Running Uptime Infrastructure Monitor with an MS SQL Server database

Contents

The up.time DataStore can run on a Microsoft SQL Server database to leverage existing SQL Server installations and take advantage of SQL Server's advanced replication, recovery and archiving options. This article assumes that you have already installed Uptime Infrastructure Monitor and have a SQL Server database available (see Supported Databases for version details).

Step 1 - Preparing your MS SQL Database

To run Uptime Infrastructure Monitor with a SQL Server database, first create a database on your SQL Server default instance that Uptime Infrastructure Monitor will be able to use. If you are unsure which databases are available to you or how to create a new database, please contact your database administrator and have a database configured for Uptime Infrastructure Monitor.

Within the SQL Server database that will hold Uptime Infrastructure Monitor configuration and historical data, create a user account that Uptime Infrastructure Monitor will use to access the database. The settings that you define for the Uptime Infrastructure Monitor database user are generally flexible with the following exceptions:

- The Uptime Infrastructure Monitor user must be the owner of the database that Uptime Infrastructure Monitor will use.
- The Uptime Infrastructure Monitor user must allow SQL authentication (Windows authentication is not currently available).

The SQL Server database should not be on the same system as the Uptime Infrastructure Monitor Monitoring Station.

Step 2 - Configuring Uptime Infrastructure Monitor

After you have created the SQL Server database and set up the Uptime Infrastructure Monitor database, you need to configure Uptime Infrastructure Monitor to access the new database. This involves editing these two configuration files (replace <uptime_dir> with the directory of your Uptime Infrastructure Monitor install):

```
<uptime_dir>\uptime.conf
<uptime_dir>\controller\resources\uptime_controller.conf
```

Starting with the uptime.conf file, add a # character at the beginning of each of the following lines to comment them out:

```
dbDriver=com.mysql.jdbc.Driver
dbType=mysql
dbHostname=localhost
dbPort=3308
dbName=uptime
dbUsername=uptime
dbPassword=uptime
```

Just below the lines listed above, there is a second group of lines in the uptime.conf file that define how Uptime Infrastructure Monitor will connect to a SQL Server database (see below). Remove the # character from the beginning of these lines and update the dbHost, dbPort, dbName, dbUsername and dbPassword to match the settings of your SQL Server database and the user that you created in Step 1.

```
#dbDriver=net.sourceforge.jtds.jdbc.Driver
#dbType=mssql
#dbHostname=10.1.1.124
#dbPort=1433
#dbName=uptime
#dbUsername=uptime
#dbPassword=password
```

Next you will need to modify the up.time Controller service configuration file in the same manner. This service provides Uptime Infrastructure Monitor with API services and is an ingress point for most of the Gadgets in the Dashboards. Update the <uptime_dir>\controller\resources\uptime_controller.conf file just as you did the uptime.conf file, commenting out the MySQL configuration lines, un-commenting the MS SQL related lines and editing them with the appropriate configuration and credential info.

Step 3 - Resetting the up.time DataStore

Reset the up.time DataStore settings to create the tables and default values required by Uptime Infrastructure Monitor on your SQL Server database. Make sure the "up.time Data Collector" and "up.time Controller" services are stopped before run the following command on your Uptime Infrastructure Monitor Monitoring Station:

This command will reset any existing Uptime Infrastructure Monitor-specific database settings. Before running this command, ensure that the settings in the uptime.conf file are correct.

```
<uptime_dir>\resetdb really
```

Step 4 - Restarting Uptime Infrastructure Monitor

After the resetdb utility has finished updating your database settings, start the "up.time Data Collector" and "up.time Controller" services. When Uptime Infrastructure Monitor starts, it will be running from your SQL Server database with an empty Uptime Infrastructure Monitor installation.

Step 5 - Disabling the up.time DataStore service

Now that Uptime Infrastructure Monitor is successfully started and running from your MS SQL database, we no longer need to have the local MySQL datastore, which means that we can disable the unneeded service to free up some resources on the monitoring station itself.
This is done via the Window Services tool typically found in the 'Administrator Tools' section of the Start Menu or from within Computer/Server Management. You'll first want to locate and stop the 'up.time Data Store' service. Once the service is stopped, you'll need to right click on the service name, and select 'Properties' from the context menu. On the general tab of the window that opens, you should see a drop down menu for 'Startup type' that currently shows 'Automatic'. You'll want to change this option to 'Disabled' , and then click OK to save your changes.