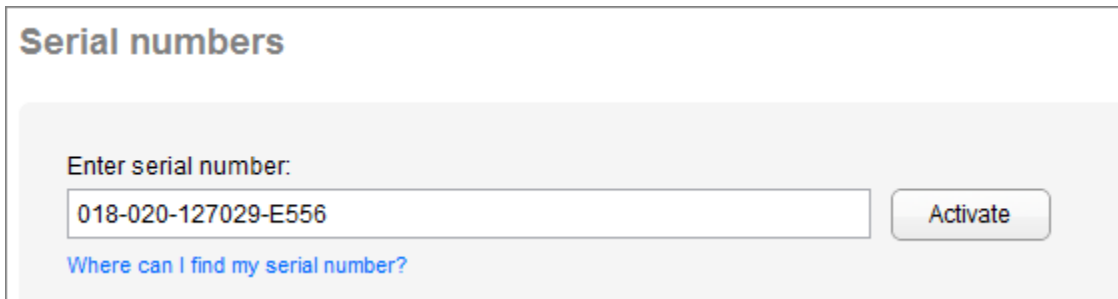
- Click on the **Enter serial number** link in the top right corner of the page:



- Go to the **Configuration** tab and under **Licensing**, select **Licensing**:



Paste a serial number into the **Enter serial number** text box and click **Activate**.

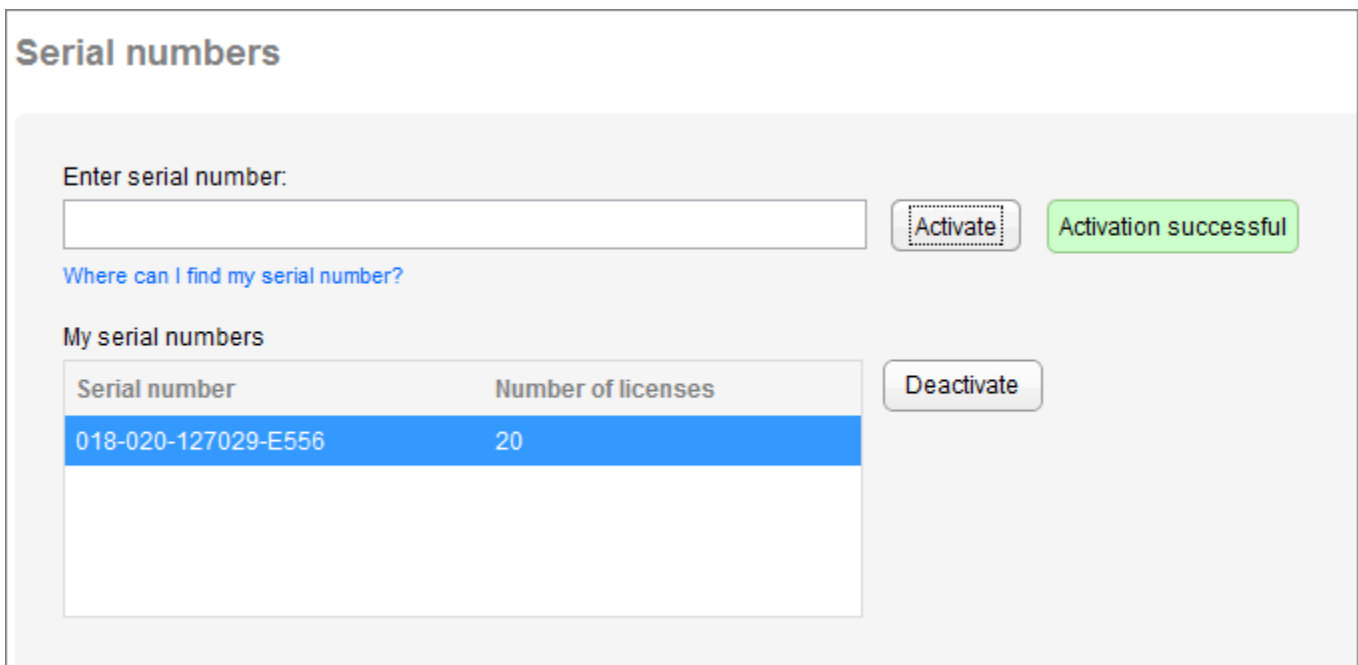


Serial numbers

Enter serial number:

[Where can I find my serial number?](#)

When a successful activation has been confirmed, details are displayed in the **My serial numbers** table:



Serial numbers

Enter serial number:

[Where can I find my serial number?](#)

My serial numbers

Serial number	Number of licenses
018-020-127029-E556	20

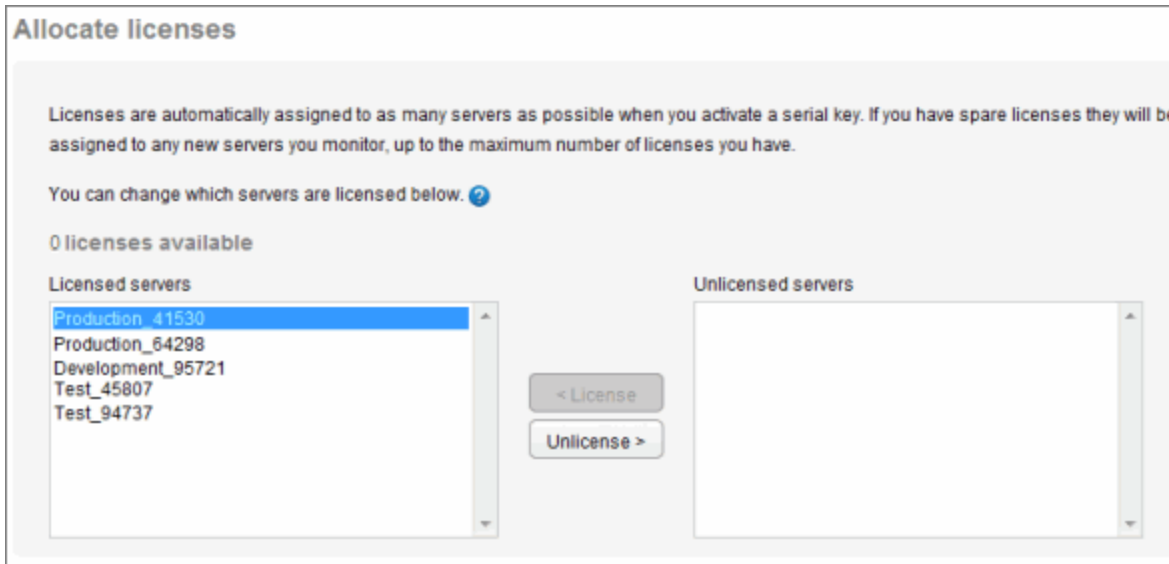
If the serial number cannot be activated, a message is displayed explaining the nature of the problem. You may need to contact Red Gate for further assistance.

Allocating licenses

SQL Monitor detects the number of servers currently being monitored and allocates available licenses to them automatically. These are displayed in the **Licensed servers** list. If during your evaluation you were monitoring more servers than you purchased licenses for, or you subsequently add more servers to monitor than you have licenses available, the servers added most recently will not be allocated licenses. These are displayed in the **Unlicensed servers** list. Unlicensed servers are not removed from the Monitored Servers list, but they will not be monitored and alerts cannot be raised on them.

You can change which servers have licenses allocated to them using the **< License** and **Unlicense >** buttons to move selected servers from one list to the other.

In the example below, five licenses have been purchased and allocated to the five servers currently being monitored. There are no spare licenses available.



Deactivating a serial number

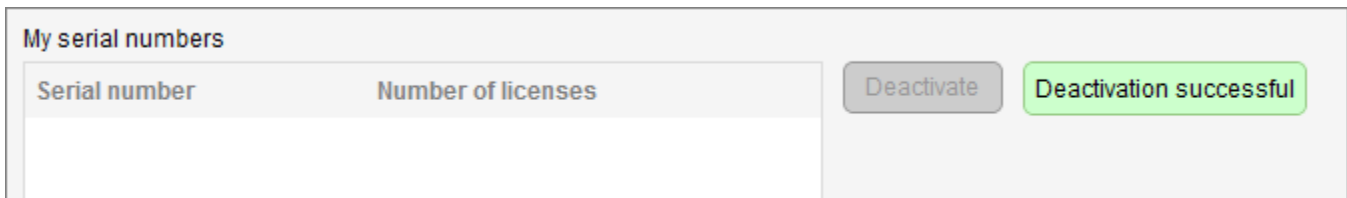
You may need to deactivate a serial number if:

- You are moving the Base Monitor service to a different machine
- You are evaluating a new version of SQL Monitor, and your existing serial number is for fewer servers that you want to monitor in trial mode; deactivating the serial number in this case will return you to trial mode (assuming you are within the evaluation period).
- Red Gate sales or support have sent you a different serial number to use

To deactivate a serial number

Select the serial number in the list and click **Deactivate**.

SQL Monitor contacts the Red Gate licensing server, and the serial number is removed from the list:



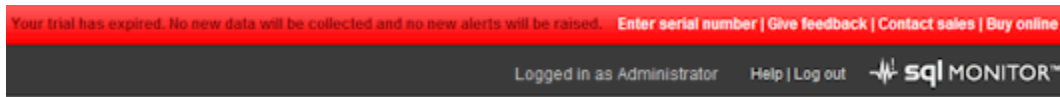
If there is a problem deactivating, a message is displayed explaining the nature of the problem. You may need to contact Red Gate for further assistance.

What happens when I deactivate a serial number?

The relevant number of servers are automatically moved from **Licensed servers** to **Unlicensed servers**. For example, if your serial number was for 10 licenses, then 10 servers will be unlicensed.

If you have licensed your servers using several serial numbers, you may have more licensed servers than are being removed. In this case, SQL Monitor removes the oldest servers first (those that were added to SQL Monitor first).

If all your servers are now unlicensed, SQL Monitor displays a warning banner:



No further monitoring data will be collected, and no new alerts will be raised, until you enter a serial number.

Troubleshooting licensing

SQL Monitor may become unlicensed without warning if a serial number is removed from the **Licensing** page. This may happen if you have more than one version of SQL Monitor installed on your network, and each version is using the same Data Repository database.

To view the details of the Data Repository database, either:

- Check the name and location of the Data Repository database: go to **Configuration > About** and select **About**. The Database entry under Base Monitor is the Data Repository database.

or,

- Run the query `sys.sp_who2` against the database, and find processes that display *SQL Monitor - Repository* in the ProgramName column. The HostName column entries for these rows confirm which machines are running SQL Monitor.

You **must** store collected data in a separate Data Repository database for each installation. If you are only using one version of SQL Monitor, simply uninstall the redundant version(s).

If you want to maintain separate versions of SQL Monitor, create a new Data Repository database for one of the installations, and update the connection string between the Base Monitor service and the database. For details of how to do this, see [Moving the Data Repository database](#).

Upgrading

Upgrading to a later version of SQL Monitor will preserve all your current monitoring data in the Data Repository. SQL Monitor will automatically modify the schema of the Data Repository database to the newer version during the installation process.

How do I know when an upgrade is available?

SQL Monitor will display a message above the product name banner when a later version is available:



Click on the link for more information about the newer version, and a link to download it.

How to upgrade

Before upgrading

Before you upgrade to a later version, we recommend that you back up your current Data Repository. Once the upgrade is completed, you will not be able to roll back to a previous version of your Data Repository database.

Upgrading from v2.x to v3.x

You should only upgrade to v3.x if your current SQL Monitor license includes a valid support and upgrades package.

If you upgrade with a support and upgrades package, the v2.x activated licenses will expire and SQL Monitor will reactivate them as v3.x licenses.

If reactivation fails, or your v2.x license does not include support and upgrades, SQL Monitor will delete the serial numbers from the Configuration > Licensing page and start a 14-day trial version of v3.x. You **cannot** roll back to the previous 2.x version, so we recommend that v2.x users without a valid support and upgrades package install a trial version of v3.x on a separate machine. When v2.x users are ready to purchase v3.x licenses, they can add the support and upgrades package to make upgrading easier.

- To find your v2.x serial numbers, go <http://www.red-gate.com/myserialnumbers> to and enter your email address and password.
- To find out whether your license includes a valid support and upgrade package, go to <http://www.red-gate.com/upgrade> and enter your serial number.
- To purchase a v3.x license with the support and upgrades package, contact dba.info@red-gate.com.

Updating the physical path in IIS

If you've upgraded SQL Monitor from v2.x to v3.x, you'll need to change the physical path to the SQL Monitor web server configured in IIS. The path needs changing from:

C:\Program Files\Red Gate\SQL Monitor 2

to:

C:\Program Files\Red Gate\SQL Monitor 3

For details on how to do this, see [Change the physical path of application content](#).

Installing the new version

1. Run the installer, and follow the wizard to install the Web Server and Base Monitor.
2. On the **Set up Data Repository** page, supply credentials with **db_owner** permissions under **Connect to Data Repository database using**:

SQL Server database for Data Repository

SQL Server: (local)
Database: RedGateMonitor

[Set up Data Repository...](#)

Connect to Data Repository database using:

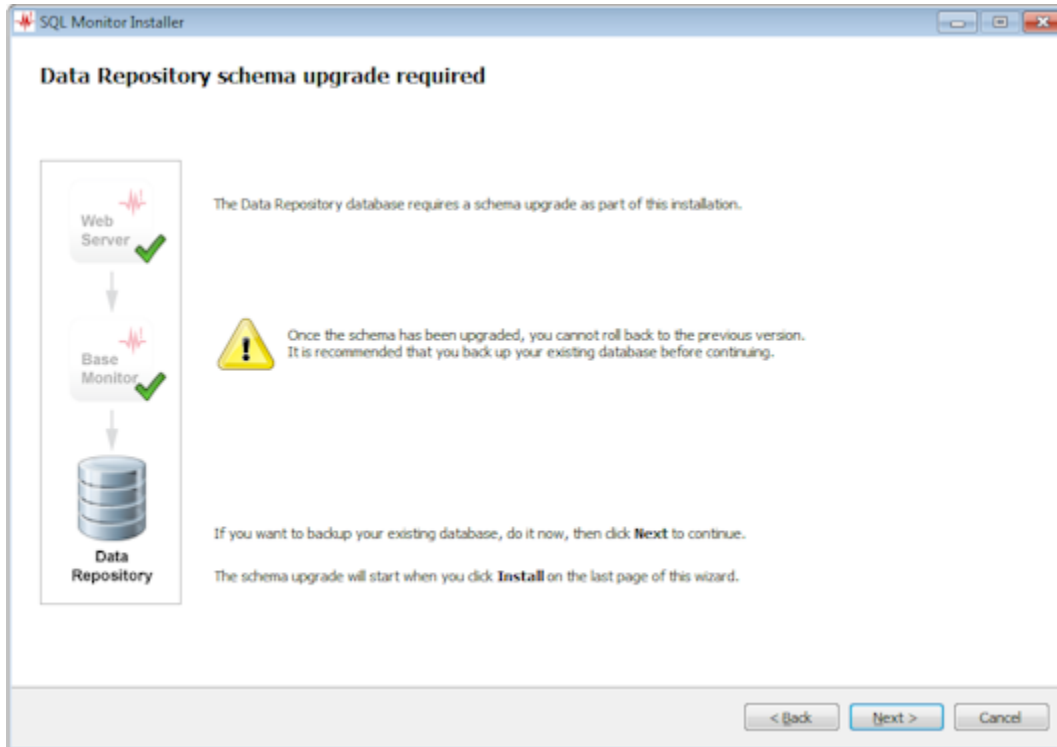
Authentication:

User name: Domain: acme-corp

Password:

[Edit connection properties...](#)

- SQL Monitor will use these credentials to upgrade the Data Repository at the end of the installation process.
3. Click **Next**. The installer will prompt you to take a backup of the Data Repository before continuing:



4. When you have successfully backed up the Data Repository database, complete the rest of the wizard and click **Install**. All the components are installed. SQL Monitor will attempt to upgrade the database schema as soon as the web service is running successfully.
5. If you open SQL Monitor in your browser before the upgrade is complete, you will see the schema upgrade progress bar:



If the upgrade is successful, your browser will then display the Enter password page or the Global Overview.

If the upgrade is not successful, an error page is displayed.

Error when upgrading?

If you encounter an error page when upgrading, please refer to the error message highlighted in the box. All error messages are those returned by SQL Server and not SQL Monitor.

The most common reason why the Data Repository cannot be upgraded is that the supplied account lacks the sufficient permissions to modify the database. Enter an account with db_owner permissions and click **Retry Upgrade**.

Note: This account is used only to upgrade the database schema. It is not required by SQL Monitor for anything else.

Using SQL Monitor for the first time

When you have installed SQL Monitor for the first time

1. Create a password to log in to SQL Monitor:



This password will be required for anyone logging in to the SQL Monitor website.

If the Create password screen is not displayed when you enter the URL for SQL Monitor, check that the details for the Base Monitor machine and port number are correct and that the Base Monitor service is running.

You can subsequently change this password by clicking the Configuration tab and selecting **Change password**:



Application options

Base Monitor connection

View details of your current Base Monitor connection; connect to a different Base Monitor.

[Change password](#)

Change the password required to log in to SQL Monitor.

2. Select the servers you want to monitor.

Click the **Monitored Servers** link in the message box, or select **Manage monitored servers** from the left side.



You are not currently monitoring any servers.

Add the servers you want to monitor on the [Monitored Servers](#) page.

3. You need to supply credentials for SQL Monitor to connect to both the host machine and SQL Server instance. Click any



icon for more information about the format for entering a server name and the credentials required.

See [Adding servers to monitor](#).

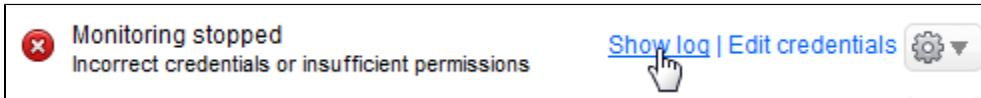
4. Check that SQL Monitor can connect to the servers using the supplied credentials.

The status of each server you add is shown in the **Status** column. When SQL Monitor is successfully monitoring the following is displayed:



Monitoring
Connected

If there is a problem collecting data from a server, click **Show log**:



Review the error message, and if required, edit the credentials, or take other remedial action. See [Account permissions required](#).

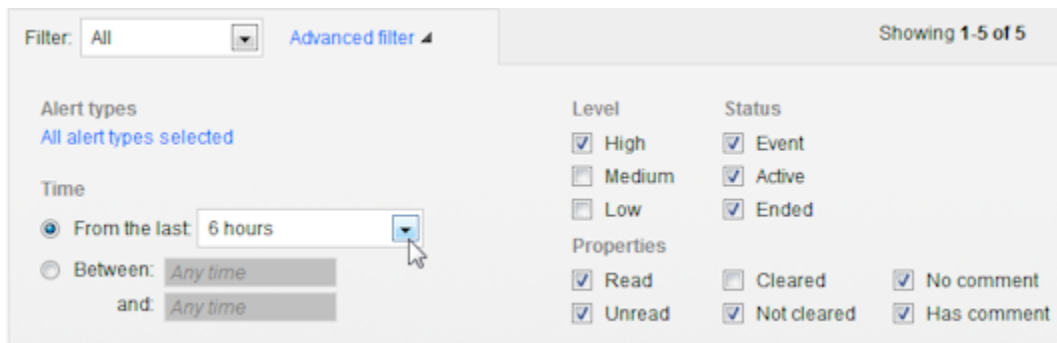
5. Go to the Global Overview page to see the current health of your monitored servers and databases. Click the **Overviews** tab. All your monitored servers should be displayed, showing performance data and any raised alerts. You can start to drill down by selecting particular instances or databases from the Monitored servers list on the left side. See [Working with overviews](#).

After you have been using SQL Monitor for a short while

1. If required, organize your monitored servers into groups. This will help you manage your development or production servers more easily in SQL Monitor.
See [Organizing monitored servers into groups](#).
2. Review any raised alerts.
You may see some alerts already raised for your servers. SQL Monitor is pre-configured with default settings for all alert types. To review any raised alerts, go to the **Alerts** tab and click on an alert in the inbox to view its details. You can then start to clear alerts that you have dealt with, or configure their settings. See [Working with alerts](#).
3. Review the default settings for all alerts.
Click the **Configure alerts** link under Monitored servers on an overview or alert page to view all the alerts that SQL Monitor can raise, and review their default settings. You can configure an alert's settings globally (at the All Servers level) or for any group, server, database or job. See [Configuring alerts](#).
4. Set up email notification for your alerts.
Go to the **Email settings** page (Configuration > Email settings), and enter your SMTP mail server settings and a global email address. SQL Monitor will send emails to this email address when alerts are raised. You can later configure each type of alert to send emails to different recipients. See [Setting up email notification](#).

When you have been running SQL Monitor overnight

1. Set up a filter to see only important alerts raised since you were last at your computer.
In the **Alert Inbox** click to expand the **Advanced filter** panel, and select a time from the **From the last** box, or enter a custom time range. You can also filter to show only alerts raised at a certain level. Click **Save as custom filter** to re-apply the filter options quickly every morning:

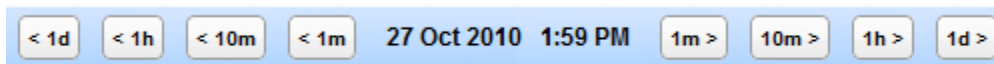


You can save any number of useful filters to help you prioritize reading alerts in the Alert Inbox.

2. Rewind time to look at the performance of your servers at a point during the night.
Go to the **Global Overview** page, drill down to a SQL Server instance or database, then click the **Rewind time** button in the top right corner:



Select a time and click **Apply**. All the sparkline graphs are updated to show their value at the time you select. Use the Back in Time bar to step forward or backward to see how these values change over time:



See [Rewinding time with Back in Time mode](#).

3. Go to the **Analysis** tab to view graphs of various performance metric values for the last few hours for your key servers. See [Working with analysis data](#).

When you have been running SQL Monitor for several days

1. Review the data purging options.

As SQL Monitor collects monitoring data from your servers, the size of the SQL Monitor Data Repository will increase. You can adjust how much historical data SQL Monitor stores in this database, to manage its size and ensure that you don't run out of disk space. See [Purging SQL Monitor data](#).

2. Continue to tweak the configuration of alerts as they are raised.

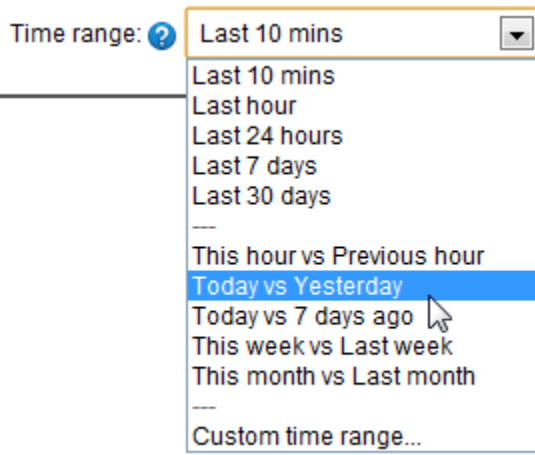
Over time you can improve the relevance of the information SQL Monitor provides by customizing more alerts, at a more granular level. When an alert is raised, open it and click the **Customize alert** link to edit the settings for only the relevant object, e.g. a specific server, database or job.

3. Set up different email recipients for different types of alerts or servers.

As part of each alert's configuration, you can enter a different email address, so that other colleagues in your organization are notified when a particular type of problem occurs, or a specific database is affected. See [Configuring email notification settings](#).

4. Compare the performance of various aspects of your server or database performance on different days.

Go to the **Analysis** tab and select the performance metric and server or database you are interested in. You can now select a comparison timeframe from the **Time range** drop-down box:

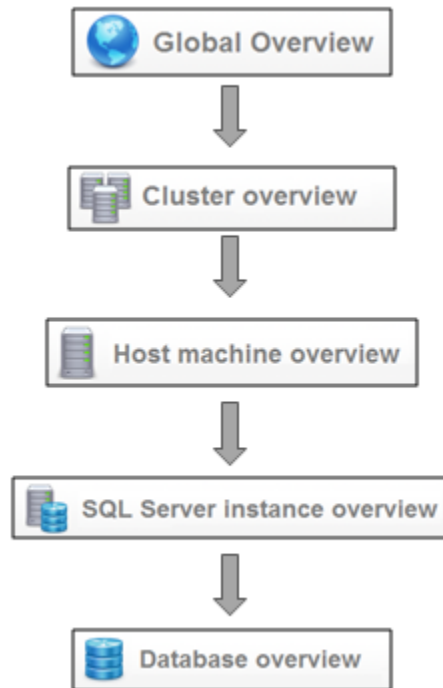


5. When your evaluation period expires, go to the **Licensing** page (Configuration > Licensing) to enter your serial numbers and choose which servers you want to license.
6. Go to the **Monitored servers** page to remove any servers you are no longer monitoring.

Working with overviews



The overviews in SQL Monitor show the current health of your monitored servers, and are arranged into a hierarchy:



The overview pages update automatically every 30 seconds. You do not need to refresh your browser.

You can drill down through the hierarchy in two ways:

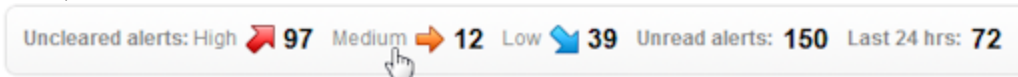
- From the **Monitored servers** list
- By clicking a hyperlink on the current overview page to take you to a lower level (e.g. from a host machine to a SQL Server instance or from an instance to a database)

Global Overview

The Global Overview is the default front page for SQL Monitor. It presents a high-level summary of the health of all your servers, showing which servers have uncleared alerts raised against them, and what their current CPU and memory usage is. Servers that require attention are color-coded. From the Global Overview, you can quickly decide which server to investigate in more detail.

Following links from the Global Overview

- Click on any object (server or database) in the **Monitored servers** list to view the overview page for that object.
- Click on a group to filter the Global Overview to show only servers in the selected group.
- Click on the name of a host machine or SQL Server instance in the **Servers** column to go to the Overview page for that server. You can also click in the **Status** column.
- Click on a number in the alerts summary panel at the top of the page to go to the Alert Inbox pre-filtered by your selection (e.g., High or Unread) across all servers.



- Click on the colored bar chart under **Uncleared alerts** to go to the Alert Inbox to view alerts for that host machine or SQL Server instance:



- Click on a sparkline graph in the **Processor time** or **Memory Used** column to go the Analysis page for those metrics, showing the last 10 minutes:



Cluster overview

The cluster overview shows all nodes in the cluster, and the SQL Server instances running on the cluster:

Cluster name and path
politics.testnet.red-gate.com

Cluster status (highest uncleared alert on all nodes and instances)
Status: **High** Uncleared alerts: 38 Unread alerts: 7 Last 24 hrs: 7

Nodes in the cluster
labour
libdem

SQL Server instances running on this cluster, and the node hosting them
(local) labour politics-sql

Status of each node (highest uncleared alert on each node)
labour: Low
libdem: High

Status of the SQL Server instance (highest uncleared alert on the instance)
(local) labour politics-sql: Medium

Name	Status	Uncleared alerts
labour	Low	10
libdem	High	25



Name	Node	Virtual instance name	Status	Uncleared alerts
(local) labour politics-sql	labour	politics-sql	Medium	3

Disk	Quorum path	Space used (GB)	Avg. read time (ms)
F:	F:\MSCS\	0.0 / 1.0	0.0

You can drill down from the Cluster overview to the Alert Inbox or Analysis page in the same way as from the Global Overview.

You can also see information about the quorum path and to which resource groups the various resources (including SQL Server instances) are allocated:

Quorum

Disk	Quorum path	Space used (GB)	Avg. read time (ms)	Avg. write time (ms)
F:	F:\MSCS\	0.0 / 1.0	 0.0	 8.3

Resources

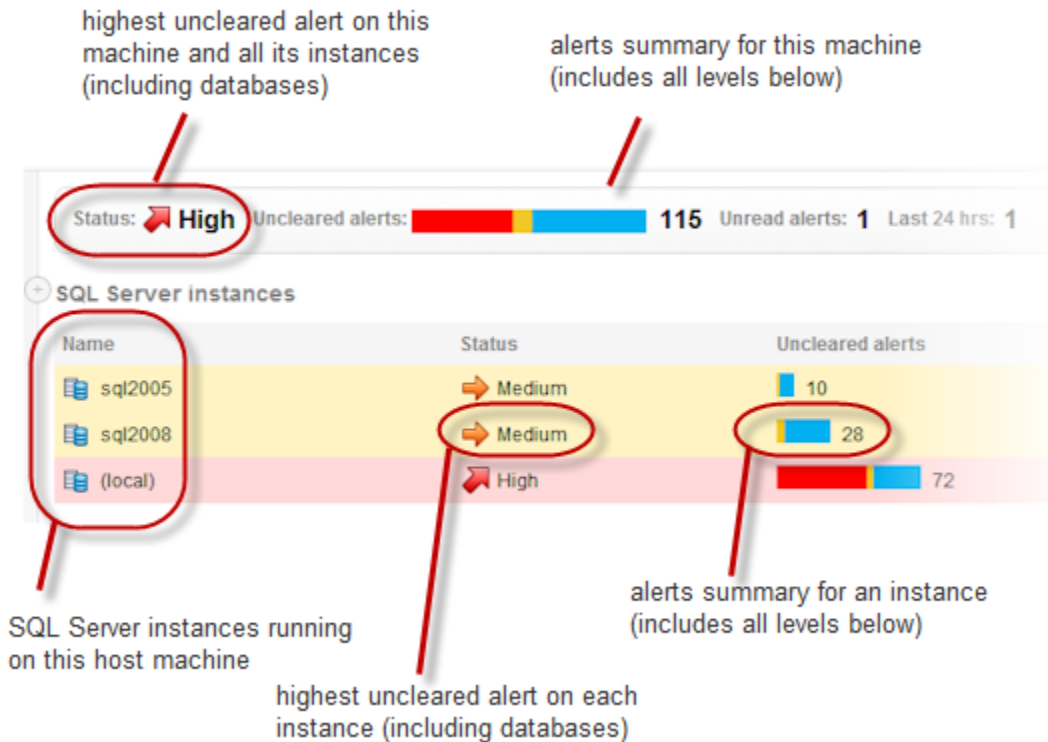
Name	Status	Active node	Resource group	Resource type
Cluster IP Address	Online	labour	Cluster Group	IP Address
Cluster Name	Online	labour	Cluster Group	Network Name
Disk F:	Online	labour	Cluster Group	Physical Disk
DTC	Online	labour	Cluster Group	Distributed Transaction Coordinator
IP	Online	labour	Cluster Group	IP Address
MonitorBaseDeploymentServiceLocal	Online	labour	SqlMonitor	Generic Service
SQL IP Address 1 (politics-sql)	Online	labour	SQLServer	IP Address
SQL Network Name (politics-sql)	Online	labour	SQLServer	Network Name
SQL Server	Online	labour	SQLServer	SQL Server
SQL Server Agent	Online	labour	SQLServer	SQL Server Agent
SQL Server Fulltext	Online	labour	SQLServer	Generic Service

- The quorum is a shared network drive that controls which nodes host which resources, and maintains the configuration data necessary for recovery of the cluster. In SQL Server terms, this equates to which node is active for a particular instance.
- The resources are various entities that are capable of being managed by a cluster. A resource can only be owned by one node at a time.
- A resource group is a collection of resources that are managed as a single unit (e.g. the SQL Server group). During a failover, the groups is moved from one node to another node.

For more information, see [Cluster fundamentals \(MSDN article\)](#).

Host machine overview

The host machine overview is similar to the cluster overview. You can see the status for the machine itself, view all its monitored SQL Server instances, and follow links to the Alert Inbox.



- Click in the **Name** or **Status** column for an SQL Server instance to drill down to overview page for the selected instance.
- Click on the colored alert bar for a SQL Server instance to go to the Alert Inbox for all uncleared alerts relating to that instance.
- Click on **Uncleared alerts**, **Unread alerts** or **Last 24hrs** in the alerts summary panel at the top of the page to go to the Alert Inbox for all alerts relating to the host machine and everything below it in the hierarchy (alerts for all SQL Server instances and databases).

What does the Status column mean?



The **Status** column shows the level of the highest uncleared alert on the machine or any level beneath. If there are no uncleared alerts then the status is **OK**.

The status also shows if the monitoring status is currently anything other than Monitoring (Connected):

SQL Server instances	
Name	Status
sql2005	Medium
sql2008	Monitoring stopped (Incorrect credentials or insufficient permissions)
sqlexpress	Unreachable
(local)	Suspended

If the Status column shows there is a problem with monitoring, for example, it displays Monitoring stopped or Unreachable, then do the following:

1. Click the **Manage monitored servers** link under the **Monitored servers** list on the left.
2. All your monitored servers are listed, and their current status. Click **Show log** for the server with the problematic status:

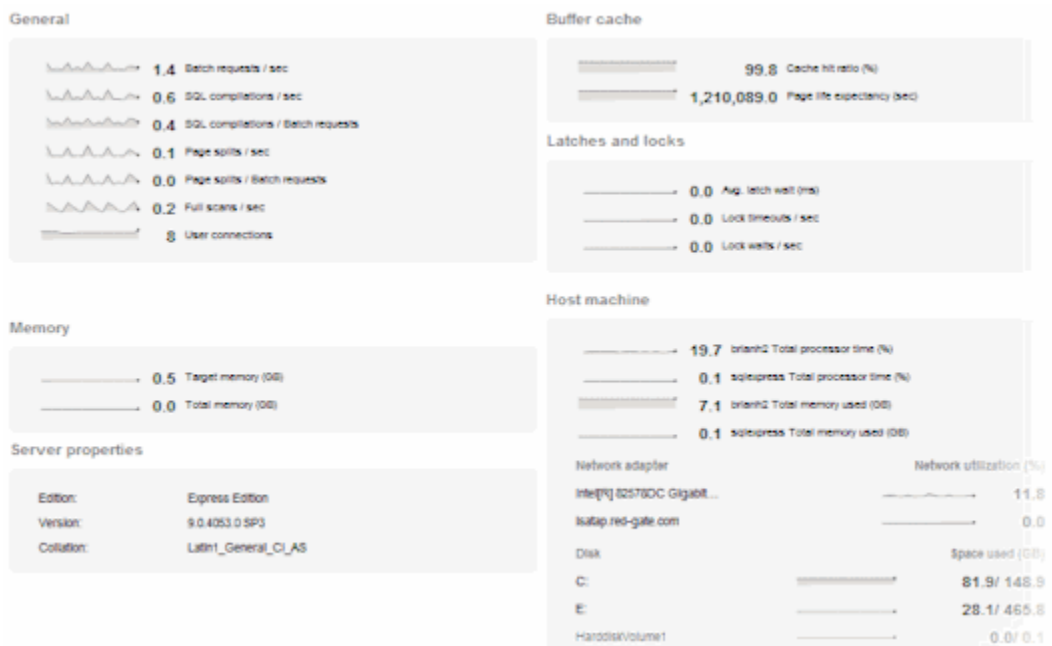

Monitoring stopped
 Incorrect credentials or insufficient permissions
 [Show log](#) | [Edit credentials](#)


3. Review the log of data collection events to investigate the problem.

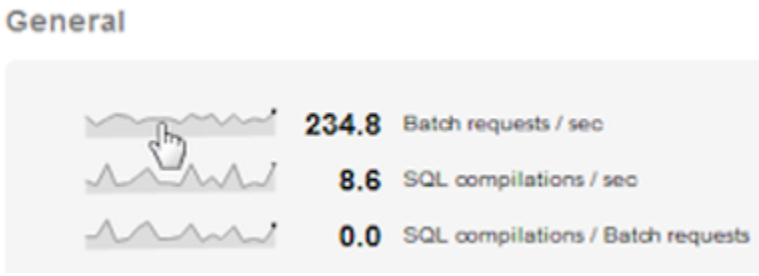
For more information about monitoring status, see [Monitoring status explained](#).

Performance data on the host machine overview

The host machine overviews display various types of current information about your machine:



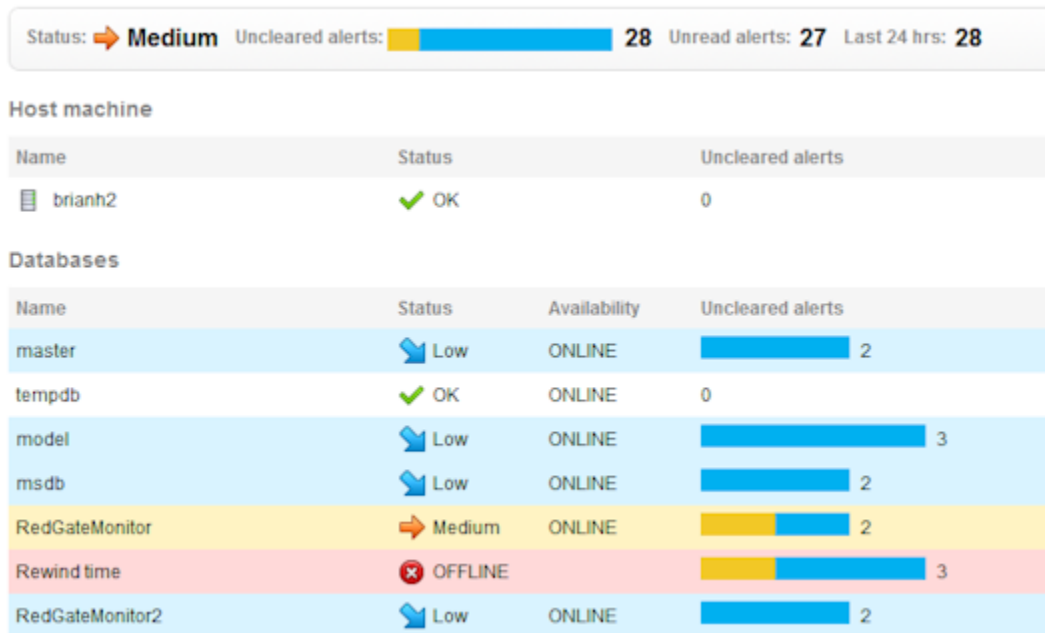
Each sparkline graph shows the current numerical value (in bold) and the last five minutes performance trend. Click on a sparkline graph to go to the Analysis page for the selected metric, where you can view a larger graph and specify a time range:



Processor time % and memory used graphs are not available for individual system processes on the Analysis page.

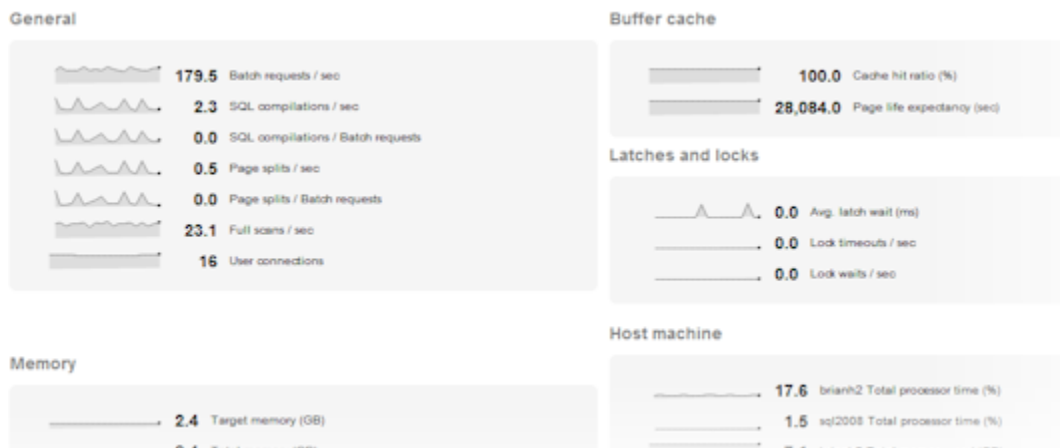
SQL Server instance overview

The SQL Server instance overview lists all the databases hosted on the instance and shows the status of each. Databases are listed in alphabetical order, with system databases listed at the top of the page:



- Click in the **Name** or **Status** column for a database to drill down to the overview page for the selected database.
- Click on the colored alert bar for a database to go to the Alert Inbox for all uncleared alerts relating to that database
- Click on **Uncleared alerts**, **Unread alerts** or **Last 24hrs** in the alerts summary panel at the top of the page to go to the Alert Inbox for all alerts relating to this SQL Server instance (and all its databases)

It shows various types of current information about the instance itself and performance data for the last five minutes for a range of useful counters. As for the other overviews, click on any sparkline graph to go to the Analysis page for that counter.



Top 10 expensive queries

Note: This feature is not available for servers running on Microsoft SQL Server 2000.

The SQL Server instance overview displays the top 10 queries that used the most resource over a selected period of time. This data helps you to evaluate the performance of queries and the efficiency of I/O usage, and can be checked to see what queries were running around the time certain alerts were triggered.

Note: Selecting one of these options does not simply change the sort order of the existing list of queries. A new table is generated according to the selected option, so different queries are likely to be displayed.

Database overview

The database overview is the lowest level in the overview hierarchy. It shows information about the database, its general properties, its data and log file sizes and locations, its top 10 most expensive queries and also any backups.

The top 10 expensive queries are not available for servers running on Microsoft SQL Server 2000.

Why aren't all my backups listed?

The backups section shows only the **most recent** backup of each type:

Backups

Type	Started	Finished
Database	2 Nov 2010 16:03	2 Nov 2010 16:03
Differential database	2 Nov 2010 15:59	2 Nov 2010 15:59

Viewing the full path to data files

The path to your data and log files is likely to be truncated on screen. To see the full path, move your mouse pointer over it to display a tooltip:

Files

Name	Type	Path
My Shiny New Database	Rows	C:\Program Files\Microso...
My Shiny New Database_log	Log	C:\Program Files\Microsoft SQL Server\MSSQL10.SQL2008\MSSQL\DATA\My Shiny New Database.mdf

Rewinding time with Back in Time mode

You can "rewind time" in SQL Monitor to enter **Back in Time mode**, where you can examine the state of any overview as it was at a time in the past.

While in Back in Time mode, you can step forward or backwards in time to see how the values on the overview page change. This is useful to examine the performance of your servers around the time that a problem occurred.

Note: Using Rewind time may be affected by how long you have opted to keep data in the Data Repository. See [Purging SQL Monitor data](#).

To enter Back in Time mode

1. Click the **Rewind time** button at the top right of any overview page, next to the time. If you know the server or database where there was a performance issue in the past, you should first [drill down](#) from the Global Overview page to the relevant overview page.



The Rewind time box is displayed, defaulted to the current date and time:

Rewind time

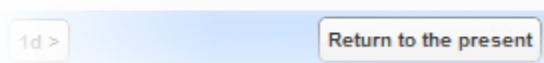
March 2012						
Su	Mo	Tu	We	Th	Fr	Sa
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

Date:

Time:

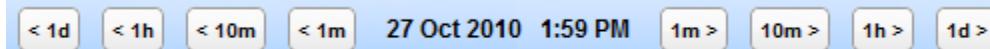
2. Enter the date and time when the problem was first identified. You may want to go back five minutes earlier, so you can watch the performance as it changed.
3. Click **Apply**.
The date and time is removed, and the page is updated at the top right to show that you are now Back in Time and no longer viewing current data:

Back in Time mode



Back in Time mode is a frozen snapshot of data at the time you specified. SQL Monitor does not automatically advance the time, so the sparkline graphs and values will not move forward in time.

4. Use the Back in Time bar buttons to adjust the time in increments of 1 minute, 10 minutes, 1 hour or 1 day:



5. While you are in Back in Time mode, you can continue navigating through the overview pages as usual. The time remains frozen until you return to the present. For example, you can first investigate the performance of the host machine, drill down to a SQL Server instance and then to a particular database, all frozen at the same time.
6. To return to normal operation, click **Return to the present**.

What is displayed in Back in Time mode?

- All performance metrics on the overview pages reflect the time entered in Back in Time mode; the sparkline graph shows the trend for 5 minutes leading up to this time. Click on a sparkline graph to go the **Analysis** page showing a performance graph of that metric for the ten minute period before the 'frozen' time. If you then select an option from the **Time range** drop-down list, Back in Time mode is cancelled and the time range is relative to the current time.
- All processes and error log entries reflect the time entered.
- Database availability is shown for the time entered in Back in Time mode. If the database was at any state other than ONLINE, then the Availability column is shown as blank.

RedGateMonitor	--	ONLINE
Seq13	--	
Eggs	--	ONLINE
Seq11	--	
SrpTuesday_3483	--	ONLINE
TEMPdb	--	ONLINE
utils	--	
SrpWednesday3507_2	--	ONLINE

These databases were not ONLINE at the time

Data not available in Back in Time mode

Alert summary information on the overview pages is **not** shown when in Back in Time mode:

- The alert summary panel at the top of the overview page is removed.
- A dash -- is shown in place of the numbers in the Uncleared alerts, Unread alerts and Last 24 hrs columns. If you click the dash, the Alert Inbox will show these alerts for the current time, not the time in the past.
- The Status column shows a dash (as the Status is determined by uncleared alerts).

Status	Uncleared alerts	Unread alerts	Last 24 hrs
--	--	--	--

Status	Availability	Uncleared alerts	Unread alerts	Last 24 hrs
--	ONLINE	--	--	--
--	ONLINE	--	--	--

What happens when I go back in time before a server was added?

Servers added recently that were not being monitored at the time you are viewing are removed from the **Monitored servers** list.

If you are already viewing an overview page, and rewind time to a point where the SQL Server instance or machine was not yet being monitored, then the page contains no data. Try advancing the time forward to a point where the sparkline graphs are displayed.

Working with alerts



SQL Monitor raises alerts when it detects problems across your servers.

What alerts does SQL Monitor raise?

SQL Monitor alerts warn you about various issues on your host machines, SQL Server instances and databases.

[List of all alert types raised by SQL Monitor](#)

Event alerts and continuous alerts

The following types of alerts are **Event** alerts, which are raised for incidents that occur at a specific point in time:

- Cluster failover
- Deadlock
- Job failed
- SQL Server error log entry

Event alerts are raised at a defined level (Low, Medium or High) which you can configure.

All other types of alert are **Continuous** alerts. Continuous alerts can have the following status:

- **Active:** the issue is still currently a problem
- **Ended:** the issue has been resolved

Depending on the type of alert, the Active duration of an alert can be quite short, for example less than a minute for long-running query alerts, or several days or even weeks for backup overdue alerts.

Like event alerts, continuous alerts are raised at a defined level (Low, Medium or High) which you can configure. For continuous alerts, however, you can configure multiple thresholds, so this level can automatically escalate or downgrade while the alert status is Active.

Alert times displayed in SQL Monitor

All alert times in SQL Monitor are displayed in the local time of your web browser, regardless of where the Base monitor server or your monitored servers are located. For example, if an alert is raised on a server in London at 18:00 and you are using SQL Monitor in New York, the alert time will be displayed as 13:00 (local time for New York).

Note: The web browser clock and Base Monitor clock need to be synchronized for SQL Monitor to work correctly.

Working with alerts

Viewing a list of raised alerts

The [Alert Inbox](#) page lists all the alerts that have been raised. You can filter the Inbox in many ways, to list only alerts that you are interested in.

Viewing the details of raised alerts

The [Alert details](#) page shows information about a single alert. Click on any alert in the Alert Inbox to view its details.

Changing alert settings

The [Alert settings](#) page allows you to disable an alert or change its level and thresholds.

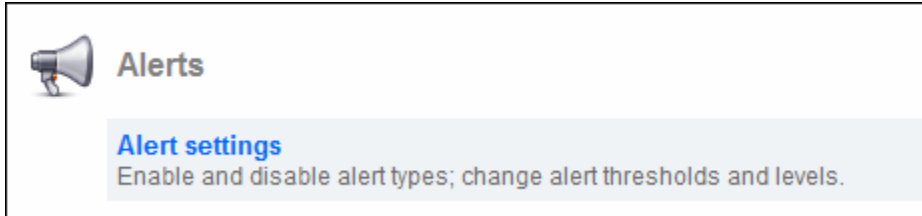
Temporarily suspending alerts

The [Configuration](#) page allows you to schedule maintenance windows during which alerting is suspended on selected servers.

List of alerts

The Alert settings page lists all the alert types that SQL Monitor can raise.

Go to the **Configuration** tab. Under **Alerts**, select **Alert settings**:



Managing alerts

For each type of alert, you can:

- disable it, so the alert will not be raised in future
- change the level at which it is raised, to either low, medium, or high
- change the thresholds that trigger the alert to be raised

You can edit the alert settings for a single SQL Server instance or across a number of instances at once (by creating a group). For job-related, disk-related or database-related alerts, you can edit the alert for a specific job, disk or database.

When an alert is raised, you can quickly change its settings by clicking **Configure alert** in the **Alert details** page.

SQL Server specific alerts

SQL Monitor raises the following types of alerts for problems on a SQL Server instance or database:

Backup overdue

Raised when:	Either of the following conditions apply: <ul style="list-style-type: none">• No entry for a full database backup of this database in the [msdb].[dbo].[backupset] system table.• The most recent entry for a full database backup of this database in the [msdb].[dbo].[backupset] system table is older than a specified time.
Configurable thresholds:	Most recent backup is older than x seconds/minutes/hours/days
Default settings:	<ul style="list-style-type: none">• Raised as Low after 7 days• Medium: not enabled• High: not enabled
Type:	Continuous Continuous alerts can have multiple thresholds and are automatically updated from Active to Ended when the condition that caused the alert to be raised no longer applies - in this case, when a full backup is detected.
Possible causes:	<ul style="list-style-type: none">• No backup job scheduled.• Backup jobs not running or not completing. Check Job failed alerts on this SQL Server instance.• SQL Server Agent Service is not started - check for any SQL Server Agent Service status alerts.

Blocked process

Raised when:	A SQL connection has been waiting for another process to release its blocking lock for longer than a specified duration.
Configurable thresholds:	SQL process blocked for longer than x seconds/minutes/hours/days
Default settings:	<ul style="list-style-type: none">• Raised as Low after 20 seconds• Escalated to Medium after 40 seconds• Escalated to High after 1 minute

Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when the condition that caused the alert to be raised no longer applies - in this case, when the block ends.
Possible causes:	<ul style="list-style-type: none"> • Long-running queries. • Using Insert, Update or Delete on large numbers of records in a single transaction. • Canceling queries, but not rolling them back.

Cluster failover

Raised when:	The active node of a cluster changes to a different node.
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Event Event alerts are raised for incidents that occur at a specific point in time; they do not change level, or update their status to Ended .
Possible causes:	The previously active node has failed or been manually switched to a different node.

Database unavailable

Raised when:	The database state is something other than Online.
Configurable thresholds:	None
Default settings:	Raised as Medium
Type:	Continuous Automatically updated from Active to Ended when the database state changes back to Online.
Possible causes:	Database has been manually removed, or has encountered a problem causing its state to change to Suspect, Emergency, Recovering or Restoring.

Deadlock

Raised when:	SQL deadlock is detected.
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Event Event alerts are raised for incidents that occur at a specific point in time; they do not change level, or update their status to Ended .
Possible causes:	<ul style="list-style-type: none"> • Inefficient application code. • Application accesses objects in a different order each time. • User input during transactions. • Lengthy transactions. • Locks not being released as early as possible.

Deadlock trace flag disabled

Raised when:	SQL Monitor is unable to turn on the deadlock trace flag on a SQL Server instance; this means no deadlock alerts can be raised.
Configurable thresholds:	None
Default settings:	Raised as High

Type:	Continuous Automatically updated from Active to Ended when SQL Monitor can enable the trace flag.
Possible causes:	Insufficient privileges for the account used to connect to the SQL Server instance. SQL Monitor requires sysadmin permissions on this account to turn on the deadlock trace flag.

Fragmented indexes

Raised when:	Both of the following conditions apply: <ul style="list-style-type: none"> Fragmentation of one or more indexes in a database exceeds a percentage threshold. The fragmented indexes contain more than a specified number of pages.
Configurable thresholds:	Percentage fragmentation level Indexes contain more than x pages
Default settings:	<ul style="list-style-type: none"> Raised as Low when fragmentation is above 15% Medium: not enabled High: not enabled <p>Raised for indexes with more than 1000 pages</p>
Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when the condition that caused the alert to be raised no longer applies - in this case, when there are no fragmented indexes (based on the thresholds defined) for the database.
Possible causes:	Regular deleting or updating existing rows or values in a table.
Notes:	<p>Checking for index fragmentation is a very resource-intensive activity.</p> <p>For this reason, SQL Monitor only checks for fragmented indexes once a week: on Sunday 02:00.</p> <p>This means that the alert may remain Active for some time after you fix the issue, and will only be updated after the next scheduled weekly check.</p>

Integrity check overdue

Raised when:	Either of the following conditions apply: <ul style="list-style-type: none"> No entry for an integrity check found following DBCC DBINFO WITH TABLERESULTS. The most recent entry for an integrity check is older than a specified time.
Configurable thresholds:	Most recent integrity check is older than x seconds/minutes/hours/days
Default settings:	<ul style="list-style-type: none"> Raised as Low after 7 days Medium: not enabled High: not enabled
Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when the condition that caused the alert to be raised no longer applies - in this case, when an integrity check of the database (DBCC CHECKDB) is detected.
Possible causes:	No integrity check has been carried out on a database, or the most recent integrity check was too long ago.

Job duration unusual

Raised when:	The job execution time is different from the baseline duration (the median of the last ten runs) by a specified percentage.
Configurable thresholds:	Percentage difference from baseline duration (either slower or quicker). Ignore jobs with run times less than x seconds.

Default settings:	<p>These defaults assume that once the baseline for a job is established, that job should not start running significantly more slowly or more quickly.</p> <ul style="list-style-type: none"> • Raised as Low when duration is 50% different to baseline • Escalated to Medium when duration is 60% different to baseline • Escalated to High when duration is 70% different to baseline • Ignore jobs that run for less than 2 seconds.
Type:	<p>Continuous</p> <p>Can have multiple thresholds applied and is automatically updated from Active to Ended when the job next completes with a duration that is within the allowed percentage of the baseline duration.</p>
Possible causes:	<ul style="list-style-type: none"> • High system load • Sub-optimal SQL execution plan • Job is waiting on an event or blocked
Notes:	<p>SQL Monitor calculates the baseline duration by using the job history to find the last ten run times.</p> <p>SQL Monitor will not raise the Job duration unusual alert until the job history contains at least ten runs.</p>

Job failed

Raised when:	Job does not complete successfully, and returns error code.
Configurable thresholds:	None
Default settings:	Raised as Medium
Type:	<p>Event</p> <p>Event alerts are raised for incidents that occur at a specific point in time; they do not change level or update their status to Ended.</p>
Possible causes:	Check the Job outcome message for a raised alert to help determine the problem.

Log backup overdue

Raised when:	<p>Either of the following conditions apply:</p> <ul style="list-style-type: none"> • No entry for a transaction log backup or a full backup of this database in the [msdb].[dbo].[backupset] system table. • The most recent entry for a transaction log backup or full backup of this database in the [msdb].[dbo].[backupset] system table is older than a specified time.
Configurable thresholds:	Most recent backup is older than x seconds/minutes/hours/days
Default settings:	<ul style="list-style-type: none"> • Raised as Low after 1 day • Medium: not enabled • High: not enabled
Type:	<p>Continuous</p> <p>Can have multiple thresholds applied and is automatically updated from Active to Ended when the condition that caused the alert to be raised no longer applies - in this case, when a transaction log backup is detected.</p>
Possible causes:	<ul style="list-style-type: none"> • No log backup job scheduled. • Log backup jobs not running or not completing. Check Job failed alerts on this SQL Server instance. • SQL Server Agent Service is not started - check for any SQL Server Agent Service status alerts.

Long-running query

Raised when:	Query has been running for longer than a specified duration.
Configurable thresholds:	<p>Query duration is longer than x seconds/minutes/hours/days</p> <p>Do not raise alerts for queries that contain certain strings (matching specified regular expressions).</p>
Default settings:	<ul style="list-style-type: none"> • Raised as Low after 1 minute • Escalated to Medium after 2 minutes • Escalated to High after 2 minutes 10 seconds

Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when the query completes.
Possible causes:	<ul style="list-style-type: none"> • Complex query • Insufficient physical memory • CPU over-utilized

Monitoring error (SQL Server data collection)

Raised when:	One of the following conditions applies continuously for 2 minutes: <ul style="list-style-type: none"> • Problems with WMI • Problems with the remote registry • File sharing issues • SQL connectivity issues
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended when data can be collected from the instance again.
Possible causes:	<ul style="list-style-type: none"> • One of the required services (WMI or remote registry) has been stopped. • Remote file access permissions have changed or hidden administrative shares have been disabled. • SQL Server Service has been stopped.

Monitoring stopped (SQL server credentials)

Raised when:	SQL Monitor is unable to collect monitoring data from the SQL Server because the credentials supplied to connect to the instance are invalid or lack permissions.
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended once the correct credentials are entered and authentication is confirmed.
Possible causes:	<ul style="list-style-type: none"> • Your user name or password has been changed. • Your permissions have changed and are no longer sufficient.

Page verification

Raised when:	PAGE_VERIFY is set to NONE (SQL Server 2005 or SQL Server 2008) or TORN_PAGE_DETECTION is set to FALSE (SQL Server 2000) for a database.
Configurable thresholds:	None
Default settings:	Raised as Low
Type:	Continuous Automatically updated from Active to Ended when SQL Monitor detects that Page verification has been turned on.
Possible causes:	New databases inherit this setting from the Model database. Check that Page Verify is turned on for the Model databases, if required.

SQL Server Agent Service status

Raised when:	SQL Server Agent Service status matches the status specified in the alert configuration.
--------------	--

Configurable thresholds:	Service status is one of the following: <ul style="list-style-type: none"> • Stopped • Stopped or paused • Started • Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	Continuous Automatically updated from Active to Ended when the service status changes to a status other than that specified.
Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started Check the list of component services on the relevant Windows machine.

SQL Server Analysis Service status

Raised when:	SQL Server Analysis Service status matches the status specified in the alert configuration.
Configurable thresholds:	Service status is one of the following: <ul style="list-style-type: none"> • Stopped • Stopped or paused • Started • Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	Continuous Automatically updated from Active to Ended when the service status changes to a status other than that specified.
Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started Check the list of component services on the relevant Windows machine.

SQL Server error log entry

Raised when:	An error message has been written to the SQL Server error log with a severity level above a specified value.
Configurable thresholds:	Error severity equal to or higher than x
Default settings:	<ul style="list-style-type: none"> • Raised as Low for severity level 17 or 18 • Raised as Medium for severity level 19 • Raised as High for severity level 20 or higher
Type:	Event Event alerts are raised for incidents that occur at a specific point in time; they do not change level, or update their status to Ended .
Possible causes:	Various Check the SQL Server error log entry area of the alert to see the error message text.

SQL Server Full Text Search Service status

Raised when:	SQL Server Full Text Search Service status matches the status specified in the alert configuration.
Configurable thresholds:	Service status is one of the following: <ul style="list-style-type: none"> • Stopped • Stopped or paused • Started • Started or paused
Default settings:	Raised as Medium when service is Started or Paused .

Type:	Continuous Automatically updated from Active to Ended when the service status changes to a status other than that specified.
Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started Check the list of component services on the relevant Windows machine.

SQL Server instance unreachable

Raised when:	The SQL Server instance cannot be reached by SQL Monitor because it is not running, or because of some other error (other than permissions).
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended when SQL Monitor can contact the SQL Server instance.
Possible causes:	<ul style="list-style-type: none"> • Host machine unreachable • SQL Server service failed • SQL Server service manually stopped Check the list of component services on the relevant Windows machine. Check for Machine unreachable alerts.

SQL Server Reporting Service status

Raised when:	SQL Server Reporting Service status matches the status specified in the alert configuration.
Configurable thresholds:	Service status is one of the following: <ul style="list-style-type: none"> • Stopped • Stopped or paused • Started • Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	Continuous Automatically updated from Active to Ended when the service status changes to a status other than that specified.
Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started Check the list of component services on the relevant Windows machine.

Host machine alerts

SQL Monitor raises the following types of alerts for problems on a host machine (Windows server):

Clock skew

Raised when:	The difference between the Base Monitor clock time and the monitored server clock time is greater than 15 seconds. (The Base Monitor is the server which is running the monitoring service)
Configurable thresholds:	None Note: This alert is always raised as High .
Default settings:	Time difference greater than 15 seconds.
Type:	Continuous Automatically updated from Active to Ended when the times are synchronized (within 15 seconds of each other).
Possible causes:	Server times across your network have not been fully synchronized.

Disk space

Raised when:	One of the following conditions apply, depending on how you configure the alert: <ul style="list-style-type: none">logical disk space used is above a percentage threshold, ORlogical disk space available is less than a fixed value.
Configurable thresholds:	You can configure this alert in two ways: <ul style="list-style-type: none">Used disk space percentage, orDisk space available (in MB or GB). Low disk space has lasted longer than x seconds .
Default settings:	Disk space available: <ul style="list-style-type: none">Raised as Low when less than 1GBEscalated to Medium when less than 800MBEscalated to High when less than 400MB For longer than: 120 seconds
Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when disk space is above the lowest defined threshold for at least the specified duration.
Possible causes:	<ul style="list-style-type: none">Database and log files may be growing too large without frequent backups.Other applications may be using the disk drive for file storage.
Note:	SQL Monitor only collects disk space data at 15-second intervals: <ul style="list-style-type: none">the minimum value for the duration is 15 seconds.when you configure the alert, change the duration value by 15 second increments.

Machine unreachable

Raised when:	The Windows server (host machine) does not respond to a Ping request from SQL Monitor.
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended when SQL Monitor can contact the host machine.
Possible causes:	<ul style="list-style-type: none">Host machine turned off or has suffered a problem.Ping request blocked by machine or a firewall. Check the log for the machine on the Monitored servers page.

Monitoring error (host machine data collection)

Raised when:	One of the following conditions applies continuously for 2 minutes: <ul style="list-style-type: none">Problems with WMIProblems with the remote registryFile sharing issues
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended when data can be collected from the host machine again.
Possible causes:	<ul style="list-style-type: none">One of the required services (WMI or remote registry) has been stopped.Remote file access permissions have changed or hidden administrative shares have been disabled.

Monitoring stopped (host machine credentials)

Raised when:	SQL Monitor is unable to collect monitoring data from the host machine because the credentials supplied to connect to the machine are invalid or lack permissions.
Configurable thresholds:	None
Default settings:	Raised as High
Type:	Continuous Automatically updated from Active to Ended once the correct credentials are entered and authentication is confirmed.
Possible causes:	<ul style="list-style-type: none"> Your user name or password has been changed. Your permissions have changed and are no longer sufficient.

Physical memory

Raised when:	One of the following conditions apply, depending on how you configure the alert: <ul style="list-style-type: none"> physical memory used is above a percentage threshold, OR physical memory available is less than a fixed value.
Configurable thresholds:	You can configure this alert in two ways: <ul style="list-style-type: none"> Used physical memory percentage, or Physical memory available (in MB or GB). Low physical memory has lasted longer than x seconds.
Default settings:	Physical memory available: <ul style="list-style-type: none"> Raised as Low when less than 400MB available Escalated to Medium when less than 200MB available Escalated to High when less than 100MB available For longer than: 60 seconds
Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when physical memory is above the lowest defined threshold for at least the specified duration.
Possible causes:	<ul style="list-style-type: none"> SQL Server has been configured with insufficient memory. Page file space running low. Other processes consuming physical memory. Not enough RAM on server.
Note:	SQL Monitor only collects physical memory data at 15-second intervals: <ul style="list-style-type: none"> the minimum value for the duration is 15 seconds. when you configure the alert, change the duration value by 15 second increments.

Processor under-utilization

Raised when:	Total processor utilization, averaged across all CPUs, is below a percentage threshold for longer than a specified duration.
Configurable thresholds:	Processor utilization less than a specified percentage. Under-utilization has lasted longer than x seconds.
Default settings:	Alert is disabled by default
Type:	Continuous Can have multiple thresholds applied and is automatically updated from Active to Ended when processor utilization is above the lowest defined threshold for at least the specified duration.
Possible causes:	Processor utilization is not as high as expected under normal operations: may indicate that SQL Server is not running normally or processing data - freeing up CPU.

Note:	<p>SQL Monitor only collects processor utilization data at 15-second intervals:</p> <ul style="list-style-type: none"> the minimum value for the duration is 15 seconds. when you configure the alert, change the duration value by 15 second increments.
-------	---

Processor utilization

Raised when:	Total processor utilization, averaged across all CPUs, is above a percentage threshold for longer than a specified duration.
Configurable thresholds:	<p>Processor utilization above a specified percentage.</p> <p>Utilization above this percentage has lasted longer than x seconds.</p>
Default settings:	<ul style="list-style-type: none"> Raised as Low when utilization is above 85% Escalated to Medium when above 90% Escalated to High when above 95% <p>For longer than: 60 seconds</p>
Type:	<p>Continuous</p> <p>Can have multiple thresholds applied and is automatically updated from Active to Ended when processor utilization is above the lowest defined threshold for at least the specified duration.</p>
Possible causes:	<ul style="list-style-type: none"> Other processes running on the server - check the System processes area of the raised alert. CPU-intensive SQL queries - if Profiler trace is turned on for the SQL Server, check the SQL statements in the SQL processes/Profiler trace area of the raised alert.
Note:	<p>SQL Monitor only collects processor utilization data at 15-second intervals:</p> <ul style="list-style-type: none"> the minimum value for the duration is 15 seconds. when you configure the alert, change the duration value by 15 second increments.

SQL Server Browser Service status

Raised when:	SQL Server Browser Service status matches the status specified in the alert configuration.
Configurable thresholds:	<p>Service status is one of the following:</p> <ul style="list-style-type: none"> Stopped Stopped or paused Started Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	<p>Continuous</p> <p>Automatically updated from Active to Ended when the service status changes to a status other than that specified.</p>
Possible causes:	<ul style="list-style-type: none"> Service failed Service manually stopped or started <p>Check the list of component services on the relevant Windows machine.</p>

SQL Server Integration Service status

Raised when:	SQL Server Integration Service status matches the status specified in the alert configuration.
Configurable thresholds:	<p>Service status is one of the following:</p> <ul style="list-style-type: none"> Stopped Stopped or paused Started Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	<p>Continuous</p> <p>Automatically updated from Active to Ended when the service status changes to a status other than that specified.</p>

Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started <p>Check the list of component services on the relevant Windows machine.</p>
------------------	--

SQL Server VSS Service status

Raised when:	SQL Server VSS Service status matches the status specified in the alert configuration.
Configurable thresholds:	<p>Service status is one of the following:</p> <ul style="list-style-type: none"> • Stopped • Stopped or paused • Started • Started or paused
Default settings:	Raised as Medium when service is Stopped or Paused .
Type:	<p>Continuous</p> <p>Automatically updated from Active to Ended when the service status changes to a status other than that specified.</p>
Possible causes:	<ul style="list-style-type: none"> • Service failed • Service manually stopped or started <p>Check the list of component services on the relevant Windows machine.</p>

Using the Alert Inbox

The Alert Inbox is an email-style inbox for all the alerts that SQL Monitor has raised. Like an email inbox, you can browse through the list, open a particular alert, mark it as read, or select multiple alerts at a time.

- The Alert Inbox updates automatically to check for new alerts every 30 seconds. You do not need to manually refresh your browser.
- By default all the alerts are pre-configured with sensible default values.

Alert Inbox : All Servers

Global filters: All alerts, Unread, Active, Cleared

Monitored servers: All Servers (52), sp.friday.testnet.re...

Filter: All | Clear | Advanced filter | Showing 125 of 101 | << Newest < Newer Older > Oldest >>

Mark as: Cleared Not cleared Read Unread Add Comment

Level	Type	Object	Status	Time	Actions
Medium	Physical memory	sp.monday.testnet.red-gate.com	Active	2:01 AM	
Low	Backup overdue	sp.monday.testnet.red-gate.co...	Ended	8:02 PM 14 May	
Low	Long running query	sp.monday.testnet.red-gate.co...	Ended	8:01 PM 14 May	
Low	Long running query	sp.monday.testnet.red-gate.co...	Ended	8:01 PM 14 May	

The status of each alert is shown in the **Status** column:

Level	Type	Object	Status	Time
Low	Log backup overdue	homer.test...	Active	10:05 PM 27 Oct
High	Deadlock	phoenix.te...	Event	8:26 PM 27 Oct
Low	Integrity check overdue	robin.red-g...	Ended	11:14 AM 27 Oct

Seeing more alerts

When you first install SQL Monitor, the Alert Inbox shows a maximum of ten alerts per page.

- To view the next ten alerts, click **Older**.
- To view the last page of alerts, click **Oldest**.

Showing 1-10 of 52 << Newest < Newer Older > Oldest >>

- To increase the number of alerts shown, use the **Rows to display** option below the inbox:

Rows to display: 10 Showing 1-10 of 52

Viewing alerts for a particular object

When you first click the **Alerts** tab, SQL Monitor always shows every uncleared alert across all your servers. Under **Monitored servers**, click on a group, host machine, SQL Server instance, or database. The Alert Inbox page title reflects the currently selected object:

Alert Inbox : Test Machines > mammoth.testnet.red-gate.com > sql2000

If you arrive at the Alert Inbox by clicking on a link from an overview page, the relevant object is pre-selected in the **Monitored servers** list. Clicking on **Uncleared alerts** in a database overview page, for example, will automatically select that database, and show only alerts related to it.

To change the list of alerts displayed, continue to drill down in the **Monitored servers** list. Alerts are shown for the currently selected level **and all levels below**. If you select a host machine, for example, you will see:

- alerts raised for the machine itself (such as disk space or memory alerts)
- alerts raised for all instances running on it (such as job failed or error log entries)
- alerts raised for all databases on those instances (such as fragmented indexes or backup overdue alerts)

The number in brackets after an object in the Monitored servers list always shows the number of **unread** alerts, regardless of the current filters applied. As you drill down in the hierarchy, the number of unread alerts at each level is displayed:

The screenshot shows a tree view of monitored servers. Red circles highlight the unread alert counts for several items, with red arrows pointing to explanatory text:

- All Servers (2948)**: total number of unread alerts
- Test Machines (1124)**: unread alerts in this group
- homer.testnet.red-gate.com (185)**: unread alerts for this server, all its instances and the databases on each instance
- sqlexpress (8)**: unread alerts for this instance and all its databases
- tempdb (2)**: unread alerts for just this database

The list of alerts is also filtered by whatever is set in the **Filter** drop-down.

How alerts are sorted in the Inbox

Alerts are sorted by the time they were raised, or for continuous alerts, whenever they changed level or were updated to **Ended**.

The screenshot shows a table of alerts. The first row is 'Ended' with a time of '2:11 PM'. A mouse cursor is clicking on the 'Active' row, which has opened a dropdown menu. The menu items are:

- 🔔 Raised Low 2:04 PM
- ⬆️ Escalated High 2:11 PM
- ⏹️ Ended - 2:13 PM

The displayed time may differ from the time used to sort the list. In the above example, the displayed alert time shows 2:11 rather than 2:04, as this is when the alert reached its highest level.

An alert's timestamp is **not** updated when:

- An alert changes to Ended
- An alert is cleared
- An alert downgrades to a lower level
- An alert escalates to a level that it has already previous reached

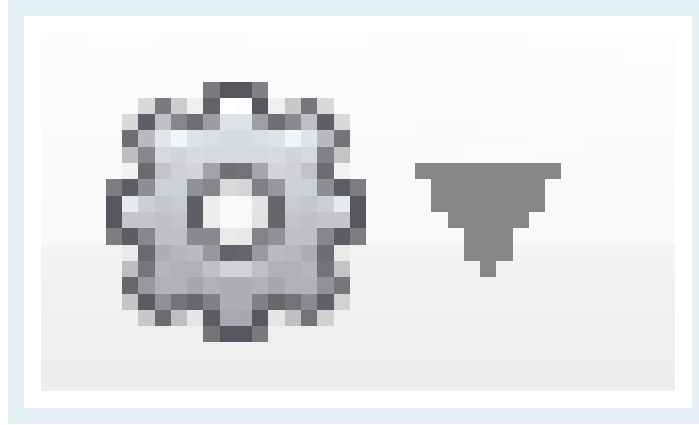
Actions you can apply to alerts

To:	Do any of the following:
------------	---------------------------------

Open an alert, in order to view its details on a separate page

Click anywhere in the row for the alert.

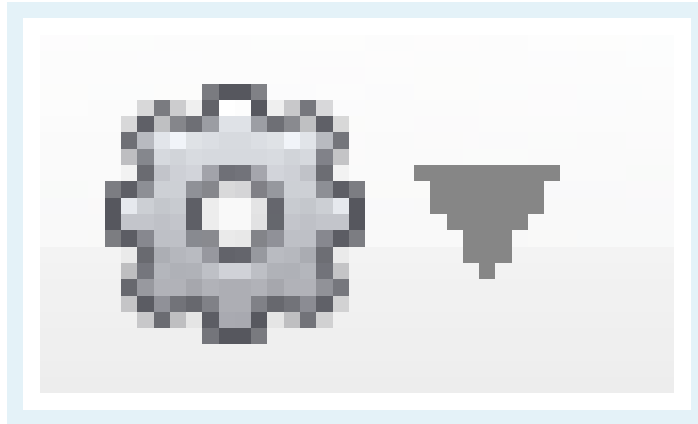
Click **Open** from the Actions list



Mark an alert as read

(only available for alerts that are currently unread)

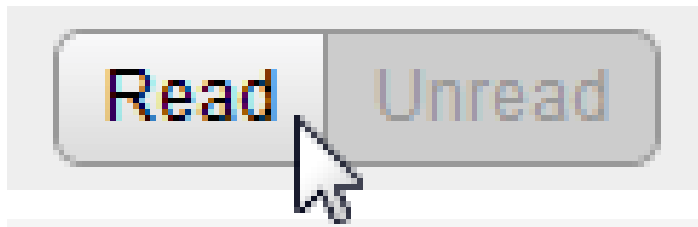
Click **Mark as read** from the Actions list



Select the alert using its check box:



then click the **Read** button:



This is useful when selecting multiple alerts to mark as read.

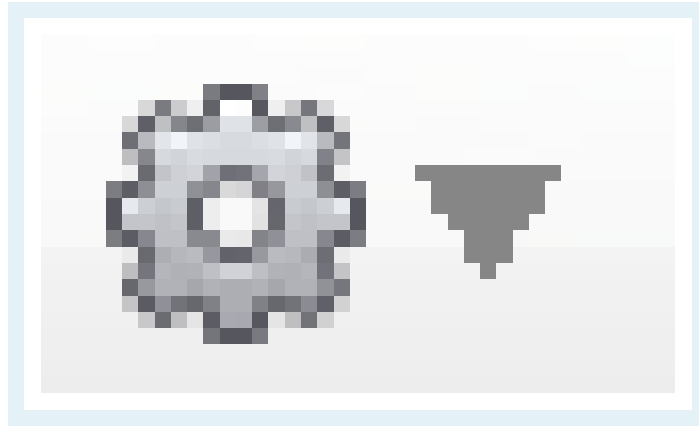
Open the alert: alert is automatically marked as read.

Clear the alert: alert is automatically marked as read.

Mark an alert as unread

(only available for alerts that are currently read)

Click **Mark as unread** from the Actions list



Select the alert using its check box then click the **Unread** button.

This is useful when selecting multiple alerts to mark as unread.

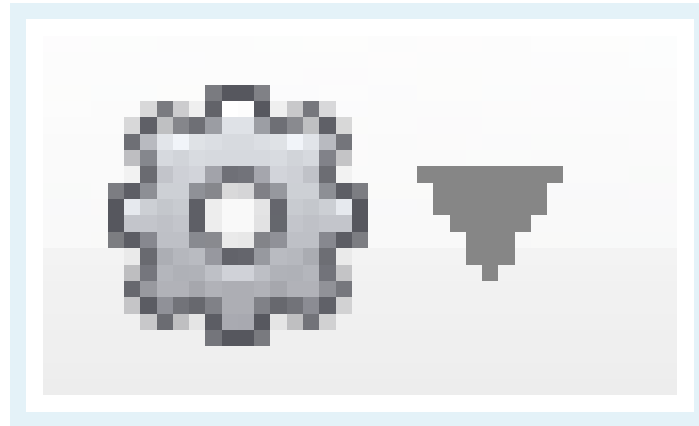
Clear an alert

(Cleared alerts are hidden from the inbox by default)

Click **Clear** in the alert row:



Click **Clear** from the Actions list



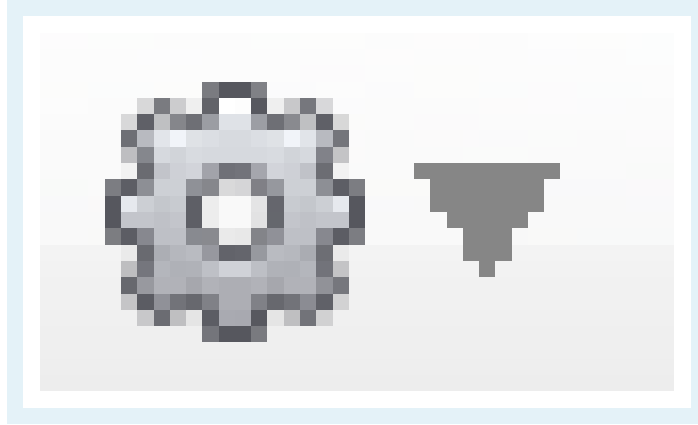
Select the alert using its check box then click the **Cleared** button.

This is useful when clearing multiple alerts.

Add a comment

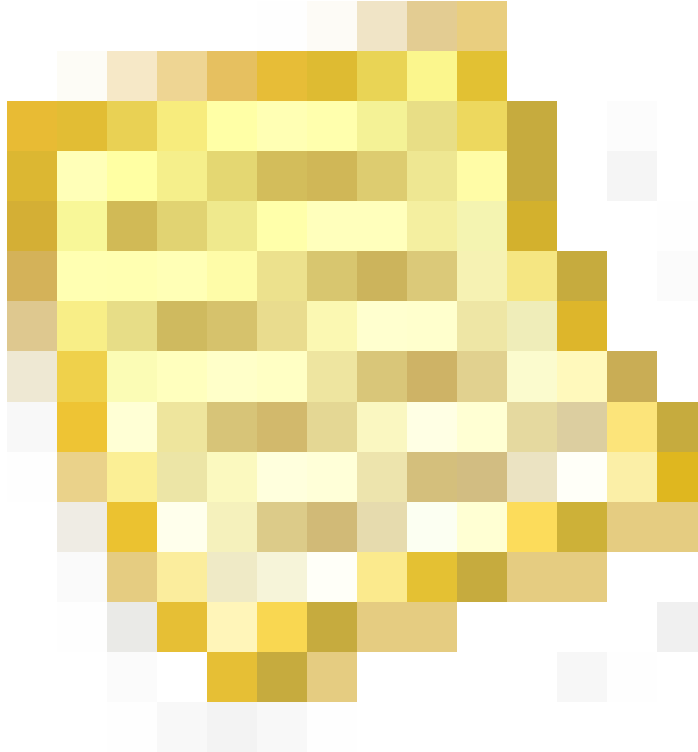
Open the alert and select the Comments tab, then use the **Add comment** button.


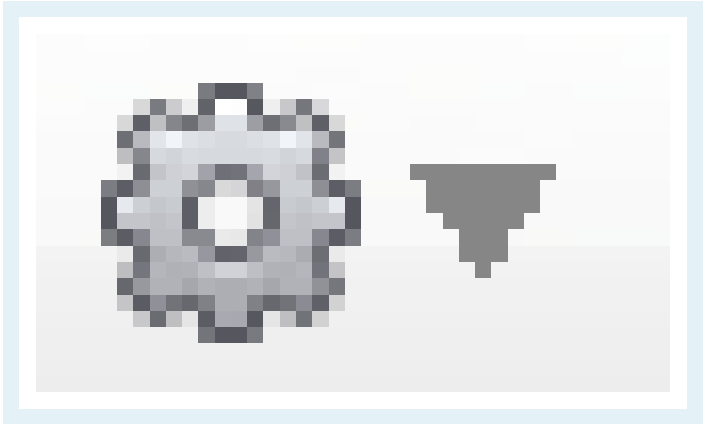
Click **Add comment** from the Actions list



Select the alert using its check box then click the **Comment** button. You can add a comment to multiple selected alerts.

Comments are displayed in the Alert Inbox. Only the most recent comment for each alert is available. To display a comment, hold the mouse pointer over the Comments icon



<p>Select one or more alerts</p>	<p>Select the check box</p>  <p>next to each alert.</p>
<p>Configure an alert</p>	<p>Click Configure from the Actions list</p>  <p>This takes you to the relevant alert settings page for the selected alert type and object.</p> <p>Under Actions on the left side, click the Configure alerts link to go to the Alert settings main page, where you can adjust the settings for any alert at any level. See Customizing alerts.</p>

Selecting alerts

Selecting alerts is a useful way to apply actions to multiple alerts at once, rather than having to perform the action for each alert in turn.

Turn on the check box

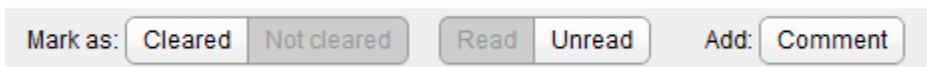


next to each alert you want to select. Use Shift + click to select a block of alerts.

Selected alerts are highlighted in the inbox by row color.

Applying actions to selected alerts

Use the command bar above the Alert Inbox to apply actions to all the currently selected alerts:



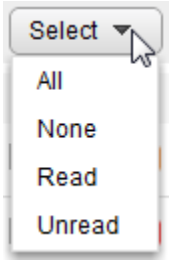
- If you select **Add Comment**, the same comment will be added to all the selected alerts.
- Clearing or reading a selection of alerts may remove them from the Alert Inbox, depending on the current filter settings.

Inline commands and actions from the Actions drop-down menu are only ever applied to the single alert:



Using the Select button

Use the **Select** drop-down button to select large groups of alerts at once:



The Select option **only applies to the current filtered view of your alerts**. For example, if you are currently filtering the Alert Inbox to show only Medium Job failed alerts, and choose **Read**, then only Medium Job failed alerts that are marked as read are selected.

The Select drop-down list is context sensitive. If you are currently filtering the Alert Inbox to view only unread alerts, the **Unread** option will not be available.

Selecting alerts across multiple pages

When there are several pages of alerts, only the current page of alerts is selected by default. You can choose to apply the selection to all alerts on all pages:

All unread alerts on this page are selected. [Select all unread alerts across all pages](#)

If you choose this option, the message is updated:

All unread alerts on all pages are selected. [Undo selection](#)

Use the **Older** and **Newer** links to confirm that alerts are selected on other pages. Clicking **Undo selection** will undo the selection on all pages except the page where you first applied the selection.

Any actions you apply at this point, for example Mark as read, Clear, or Add comment will be applied to all selected alerts on all pages.

Filtering alerts

You can filter which alerts appear in the Inbox by any of their attributes:

- by level (Low, Medium, High)
- by status (Event, Active, Ended, Cleared)
- by type (e.g. Job failed, Backup overdue, or any selection)
- by action taken (Read, Unread, Commented, No Comment)
- by time

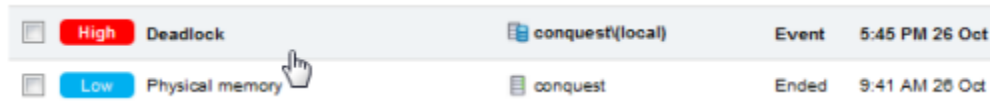
SQL Monitor has built-in filters for the most common options you may want. You can also create your own filters using the Advanced filter pane.

See [Filtering alerts](#).

Viewing details of an alert

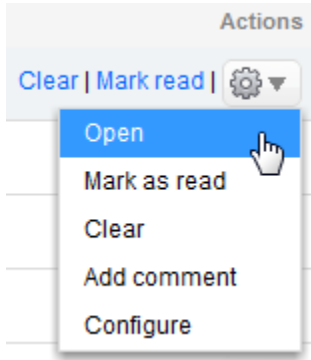
To view the details of raised alert, from the Alert Inbox, do one of the following:

- Click anywhere in the row for that alert:



<input type="checkbox"/>	High Deadlock	conquest(local)	Event	5:45 PM 26 Oct
<input type="checkbox"/>	Low Physical memory	conquest	Ended	9:41 AM 26 Oct

- Select **Open** from the Actions drop-down menu:



The alert is opened in its own page. When you open an alert, it is automatically marked as read.

To open the alert on a separate page, use your browser's **Open in new window** context menu command. (This only works for the Actions menu **Open** command.)

What information is provided for an alert?

Alert details are organized into three main areas, from top to bottom:

Summary pane

Status: **Ended** Time raised: **27 Oct 2010 1:53 PM** Time ended: **29 Oct 2010 10:59 AM**

Shows the current status of the alert and its raised time and (if applicable) its ended time.

Alert details area





Details	Database files	Comments	Alert history	Occurrences	Description
Disk space used: 0.00GB (1.20%)					
Disk space available: 0.24GB (98.80%)					
Total disk size: 0.24GB					

The alert details area is organized into several tabs:

Details: general data showing the nature of the issue.

Comments: all comments that have been added for this occurrence of the alert.







Alert history: the lifecycle of this occurrence of the alert, displayed as a list of entries representing each significant change:

 Raised	Low	1:55 AM
 Escalated	High	1:56 AM
 Ended	-	2:06 AM
 Cleared	-	2:56 PM

For continuous alerts that are frequently being escalated and downgraded over time, there may be a long list of entries. The maximum number of entries displayed in the Alert history page is 200. In the unlikely event that the number of entries exceeds this limit, the initial 100 entries and the most recent 100 entries are displayed.

Occurrences: lists all alerts of this type raised for the current object, in order of time raised. Click on any occurrence to view its details.

6 occurrences (most recent first)

	Low	15 Oct 2010 at 03:21
	Low	15 Oct 2010 at 03:21
	High	15 Oct 2010 at 02:49
	High	15 Oct 2010 at 02:29
	High	15 Oct 2010 at 02:16
	High	15 Oct 2010 at 01:59

Description: explanation of this alert, when it is raised, and links to further information.

Some alert types have an additional tab that contains more detail about the issue or related information.

Performance data area

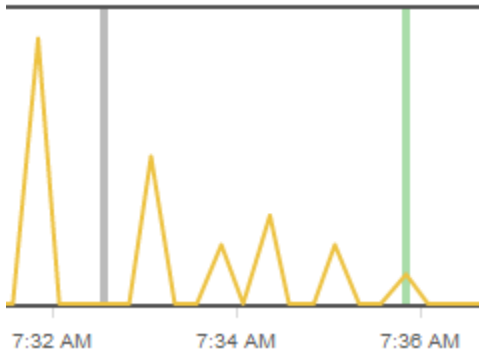
The performance data area shows various performance counter values and process information captured around the time the alert was raised.



Host machine: shows graphs for metrics relevant to the host Windows machine (e.g. processor time, disk queue length).

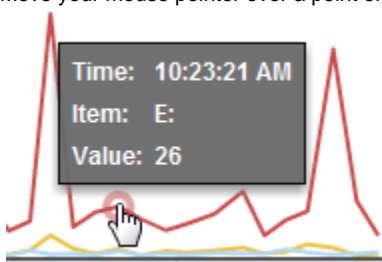
SQL Server: shows graphs for metrics relevant to the SQL Server instance (e.g. user connections, buffer cache hit ratio).

The alert raised time is represented by the vertical gray line. For alerts that have ended, the ended time is shown as a green line:



For each graph, you can do the following:

- Click anywhere on the graph to go to the **Analysis** page, with the metric and object pre-selected, so you can expand the time range.
- Move your mouse pointer over a point on the graph to read the value:



The start and end times of all these graphs relative to the alert raised time will vary depending on the nature of the alert and when SQL Monitor last collected monitoring data.

System processes: A snapshot of the processes that were running at the time the alert was raised. The processes are sorted by name.

SQL processes/Profiler trace: A snapshot of the SQL processes that were running at the time the alert was raised.

If you have enabled trace on the server on which this alert was raised, you can click on a SQL process in the list to view the detailed trace output:

65	SQL Monitor - Repository
75	SQL Monitor - Repository
54	Microsoft SQL Server Management Studio - Query
52	SQL Monitor - Repository

Time	Statement
29 Oct 2010 16:37:24	<pre>RAISERROR (N'Message text.', -- Message text. 18, -- Severity, 1, -- State, N'number', -- First argument. 5) with log; -- Second argument. -- The message text returned is: This is message number 5.</pre>
29 Oct 2010 16:37:32	<pre>RAISERROR (N'Message text.', -- Message text. 18, -- Severity, 1, -- State, N'number', -- First argument. 5) with log; -- Second argument. -- The message text returned is: This is message number 5.</pre>

Top 10 expensive queries: This tab is available for SQL Server alerts only. It displays the top 10 queries that used the most resource for a 15 minute period, starting 10 minutes before the alert was triggered.

Note: This feature is not available for servers running on Microsoft SQL Server 2000.

The following query data is displayed:

- Execution count - the number of times this query statement was executed. By default, queries are listed in descending order according to this metric.
- Duration - how long it took in milliseconds to execute the query.
- CPU time - how much processor time in milliseconds was used to execute the query.
- Physical reads - the number of times a page is read into the buffer cache. If the page is in the cache already, it uses the page already in memory and does not generate a physical read.
- Logical reads - the number of times the database engine requested a page from the buffer cache.
- Logical writes - the number of times data is modified in a page in memory. If a page stays in memory for an extended period, more than one logical write may be required before it is physically written to disk.

The database that each query was run against is displayed in the Database column.

You can also update the list as follows:

- Click **Totals** to display queries based on total values.
- Click **Avg. per execution** (selected by default) to display queries based on average metrics over the time period selected. Note: This does not affect Execution count which always displays total values.
- Click on a different column heading to display queries based on that metric in descending order.

Selecting one of these options does not simply change the sort order of the existing list of queries. A new table is generated according to the selected option, so different queries are likely to be displayed.

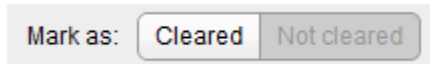
Why is there no trace data?

- The selected SPID may not have any trace output associated with it.
- You have not enabled Profiler trace for this server. See [Configuring Profiler trace](#).

Working with raised alerts

When you have read the details of a raised alert, you may want to do one of the following:

- **Clear the alert**



Clearing an alert hides it from the default "All" filter in the Alert Inbox. Clear alerts that you have investigated, or are not important to you. Cleared alerts are not included in the alert summary counts on the overview pages.

To view cleared alerts in the Alert Inbox, use the **Cleared** filter in the Advanced filter pane.

- **Add a comment**



Adding a comment allows you to make a note about this occurrence of the alert.

Add comment ✕

Author:

Comment:

This query needs optimizing. Passed to dev team on 11/10/10 for analysis.

The comment is displayed in the Alert Inbox. Move your mouse pointer over the comment to view a tooltip containing the full comment text:

Comment	Actions
<p>This query needs optimizing. Passed to de...</p>	<p>Clear Mark unread ▼</p>

This query needs optimizing. Passed to dev team on 11/10/10 for analysis.

Only the most recent comment for each occurrence is displayed in the Alert Inbox. To view all alerts with comments, use the **Has comment** filter in the Advanced filter pane.

You can add further comments to an alert, and delete or edit an existing comment. Open the alert, and use the **Comments** tab to manage comments:

Details
Comments
Alert history
Occurrences

Mike D

Nov 3 2010 at 11:26 AM

This query needs optimizing. Passed to dev team on 11/10/10 for analysis.

[Edit](#) | [Delete](#)

If you edit an existing comment, the timestamp for the comment is not updated.

- **Customize the alert**

Customizing the settings for an alert is useful if alerts are being raised that you do not want to be alerted about in future, or if you want to adjust the thresholds that trigger the alert, or set up email notifications to be sent to colleagues. Click **Configure alert**:

Low **Physical memory** | srp-monday.testnet.red-gate.com

Status: **Ended** Time raised: 14 May 2012 3:38 PM Time ended: 14 May 2012 3:39 PM

Alert Inbox Newer | Older Mark as: Cleared Not cleared Read Unread Add: Comment **Configure alert...**

SQL Monitor takes you straight to the configuration settings page for the specific object, but you can choose to configure the alert at any level in the hierarchy. See [Configuring alerts](#).

Filtering alerts

You can filter the alerts that are displayed in the Alert Inbox in two basic ways:

- by the selection in the **Monitored servers** list
- by alert attributes (using system or custom filters)

These two methods of filtering the Alert Inbox always work in combination.

The Monitored servers list

The Monitored servers list controls the objects for which alerts are displayed. For example, if you select a database in the list, then only alerts relating to that database are listed.

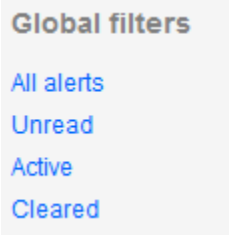
The current object selection is reflected in the page heading:

Alert Inbox : Personal Test Machines > abyss.testnet.red-gate.com

In this example, the Alert Inbox is displaying only alerts on a selected host machine. The heading includes the full path to the selected object, showing where it is in the hierarchy of groups, machines, instances and databases.

Global filters

The Global filters are available above the Monitored servers list. They are a quick way of getting back to a sensible starting point.



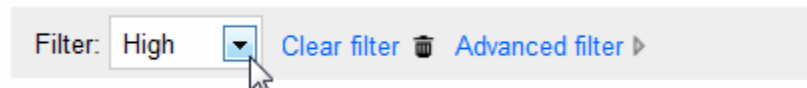
The Global filters are shortcuts that carry out two actions at once:

1. Reset the Monitored servers list back to All Servers.
2. Apply a system filter to the **Filter** drop-down list.

The **All alerts** filter excludes cleared alerts. Cleared alerts are hidden by default so they do not clutter up your Alert Inbox.

System filters

The **Filter** drop-down above the Alert Inbox contains a number of predefined filters, covering typical scenarios for filtering the Alert Inbox. For example, you can view only Low, Medium, or High level alerts, or see all Active alerts.



As soon as you select a filter, the Alert Inbox is updated.

Select the **Cleared** filter to view all cleared alerts. All the other filters (including **All**) hide cleared alerts from the inbox.

Custom filters

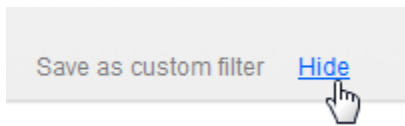
Click **Advanced filter** to expand the filter panel, where you can apply any combination of filter options you want.

Filter: High

Alert types All alert types selected	Level <input checked="" type="checkbox"/> High <input type="checkbox"/> Medium <input type="checkbox"/> Low	Status <input checked="" type="checkbox"/> Event <input checked="" type="checkbox"/> Active <input checked="" type="checkbox"/> Ended
Time <input checked="" type="radio"/> From the last: (no filter) <input type="button" value="v"/> <input type="radio"/> Between: Any time and: Any time	Properties <input checked="" type="checkbox"/> Read <input type="checkbox"/> Cleared <input checked="" type="checkbox"/> No comment <input checked="" type="checkbox"/> Unread <input checked="" type="checkbox"/> Not cleared <input checked="" type="checkbox"/> Has comment	

To collapse the filter panel:

- Click **Advanced filter** again, or
- Click the **Hide** hyperlink at the bottom right corner of the panel



The Advanced filter panel allows you to filter alerts in a number of ways:

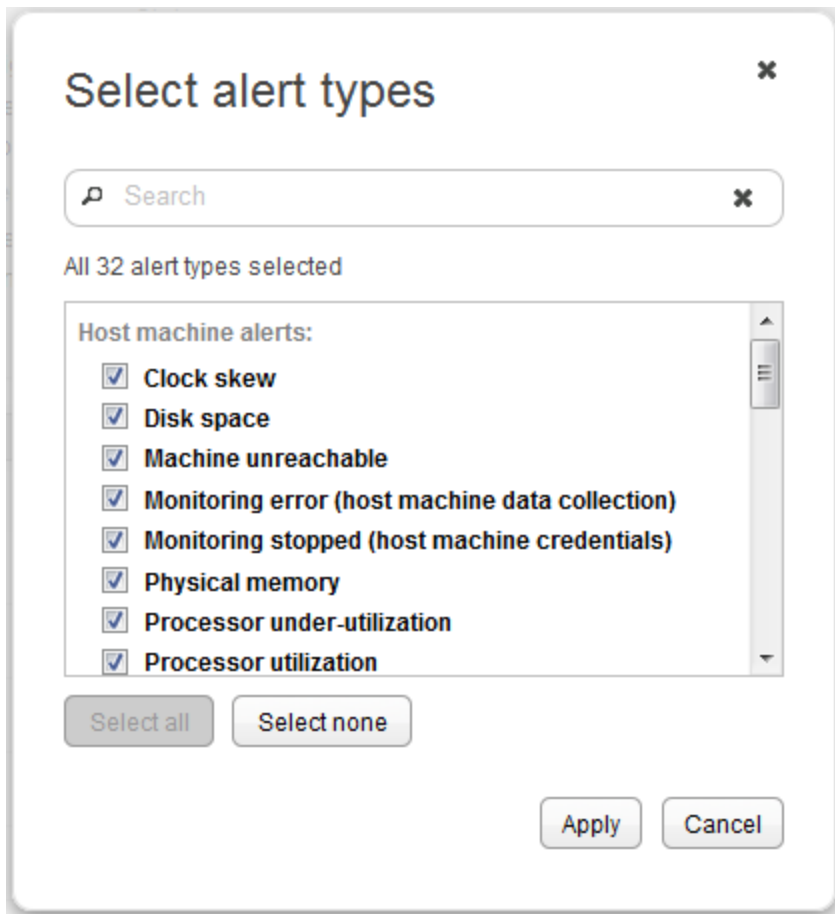
- By type
- By time (when the alert was raised)
- By level
- By status
- By properties

All these options work in combination. You can be very specific about which alerts you want to see, for example, High level deadlocks that are unread in the last hour, or all Active alerts with comments except Low level alerts and Job failed alerts between last night and today.

You can then save and apply these custom filters to all objects or selected objects only.

Filtering by type

Click **All alert types selected**. The Select alert types box is displayed:

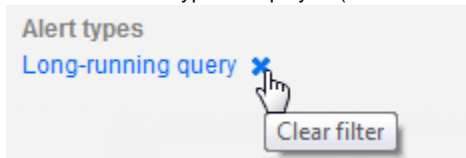


Select the check box for each alert type you want to show in the Alert Inbox. Use the **Select none** button first to clear the selection, if you only want to choose one or two types to display.

If you are familiar with the alert names, type the first few characters of an alert name in the **Search** box. Only matching alerts are displayed. (You may need to choose **Select none** and then **Select all** if you want to select only the matching alerts.)

As soon as you click **Apply**, the Alert Inbox is filtered by your selection, and the Advanced filter panel is updated:

- The selected alert type is displayed (or the number of types selected, if more than one).



- Click the **x** to remove the alert type filter; the link is reset to **All alert types selected**.
- You can see how many alerts match the current filter options by looking at the **Showing....** number.



Filtering by time

The time filter applies to the time each alert was raised. Under **Time**, select how you want to filter the alerts:

- **From the last:** select a fixed time period from the drop-down list
- **Between x and y:** click in each box to choose a date and time. You can use only one box if required, for all alerts that were raised up to a fixed point in time, or all alerts starting from a fixed point in time.
Note: Alerts will be shown that were raised in the minute matching the "End" time; for example, if you want to view alerts between 10:00 and 14:00, alerts that were raised up till 14:01 will be shown.

Click anywhere in the alert panel outside of the boxes to apply the time filter options. The number of alerts is updated instantly, and the **Showing..** number changes accordingly.

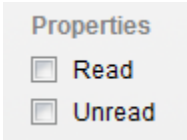
Click the **x** to remove the time filter (reset it to "Any time").

Filtering by other properties

Choose the required check boxes under **Level**, **Status**, and **Properties** to filter the Alert Inbox. As soon as you clear or select a check box, the list of alerts is updated accordingly to match your selection.

Why have all the alerts vanished from the list?

You can select combinations of filter options that will always result in zero matching alerts:



The image shows a grey rectangular box with the title "Properties" in blue. Below the title are two checkboxes, each with a small square icon to its left. The first checkbox is labeled "Read" and the second is labeled "Unread". Both checkboxes are checked, with a small dark square inside each box.

In this example, you are telling SQL Monitor that you don't want to see alerts that are either Read or Unread. However, all alerts must be either Read or Unread; therefore no alerts will ever match this set of options.

Similarly, you can choose both **Has comment** and **No comment** or **Cleared** and **Not cleared**. In each case, no alerts can possibly match the selection.

The Alert Inbox may also be empty because no alerts happen to match a legitimate combination of options.

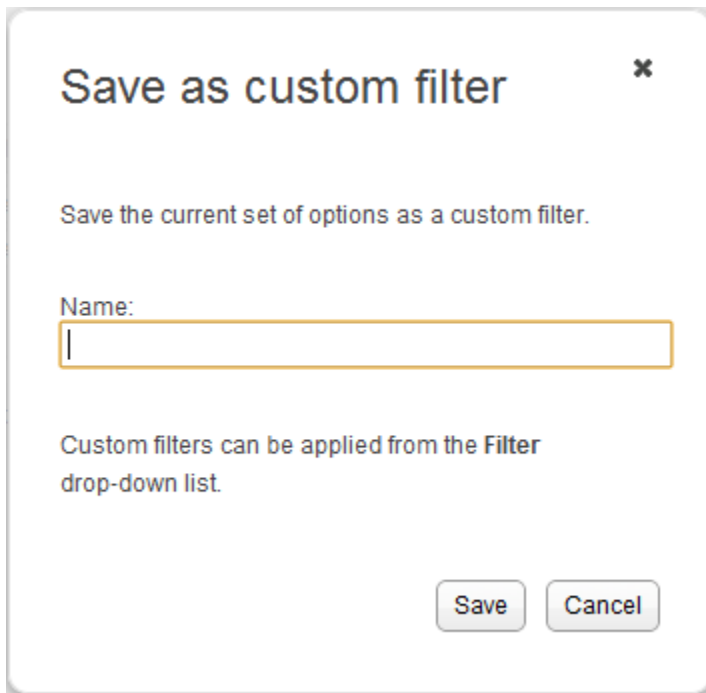
Saving advanced filter options as a custom filter

When you have selected the advanced filter options you want, you can save them as a custom filter.

This allows you to quickly reapply all the filter options you have selected, without having to reselect them again each time. You can save any number of custom filters, and this may help you prioritize which alerts you investigate - Deadlocks first, followed by all High alerts except deadlocks, then all alerts within the last two hours, and so on.

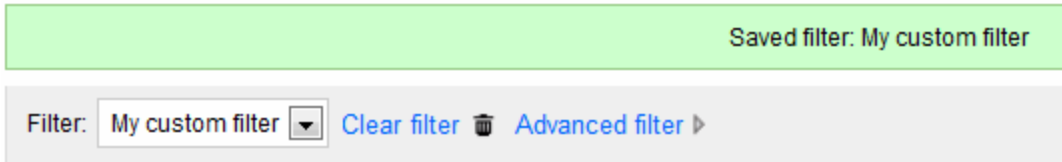
To save your filter options as a custom filter:

1. Click the **Save as custom filter** hyperlink. The Save as custom filter box is displayed:



The image shows a dialog box titled "Save as custom filter" with a close button (x) in the top right corner. Below the title is the instruction "Save the current set of options as a custom filter." followed by a "Name:" label and an empty text input field. Below the input field is the text "Custom filters can be applied from the Filter drop-down list." At the bottom of the dialog are two buttons: "Save" and "Cancel".

2. Type a name for the custom filter. You cannot use the name of a system filter.
3. Click **Save**. Your custom filter is now available from the **Filter** drop-down list:



To delete a custom filter

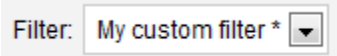
1. Select the filter you want to delete from the **Filter** list.
2. Click the delete button



The Filter list defaults back to All, but the custom filter options are persisted until you explicitly select something else.

To edit a custom filter

1. Select the filter you want to edit from the **Filter** list.
This filter is immediately applied to the Alert Inbox.
2. Click **Advanced filter** to display the advanced filter pane, showing the saved options for this filter.
3. Edit the options. An asterisk * next to the custom filter name identifies it as being modified:



4. Select **Save as custom filter**. The filter name is pre-filled in the **Name** box:
 - To save the edited custom filter, leave the existing name and click **Save**.
 - To create a new custom filter, enter a new name for the custom filter. The previous custom filter remains unchanged, and a new custom filter is created.

Configuring alerts

SQL Monitor provides sensible factory defaults for alert types so that alerts can be raised as soon as you've installed the application. You can use the **Alert settings** page (Configuration > Alert settings) to customize individual alert types for specific jobs, disks, databases, servers, clusters and groups.

How are alert settings applied?

Alert settings are applied in a hierarchy; an object inherits its alert settings from the level above in the hierarchy. For example, a database inherits its settings from the instance, which in turn inherits settings from the host machine, which inherits from the group to which it belongs.

When you first add a server to monitor, it inherits default alert settings for every type of alert from the All Servers level, which is the highest level in the hierarchy.

What can be configured?

For each type of alert, you can:

- disable it, so the alert will not be raised in future
- change the level at which it is raised, to either Low, Medium, or High
- change the thresholds that raise an alert (certain alert types only)
- change the email recipient

Each type of alert can be customized for each object you are monitoring (for example, a specific server, job, or database). You can also customize alerts across groups of objects (all databases on an instance, all servers in a group), or change everything at once using the All Servers setting.

To set up and configure global email notification settings, see [Setting up email notification](#).

Configuring alert settings

Go to the **Configuration** tab. Under **Alerts and metrics**, select **Alert settings**:



Alerts and metrics

Alert settings

Enable and disable alert types; change alert thresholds and levels.

Email settings

Set up and manage email notification for alerts.

Custom metrics

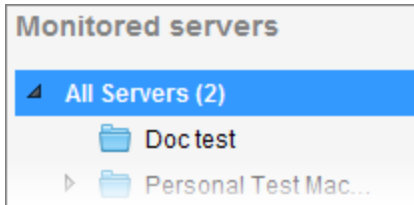
Create and manage custom metrics and custom alerts.

To configure an alert type for a group, host machine, SQL Server instance, database or job, [drill down](#) to select the required object from the **Monitored servers** list. The alert types relevant for the selected object are listed. For example, if you select a database, only the six database-specific alert types are displayed.

Click on the alert type in the list that you want to edit and follow the instructions in *Customizing settings* below.

Changing the default alert settings (the All Servers level)

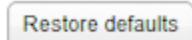
By default, when you first go to the Alert settings page, you are at the All Servers level. Changing the settings at the All Servers level applies the settings to everything you're monitoring:



Click on the alert type in the list that you want to edit.

Note: Changing alerts at the All Servers level does not affect any alerts you have already customized at a lower level (for a specific server or database, for example).

Click **Restore defaults**

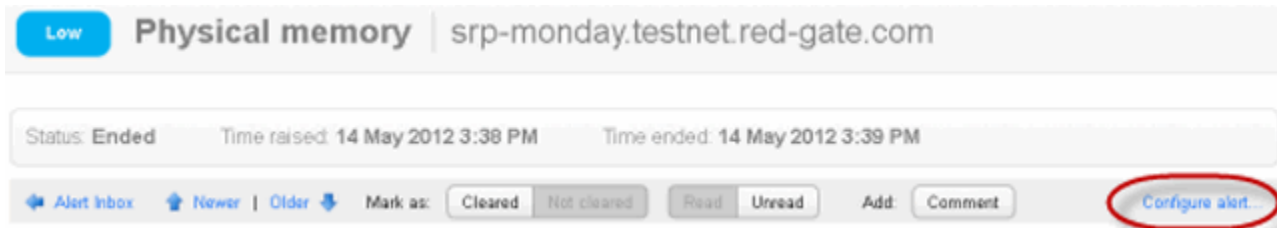


to reset an alert type back to its factory settings.

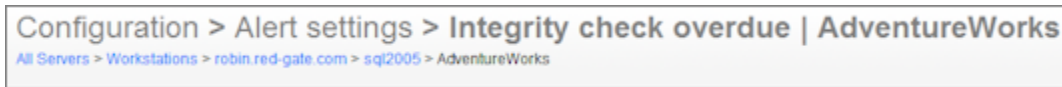
Configuring the settings for a raised alert

When you click **Configure alert...** for a raised alert (from the Alert Inbox or from an individual alert page), SQL Monitor takes you straight to the alert configuration settings page for that object.

For example, configuring an **Integrity check overdue** alert will take you to the alert settings page for the specific database for which the alert was raised:

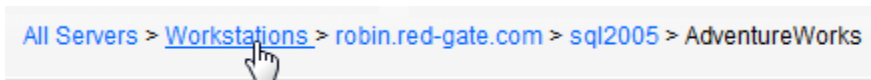


On the alert settings page, the page title and hyperlinked breadcrumbs beneath identify exactly what settings are displayed:



In the example above, the alert settings page is for the AdventureWorks database on the sql2005 instance only. If you edit the settings at this level, you will be changing the alert for just that database.

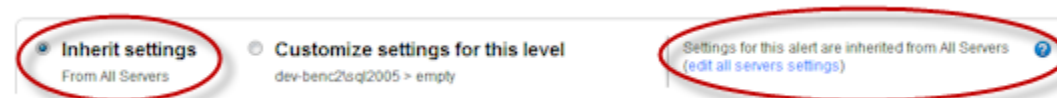
To edit this alert so that it applies to a broader scope of objects, click on the appropriate higher level in the breadcrumbs. For example, to edit the settings for all databases in the "Workstations" group:



The page title is updated to reflect the fact that you are editing the alert settings for a group.

Customizing settings

The level from which the selected object currently inherits its settings for this alert type is displayed at the top of the page. The rest of the configuration options are grayed out.



Select **Customize settings for this level** to configure this alert type for the current level and all levels beneath. Once selected, the other options on this page become available. After you have customized a particular level, this level *will not* inherit subsequent changes made to the levels above, including changes applied at the All Servers level.

Disabling an alert

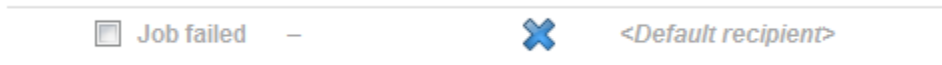
Every alert type (except Processor under-utilization) is enabled by default for all servers when you first install SQL Monitor. This should ensure that you're aware of any potential problems with your servers.

Select **Disabled** to stop raising alerts of the current type for the selected object. This will prevent any future occurrences of this type of alert appearing in the Alert Inbox when the conditions that would have triggered this alert are encountered. The rest of the configuration options on the page are grayed out.

Click **Apply changes** near the bottom of the page. The alert is grayed out in the Alert settings page and the **Enabled** column displays a



disabled icon:



If you disable an alert type when there are currently Active alerts of that type, these alerts are automatically updated to Ended when SQL Monitor next polls your servers.

Setting alert levels

All alert types can be raised at either Low, Medium or High, depending on how critical they are to your systems. You can change this level to suit your priority levels.

For example, if a SQL deadlock occurs, by default the **Deadlock** alert is raised as a High alert because in most cases, a DBA would treat this as priority and want to investigate it immediately. For some servers, however, you may want to drop the level to Medium or Low, so that you can filter them out when viewing your most critical High level alerts.

Alert level

When a SQL deadlock occurs, raise this alert at the following level:

High

Medium

Low

Setting alert thresholds

Some types of alerts have multiple thresholds that you can match to Low, Medium or High level. The alert is raised when it passes the lowest-defined threshold, and can then escalate to a higher level when it passes another threshold. You do not have to enable all these levels.

In the example below, the **Long-running query** alert is configured to be raised as a Low alert after 45 seconds, be automatically escalated to a Medium alert after 90 seconds, and then to High after 2 minutes. When the query finally ends it will be marked as Ended.

Alert thresholds

Raise this alert when a query runs longer than:

<input checked="" type="checkbox"/>	High	2	minutes
<input checked="" type="checkbox"/>	Medium	90	seconds
<input checked="" type="checkbox"/>	Low	45	seconds

In the Alert Inbox, the alert level will change automatically as the different thresholds are passed. This allows you quickly to keep track of how

serious this problem is, and prioritize queries for investigation that have escalated to High.

In the example below, only the Medium level has been enabled for the **Blocked process** alert. In this case, no alert will be raised until the block lasts for at least 45 seconds, and the alert will be raised as Medium. It will remain as a Medium alert however much longer the block lasts. When SQL Monitor detects that the block is over, the alert is marked as Ended.

Alert thresholds

Raise this alert when a SQL process is blocked for longer than:

<input type="checkbox"/>	High	2	minutes
<input checked="" type="checkbox"/>	Medium	45	seconds
<input type="checkbox"/>	Low	30	seconds

You can see that the thresholds are multiples of 15 seconds in both examples. SQL Monitor only collects data about most problems on your servers every 15 seconds.

Configuring email notification

Until you set up your mail server settings in SQL Monitor, no emails will be sent, regardless of the settings you apply on any Alert settings page. Click **Configure email settings** to enable email notification across the application. See [Setting up email notification](#).

Notification

Each time this alert is raised, SQL Monitor can send an email to one or more recipients. ?

Use default email recipient Specify an email recipient

Send email notifications to:

developers@mycompany.org

Use a comma to separate multiple email addresses.

[Configure email settings](#)

There are two options for sending email notifications:

- **Use default email recipient**
Sends email notifications for this alert to the email address entered in the **Send emails to** box on the **Email settings** page. If you edit the global email address on the Email settings page, then this alert will automatically pick up the new global setting.
- **Specify an email recipient**
Customize email notifications for this alert:

To disable emails for this alert, clear the **Send email notifications to** box.

To send emails for this alert to one or more recipients, type the email addresses you want in the box.

Configuring multiple alerts

To configure multiple alert types simultaneously, select them using the check boxes or the **Select** drop-down list, and click **Configure alerts**:

Alert type	Inherited from ?	Enabled	Email notification ?
<div style="display: flex; justify-content: space-between; border-bottom: 1px solid #ccc; padding-bottom: 5px;"> Select ▼ Configure alerts... Remove overrides </div>			
<div style="background-color: #e6f2ff; padding: 2px;"> ▲ All SQL Server alerts </div>			
<input checked="" type="checkbox"/> Backup overdue	–	<input checked="" type="checkbox"/>	<Default recipient>
<input checked="" type="checkbox"/> Blocked process	–	<input checked="" type="checkbox"/>	<Default recipient>
<input checked="" type="checkbox"/> Cluster failover	–	<input checked="" type="checkbox"/>	<Default recipient>
<input type="checkbox"/> Database unavailable	–	<input checked="" type="checkbox"/>	<Default recipient>
<input type="checkbox"/> Deadlock	–	<input checked="" type="checkbox"/>	<Default recipient>
<input type="checkbox"/> Deadlock trace flag disabled	–	<input checked="" type="checkbox"/>	None

For multiple alert types, you can only configure whether all the selected alerts are enabled or disabled, and apply email notification properties. Configurable thresholds either don't exist or vary greatly for certain alert types, so these changes must be carried out on an individual basis.

Configuring alert settings for individual disks

For the Disk space alert, you can configure alert settings for individual drives on your machine. Click on the arrow before the alert to display the drives, and select individual or multiple drives for configuration:

▲ Host machine alerts	
<input type="checkbox"/> Clock skew	All Servers
▲ <input type="checkbox"/> Disk space – (all disks)	
<input type="checkbox"/> politics.testnet.red-gate.com> C:	All Servers
<input type="checkbox"/> politics.testnet.red-gate.com> D:	All Servers
<input type="checkbox"/> Machine unreachable	
	All Servers

Finding customized alerts

You can see which alerts have been customized at which levels in the **Monitored servers** list; they are indicated by the number in brackets:

Monitored servers	
▲ All Servers	
▲ brianh2 (1)	
▶ (local)	
▶ sql2008 (2)	
▶ sqlexpress	
▶ dev-benc2	

In the above example, one alert has been customized at the machine level, and two at the SQL Server instance level. When you select a level in the **Monitored servers** list, those alerts that have been customized at the selected level are identified by **<This level>** and highlighted in bold:

<input type="checkbox"/> Backup overdue	All Servers
<input type="checkbox"/> Blocked process	<This level>
<input type="checkbox"/> Cluster failover	All Servers

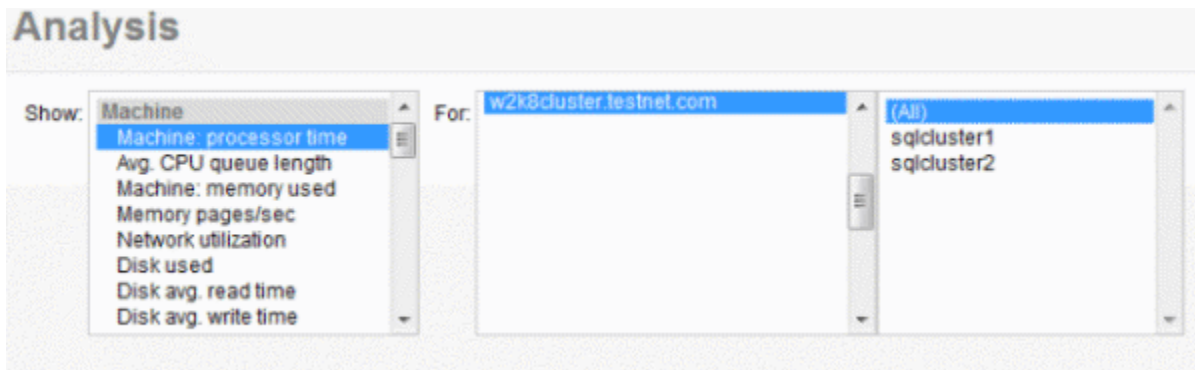
Working with analysis data




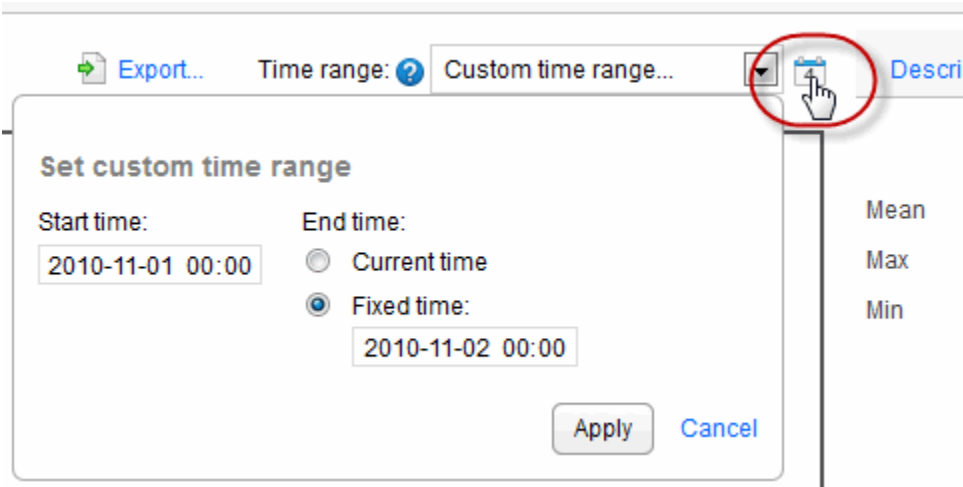
The Analysis page charts performance data for monitored servers and databases. You can display graphs containing data points that represent points in time when performance data was collected

To view an analysis graph

1. Choose the performance metric you want to plot from the **Show** drop-down list. Metrics are grouped into Machine, SQL Server, and Database categories.
2. Select the object to plot from the **For** list.
For some types of metric, you can plot all the objects together. The example below shows Machine: processor time for all machines in the clusters on the selected server.



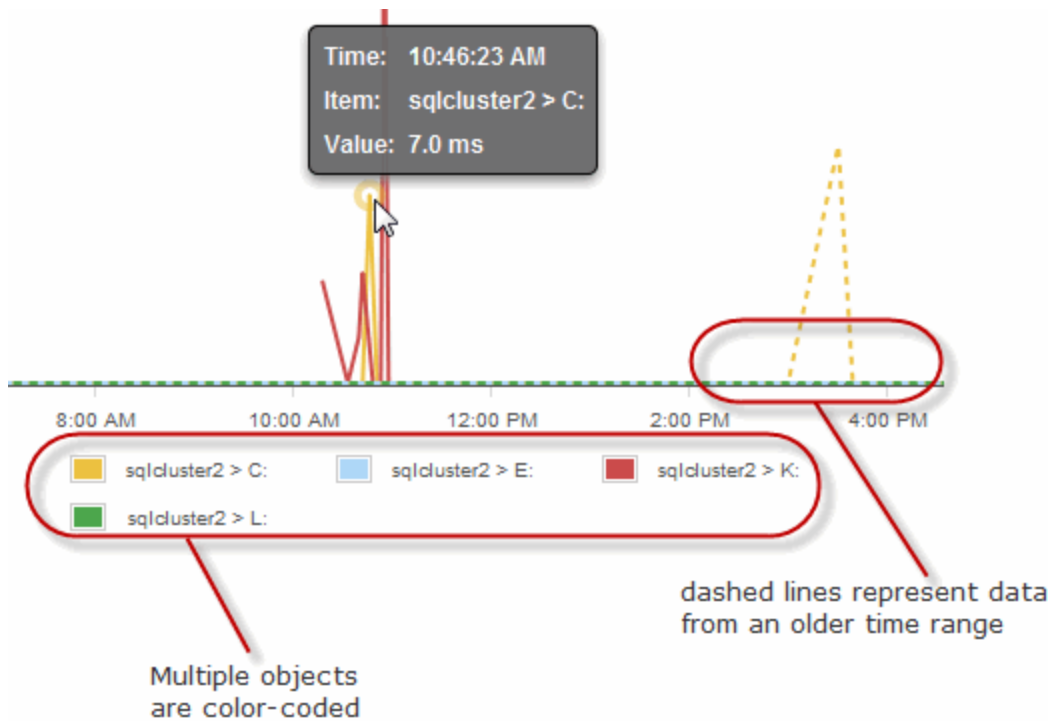
3. Choose a time range. By default, the graph shows the last 10 minutes, but you can select a different fixed value from the **Time range** drop-down list.
4. To specify a particular time range, click the **Set custom time range** icon  and enter the start and end time.



Viewing values on the graph

Each graph line represents the performance of a selected object. SQL Monitor plots up to a maximum of 500 data points on each graph and draws straight lines between consecutive points. For short time ranges (less than 10 minutes) the graph displays fewer data points. For longer time ranges, more data points are displayed, so that these graphs can still provide a detailed overview of changes in values. Move your mouse pointer over a line on the graph to view a tooltip showing the time, object and value at that data point.

If you are comparing data for different time ranges, the graph line for the older time range is displayed as a dashed line. If you are comparing data for multiple objects, a color-coded key is displayed below the graph.



Viewing statistics

Click the **Statistics** tab to the right of the graph to display the mean, maximum and minimum performance counter values for the selected time range. If you're comparing data at different time ranges, the difference between each value is also displayed.

Exporting the data

Click **Export...** to open or save performance data as a .csv file for the object and time range specified. This means you can record object data for use in reports on performance metrics. The export.csv file contains:

- the name of the selected performance counter and host machine.
- the type of value displayed if this doesn't form part of the performance counter name, for example, bytes, milliseconds or percentages.
- a column containing the date and time of every data point displayed on the analysis graph. The time is displayed as Coordinated Universal Time (UTC).
- columns containing details of every data point for each selected object.

Note: If necessary, you may consider adjusting the UTC times displayed in the file according to the time zone in which your servers are located.

Finding more information about performance counters

Click the **Description** tab to the right of the graph to display detailed information about the selected performance counter, including:

- the name of the equivalent PerfMon counter
- an explanation of what this performance counter measures
- guidelines for acceptable values, and values that may suggest a problem with your object
- related performance counters that should also be checked
- suggestions for solutions to problems caused by unacceptable values
- links to articles or white papers containing additional information

The screenshot shows a software interface with two tabs: 'Description' (selected and circled in red) and 'Statistics'. Below the tabs, the title 'SQL Server: processor time' is displayed. The main content area contains the following text:

The processor utilization (%) for the SQL Server service running on the host Windows machine.

Equivalent PerfMon counter: Process(sqlservr): % Processor Time

Explanation: This value tells you what percentage of processor time is being used to run the sqlservr process (the database engine). It excludes other SQL Server-related processes, such as the SQL Server Agent, SSIS, Reporting Services, Analysis Services, and Full Text Search. Because **SQL Server: processor time** is a subset of **Machine: processor time**, the **SQL Server: processor time** value will be lower.

Not enough data to plot one or more items?

Data cannot be plotted on the graph for the time range selected for the following reasons:

- there aren't enough data points on the graph; there must be at least two data points for plotting to take place. The frequency of data points depends on the counter and time range selected, for example, if Last 10 mins is selected for Machine: processor time, data points are plotted at 15 second intervals. Plotting is less frequent for some other counters and for certain longer time intervals.
- data for that time range has been purged
- servers had not been added to the Monitored servers page, or monitoring was paused

If data cannot be plotted, a blank area is displayed on the graph.

Configuring SQL Monitor



Overviews



Alerts



Analysis



Configuration

- Configuring monitored servers
- Setting up email notification
- Configuring Profiler trace
- Organizing monitored servers into groups
- Purging SQL Monitor data
- Changing the Base Monitor connection
- Setting maintenance windows

Configuring monitored servers

Use the **Monitored servers** page to add more servers to monitor, remove existing servers from SQL Monitor, suspend monitoring on servers, or change the connection credentials for one or more servers. Go to the **Configuration** tab. Under **Monitoring**, select **Monitored servers**:



Monitoring

Monitored servers

Add servers to monitor; edit connection credentials; remove or suspend monitored servers; configure maintenance windows.

Trace

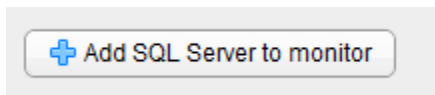
Enable or disable Profiler trace on selected servers.

Groups

Organize your monitored servers into groups.

Adding servers to monitor

1. Click **Add SQL Server to monitor** button to expand the Add SQL Server panel:



2. Enter the name of the server and supply credentials for connecting to the host Windows machine and SQL server instance:

3. Click **Add**.

For more information about adding servers, see [Adding servers to monitor](#).

Editing the credentials or properties of added servers

You can configure some or all of your servers or instances at once:

- to select multiple servers or instances, use the check boxes to the left of each row in the table.
- to select all servers or instances, choose from the **Select** drop-down button near the top of the page.

Editing credentials

You may be forced to change the credentials used to log in to the Windows server on which the SQL Server instance is running, or the authentication details to log into the SQL Server instance:

- to change the credentials for a single server or instance, move your mouse pointer over a row and click **Edit credentials**.
- to change the credentials for multiple servers or instances, select them and click the **Edit credentials** button near the top of the page.

Once you've applied your changes to the existing credentials, SQL Monitor will automatically retry the connection using the updated credentials.

Editing connection properties

You can change the default properties used to connect to the SQL Server instance, for example, you can supply a port number, change the network protocol, and choose to encrypt the connection.

1. Select the instances whose properties you want to edit and click **Edit credentials**.
2. Click **Edit properties** near the bottom of the dialog and update the connection details. Once you've edited these settings and clicked **OK**, the Connection type will be displayed as Custom, not Default.

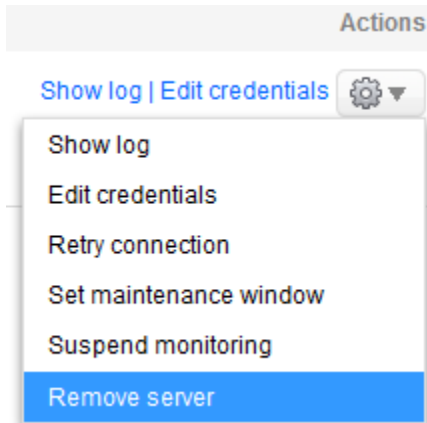
Removing a monitored server

You may want to remove a server from SQL Monitor for a number of reasons:

- server name typed incorrectly; you will need to remove the server before adding it again with the correct name
- the server is no longer in use in your organization
- your trial has expired and you do not wish to license this server

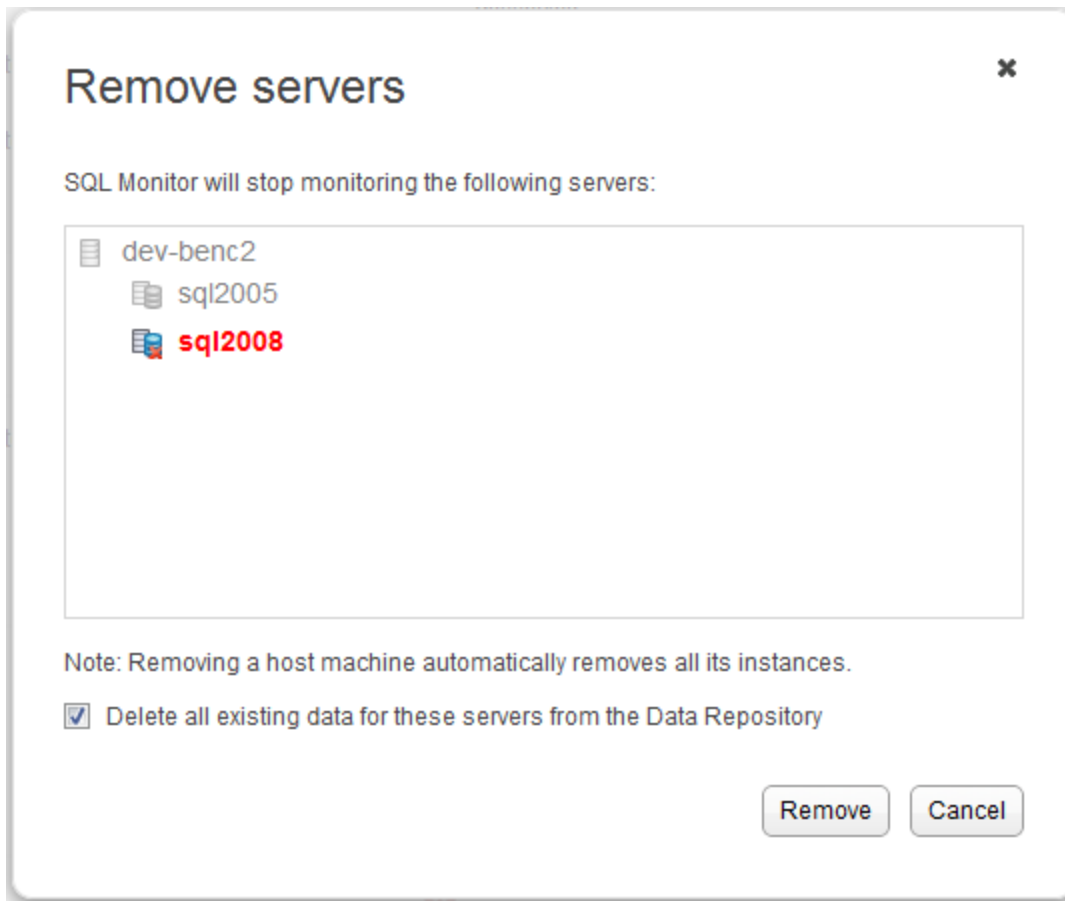
To remove a server:

1. In the row for the server you want to remove select **Remove server** from the Actions list:



You cannot remove an individual node in a cluster. You should remove the cluster management node itself.

2. The Remove servers box is displayed, with the servers to be removed highlighted in the list:



If you remove a host machine, all its instances are also removed.

3. To remove data that is currently stored in the Data Repository for the selected servers, select the **Delete all existing data...** check box.

The server is removed from the **Monitored servers** list; if you have selected to remove existing data, all alerts for that server are removed.

Suspending a monitored server

You may want to stop monitoring temporarily, for example, if one of your servers requires immediate, one-off maintenance and you don't want SQL Monitor to connect to it during this time, to avoid raising redundant alerts.

If routine maintenance is carried out on your servers at regular intervals, you can schedule maintenance windows. During these windows, SQL Monitor remains connected to the servers but alerting is suspended. See [Setting maintenance windows](#).

To suspend monitoring:

- for a single server or instance, click the







- **Actions** button at the end of the row and select **Suspend monitoring**
- for multiple servers or instances, select them and choose **Suspend monitoring** from the **More actions** drop-down button near the top of the page

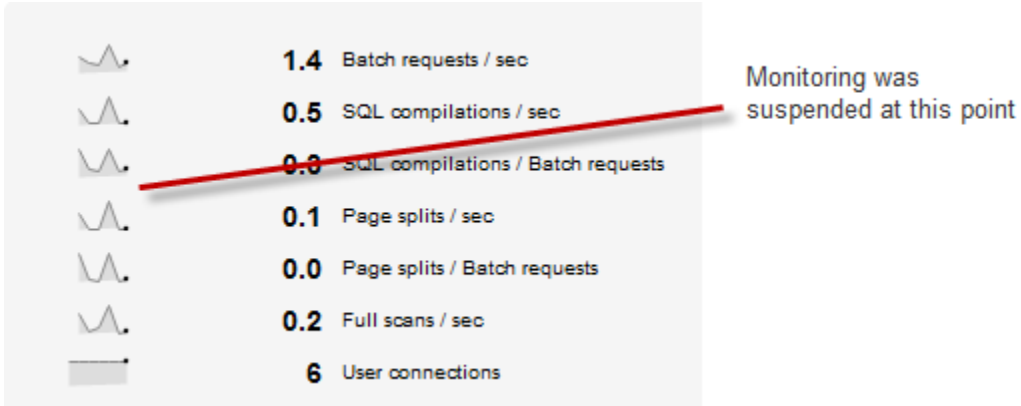
If you suspend monitoring on a host machine, all its instances will also be suspended.

What are the effects of suspending monitoring?

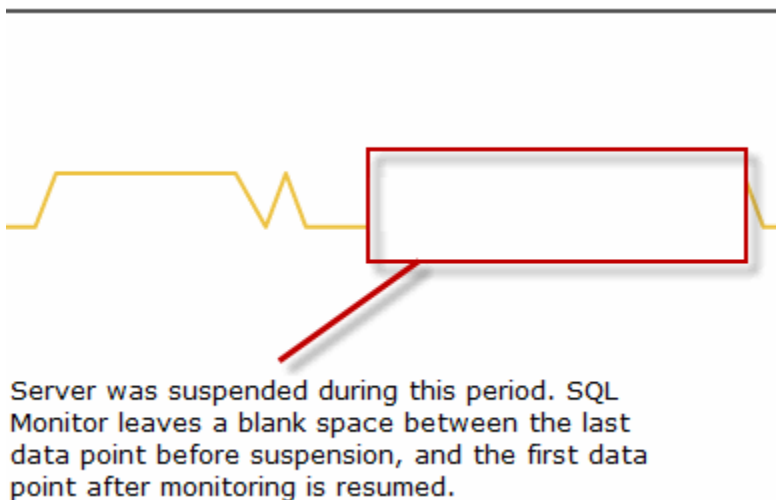
- SQL Monitor will not attempt any connection to the suspended server; all data collection methods are therefore inactive.
- The suspended host machine or instance remains in the **Monitored servers** list, but is identified on all overview pages as being suspended:

SQL Server instances	
Name	Status
 sql2008	 Medium
 (local)	 Suspended

- Values and sparkline graphs are not updated on the overview pages:



- The Analysis page for counters on this server will plot a graph if possible, but fewer data points will be available:



Resuming monitoring on a suspended server

Go to the Monitored servers page, click the



Actions button at the end of the row and select **Resume monitoring**.

Setting maintenance windows

You can configure SQL Monitor to temporarily suspend alerts during set weekly durations, allowing you to carry out maintenance on your servers without raising redundant alerts. For details, see [Setting maintenance windows](#).

Monitoring status explained

SQL Monitor displays one of the following statuses on the Configuration > Monitored servers page for each host machine, cluster node or SQL Server instance:

Monitoring (Connected)

Displayed when:	All monitoring data is being collected successfully.
Suggested action:	Click Show log and select All events to view all data collection events in the last five minutes.

Monitoring stopped (Incorrect credentials or insufficient permissions)

Displayed when:	SQL Monitor is currently unable to collect monitoring data because the credentials it is using have failed authentication. A Monitoring stopped ([host machine SQL Server] credentials) alert is also raised.
Suggested action:	<ul style="list-style-type: none">For host machine authentication failures, check that the credentials supplied to log in are valid and have sufficient permissions to collect WMI data, read the registry, and access files.For SQL Server instance authentication failures, check that the user name and password are correct and the log in has the required permissions. For more information, see Account permissions required. <p>After an authorization error, by default SQL Monitor will not attempt to reconnect until you either change the credentials specified in SQL Monitor, or change the account permissions (in this case, you will need to click Retry connection on the Monitored Servers page). This prevents SQL Monitor from sending a potentially large number of failed login messages to your security log, and triggering Intrusion Detection System in the process.</p> <p>If the potential impact on your security log will not cause significant problems to your systems, you can change this default setting so that SQL Monitor will automatically attempt reconnection. See Reconnecting automatically after monitoring stopped.</p> <p>Once the correct credentials are entered and authentication is successful, monitoring will resume and the Monitoring stopped ([host machine SQL Server] credentials) alert will be marked as Ended.</p>

Unreachable (Cannot connect)

Displayed when:	One of the following reasons applies: <ul style="list-style-type: none">The host machine or cluster node is not responding to Ping requests; either the machine name is incorrect or there is a network problem. A Machine unreachable alert is also raised, and a SQL Server instance unreachable alert is raised for each instance on the machine.The SQL Server service has been stopped or there is a problem with SQL connectivity. A SQL Server instance unreachable alert is also raised.The Remote Registry service is not started or remote file access is not set up. If this problem occurs continuously for 2 minutes, a Monitoring error ([host machine SQL Server] data collection) alert is also raised.
Suggested action:	<ul style="list-style-type: none">Check that you have entered the server name correctly, and that your machine is not behind a firewall.Click Show log and check the <i>Exception</i> and <i>Exception message</i> columns displayed in the table of data collection events to determine the specific problem. <p>For more information, see How do I test data collection methods?.</p>

Connection failed (Internal SQL Monitor error)

Displayed when:	An unknown application error has occurred. If this problem occurs continuously for 2 minutes, a Monitoring error ([host machine SQL Server] data collection) alert is also raised.
Suggested action:	Please contact Red Gate support with information about your error, including relevant log file details. See Viewing component information and log files .

Connection failed (Bad data)

Displayed when:	<p>SQL Monitor cannot process some data collected from the host machine or cluster node. Possible reasons are:</p> <ul style="list-style-type: none"> • The performance counter library may be corrupt and needs rebuilding. For information about rebuilding your PerfMon library, see http://support.microsoft.com/kb/300956. • A WMI provider may be corrupt and needs rebuilding. For information about troubleshooting WMI, see http://technet.microsoft.com/en-us/library/ff406382.aspx. • You are running a 32-bit SQL Server instance on a 64-bit Windows machine, and some performance counter objects are not available. For more information, see this MSDN article on 64-bit Support.
Suggested action:	Click Show log and refer to the <i>Exception</i> and <i>Exception message</i> columns in the table of data collection events to determine the specific problem.

Connection failed (Cannot connect)

Displayed when:	<p>One of the following reasons applies:</p> <ul style="list-style-type: none"> • Remote file access is not set up (hidden administrative shares should be enabled). • One of the required services (WMI, Remote Registry) is not started. • There is a problem with SQL connectivity. If this problem occurs continuously for 2 minutes, a Monitoring error ([host machine SQL Server] data collection) alert is also raised.
Suggested action:	<p>SQL Monitor will automatically retry the connection every few minutes. Click Show log and refer to the <i>Exception</i> and <i>Exception message</i> columns in the table of data collection events for information about the type of problem.</p> <p>For more information, see How do I test data collection methods?.</p>

Unlicensed

Displayed when:	Your evaluation period has expired and the machine, cluster node or SQL Server instance has not yet been licensed.
Suggested action:	<p>Enter a serial number or allocate licenses on the Configuration > Licensing page. One license is required for each host machine or cluster node - this will license all SQL Server instances on the machine or node.</p> <p>For more information, see Licensing.</p>

Suspended

Displayed when:	Monitoring on this machine, cluster node or SQL Server instance has been manually suspended. No data will be collected until you resume monitoring.
Suggested action:	<p>Go to the Configuration > Monitored servers page, click the Actions button</p> <div data-bbox="305 1241 1003 1661" data-label="Image"> </div> <p>at the end of the row and select Resume monitoring.</p> <p>For more information, see Configuring monitored servers.</p>


Maintenance window (Alerting suspended)



Displayed when:	The host machine or node is currently in a maintenance window. SQL Monitor is still monitoring servers, but alerting is temporarily suspended. Once the maintenance window duration has ended, alerting will resume.
-----------------	--


Suggested action:	Go to the Configuration > Monitored servers page to check when the current maintenance window is due to end. For more information, see Setting maintenance windows .
-------------------	---

Viewing the connection events log

For some statuses, it is recommended that you check the log for more details. To do this, on the Configuration > Monitored servers page, click **Show log** for the server or instance experiencing connection problems:

 **Monitoring stopped**
Incorrect credentials or insufficient permissions

[Show log](#) | [Edit credentials](#)  



The page contains a table listing recent errors or data collection events that can help you to determine the specific problem.

Reconnecting automatically after monitoring stopped

If the Configuration > Monitored servers page displays **Monitoring stopped (Incorrect credentials or insufficient permissions)**, by default, SQL Monitor will not attempt to reconnect until you do one of the following:

- change the credentials specified in SQL Monitor
- change the account permissions (in this case, you will need to click **Retry connection** on the Monitored Servers page)

This prevents SQL Monitor from sending a potentially large number of failed login messages to your security log, and triggering Intrusion Detection System in the process.

If the potential impact on your security log will not cause significant problems to your systems, you can change this default setting so that SQL Monitor will automatically attempt reconnection.

Configuring automatic reconnection after monitoring stopped in v3

1. Stop the Base Monitor service. To do this, go to Services (select **Start > Run**, enter **services.msc** and click **OK**) and stop **SQL Monitor 3 Base Monitor**.
2. Open the **RedGate.Response.Engine.Alerting.Base.Service.exe.settings** file in Notepad. By default, this is saved in C:\ProgramData\Red Gate\SQL Monitor 3.
3. Find the `reconnectAfterAuthorizationError` section. This contains a setting called `reconnectAfterAuthorizationError="false"`:

```
reconnectAfterAuthorizationError
*****

whether SQL Monitor should try to reconnect and continue monitoring after an
AuthorizationError is raised
-->
  <repositorySettings slaveMode="false">
    <!-- valid databaseEngines: "Mssql"-->
    <dataStore databaseEngine="Mssql" autoUpgrade="true">
      <!-- valid versions: "Auto", "SQL2005", "SQL2008" -->
      <mssql version="Auto"
connectionStringName="DataConnectionString"/>
    </dataStore>
  </repositorySettings>
  <collectionSettings maxTraceFileAge="00:15:00"
maxTraceFileDiskUsage="1024" reconnectAfterAuthorizationError="false">
  <collectionSchedules>
    <!-- Four schedules are available:
```

4. Replace "false" with "true" for this setting, and save the file.
5. Restart the **SQL Monitor 3 Base Monitor** service.

Configuring automatic reconnection after monitoring stopped in v2

Follow the steps above to configure earlier version of SQL Monitor. Note that the Base Monitor service is called **SQL Monitor 2 Base Monitor**, and the settings file is saved in C:\ProgramData\Red Gate\SQL Monitor 2.

Setting up email notification

About email notification

You can set up SQL Monitor to send an email whenever an alert is raised.

Notification behavior is entirely configurable. You can set up SQL Monitor:

- to send an email for every type of alert
- not to send emails at all
- to send emails only for those alerts you consider important

Once you have set up your basic email notification settings, you can also tweak email notifications at a more granular level - to use a different recipient for an alert on a particular server, for example, or disable notifications for one or more types of alert.

What is in the email?

Each email message sent by SQL Monitor contains information to help you understand what the problem is before you log in to [view details of the alert](#). This includes:

- the machine and object against which it has been raised
- the alert type and level it was raised at
- the unique ID number that forms part of the URL for this alert
- a hyperlink so that you can open this alert in your browser

Emails sent when a **Job failed** or **Job backup overdue** alert is raised will also contain the job name.

If you send emails to other people, they will be prompted for the SQL Monitor password when they follow the link.

When does SQL Monitor send emails?

Each time an alert is raised, it is considered an occurrence of that type of alert, and triggers an email (if email notification is enabled for that type of alert and for that monitored object). For example, if 200 deadlocks occur on one of your instances, 200 emails will be sent, one for each raised alert.

Some types of alerts are continuous (they are raised as **Active**) and can have a duration. For example, a **Backup overdue** alert is considered Active until the backup is performed. In this case, only a single email will be sent when the problem is first detected. If the problem escalates, you can choose to receive further emails. See [Also send emails when](#) below.

Limiting the number of emails sent by SQL Monitor

SQL Monitor limits the number of emails sent to ensure it does not flood your mail servers. It will not send more than 1000 emails per 24 hour period. If this limit is reached, a warning email is sent with advice on how to reduce the volume of emails. This limit is for all emails sent by SQL Monitor, regardless of recipient.

The warning email also contains instructions for adjusting the daily email limit.

To set up email notification

Go to the **Configuration** tab. Under **Alerts and metrics**, select **Email settings**:



Alerts and metrics

Alert settings

Enable and disable alert types; change alert thresholds and levels.

Email settings

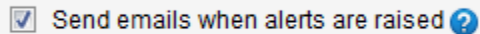
Set up and manage email notification for alerts.

Custom metrics

Create and manage custom metrics and custom alerts.

Enabling or disabling all emails

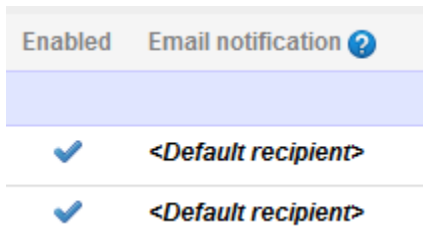
If you want to use email notification in SQL Monitor, select the **Send emails when alerts are raised** check box:



If this check box is cleared, SQL Monitor will not send **any** emails, regardless of all other settings.

Setting up the default email address

Enter an email address that all SQL Monitor emails will be sent to by default in the **Send emails to** box. This is the email address that will be shown on the Alert settings page as the **<Default recipient>**:



To add multiple email addresses, separate each with a comma. You can later customize the email recipient for different alert types or servers. If you subsequently change the email address in the **Send emails to** box, this will be automatically applied to any alerts that use <Default recipient>. Alerts where you have specified a different email recipient will not be affected.

Entering a send from email address

The **Send from email address** is the email address that will appear in the From field for all email messages sent by SQL Monitor when an alert is raised.

Enter the email address in the format: username@domain.com. You can only send a test email when a valid address in the right format has been entered. You can enter the **Send emails to** and **Send from** addresses before supplying your mail server details, but you won't be able to send the test email until your mail server details are provided. See **Enter your mail server settings** below.

Also send emails when

These are global options that apply to alerts that are raised with a status of **Active**.

Some types of alerts have a status of **Event** - their level cannot increase, and their status does not change to ended, so these options do not apply. For more information, see [List of alerts](#).

An alert level increases

Some alerts can be configured with multiple thresholds: Low, Medium and High. These alerts are raised with a status of Active and can escalate automatically to a higher level when another threshold is passed. Select **An alert level increases** to specify that if this happens, a subsequent email should be sent.

No emails will be sent if:

- the level of an Active alert is automatically downgraded (for example, drops from High to Medium)
- an Active alert is downgraded and then subsequently escalates to its previous higher level

An alert's status changes to Ended

Active alerts automatically change to **Ended** when the condition that triggered them no longer applies; for example if memory used falls below the defined threshold for the alert, or if a backup is performed. Select **An alert's status changes to Ended** to specify that an email should be sent to inform the recipient that this particular occurrence of the alert has ended.

About PagerDuty

[PagerDuty](#) is an alarm aggregation and dispatching service. It collects alerts from your monitoring tools in a single repository, and notifies users via SMS, phone calls or emails. It enables to you to add alert severity filters, on-call scheduling, escalation policies and incident tracking to SQL Monitor.

How do I integrate SQL Monitor with PagerDuty?

There are integration guidelines in PagerDuty's [Red Gate SQL Monitor Integration Guide](#). The only configuration change required in SQL Monitor is to add the PagerDuty integration email address to the **Send emails to** section of the Configuration >Email settings page.

Once you have created a SQL Monitor service in PagerDuty, you can edit the service settings. By default, PagerDuty will notify users of alerts with a status of high or medium, and low status alerts will be ignored. You can choose to be notified of all alerts, or high status alerts only.

Enter your mail server settings

If you don't know the DNS name or the IP address and port number of your outgoing SMTP mail server, check with your system administrator and enter them in the relevant text boxes.

Select **Require a secure (SSL or TLS) connection** to enable explicit SSL email. Once the SMTP connection is established, SSL is requested and the message delivered using the supplied credentials.

Also check with your administrator to find out whether your mail server requires a user name and password. If this is the case, select the **Mail server requires a user name and password** check box and enter details.

URL settings

When SQL Monitor sends an email notification, it includes a hyperlink to the alert. This hyperlink contains the name of your website. In some situations, this link will not work for other people (for example if the URL is localhost:<portnumber> this will only open SQL Monitor on your local computer).

If this is the case, you can enter the fully qualified URL to be included in the email hyperlink. Enter the URL in the **Fully qualified website URL** box.

Configuring Profiler trace

Enabling trace allows you to continuously capture SQL Profiler trace data on the selected instance. When an alert is raised, the SQL statements that were executing around the time of the alert are displayed under SQL Processes in the Performance data section for that alert.

Configuring trace

1. Go to the **Configuration** tab. Under **Monitoring**, select **Trace**:



Monitoring

Monitored servers

Add servers to monitor; edit connection credentials; remove or suspend monitored servers; configure maintenance windows.

Trace

Enable or disable Profiler trace on selected servers.

Groups

Organize your monitored servers into groups.

2. For each SQL Server instance you want to store trace data for, Select **On** from the **Trace** drop-down list, then click the **Save settings** button.

Configuration > Trace

Enable trace to continuously capture SQL Profiler trace data on the selected instance. When an alert is raised, the SQL statements that were executing around the time of the alert are displayed under SQL Processes in the Performance data section for that alert.

We recommend using trace only when required, as a diagnostic aid to help investigate the monitored server by up to 5%. Enabling trace for multiple instances can also slow down the server.

SQL Server instance	Trace
---------------------	-------

tc-catherinek(local)	
----------------------	--

	Off
	Off
	On


Save settings


or **Cancel**

If trace is enabled for a SQL Server instance, the trace status is displayed on the Global overview and Host machine overview pages:

Read alerts ?	Unread alerts ?	Last 24 hrs	Trace	Process
365	324	9	On	
156	136	1		
111	93	3		
	8	0		

Global overview
Click to display the Trace page

Help | Log out 

11:06 AM
30 Nov 2010 

365 Unread alerts: **324** Last 24 hrs: **9** [Trace on](#)

Host machine overview
Click to display the Trace page

What trace data is captured?

The following trace information is captured (SQL Server event number in brackets):

- RPC_Starting (11)
- SQL_BatchStarting (13)
- Audit_Login (14)
- Attention (16)
- SQL_Exception (33)

Storing and purging trace data

If trace is enabled, a trace data file is stored locally on the monitored server. Sections of the data file from around the time an alert was raised are copied to the SQL Monitor Data Repository. When you click on an alert in the Alert Inbox to display more details, the trace data is retrieved from the Data Repository.

Note: To prevent the trace data file from taking up excessive storage space on the monitored server, it is automatically deleted after it has remained on the server for 15 minutes, or reaches 1 GB in size.

Trace data stored in the Data Repository is categorized as **SQL Server data** for the purposes of purging. To specify how long to keep stored trace data in the Data Repository, use the **SQL Server data** purging option on the **Data purging** page (Configuration > Data purging). Trace data stored in the Data Repository is categorized as **SQL Server data** for the purposes of purging. To specify how long to keep stored trace data in the Data Repository, use the **SQL Server data** purging option on the **Data purging** page (Configuration > Data purging).

See [Purging SQL Monitor data](#).

Organizing monitored servers into groups

Use the **Groups** page (Configuration > Groups) to organize your servers into groups sharing similar properties.

Why create a group?

Monitored servers can be grouped together so you can apply the same alert configuration settings to them more easily. For example, you may have a number of production servers on different host machines but you want them all to use the same alert settings.

You can also filter the Global Overview and Alert Inbox by group, to look at information for groups of servers at a time.

Groups are identified in the **Monitored servers** list by the group



icon.

Creating a group

1. Go to the **Configuration** tab. Under **Monitoring**, select **Groups**:



Monitoring

Monitored servers

Add servers to monitor; edit connection credentials; remove or suspend monitored servers; configure maintenance windows.

Trace

Enable or disable Profiler trace on selected servers.

Groups

Organize your monitored servers into groups.

You can also use the **Manage groups** link on the left of any overview page or the Alert Inbox.

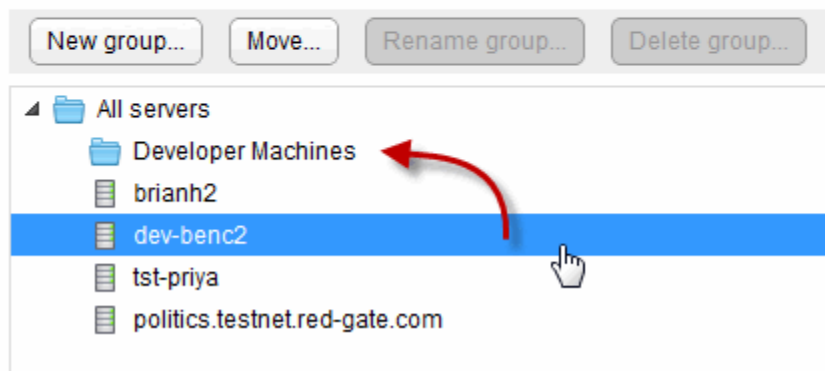
2. Click **New group**, enter a name and click **Create**.

To add servers to the group, drag and drop each one in turn onto the group name. (You cannot drag multiple selected servers.)

In the example below, the server dev-benc2 is about to be moved to the "Developer Machines" group:

Configuration > Groups

Grouping servers with similar properties (e.g. "Production" or "Dev") allows you to view and configure Alert settings for all servers in the group at once.



Only host machines are listed:

- You cannot move individual SQL Server instances; they are moved automatically with their host machine

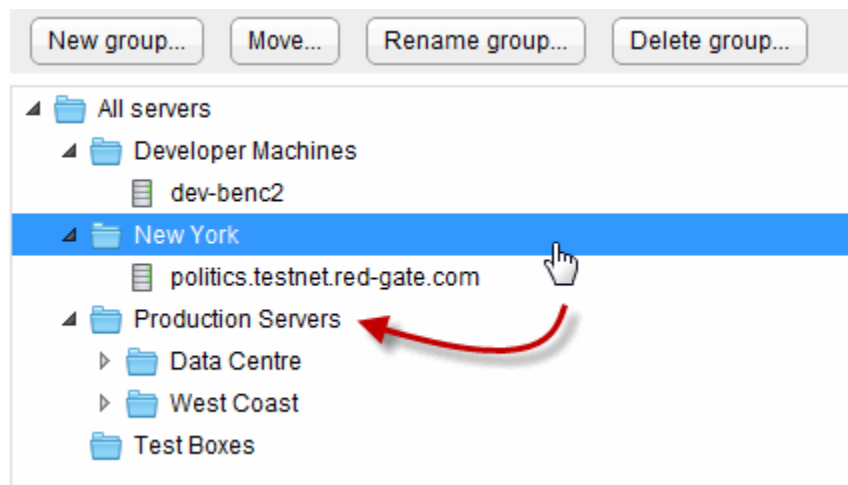
- If you move a cluster into a group, all its nodes and instances are moved with it

You can also select a server, and click **Move**. Select the destination group in the Move box.

Creating a hierarchy

You can move one group into another. For example, if you want to further organize your Production servers into several regions, you should first create your Production group, then create a group for each region and move those groups into "Production".

In the example below several sub-groups have been created for the "Production servers" group. Moving the "New York" group will also move any servers in that group.



By default, each group inherits all alert settings from the group above; so this allows you to create alert settings for all your production servers, which are applied to each region. However, you can then tweak the settings for just one of those regions, if required.

Removing a server from a group

To remove a server from a group, do one of the following:

- Drag it to **All servers**
- Select the server, click Move and then select **All servers**

Working with groups

Use the buttons above the list of groups to rename, delete or move a selected group.

Deleting a group doesn't delete the servers it contains; instead the servers are moved to the next level up in the server hierarchy.

Configuring alerts by group

One of the main benefits of adding servers to a group is that you can configure alert settings at the group level. Everything in that group inherits these settings (unless specifically customized at a lower level). Each group in the hierarchy can be customized separately.

In the example below, on the **Alert settings** page, different types of alerts have been customized at various levels; the **Inherited from** column shows which level each type of alert is customized at.

Monitored servers

- ▲ All Servers
 - ▶ Developer Machines
 - ▶ New York
 - ▲ Production Server...
 - ▲ Data Centre (1)
 - ▶ **brianh2 (1)**
 - ▶ West Coast (1)
 - Test Boxes

Alert type	Inherited from ?
▲ All SQL Server alerts	
<input type="checkbox"/> Backup overdue	Production Servers
<input type="checkbox"/> Blocked process	<This level>
<input type="checkbox"/> Cluster failover	All Servers
<input type="checkbox"/> Database unavailable	Data Centre
<input type="checkbox"/> Deadlock	Production Servers

When editing the alert settings, the breadcrumbs show the full inheritance path for this type of alert. Any level with an asterisk * has been customized. In the example below, the Blocked process alert is being customized for a specific machine, but it has also been set more generically for the "Production servers" group.

Configuration > Alert settings > Blocked process | brianh2

All Servers > Production Servers * > Data Centre > brianh2 *

- Inherit settings**
 From Production Servers
- Customize settings for this level**
 brianh2

For more information, see Customizing alerts.

Purging SQL Monitor data

SQL Monitor stores a large amount of high, medium and low volume data in the Data Repository. To prevent the Data Repository database using all your hard disk space, SQL Monitor checks for old data every hour, and purges data that is older than the time specified for each category in your purging policy.

Configuring data purging

Go to the **Configuration** tab. Under **Data Management**, select **Data purging**:



Data Management

[Data purging](#)

Specify how long SQL Monitor keeps historic data in its Data Repository.

Data is stored in three separate categories: Machine data, SQL Server data and Alert data. Each category is broken down further to make it easier to identify whether the data could become high, medium or low volume, and where the data is displayed in SQL Monitor:

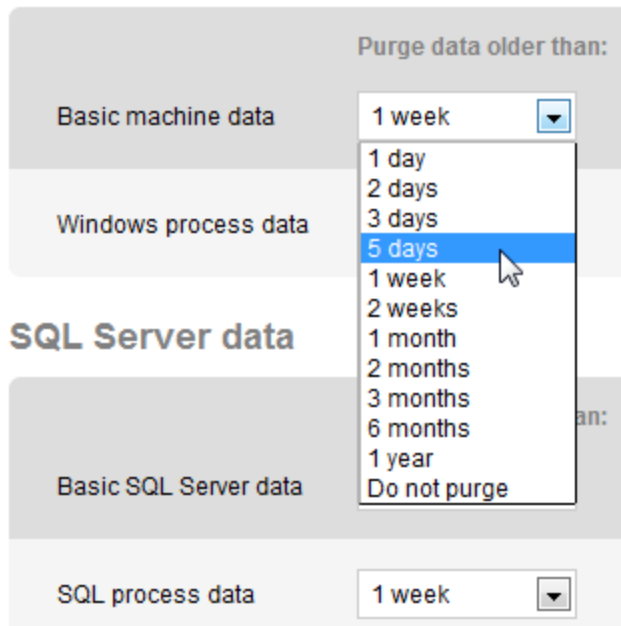
- **Machine data:**
 - Basic machine data (low volume data) - displayed on the Host machine, Cluster, and SQL Server instance overview pages, and as metrics listed under Machine on the Analysis page.
 - Windows process data (high volume data) - displayed as the System processes (top 10) on the Host or Cluster machine overview pages.
- **SQL Server data:**
 - Basic SQL Server data (low volume data) - displayed on the SQL Server instance overview pages, and as metrics listed under SQL Server on the Analysis page.
 - SQL process data (high volume data) - displayed as SQL user processes (top 10 by CPU usage) on the SQL Server instance overview pages, and as alert details when Trace is turned off.
 - Top 10 queries data (high volume data) - displayed as the Top 10 expensive queries on the SQL Server instance and Database overview pages.
 - Database performance metric data (medium volume data) - displayed on the Database overview pages, and as metrics listed under Database on the Analysis page.
- **Alert data:**
 - Basic alert data (low volume data) - displayed on the Alert pages of raised alerts and including alert details, history of occurrences and performance data.
 - Trace data (high volume data) - displayed under SQL Processes/Profiler trace in the Performance data section of the Alert pages when Trace is turned on.

The default purge settings ensure that all data older than 1 week is purged. To change these settings, select a different time limit from the drop-down list for each category and click **Save settings**.

Machine data

Purge data older than:

Basic machine data	1 week
Windows process data	5 days
SQL Server data	1 week
Basic SQL Server data	1 week
SQL process data	1 week



How frequently should I purge data?

Your purging frequency depends on:

- how important it is for you to retain historic information about each category. For example, for auditing purposes, you may need to keep data about your machines for longer than alert data.
- the amount of disk space you have available in your Data Repository to store collected data. If space is an issue, you may consider purging high volume data on a more frequent basis.

There is a **Do not purge** option for each category, which means data will be stored indefinitely in your Data Repository, but you should ensure that you have enough disk space available to accommodate potentially high volumes of data. For more information, see [Growth in size of Data Repository database](#).

What is the effect of purging data?

- For overview data, a dash (--) or (no data) will be displayed on the overview pages next to a value if you rewind time to a point for which data does not exist because it has been purged.
- For alert data, purging will result in older alerts no longer being displayed in the application.
- For analysis graphs, a blank area will be displayed on the graph for the duration purged.

Changing the Base Monitor connection

The Base Monitor Service account is the Windows account you specified during installation to connect to the Data Repository (the SQL Server database that stores all collected data). See [Installing SQL Monitor components](#).

You may need to disable the machine on which the Base Monitor Service account is running, but you do not want to stop monitoring data. You can do this by installing the Base Monitor on another machine and then changing the connection.

Configuring the Base Monitor connection

Go to the **Configuration** tab. Under **Application options**, select **Base Monitor connection**:



Application options

Manage user roles

Manage passwords for Administrator, Standard and Read-only user roles.

Base Monitor connection

View details of your current Base Monitor connection; connect to a different Base Monitor.

Installing the Base Monitor on another machine

1. Click the Base Monitor installer link to download the SQLMonitorBaseMonitorInstaller.exe file.
2. Run the .exe file. Click **Next** to display the **Set up Base Monitor** page.
3. Specify where to install the files that will run the service, and provide a writable folder for the configuration file.
The configuration file stores the location of the Data Repository and the connection string for accessing the database. The configuration file requires a writeable location so that it can be updated by SQL Monitor if you move the Data Repository database to a different SQL Server instance.
4. Select a TCP port to use for communicating with the Web Server and the Data Repository. Click **Next**.
5. By default, the SQL Server database set up when you installed will still be used as the Data Repository. If you're happy to continue using this, go to the instructions below on *Connecting to the database*.

If you want to create a new database or use an existing one, first follow the instructions below.

Creating a new database for the Data Repository

1. Click **Set up Data Repository** and then select **Create a new SQL Server database**.
2. Click **Next**, then enter the SQL Server instance name.
You can type an instance name or IP address directly into the **SQL Server** box if the instance name does not appear in the drop-down list.
3. By default, SQL Monitor will create a database called RedGateMonitor. To use a different name, in the **Database** box, type a name for the database.
4. Under **Create database using**, one of the following:
 - **Use current credentials** to use the account that you are currently logged into.
 - **Specify an account**. If you select Windows authentication, SQL Monitor assumes this account is in the current domain. To use a different domain account, enter credentials in the format **username@domain-name** or **domain-name\username**. Enter the password.

The specified account must have Create Database permissions.

5. Click **Create Now** to create the database. Once database creation has been confirmed, click **Close**. If you go back a page in the wizard, this will not undo the creation of the database.

The database is created using all the default settings for creating a new database on the SQL Server instance you selected. If you encounter an error message when creating the database, see [Account credentials required](#).

Using existing database for the Data Repository

You should only use an existing database if you want to create the database outside of the SQL Monitor installer using a SQL Server management tool, in order to specify options such as filegroups, autogrowth, and so on.

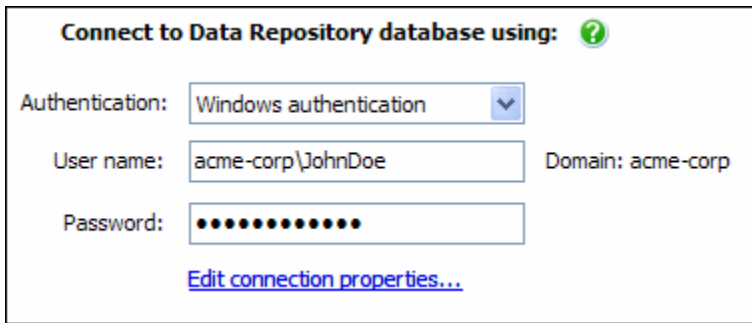
To specify your Data Repository database:

1. Click **Set Up Data Repository** and then select **Use an existing SQL Server database**.
2. Click **Next**, then select the SQL Server instance.
3. In the **Database** box, select the database you want to use; databases on the SQL Server should be automatically detected, and appear in the list.
You can type an instance name or IP address directly into the Server box if the instance name does not appear in the drop-down list.
4. Click **Next**. SQL Monitor checks the database. Once the database connection has been confirmed, click **Close**.

If you encounter an error message when connecting to the database, see [Account credentials required](#).

Connecting to the database

The Base Monitor service requires credentials to connect to the database you just specified in order to store and retrieve collected data. This account must have **administrator privileges** on the database.



Windows authentication

If you connect to the database using Windows authentication, the Windows account you select will also be used to run the SQL Monitor Base Monitor service. If the account does not have permissions to run a Windows service, then it will be granted these permissions.

The user name defaults to your current domain. You can change the domain in the **User name** box using either of the following formats:

- mydomain\username
- username@mydomain

SQL Server authentication

You can connect to the database using SQL Server authentication. If you provide SQL Server login credentials, the Base Monitor service will run under the Local Service account.

[Read more information about the Local Service account.](#)

Error reporting

SQL Monitor can send data back to Red Gate about the features you use and any application errors you encounter. This helps us to improve SQL Monitor for future releases.

All data is anonymized before we receive it, and no confidential information is sent to us.

If you are happy to allow this data to be sent, select **Send error reports**, and optionally enter your email address. We will only contact you in the event of an error for which we require further information, to help us eliminate bugs.

Summary

1. Review all your installation options on the Summary page. If you want to change anything, use the **Back** button to edit the required page.

Note: Once your database for the Data Repository has been created, it will not be deleted if you go back and set up a different Data Repository.

2. Click **Install** to start installing SQL Monitor using the selected settings.
3. When installation has completed, click **Finish** to launch SQL Monitor.

To start using SQL Monitor, you first need to create a password to use when logging in to the SQL Monitor pages.

Changing the Base Monitor connection

Once you've installed the Base Monitor on another machine, you can change the connection:

1. Enter the full domain name or IP address for the machine that has the Base Monitor installed on it.
2. Select the TCP port you specified during installation to use for communicating with the Web Server and the Data Repository.
3. Click **Change connection**.

Setting maintenance windows

About maintenance windows

If routine maintenance is carried out on your server at regular intervals, you may want to temporarily stop SQL Monitor from raising alerts during those periods. SQL Monitor allows you to schedule maintenance window time periods, during which alerting is suspended on selected servers.

During the maintenance window time period, no alerts are raised or queued, and alerting only resumes once the specified maintenance window duration has ended. Monitoring continues as normal, so you can still check the health of your servers using the Overview pages and examine performance data using the Analysis page.

The maintenance window schedule is entirely configurable. You can set a maintenance window to apply weekly:

- at a certain time of day
- for a specified duration
- on any day of the week

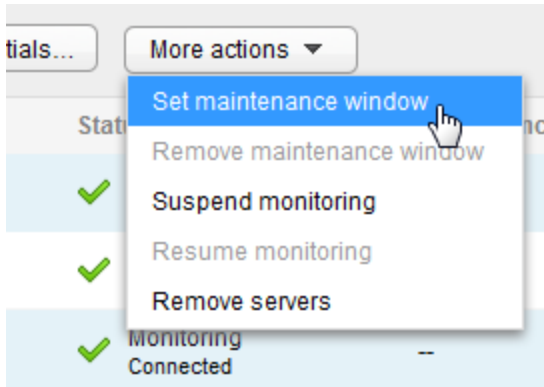
Once you have set up your maintenance window, you can edit or remove it at a later date.

Setting a maintenance window

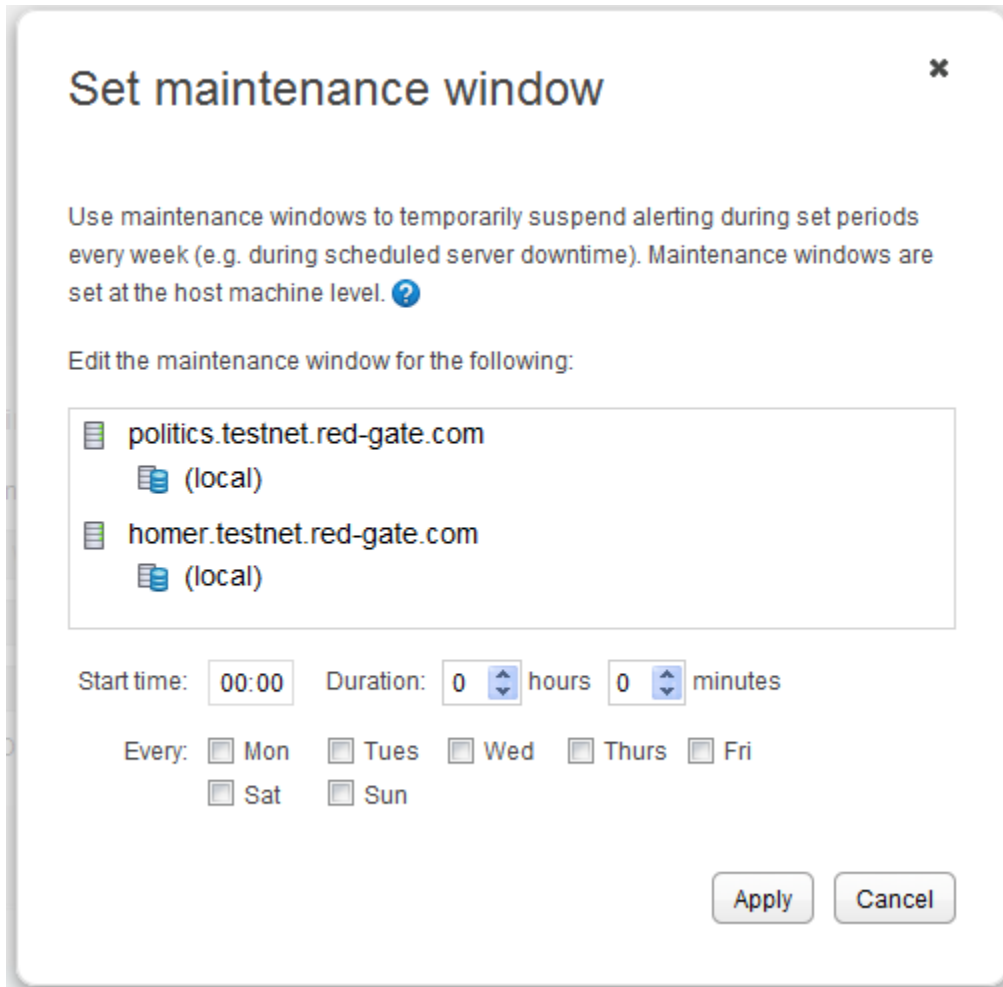
1. Go to the **Configuration** tab and under **Monitoring**, select **Monitored servers**.
2. Use the check boxes in the **Server** column to select the servers or clusters that need a maintenance window. If you select an instance, the maintenance window will be set for the host machine and all sibling instances and cluster nodes.

Server	Status
<input checked="" type="checkbox"/> homer.testnet.red-gate.com	Monitoring Connected
<input type="checkbox"/> (local)	Monitoring Connected
<input checked="" type="checkbox"/> politics.testnet.red-gate.com	Monitoring Connected
<input type="checkbox"/> labour	Monitoring Connected
<input type="checkbox"/> libdem	Monitoring Connected
<input type="checkbox"/> (local)	Monitoring Connected

3. Select **Set maintenance window** from the **More actions** list:















The machines and instances affected by the maintenance window are displayed in the dialog:











4. Enter the time that you want the maintenance window to start, and its duration in hours and minutes. The start time uses the 24 hour clock, so the range is between 00:00 and 23:59.

The start time uses the same time zone as the Base Monitor.

5. Use the check boxes to select the day or days on which you want the maintenance window to be set, then click **Apply**. You can see details for the affected machines in the the Maintenance window column of the the **Configuration > Monitored servers** page:

Server	Status	Maintenance window
<input checked="" type="checkbox"/>  homer.testnet.red-gate.com	 Monitoring Connected	Every Friday Start: 22:00, Duration: 1 hour 30 mins
<input type="checkbox"/>  (local)	 Monitoring Connected	(Same as host)
<input checked="" type="checkbox"/>  politics.testnet.red-gate.com	 Monitoring Connected	Every Friday Start: 22:00, Duration: 1 hour 30 mins
<input type="checkbox"/>  labour	 Monitoring Connected	(Same as host)
<input type="checkbox"/>  libdem	 Monitoring Connected	(Same as host)
<input type="checkbox"/>  (local)	 Monitoring Connected	(Same as host)

For the duration of the maintenance window (between 22:00 and 23:30 every Friday in this example), the Status of the affected machines changes:

Server	Status	Maintenance window
<input checked="" type="checkbox"/>  homer.testnet.red-gate.com	 Maintenance window Alerting suspended	Every Friday Start: 22:00, Duration: 1 hour 30 mins
<input type="checkbox"/>  (local)	—	(Same as host)
<input checked="" type="checkbox"/>  politics.testnet.red-gate.com	 Maintenance window Alerting suspended	Every Friday Start: 22:00, Duration: 1 hour 30 mins
<input type="checkbox"/>  labour	—	(Same as host)
<input type="checkbox"/>  libdem	—	(Same as host)
<input type="checkbox"/>  (local)	—	(Same as host)

The status displayed on the Overview pages for the affected machines also changes to **Maintenance window**.

Editing a maintenance window

You may want to reschedule the maintenance window set on a particular server. If you initially set a maintenance window schedule for several servers, but later edit the schedule for one of those servers, the new schedule will *only* apply to the single edited server and its instances. To edit the maintenance window settings:

- for a single machine, click the



Actions button at the end of the row and select **Edit maintenance window**. Make your changes, then click **Apply**.

- for multiple machines, select them using the check boxes in the Server column and select **Set maintenance window** from the **More actions** list. Make your changes, then click **Apply**.

Removing a maintenance window

To remove a maintenance window:

- for a single machine, click the



Actions button at the end of the row and select **Remove maintenance window** and click **Remove**.

- for multiple machines, select them using the check boxes in the Server column and select **Remove maintenance window** from the **More actions** list, then click **Remove**.

Managing security and user roles

SQL Monitor is a web application that is accessed using a web browser within your LAN. The following aspects of SQL Monitor may therefore require security considerations:

Encryption between the Base Monitor service and Web Server

The communication between the Base Monitor service and the Web Server is encrypted using a self signed certificate.

Where does SQL Monitor store credentials for host machines and SQL Server instances?

When you install SQL Monitor, it creates a single Data Repository database in which all monitoring data, alert information and configuration settings are stored.

When you add servers to monitor, the login and password you provide for each host Windows machine and SQL Server instance are stored in settings tables inside the Data Repository.

Passwords are obfuscated before they are stored in the Data Repository.

Protecting the configuration file

The configuration file referenced above may contain password information in plain text if you specify SQL Server authentication as part of the connection string. You should ensure that unauthorized users are unable to view the contents of this file, for example, by denying then access to the folder.

The Base Monitor service account needs access to the configuration file.

Log files

There is no sensitive information logged in the log files created by the Base Monitor service or the Web Server.

Password for accessing the SQL Monitor website

When you first install and run SQL Monitor, you will be prompted to create a password that will be required for anyone accessing SQL Monitor web pages.

There are no complexity restrictions for the password.

Resetting your SQL Monitor password

To reset your SQL Monitor password, you will need to delete the current password stored in the SQL Monitor Data Repository database:

1. Log in to the SQL Server instance hosting your SQL Monitor Data Repository database.
In SQL Server Management Studio (or equivalent SQL editing tool) run the following query:

```
USE RedGateMonitor
GO
DELETE FROM settings.UserAccount
```

(where `RedGateMonitor` is the name of the database you are using for the Data Repository).

2. Paste the URL for SQL Monitor into your browser address bar, e.g. <http://MyServer.mydomain:8080>.

If your browser displays a message asking if you want to repost the data, ignore it. Reposting will enter the password you have just deleted.

3. The **Create password** page is displayed. Enter a new password.

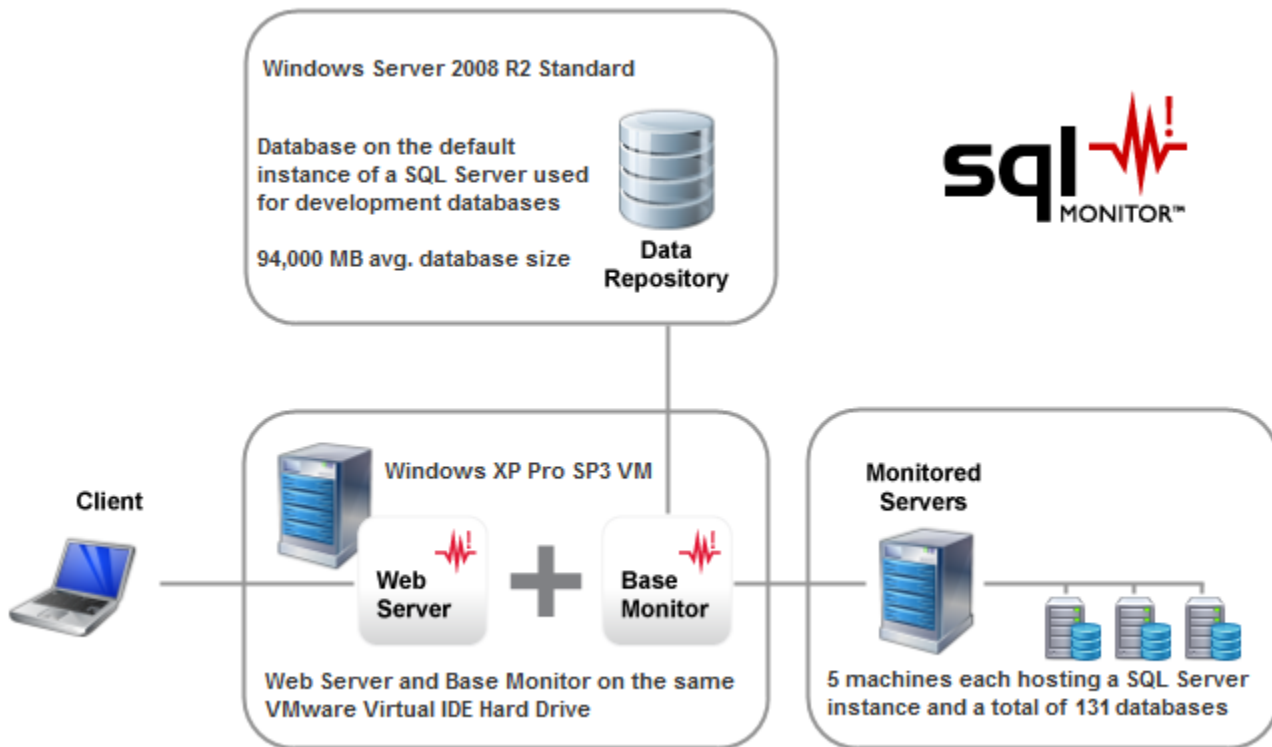
Worked examples

Learn more about using SQL Monitor by following these examples:

- [Example configuration using virtualization software](#)

Example configuration using virtualization software

The network configuration described below is a working example of how SQL Monitor has been deployed by John Henderson, DBA/Senior Programmer at Smith Travel Research (STR).



Base Monitor and Web Server configuration

The Base Monitor runs the service that monitors your machines and instances, and the Web Server delivers the pages for the SQL Monitor web interface. In this configuration, the Web Server and Base Monitor are on the same VMware Virtual IDE hard drive of a machine running Windows XP Pro SP3 VM with .NET 3.5 SP1 installed.

- Physical memory: There is 2 GB of physical memory on the machine. The RAM utilization is:
Base Monitor - RedGate.Response.Engine.Alerting.Base.Service.exe = ~320 MB
Web Server - RedGate.Response.UI.Server.Service.exe = ~16 MB
- Processor: There are 2 cores allocated to the VM, with the operating system handling processor distribution.

The Base Monitor service uses default TCP port 7399 to communicate with the Web Server.

The Web Server is SQL Monitor's own standalone XSP Web Server that uses .NET 3.5 runtime and default TCP port number 8080 for incoming connections. The client machines accessing the Web Server are using mostly Firefox 4.0 but occasionally Internet Explorer 8 and Chrome 12 Development Channel. Firefox is preferred because earlier version of SQL Monitor rendered much more consistently in this browser, although more recent versions render just as well on Chrome.

Data Repository configuration

The Data Repository is an MS SQL Server database that stores the data collected by the Base Monitor service. In this configuration, the Data Repository database runs as the default instance on a MS SQL Server 2005 64-bit (x64) Enterprise database. The machine is running Windows Server 2008 R2 Standard with .NET version 2.0 to 4.0 installed. There are other development databases hosted on this machine, but there is sufficient memory and processor capacity to accommodate this:

- Physical memory: The total available RAM is 8 GB
- Processor: There are 4 x 3 GHz dual cores available

Monitored servers

This configuration contains five machines each hosting a SQL Server instance with a total of 131 databases across all instances. A breakdown of machines, instances and databases being monitored is as follows:

- Machine A hosting 1 SQL Server instance and 14 databases
- Machine B hosting 1 SQL Server instance and 28 databases
- Machine C hosting 1 SQL Server instance and 37 databases
- Machine D hosting 1 SQL Server instance and 16 databases
- Machine E hosting 1 SQL Server instance and 36 databases

One of the machine is in a DMZ, so there is a firewall between it and the Base Monitor. TCP ports 135, 445 and 1433 have been opened to allow remote file access, remote registry and WMI access.

Performance, trace and purge settings

The following metrics obtained from SQL Monitor's Analysis page give an idea of how busy the SQL Server instances are on average in this network for the last 30 days. The equivalent PerfMon counter values are displayed in brackets (for more information, see [Using Performance Monitor](#)):

SQL Server: processor time	(Process (sqlservr): % Processor Time)	11.22%
SQL Server: user connections	(SQLServer: General Statistics: User Connections)	52.56
Transactions/sec	(SQLServer: General Statistics: User Connections)	8.26

The average metrics for each instance are as follows:

	SQL Server: processor time	SQL Server: user connections	Transactions/sec
Machine A	11.8%	23.5	11.5
Machine B	10.0%	87.4	02.3
Machine C	10.8%	52.7	09.7
Machine D	12.3%	77.5	05.1
Machine E	11.5%	21.6	12.7

Trace data collection is enabled and used to track down potential causes of blocks and long running queries, and for general troubleshooting.

Finally, the DBA wants to do 90 days' worth of trending, but they need to make sure that the Data Repository growth rate does not impact performance. The purge settings have been customized as follows:

- Machine Data: 3 months
- Alert Data: 2 weeks
- SQL Server Data: 3 months

Troubleshooting

Error messages

- SQL Monitor browser error: Cannot display webpage; connection timed out; webpage not available; could not locate remote server
- SQL Monitor connection error: Cannot generate SSPI context

Issues with IIS, WMI and ASP.NET

- Global Overview doesn't display any details in your browser
- Issues caused by clock skew
- tempdb size increases when clearing purge backlog

Technical questions

- Accessing SQL Monitor through a firewall
- How do I test data collection methods?
- Moving the Data Repository database
- Viewing component information and log files

Error messages

- SQL Monitor browser error: Cannot display webpage; connection timed out; webpage not available; could not locate remote server
- SQL Monitor connection error: Cannot generate SSPI context

SQL Monitor browser error: Cannot display webpage; connection timed out; webpage not available; could not locate remote server

If you are unable to view the SQL Monitor web page, and your browser displays one of the following messages:

- **Internet Explorer:** Internet Explorer cannot display the webpage
- **Firefox:** The connection has timed out
- **Chrome:** This web page is not available
- **Opera:** Could not locate remote server

Please check the following:

- The host machine name and port number are correct.
The URL should be in the format **http://mymachine.domain:8080/** or **http://127.0.0.1:8080/**.
- The SQL Monitor web server has been installed and the service is running.
Under local services on the web server machine, check that the **SQL Monitor Web Service** is started.
- If there is a firewall on the web server machine, ensure that it is configured to allow connections on the SQL Monitor port.
See [Accessing SQL Monitor through a firewall](#) for more information.
- You have internet connectivity.
Check that you can access other websites.

SQL Monitor connection error: Cannot generate SSPI context

If the status of a SQL Server instance displayed on the Monitored servers configuration page is **Connection failed - Cannot connect**, and the exception message in the log file contains "Cannot generate SSPI context", do the following:

Stop your SQL Server service

1. Open the Services Microsoft Management Console (MMC) (**Control Panel > Administrative Tools > Services**).
2. Double-click on the SQL Server service and on the General tab, click **Stop**.

Restart using the Local System account and then stop it again

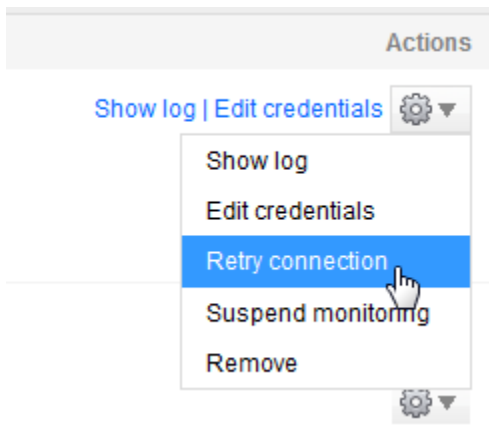
1. In the Log On tab, select **Local System account** and click **OK**.
2. On the General tab, click **Start**.
3. Once the Service status is confirmed as Started, click **Stop**.

Switch back to your domain account and restart

1. In the Log On tab, select **This account**.
2. Enter your SQL Server domain account details and click **OK**.
3. On the General tab, click **Start**.

Retry the connection in SQL Monitor

At the Monitored servers page (Configuration > Monitored servers), select **Retry connection** from the Actions list for the SQL Server instance:



For more information, see http://blogs.msdn.com/b/sql_protocols/archive/2005/10/15/481297.aspx.

Issues with IIS, WMI and ASP.NET

- Issues with IIS and ASP.NET
- Moving from SQL Monitor Web server to IIS
- WMI leaks memory on Windows 7 and Windows Server 2008 R2

Issues with IIS and ASP.NET

Troubleshooting ASP.NET

Installation

If you installed ASP.NET using the Add or Remove Programs dialog or the Configure Your Server Wizard, ASP.NET should be automatically enabled in IIS. If you installed it as part of another application, downloaded it from the Web, or installed it prior to installing IIS, you must enable ASP manually. Check that your IIS server has ASP.NET enabled before installing SQL Monitor.

Use the IIS Manager (see [How to Open IIS Manager](#)) to access and enable ASP.NET settings. For more information, see [ASP.NET and IIS Configuration](#).

Enable ASP.NET impersonation

IIS needs to provide a Windows identity in order to access protected Windows resources or certain files or database servers. ASP.NET needs access to these too, but by default it does not use the same Windows identity. You need to configure ASP.NET to impersonate the IIS Windows identity.

For details of how to do this in IIS 6, see [Impersonation in ASP.NET \(IIS 6\)](#).

For details of how to do this in IIS 7, see [Configure ASP.NET Impersonation Authentication \(IIS 7\)](#).

Security settings required

SQL Monitor requires the follow security permissions to be enabled in IIS 7:

- Anonymous authentication
- Forms authentication

In IIS 7.5, security has been tightened so that w3wp.exe (the IIS application pool process) runs by default under the ApplicationPoolIdentity account. In Windows Server 2008 R2 and Windows 7 this is a Managed Service Account. Managed Service Accounts are a new concept in Win 7 / Windows 2008 R2. For more information, read the following article:

<http://technet.microsoft.com/en-us/library/dd367859.aspx>

The ApplicationPoolIdentity account does not necessarily have access permissions to the SQL Monitor website files that are installed by default to C:\program files\red-gate\sql Monitor 3\website. You can fix this by manually giving permissions to the SQL Monitor website folder; to do this, run the following command:

```
C:\>icacls "c:\program files\red-gate\sql Monitor 3\website" /grant "IIS APPPOOL\Monitor":(OI)(RX)
```

IIS APPPOOL\Monitor is the Managed Service Account automatically generated for the "Monitor" application pool; the Monitor application pool is created automatically when adding a website called "Monitor".

In IIS Manager, go to the Monitor website, double click **Authentication** and then set **Anonymous Authentication** to use **Application Pool Identity** (i.e IIS APPPOOL\Monitor) rather than IUSR (IUSR is the default user identity for anonymous authentication).

Note: SQL Monitor will not work with any application pool using Classic Managed Pipeline Mode. You will need to ensure that SQL Monitor is added to an application pool running Integrated Pipeline mode.

IE error messages

By default, Internet Explorer replaces error messages with friendly versions that do not contain the details required to help diagnose problems in IIS 7. If you are receiving standard, "Internet Explorer cannot display the webpage", disable friendly HTTP error messages and enable IIS 7 detailed errors. For details of how to do this, see [Troubleshooting IIS7 errors like a pro](#).

HTTP Error 500.19 - Internal Server Error messages

The SQL Monitor web server may display the following message (where ### is the identifier for that specific error):

```
HTTP Error 500.19 - Internal Server ErrorHRESULT: 0x80070###
```

Description of HRESULT

The requested page cannot be accessed because the related configuration data for the page is invalid.

You can find possible causes and solutions for errors by checking the HRESULT identifier against the list here: <http://support.microsoft.com/kb/942055>.

Troubleshooting IIS 6

If SQL Monitor is not working on your IIS 6 server, check the following:

- Anonymous authentication is enabled.
- Wildcard mapping - Ensure you have set up wildcard mapping as described in [Using SQL Monitor with IIS](#).
- Port is different from the Default website port - Ensure that the port number specified for SQL Monitor is not the same as the number used for the Default website.
- The version of the SQL Monitor website process matches that of your Windows OS. Using the 32-bit version of ASP.NET on a system configured to support the 64-bit version may cause ASP.NET to become corrupted. Similarly, using the 64-bit version of ASP.NET on a system that is configured to support the 32-bit version would cause ASP.NET to become corrupted.

To run the SQL Monitor website as a 64 bit process on a 64 bit machine:

1. Run the following command on your web server:

```
cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SET  
W3SVC/AppPools/Enable32bitAppOnWin64 0
```

2. Under Web extensions in IIS Manager, set the 64 bit ASP.NET Web Service extension to **Allowed**, and set the 32 bit ASP.NET process to **Prohibited**.

To run the SQL Monitor website as a 32 bit process on a 64 bit machine (not recommended):

1. Run the following command on your web server:

```
cscript %SYSTEMDRIVE%\inetpub\adminscripts\adsutil.vbs SET  
W3SVC/AppPools/Enable32bitAppOnWin64 1
```

2. Under Web extensions in IIS Manager, set the 32 bit ASP.NET Web Service Extension to **Allowed**.

Moving from SQL Monitor Web server to IIS

If you already have SQL Monitor installed and are using the default XSP web server, you can later move hosting to IIS.

Note: Once you have installed IIS, make sure that ASP.NET is both installed and enabled. See [Issues with IIS and ASP.NET](#).

Configuring an existing IIS Web Server

Use IIS Manager (see [How to Open IIS Manager](#)) to carry out the changes described below.

Configuring IIS 7

1. Select **Add web site** and choose a name for the website.
2. Enter the physical path for the website folder - this is the location of the XSP website folder you specified during installation of SQL Monitor. By default this is:
`C:\Program Files\Red Gate\SQL Monitor 3\Website`
3. Specify an available port number and apply any other options, such as binding or certification that you need.
4. Click **Browse** to test that the website is working; you should see the SQL Monitor front page.

A new application pool is automatically created for the SQL Monitor website. If you want to add SQL Monitor to an existing application pool, edit the **Advanced properties** for the site to change its application pool.

Read more about setting up your IIS website ([Microsoft KB article](#))

Configuring IIS 6

1. Right-click **Web Sites** and click **New > Web site**.
2. In the Web Site Creation Wizard, type the name of the website (for example "SQL Monitor") in the **Description** box.
3. Leave the IP address of the IIS server as **(All Unassigned)**, and specify an available port (80 is the default port).
4. Leave the Host header box empty, unless you have a specific requirement for a host header.
5. In the Web Site Home Directory page, enter the path of the XSP website folder you specified during installation of SQL Monitor into the **Path** box. By default this is:
`C:\Program Files\Red Gate\SQL Monitor 3\Website`
6. Ensure that **Allow anonymous access to this Web site** is checked. SQL Monitor requires anonymous access to the website. For more information about website authentication, please read the following Windows KB article for more information: <http://support.microsoft.com/kb/324274> (How to configure IIS Web site authentication in Windows Server 2003).
7. Leave the default permissions on the Web Site Access Permissions page.

You also need to enable wildcard mapping, because SQL Monitor is an ASP.NET application that uses URLs without extensions. For details of how to do this, see [Using SQL Monitor with IIS](#).

Stopping the XSP Web Server

To stop the XSP server, display Windows Services (select **Start > Run**, enter `services.msc` and click **OK**) and stop **SQL Monitor Web Service**.

It is possible to keep both IIS and XSP web servers on the same machine. You can uninstall the SQL Monitor Web Service, but for testing purposes, it may prove useful to restart the XSP web server should you have any problems with the IIS configuration.

WMI leaks memory on Windows 7 and Windows Server 2008 R2

There is a known Microsoft issue where applications using the WIN32_Service WMI class suffer memory leaks when running on Windows 7 or Windows Server 2008 R2. Using SQL Monitor, you may experience:

- Up to 1 GB of wasted memory on the entity you are monitoring. There may also be problems collecting data from this entity using WMI.
- Monitoring connection errors on the Configuration > Monitored servers page.
- 'Network path not found' or 'Out of memory' exception messages on the Log page.

There is a hotfix available from Microsoft to fix this issue. See <http://support.microsoft.com/kb/981314>.

Unexpected behavior

- Global Overview doesn't display any details in your browser
- Issues caused by clock skew
- tempdb size increases when clearing purge backlog

Global Overview doesn't display any details in your browser

If the Global Overview page doesn't contain any data, please check the following:

For Internet Explorer on Windows Server 2008 and 2008 R2, ensure that the SQL Monitor website has been added to the list of allowed local intranet sites:

1. Go to **Tools > Internet Options > Security**.
2. Under **Zones**, select **Local intranet**.
3. Click **Sites** (For IE 8, click **Advanced > Sites**).
4. Add the SQL Monitor website. (This should be pre-selected.)

For all browsers: Check that Javascript is not disabled on your browser.

Issues caused by clock skew

Clock skew alerts are raised when the difference between the clock setting on the computer hosting the SQL Monitor Base Monitor and the monitored SQL Server host machine is greater than 15 seconds. Windows Server uses Windows Time service (W32Time) to maintain accurate date and time synchronization. For more information, see [How the Windows Time Service Works](#).

How clock skew can occur on virtual machines

The hardware clock used by the virtual machine's operating system is based on counting interrupts and ticks from the virtual hardware, which makes it susceptible to clock skew when the virtual machine or its host is under significant load. The techniques used to minimize timing performance differences can still cause inaccuracies and raise clock skew alerts.

How clock skew affects SQL Monitor

It is important to ensure that the machines hosting components of SQL Monitor, and the machines being monitored, have synchronized clock settings. If not, you may experience the issues listed below.

I can't log into my servers

Windows uses the Kerberos authentication protocol to verify the user requesting authentication and the server providing the requested authentication. The date and time set on the key distribution center (KDC) and on the client making the requests must be synchronized. If there is a significant time difference between the two, authentication can't function properly and you may be prevented from logging into your servers. For more information, see [http://technet.microsoft.com/en-us/library/cc780011\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc780011(WS.10).aspx).

SQL Monitor pages aren't displayed correctly

If any of the machines hosting components of SQL Monitor are out of sync, certain pages of SQL Monitor may display incorrect time stamps for performance data, or not display data at all. Pages that display details from multiple clocks, including the Global overview page and the Alert details page, are most likely to be affected.

An exception is raised when I click on an alert

If the clock setting on your SQL Server host machine is running ahead of the time set on your SQL Monitor Web Server, alerts will be raised with a timestamp set sometime in the future. How far in the future depends on how far out of sync the host machine is. For example, if the monitored machine is ahead by 5 minutes, an alert can be raised with a timestamp of 12:05 even though the time according to the Web Server is 12:00.

SQL Monitor can only support alerts with a timestamp equal to or earlier than the Web Server time. If you click on the alert before the Web Server time has reached the alert timestamp (between 12:00 and 12:05 in this example), an exception is raised. If you click on the alert after the Web Server time has reached the alert timestamp, SQL Monitor works as expected.

tempdb size increases when clearing purge backlog

Improvements were made to the way data was purged from the Data Repository database in SQL Monitor 2.2 and later. Purging is now performed as a faster, background task that reduces the affect on general performance of clearing a backlog of purged data. To facilitate this, SQL Monitor 2.2 and later has increased its usage of tempdb, so you may see tempdb grow by up to half the size of your Data Repository database when SQL Monitor is clearing a large backlog.

Once the backlog of data has been purged successfully, restart the SQL Server service and tempdb's size will revert back to normal.

A significant backlog of data that needs purging may build up if you perform any of the following actions:

- change your current purging policy to keep data in the Data Repository for a shorter amount of time
- update a large SQL Monitor database to version 2.2 from an earlier version
- remove a monitored server and delete all existing data for the server from the Data Repository

While the Data Repository database is busy with these actions, you may notice reduced performance while using the SQL Monitor interface.

Technical questions

- Accessing SQL Monitor through a firewall
- How do I test data collection methods?
- Moving the Data Repository database
- Viewing component information and log files

Accessing SQL Monitor through a firewall

If you are unable to view the SQL Monitor web page, and your browser displays a "cannot connect" type message, then you may need to check that SQL Monitor is not being blocked by the firewall on the web server machine.

If possible, install the Base Monitor on a server that does not need to go through a firewall to access the SQL Servers you want to monitor. If your network configuration prevents this, then we recommend that you use a VPN link between the Base Monitor and the monitored SQL Servers.

If your organization uses Network Address Translation (NAT), then you may not be able to monitor SQL Servers that are subject to it.

To allow access to SQL Monitor through the firewall, follow the steps below.

Allow access to TCP ports and WMI

SQL Monitor requires access to:

- TCP port 135 used by the Remote Procedure Call (RPC) service. Also make sure that the remote registry service is started on the server.
- TCP port 445 used by the Server Message Block (SMB) service that allows remote file access.
- TCP port 1433 used as the default registered address for the SQL Server Database Engine. If your SQL Server uses a different TCP configuration, use that port number instead.
- WMI. You will need to configure each server you want to monitor separately. See the following:
[Connecting Through Windows Firewall](#)

[Setting Namespace Security with the WMI Control](#)

[Security a Remote WMI Connection](#)

[Setting Up a Fixed Port for WMI](#)

For XP, Vista and Windows Server 2003 and 2008

1. Open the **Control Panel** on the machine where you installed the SQL Monitor Web Server.
2. Go to **Security Center** and click **Windows Firewall**.
3. Click **Change settings**.
4. Under **Exceptions**, click **Add port** and then enter a name and the port number. You can use any name to identify that this exception is for SQL Monitor.
The port number should be your SQL Monitor web server port number, specified during installation; this is 8080 by default.
5. Click **OK**.

For Windows 7

1. On the machine where you installed the SQL Monitor Web Server, open **Windows Firewall with Advanced Security**:
 - Type "Firewall" into the Search programs and files box, or
 - From the Control Panel, select **System and Security** then **Windows Firewall** then **Advanced settings**.
2. Click **Inbound Rules**.
3. Click **New Rule**.
4. Select **Port** as the type of rule you want to create, then click **Next**.
5. Choose **TCP** as the port type, then under **Specific local ports**, enter the SQL Monitor port number.
Your SQL Monitor web server port number is specified during installation; this is 8080 by default.
6. Click **Next**, then select **Allow the connection**.
7. Click **Next**, then choose the type of profile to apply the rule to.
8. Click **Next**, then enter a name to identify that this exception to your firewall is for SQL Monitor.
9. Click **Finish**.

Further information

For XP, see the following Microsoft KB article: <http://support.microsoft.com/kb/842242>

For Windows 7, see the following Microsoft article: <http://windows.microsoft.com/en-us/windows7/Allow-a-program-to-communicate-through-Windows-Firewall>

How do I test data collection methods?

SQL Monitor collects data from your monitored servers using the following methods:

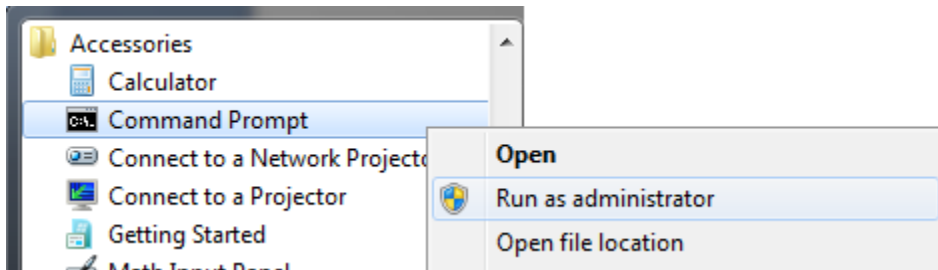
- Ping
- Remote registry (PerfMon & registry access)
- WMI
- Remote file access
- SQL

If you are experiencing connection issues in SQL Monitor, you can start to diagnose where the problem occurs by testing each type of connection independently.

Opening the command prompt

To test each data channel, you will first need to open a command prompt as an administrator:

1. Log in to the machine that is running the SQL Monitor Base Monitor service. See the **About** page (Configuration > About) for details of which machine is running the Base Monitor service.
2. Run the command prompt as an administrator:
 - a. From the **Start** menu, select **All Programs > Accessories**.
 - b. Right-click on Command Prompt and select **Run as administrator**.



- c. A security warning may be displayed. Click **Yes** to continue.

Testing ping

At the command prompt, run:

```
ping myserver.example.com
```

where `myserver.example.com` is the name of the Windows server you are attempting to monitor.

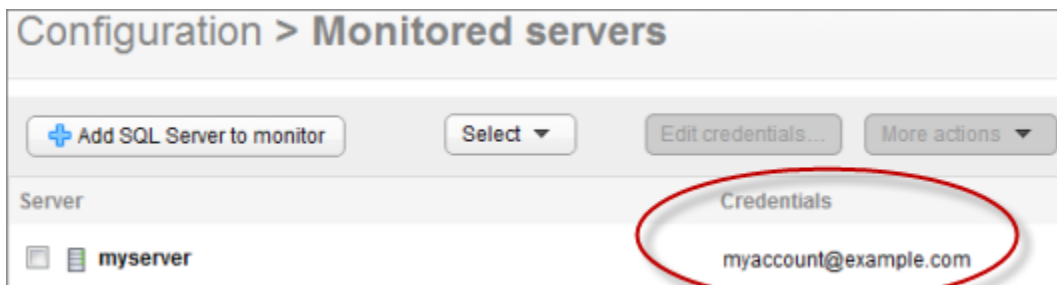
For more information, including details about the most common error messages that may be displayed, see [Testing Network Connections with Ping](#).

Testing remote registry (PerfMon)

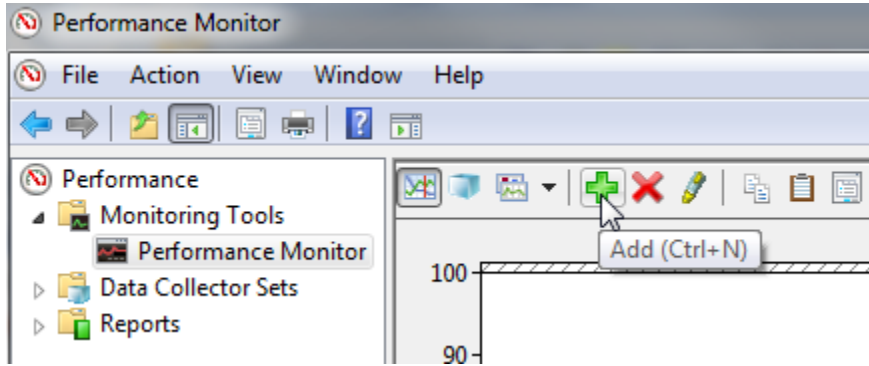
1. At the command prompt, run:

```
runas /netonly /user:example.com\myaccount "perfmon"
```

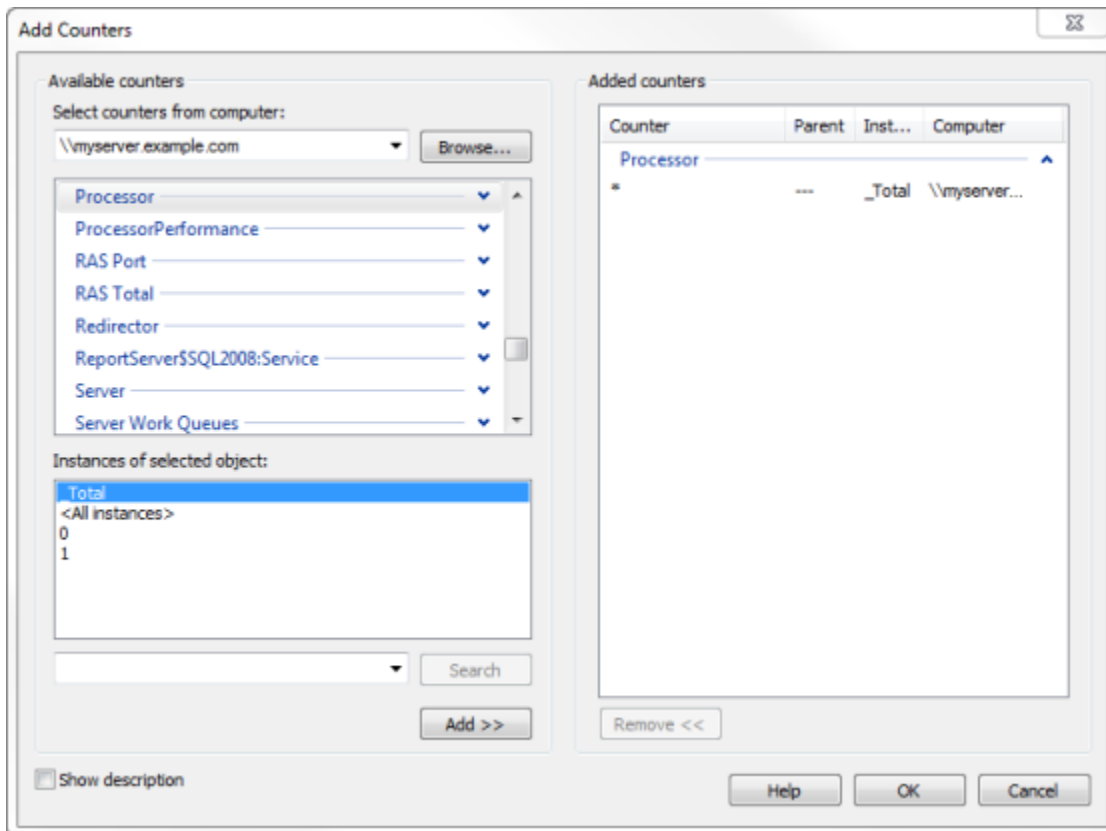
where `example.com\myaccount` is the Windows account you are using to monitor the server. This is the account listed under Credentials on the **Monitored servers** page in SQL Monitor (Configuration > Monitored servers), as shown below:



- When prompted, enter the password for this account.
- Click on Performance Monitor in the left pane, then click the **Add** button:



- In the **Select counters from computer** box, enter the name of the server you are attempting to monitor.
- By default, **Processor** is selected in the list of counters and **_Total** in the **Instances of selected object** box.
- Click **Add>>**. The selected counter appears in the Added counters box on the right:



- To check another server you are attempting to monitor, type its name in the **Select counters from computer** box and repeat the steps above.

The connection test can fail if the remote registry is not running, not responding, or the interprocess communications (IPC) pipe is blocked by a policy or registry change. The affected server will report either of the following errors:

- (PerfMon) Unable to connect to specified machine or machine is offline (-2147481648)
- (Registry) The network path was not found (53)

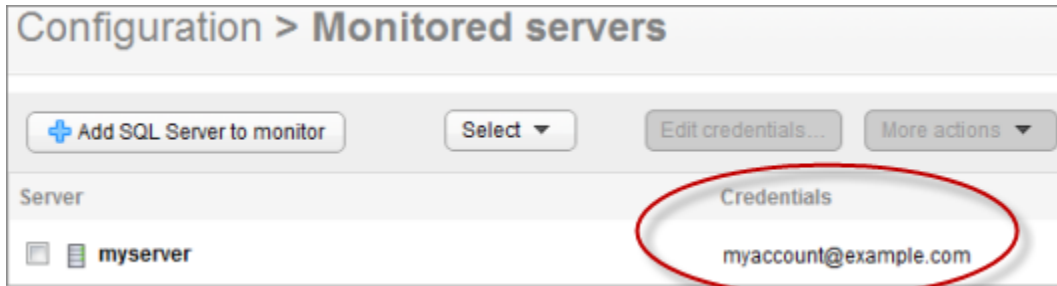
To troubleshoot this, try the following solutions in turn and retest the remote registry connection after each attempt:

- Check that the remote registry service is running. Click **Start > Control Panel > Administrative Tools > Services**, and make sure that the Remote Registry Startup Type setting is **Automatic**, and the Status is **Started**.
- Restart the remote registry service.
- Check that you have the necessary permissions to access the remote registry. For more information, see [How to Manage Remote Access to the Registry](#).
- Check that the Active Directory Domain Services do not contain a policy that blocks access to the Remote Registry service. See [Active Directory Domain Services](#).
- If your PerfMon library has become corrupt, rebuild it. For details, see <http://support.microsoft.com/kb/300956>.

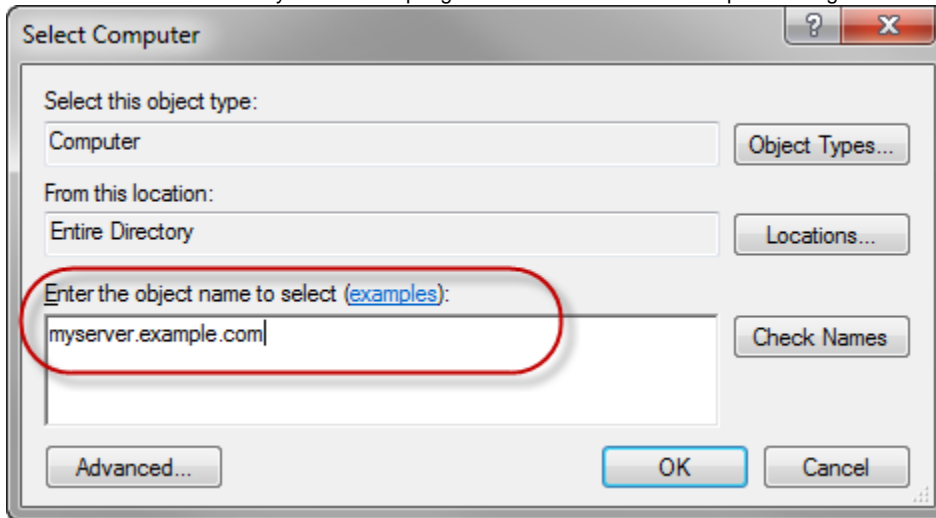
Testing remote registry

1. At the command prompt, run:
`runas /netonly /user:example.com\myaccount "regedit"`

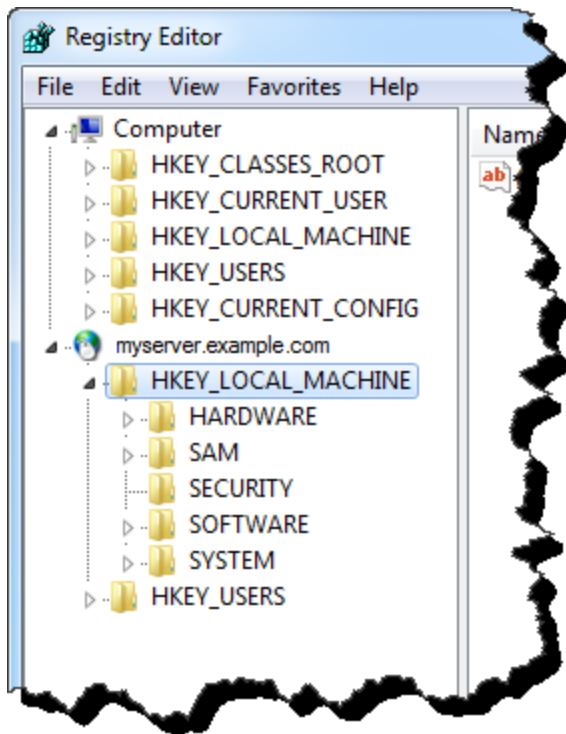
where `example.com\myaccount` is the Windows account you are using to monitor the server. This is the account listed under Credentials on the **Monitored servers** page in SQL Monitor (Configuration > Monitored servers), as shown below:



2. When prompted, enter the password for this account.
3. In the Registry Editor, select **File > Connect Network Registry**.
4. Enter the name of the server you are attempting to monitor in the Select Computer dialog box:



5. Click **OK**. The name of the server should be displayed in the left pane.
6. Beneath the name of the server, expand HKEY_LOCAL_MACHINE. You should see a list of registry keys:



7. Repeat the steps above for other servers you want to test.

The connection test can fail if the remote registry is not running, not responding, or the interprocess communications (IPC) pipe is blocked by a policy or registry change. The affected server will report the following error:

- The network path was not found (53)

To troubleshoot this, try the same solutions described for Testing remote registry (PerfMon) above.

Testing WMI

1. At the command prompt, run:
`wbemtest`
2. In the Windows Management Instrumentation Tester dialog box, click **Connect**.
3. In the **Namespace** box, type `\\myserver.example.com\root\cimv2` where `myserver.example.com` is the name of the Windows server you are attempting to monitor.
4. Under **Credentials**, enter the Windows account you are using to monitor the server and the password. This is the account listed under Credentials on the **Monitored servers** page in SQL Monitor (Configuration > Monitored servers).

Connect

Namespace: Connect
Cancel

Connection:

Using: ▼

Returning: ▼ Completion: ▼

Credentials

User:

Password:

Authority:

Locale:

How to interpret empty password

NULL Blank

Impersonation level

Identify

Impersonate

Delegate

Authentication level

None Packet

Connection Packet integrity

Call Packet privacy

5. Click **Connect**.
6. Click **Query** and enter the following query:

```
SELECT Name FROM Win32_Service
```

7. Click **Apply**. You should see a list of results similar to those shown below:

Query Result

WQL: SELECT Name FROM Win32_Service Close

128 objects max. batch: 10 Done

Win32_Service.Name="AeLookupSvc"

Win32_Service.Name="Alerter"

Win32_Service.Name="ALG"

Win32_Service.Name="AppMgmt"

Win32_Service.Name="aspnet_state"

Win32_Service.Name="Ati HotKey Poller"

Win32_Service.Name="ATI Smart"

Win32_Service.Name="AudioSrv"

Win32_Service.Name="BITS"

Win32_Service.Name="Browser"

Win32_Service.Name="CtSvc"

Win32_Service.Name="ClipSrv"

Add Delete

If connection fails and an "Access Denied" error message is displayed, see [WMI Troubleshooting and WMI Isn't Working!](#).

Testing remote file access

1. At the command prompt, run:

```
runas /netonly /user:example.com\myaccount "explorer"
```

where `example.com\myaccount` is the Windows account you are using to monitor the server. This is the account listed under Credentials on the **Monitored servers** page in SQL Monitor (Configuration > Monitored servers).

2. When prompted, enter the password for this account.
3. In the Explorer window address bar, type `\\myserver.example.com\c$` where `myserver.example.com` is the name of the server you are attempting to monitor. You should see the contents of that server's C: drive displayed.

If remote file access fails and an "Access Denied" error message is displayed:

- Check that the server you are attempting to monitor has security policy permissions set to allow remote access.
- Check that the files you are attempting to access on the server's C: drive have file sharing permissions set to allow access.

Testing SQL connection

You can test SQL connectivity using SQL Server Management Studio.

Testing SQL connection using SQL Server authentication:

1. From the **Start** menu, select **All Programs > Microsoft SQL Server 2005/2008/2012 > SQL Server Management Studio**.
2. At the Connect to Server dialog, enter the fully qualified name of the SQL Server instance you are attempting to monitor in the **Server name** box.
3. Select **SQL Server Authentication**.
4. Enter the SQL Server login and password you are using to connect to the SQL Server instance and click **Connect**.

Testing SQL connection using Windows authentication:

1. From the **Start** menu, select **All Programs > Microsoft SQL Server 2005/2008/2012** to display a sub-menu.
2. Hold down the Shift key, right-click on **SQL Server Management Studio** and select **Run as different user**.
3. At the Windows Security dialog, enter the Windows user name and password you are using to connect to the SQL Server instance and click **OK**.
4. At the Connect to Server dialog, enter the fully qualified name of the SQL Server instance you are attempting to monitor in the **Server name** box, and click **Connect**.

If connection fails, one of the following error messages is displayed:

- Timeout expired
- An existing connection was forcibly closed by the remote host
- No process is on the other end of the pipe
- Login failed for user <x>

A connection failure may also occur if the system administrator privileges used to access the database engine are inadvertently deleted. To resolve these issues, see [Troubleshooting Database Engine Connectivity](#).

Moving the Data Repository database

If you want to move the SQL Monitor Data Repository, follow these steps:

1. Stop the Base Monitor service.
To do this, go to Services (select **Start > Run**, enter **services.msc** and click **OK**) and stop **SQL Monitor 3 Base Monitor**.
2. In your SQL Server management tool, back up your current Data Repository database and then restore it to your preferred location. If you did not change the default name when installing, the Data Repository database will be called **RedgateMonitor**.
3. Edit the SQL Monitor configuration file to update the connection string that locates the SQL Monitor database.
Navigate to the folder you specified for the configuration file during installation. By default, this is C:\Documents and Settings\All Users\Application Data\Red Gate\SQL Monitor 3 for Windows XP and Windows Server 2003, or C:\ProgramData\Red Gate\SQL Monitor 3 in Windows Server 2008, Windows Vista and Windows 7. Open **RedGate.Response.Engine.Alerting.Base.Service.exe.settings.config** in a text editor or XML editor. The Data Repository database is specified in the **<connectionStrings>** section near the end of the file:

```
<connectionStrings><add name="DataConnectionString" connectionString="Data
Source=MACHINENAME\SQLInstancename;Initial Catalog=RedgateMonitor;Integrated
Security=True;Application Name=&quot;SQL Monitor - Repository&quot;" />
</connectionStrings>
```

The Data Repository comprises a single database schema. The **DataConnectionString** entry specifies the database that will contain all SQL Monitor monitoring and alert data.

By default, this entry points to the Data Repository database that you specified during installation. To move the Data Repository to a different database, edit this entry to point to a different Data Source (SQL Server instance) and Catalog (database name). For example:

```
<add name="DataConnectionString" connectionString="Data
Source=SECUREMACHINENAME\MyInstance;Initial Catalog=RedgateMonitor;Integrated
Security=True;Application Name=&quot;SQL Monitor - Repository&quot;" />
```

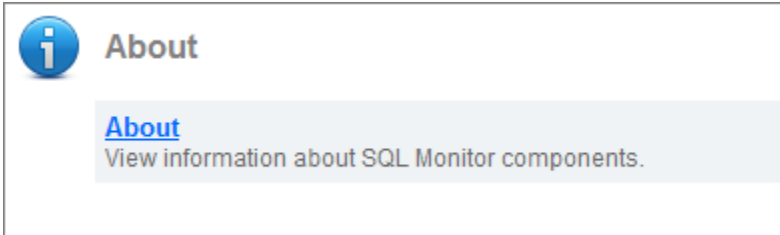
4. Save the configuration file.
5. Restart the **SQL Monitor 3 Base Monitor** service.
6. Launch SQL Monitor in your browser. To check that your change has been made, go to the **About** page (Configuration > About). Under **Base Monitor**, ensure that the settings for **Database** have been updated to the new database name.

Viewing component information and log files

Log files collect information about SQL Monitor while you are using it. These files are useful to us if you have encountered a problem.

Viewing log files

Go to the **Configuration** tab. Under **About**, select **About**:



The page displays useful information about the Web Server, Base Monitor and Client Browser.

Click the **View Web server log file** or **View Base Monitor log file** links to display detailed logs of the activities for each of these SQL Monitor components. Should you need to contact Red Gate support, they may ask you to email log details to them. Attach a copy of the original log file to your email. This maintains the initial log format, which can be checked more quickly and easily than reformatted content.

Configuration > About

Web Server

- Version: 2.1.0.153
- Web Server: Mono.WebServer2/0.2.0.0
- Server Name: http://WAR:8080
- Server HOST .NET: .NET 3.5 SP 1
- Server Path: c:\Program Files\Red Gate\SQL Monitor 2
- Hosted Environment: 32Bit
- User: NT AUTHORITY\NETWORK SERVICE

[View Web Server log file](#)

Base Monitor

- Version: 2.1.0.153
- Base Monitor: WAR.TESTNET.RED-GATE.COM:7399
- Install Location: c:\Program Files\Red Gate\SQL Monitor 2
- Host Environment: 32Bit
- User: RED-GATE\sqlresponsetest
- Domain: RED-GATE
- Database: CONQUEST.TESTNET.RED-GATE.COM

[View Base Monitor log file](#)

Client Browser

- Client Agent: Mozilla/5.0 (Windows; U; Windows NT 6.1

When you click on a link to view the log file, the contents of the log file are displayed in your browser, and a link to the location of the file on the Base Monitor computer is displayed.

Configuration > Web Log File

Local file name: C:\ProgramData\Red Gate\Logs\SQL Monitor 2\Website.log

```
[Red Gate Monitor (2.1.0.178), Running as: NT AUTHORITY\NETWORK SERVICE (Service), Working Dir: C:\ProgramData\Red Gate\Logs\SQL Monitor 2\Website.log]
[Running on: BRIANH2 (brianh2.red-gate.com), OS Version: Microsoft Windows NT 6.1.7600.0, .NET Framework 4.0.30319.1]
[Start Date: 2010-11-30 10:49:31Z, Current Date: 2010-11-30 11:38:13Z, Elapsed Time: 00:48:42.866]
2010-12-01 09:32:02,467 [ 9] WARN RedGate.Response - Getting Log file
2010-12-01 11:06:59,404 [ 10] WARN RedGate.Response - Getting Log file
```

Viewing connection information for monitored servers

If you have connection problems when monitoring some of your servers, you can view logged connection errors:

1. Go to the **Configuration** tab and under **Monitoring**, select **Monitored servers**.



Monitoring

Monitored servers

Add servers to monitor; edit connection credentials; remove or suspend monitored servers; configure maintenance windows.

Trace

Enable or disable Profiler trace on selected servers.

Groups

Organize your monitored servers into groups.

2. For the server you want to check, click the inline **Show log** link.
3. By default, only recent errors are displayed. Select **All events** to view all recent connection events for the selected server:

Configuration > Monitored servers > myserver

Status: ✘ Connection failed (Bad data)

Show: Errors All events

Date	Group	Event	Outcome	Exception	Exception message
14 Oct 2010 10:34 AM	General	Perfmon	Bad data	#Rrv	Perfmon data is missing objects: SQLServer.Access Methods, SQLServer.Buffer Manager, SQLServer.Databases, SQLServer.General Statistics, SQLServer.Latches, SQLServer.Locks, SQLServer.Memory Manager, SQLServer.SQL Statistics. Possible causes include performance counter library corruption, or a 32-bit/64-bit mismatch between Windows and the performance counter provider (e.g. SQL Server).

Release notes and other versions

Version 5.0	January 6th, 2016 (latest)	Release notes	Documentation	
Version 4.3	June 18th, 2015	Release notes	Documentation	
Version 4.2	May 27th, 2015	Release notes		
Version 4.1	April 7th, 2015	Release notes		
Version 4.0	January 8th, 2014	Release notes		
Version 3.10	May 27th, 2015	Release notes		
Version 3.5	July 23rd, 2013	Release notes	Documentation	
Version 3.4	May 21st, 2013	Release notes		
Version 3.3	March 25th, 2013	Release notes		
Version 3.2	August 7th, 2012	Release notes		
Version 3.1	May 24th, 2012	Release notes		
Version 3.0	March 22nd, 2012	Release notes		
Version 2.3	May 26th, 2011	Release notes		Documentation
Version 2.2	February 16th, 2011	Release notes		Documentation (PDF)
Version 2.1	December 10th, 2010	Release notes	Documentation (PDF)	
Version 2.0	November 4th, 2010	Release notes	Documentation (PDF)	
Version 1.3	May 20th, 2009	Release notes	See SQL Response documentation	
Version 1.2	March 18th, 2009	Release notes		
Version 1.1	November 11th, 2008	Release notes		
Version 1.0	September 30th, 2008	Release notes		

If you need to install an old version of SQL Monitor, go to [Download old versions of products](#).

SQL Monitor 2.3 release notes

May 26th, 2011

New features

- Top 10 expensive queries – the 10 queries that use the most resource based on Execution Count, Duration, CPU Time, Physical Reads, Logical Reads and Logical Writes are now displayed on the SQL Server Instance and Database overview pages. This data is also available in the Alert Details page for SQL Server alerts.
- Data purging – SQL Monitor now supports explicit purge options for high volume data channels. **Note:** Purge options that were customized in earlier versions will be translated accordingly. You should review your purge settings after upgrade.
- Analysis page – The Analysis page now supports high resolution graphs and can plot a maximum of 500 data points.
- Performance improvements have been made to the Base Monitor and UI.
- SQL Monitor now works with [PagerDuty](#).
- Emails for *Job duration unusual* and *Job failure* alerts now contain job name.
- *Monitoring error (host machine data collection)* and *Monitoring error (SQL Server data collection)* alerts now have a two minute time tolerance before they are raised.
- *Long running query* alert now contains database name and query duration.
- Behavior change for *Job duration unusual* alert. It now validates the “Ignore less than” value against the current run job duration instead of the base line.

Fixes

- **SRP-3645:** *Monitoring error (host machine data collection)* and *Monitoring error (SQL Server data collection)* alerts are too sensitive.
- **SRP-3614:** Cannot set a SQL connection port greater than 32767.
- **SRP-3483:** *Processor utilization* alert can be raised for disabled alert levels.
- **SRP-2905:** Include query duration for Long running query alert.
- **SRP-3688:** Analysis graph plots data but not the lines when the custom time range goes further back than the data available..
- **SRP-3036:** Trace statements in the SQL Processes / Profiler trace tab are not chronologically ordered.
- **SRP-2969:** Trace information in 2.2 less than 1.3.
- **SRP-3090:** Offline alerts: include seconds on alert timings.
- **SRP-2472:** *Job failed* and *Job duration unusual* alert email does not contain job name.
- **SRP-2924:** Allow more than 100 percent for job duration unusual configuration.
- **SRP-3399:** *Blocked process* and *Long running query* alert pages do not display database name.
- **SRP-3586:** *Log backup overdue* alert is being raised wrongly after maintenance window.
- **SRP-1294:** The list of SQL processes in an alert does not include the process which caused the alert.

Known issues

- **SRP-495:** If the Base Monitor loses its connection to the SQL Monitor Data Repository then you may see the following error, “System.Data.SqlClient.SqlException was thrown by method Query on service DataPresenterService: Timeout expired. The timeout period elapsed prior to completion of the operation or the server is not responding”.
- **SRP-1443:** An internal error occurs if the database name contains a non-latin character.
- **SRP-2167:** A user will see a “script stack space quota is exhausted” JavaScript error on Firefox when there is a lot of performance data in an alert details page. Use an alternative browser to view SQL Monitor.
- **SRP-2310:** “Back in time” can sometime crash with an error when using IE 7. Use an alternative browser to view SQL Monitor.
- **SRP-2702:** Long running query process ID not appearing in the SQL Processes/Trace tab.

SQL Monitor 2.2 release notes

February 16th, 2011

New features

- Maintenance window - you can now set maintenance windows to temporarily suspend alerting during set periods.
- Four new alerts - you will now receive alerts when connection issues occur.
- *Machine offline* and *SQL Server instance offline* alerts have been renamed *Machine unreachable* and *SQL Server instance unreachable* respectively.

Fixes

- **SRP-2134:** The *Fragmented index* alert is broken.
- **SRP-2927:** Alert descriptions for *Backup overdue*, *Log backup overdue* and *Integrity check overdue* alerts need changing.
- **SRP-3135/ SRP-3136:** The time displayed in the Occurrences tab of the Alert details page is out by 12 hours or is in the wrong format.
- **SRP-3018:** Purge behavior does not work as expected; a backlog of data due to be purged builds up, and attempts to purge the backlog affects performance. To fix this, SQL Monitor has increased its usage of tempdb, so you may see tempdb grow by up to half the size of your Data Repository database. Once the backlog of data has been purged successfully and your SQL Server service has been restarted, tempdb size will revert back to normal.
- **SRP-3140:** The *Disk used* alert description is not entirely accurate.
- **SRP-3090:** Offline alerts: include seconds on alert timings.
- **SRP-2924:** Allow a threshold of more than 100% for the *Job duration unusual* alert configuration.
- **SRP-2202/ SRP-1769:** The availability of button in the Alert Inbox should not be determined by the visible page.
- **SRP-1137:** Enable sorting of the SQL Process by ID to make it easier to find a particular process.
- **SRP-3243:** On the Alert details page, trace data is retrieved only when users click a specific SQL process.

Known issues

- **SRP-495:** If the Base Monitor loses its connection to the SQL Monitor Data Repository then you may see the following error, "System.Data.SqlClient.SqlException was thrown by method Query on service DataPresenterService: Timeout expired. The timeout period elapsed prior to completion of the operation or the server is not responding".
- **SRP-1443:** An internal error occurs if the database name contains a non-latin character.
- **SRP-2167:** A user will see a "script stack space quota is exhausted" JavaScript error on Firefox when there is a lot of performance data in an alert details page. Use an alternative browser to view SQL Monitor.
- **SRP-2310:** "Back in time" can sometime crash with an error when using IE 7. Use an alternative browser to view SQL Monitor.
- **SRP-2702:** Long running query process ID not appearing in the SQL Processes/Trace tab.

SQL Monitor 2.1 release notes

December 10th, 2010

New features

- License deactivation is now supported.
- Long running query alert now supports excluding by process name.
- Improvements to email format.
- Improvement to Job duration unusual alert.
- Performance improvement when trace is running.
- Long running query: Regex on new lines are treated as distinct new regex rather than a single regex with a new line in it.

Fixes

- **SRP-2126/ SRP-2522:** Database creation fails during installation and displays the following error message, "invalid sequential schema version". This can happen when there are user objects in the model database.
- **SRP-1891:** The breadcrumbs for any job-specific alert configuration, e.g., "Job failed" alert, contain the name of the instance twice.
- **SRP-753/ SRP-425:** Sometime XSP webserver crashes with socket exception.
- **SRP-2547:** Trace not working as expected for SQL Server 2000.
- **SRP-2624:** "Database **** cannot be opened. It is in the middle of a restore." was wrongly categorized as an authentication error.
- **SRP-2489:** Analysis page not plotting Total Size, Data Size and Log Size graphs.
- **SRP-2398:** Invalid default SQL Server Log path causes connection failure.
- **SRP-2475:** TypeAccessException thrown when viewing alert inbox.
- **SRP-2604:** Installer fails to start website service on French Windows XP & 2003 when on a work group.
- **SRP-2512:** SQL fragment truncated for Long running query, Blocked process, Deadlock and SQL Server error log entry alert.

Known issues

- **SRP-495:** If the Base Monitor loses its connection to the SQL Monitor Data Repository then you may see the following error, "System.Data.SqlClient.SqlException was thrown by method Query on service DataPresenterService: Timeout expired. The timeout period elapsed prior to completion of the operation or the server is not responding".
- **SRP-1443:** An internal error occurs if the database name contains a non-latin character.
- **SRP-2167:** A user will see a "script stack space quota is exhausted" JavaScript error on Firefox when there is a lot of performance data in an alert details page. Use an alternative browser to view SQL Monitor.
- **SRP-2310:** "Back in time" can sometime crash with an error when using IE 7. Use an alternative browser to view SQL Monitor.
- **SRP-2702:** Long running query process ID not appearing in the SQL Processes/Trace tab.
- **SRP-2735:** The Fragmented index alert is broken in build 2.1.0.380. To obtain a patch to fix this problem, contact [Red Gate support](#).

SQL Monitor 2.0 release notes

November 4th, 2010

Known issues

- If the Base Monitor loses its connection to the SQL Monitor Data Repository then you may see the following error, "System.Data.SqlClient.SqlException was thrown by method Query on service DataPresenterService: Timeout expired. The timeout period elapsed prior to completion of the operation or the server is not responding".
- A user will see a "script stack space quota is exhausted" JavaScript error on Firefox when there is a lot of performance data in an alert details page. Use an alternative browser to view SQL Monitor.
- Database creation fails during installation and displays the following error message, "invalid sequential schema version". This can happen when there is a user object in the model database.
- The breadcrumbs for any job-specific alert configuration, e.g. "Job failed" alert, contain the name of the instance twice.
- "Back in time" can sometime crash with an error when using IE 7. Use an alternative browser to view SQL Monitor.

SQL Response 1.3 release notes

May 20th, 2009

New features

- Alert Repository performance improvements.
- Recommendations are now checked every five hours.
- SQL Response can now handle a very high number of SQL and Win Processes.

Fixes

- Fix for generic GDI+ error.

SQL Response 1.2 release notes

March 18th, 2009

Fixes

- Improved WMI support.
- Reduced load on monitored SQL Servers.
- Data grids in the Alert and Recommendation details can now be sorted.
- Deadlock trace flag is now set only when the deadlock alert is enabled.

SQL Response 1.1 release notes

November 11th, 2008

Fixes

- Trial licensing now works with time stamps.
- Email links now work in all situations.
- Performance counters are correctly reconnected when box is restarted
- Installer for SQL Response Alert Repository now allows "LocalSystem".

SQL Response 1.0 release notes

September 30th, 2008

Original release.

Learn more about database administration

This page lists articles which can help you learn more about database administration and monitoring your database.

[SQL Monitor's data repository: Monitoring data \(Simple Talk\)](#)

[SQL Monitor - Using the performance data smartly \(Simple Talk\)](#)

[Monitoring your business, not just your servers \(Red Gate webinar\)](#)

[SQL Monitor Metrics - online resource for custom metrics \(sqlmonitormetrics.com\)](#)

[The Ten Commandments of SQL Server Monitoring \(Simple Talk\)](#)

Navigating in SQL Monitor

Top navigation bar

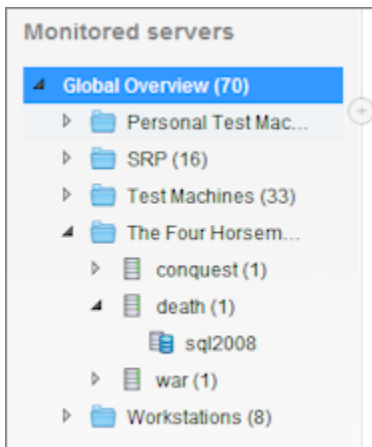
SQL Monitor contains four main areas, shown as tabs across the top of the screen:




When you click on a tab in the top bar, the landing page for the selected area is displayed. For example, clicking on Overviews takes you to the Global Overview, and clicking on Alerts takes you to the Alert Inbox. You can then drill down to view more detailed information.

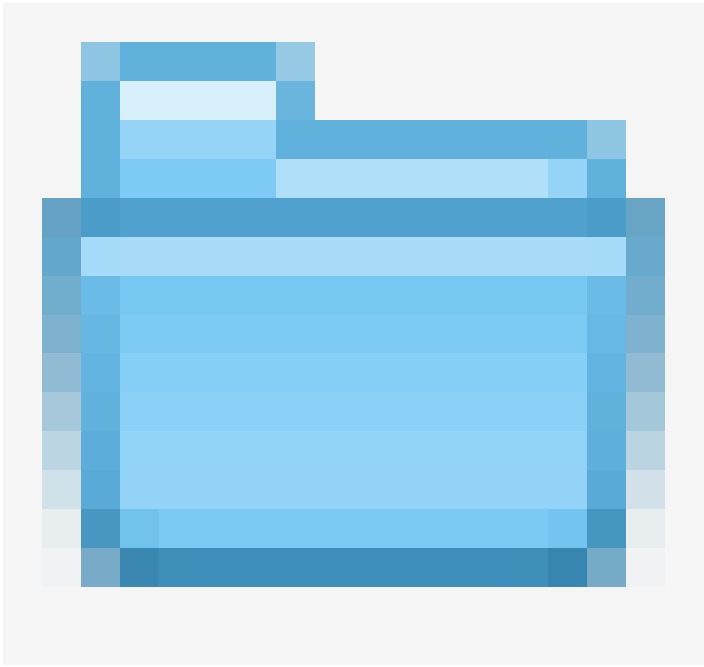
Note: If you have clicked on a different tab, and want to return to your previous page instead of the landing page, use your browser's **Back** button. Clicking on a top level tab always starts you at the landing page. You can open pages in new tabs or windows if you want to view multiple pages simultaneously.

Monitored servers list

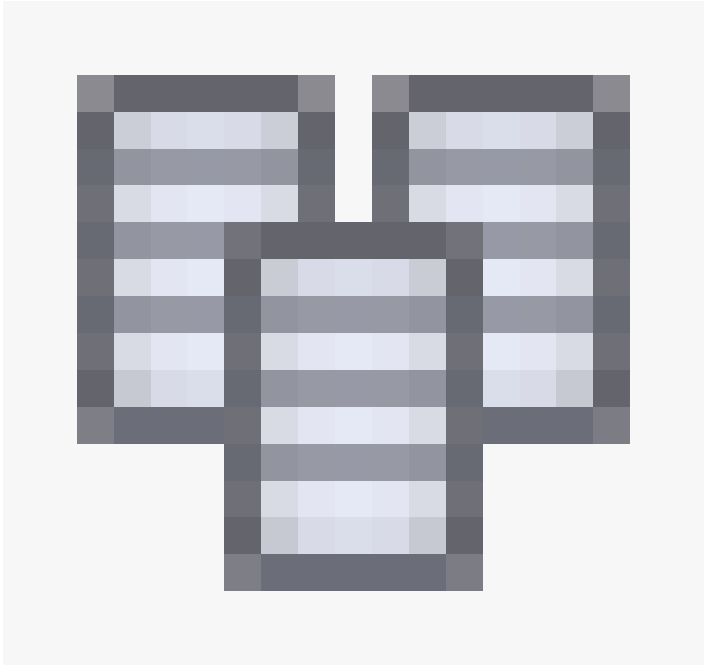


The servers you are monitoring, and any groups you have created, are shown on the left side of the screen in various places in SQL Monitor: the overviews, the Alert Inbox and the alert settings pages.

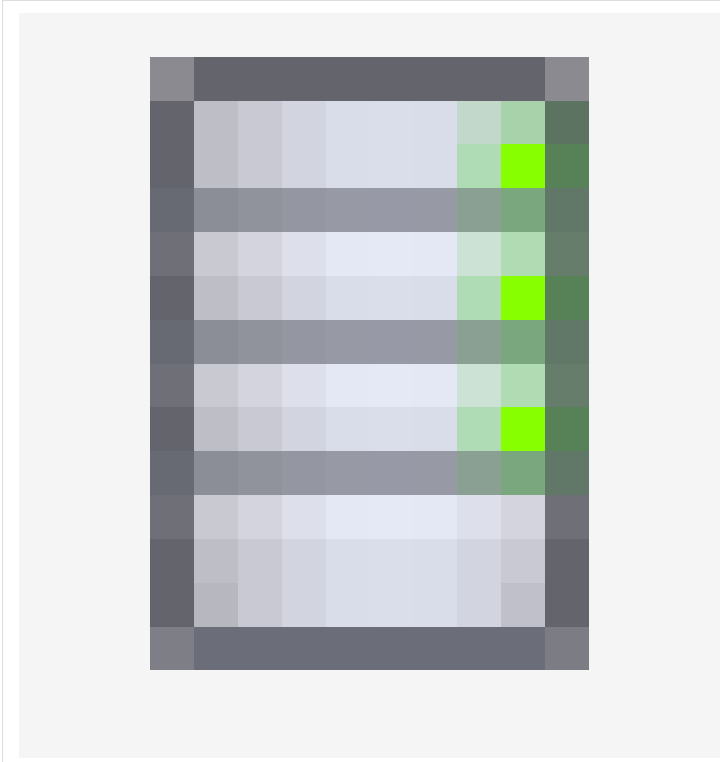
Click on an object in the list to view the relevant overview or show only alerts/alert settings for the selected object. Click the arrow  next to an object name to view levels beneath.



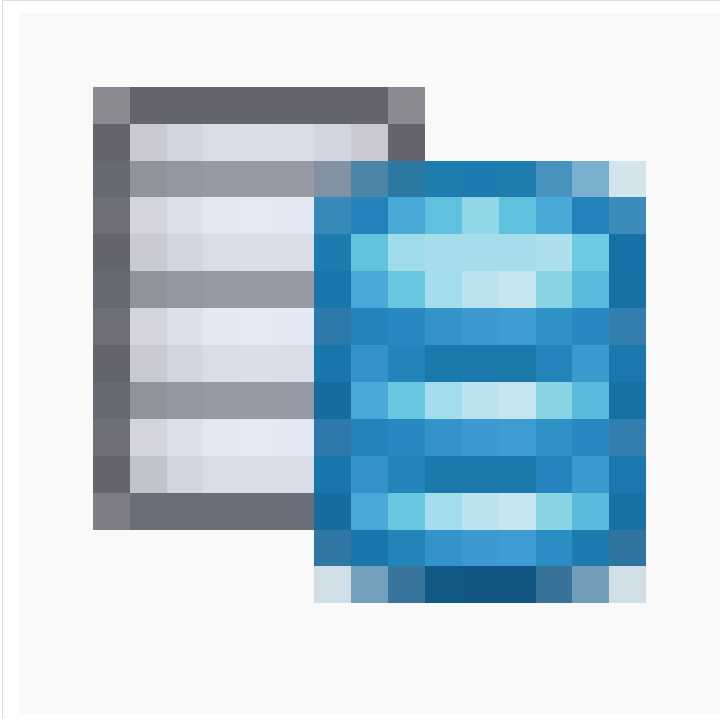
Group



Cluster




Host machine (Windows server)



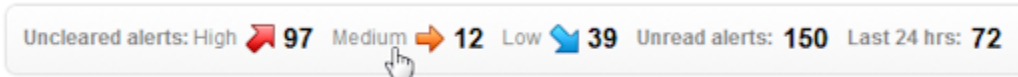
SQL Server instance

- In the Alert Inbox, you can continue to drill down to view databases. Clicking on a database shows alerts for the selected database.
- When configuring alerts, you can also drill down to view individual jobs.
- On the overviews, the number (n) after a group or server name shows the number of objects beneath. In the Alert Inbox, the number in brackets indicates the number of **unread** alerts.

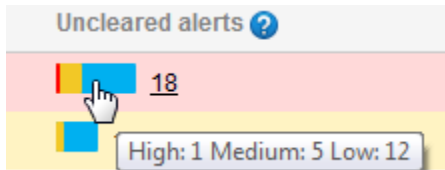
Tip: If the names of your servers are truncated because they are too long, click  to expand the width of the left pane.

Navigating in the Global Overview

- Click on any object (server or database) in the **Monitored servers** list to view the Overview page for that object.
- Click on a group to filter the Global Overview to show only servers in the selected group.
- Click on the name of a host machine or SQL Server instance in the **Servers** column to go to the Overview page for that server. You can also click in the **Status** column.
- Click on a number in the alerts summary panel at the top of the page to go to the Alert Inbox pre-filtered by your selection (e.g. High or Unread) across all servers.



- Click on the colored bar graph under **Uncleared alerts** to go to the Alert Inbox to view alerts for that host machine or SQL Server instance:

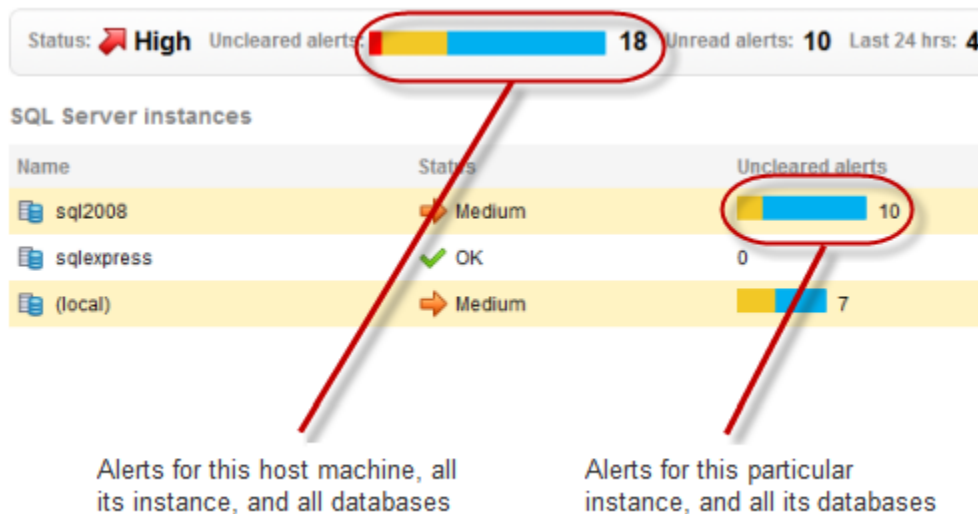


- Click on a sparkline graph for Processor time or Memory Used to go the Analysis page for those counters



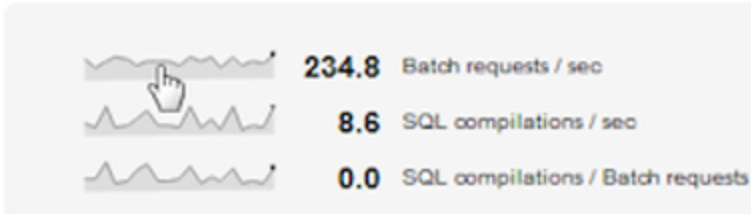
Navigating in a server or database overview page

- For a host machine, its monitored SQL Server instances are listed. Click in the **Name** or **Status** column to drill down to the SQL Server overview page.
- For a SQL Server instance, all its databases are listed. Click in the **Name** or **Status** column for a database to drill down to the database overview page.
- Click on **Uncleared alerts**, **Unread alerts** or **Last 24hrs** in the alerts summary panel at the top of the page to go to the Alert Inbox for all alerts relating to this object and everything below it in the hierarchy (e.g. for a host machine, it will show alerts for the machine as well as SQL Server instances and databases).
- Click on **Uncleared alerts**, **Unread alerts** or **Last 24hrs** for a SQL Server instance or database to go to the Alert Inbox for alerts relating only to the specific object (e.g. a SQL Server instance) and levels below it, if applicable:



- Click on any sparkline graph to go to the Analysis page for the selected metric:

General



Navigating in the Alert Inbox

In the Alert Inbox, you can filter the alerts that are displayed in two ways:

- Select a server or database in the **Monitored servers** list on the left, and
- Filter which alerts are shown, using a Global filter or a filter in the Filter panel

These two filters work in combination. For example, you can view all High level unread alerts on one instance raised in the last hour, or all Cleared alerts across all servers.

The number in brackets after a server or database name in the Monitored servers list always shows the number of **unread** alerts, regardless of the current filters applied. As you drill down in the hierarchy, the number of unread alerts at each level is displayed:

Monitored servers

- All Servers (2948) — total number of unread alerts
- Doc test
- Personal Test Machines (614)
- SRP (669)
- Test Machines (1124) — unread alerts in this group
 - behave.testnet.red-gate.com (59)
 - homer.testnet.red-gate.com (185) — unread alerts for this server, all its instances and the databases on each instance
 - (local) (49)
 - sql2005 (38)
 - sql2008 (59)
 - sqlexpress (8) — unread alerts for this instance and all its databases
 - master (1)
 - model (1)
 - msdb (1)
 - tempdb (2) — unread alerts for just this database
 - itanium2.testnet.red-gate.com (82)
 - lisa.testnet.red-gate.com (112)

Click on any level to filter the list of alerts for that level and all levels below. (The alerts displayed will also depend on the filters you have selected.)

Viewing more alerts

When you first install SQL Monitor, the Alert Inbox shows a maximum of ten alerts per page. You can navigate through this list using the **Newer** and **Older** links:

Showing 1-10 of 52 << Newest < Newer Older > Oldest >>

The total number of alerts shown (...of n) is useful to indicate how many alerts match the current set of filters. Note that this will not necessarily match the number of unread alerts in the Monitored servers list.

To increase the number of alerts shown, use the **Rows to display** option below the inbox:

Rows to display: 10 ▾

Showing 1-10 of 52

The **Rows to display** value is stored as a cookie in your browser.

Viewing alert details

To view the details for an alert, click anywhere in the row for that alert:

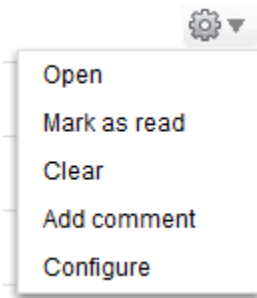
<input type="checkbox"/>	High Deadlock	conquest(local)	Event	5:45 PM 26 Oct
<input type="checkbox"/>	Low Physical memory	conquest	Ended	9:41 AM 26 Oct

The alert opens in its own page, where you can review the detailed alert information and performance data. You can also apply various actions, for example, mark it as Cleared or add a comment.

Click



in the **Actions** column to apply an action to an alert:



If you click **Clear** or **Mark as read**, the alert may disappear from the list, depending on your current set of filters.