GUIDE

Choosing a SQL Server monitoring tool

redgate

Contents

Introduction

The increasing size of SQL Server databases, along with the growing complexity of SQL Server estates, has led more organizations to consider a third-party monitoring tool.

If you're using custom scripts, you are most likely well aware of how time consuming monitoring your SQL Server estate can be. For a growing environment, it can take several hours each day. In addition, while custom scripts can provide basic information (e.g. SQL Server wait stats and memory utilization), these are often not sufficient to proactively identify trends and issues before problems arise for you and your users.

The ability to see trends in data collected by a third-party monitoring tool can also warn you about I/O, memory, or disk space issues, long before they become a crisis, in addition to more acute issues and abnormal resource patterns, such as sustained CPU spikes or failed backups.

Furthermore, investing in proactive monitoring is easily justified given the rapidly increasing costs associated with downtime and data breaches. Organizations can no longer afford to miss signs of unusual activity that could potentially impact performance or signal unauthorized access.

Having said that, selecting the best monitoring tool for your unique needs and budget can be an enormous challenge. In order to make it a bit easier, Redgate has put together this brief guide to choosing a SQL Server monitoring tool. We have captured criteria from subject matter experts both inside and external to Redgate.

As Redgate is a leading provider of SQL Server tools and solutions, we certainly hope that you'll choose to work with us. However, our ultimate goal is to help you define your needs, prioritize the selection criteria based on those needs, and determine the best options within your budget.

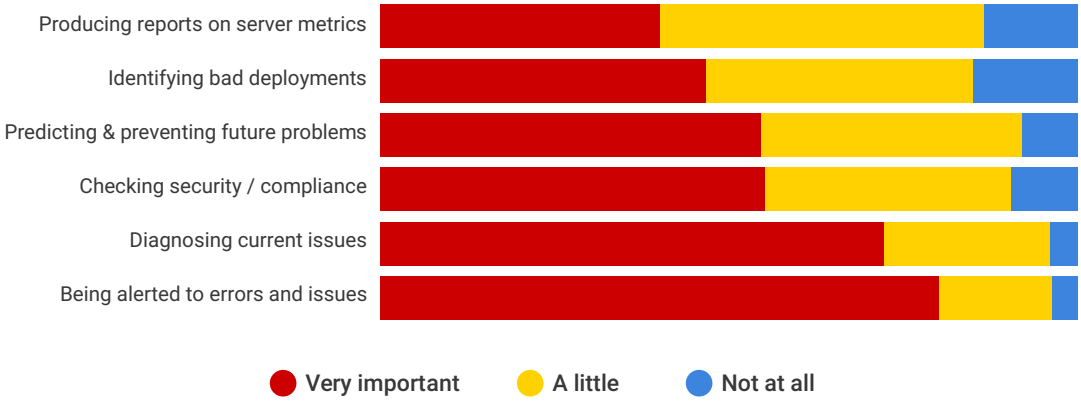
Defining the evaluation criteria

With larger, more complex environments and an accelerating pace of change, further complicated by database sprawl, understanding your environment is more difficult than ever.

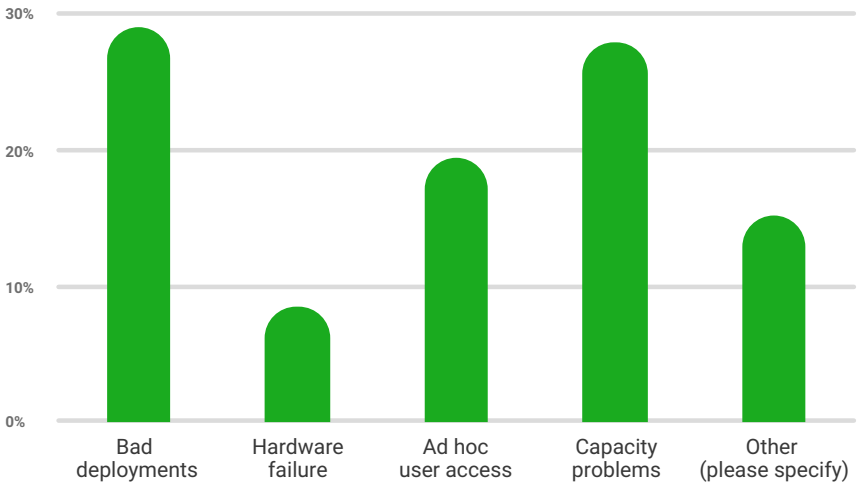
To develop criteria for evaluating a monitoring solution, it is important to understand your needs and requirements with regard to the most frequently occurring issues and those with the highest potential impact on your business.

The charts on this page represent responses to these questions from 626 technology professionals who work in organizations that use SQL Server. This recent survey was conducted by Redgate and 1105 Media.

How important are each of these aspects of SQL Server monitoring to your business?



Which of these causes issues the most frequently?



The most frequently cited causes of problems – running out of capacity, bad deployments, and ad hoc user access – suggest that respondents too often find themselves fire-fighting, rather than proactively monitoring and managing database activity.

This is further supported by the importance ratings of SQL Server tasks. Any third-party monitoring tool should provide alerting around each of these areas, as well as long-term tracking of trends in areas such as database size growth – providing ample warning before problems become critical.

Compatibility and scalability

The first thing to look for when choosing a new database tool is ensuring that it will be compatible with your current and future environment. This means choosing a tool that can support the number of servers that you currently use and plan to use in the future. If you have plans to move to the cloud, evaluate the tool's ability to handle hybrid SQL environments such as Azure DB, Managed Instances, and Cloud hosted IaaS.

Many tools are capable of providing reports and real-time monitoring of hardware usage and performance across a large number of servers, including disk space, CPU usage, memory, and availability. Finding a tool that can provide all of these services across a large number of servers is key for growing businesses.

Determining the fit

Once you've determined your specific needs and requirements, as well as how your estate may change and grow, you can begin obtaining information on third-party tools and map these to your needs and goals.

Three key questions to ask are:

1. Will you be able to prevent problems before they impact users?
2. Will you spend less time monitoring?
3. Will you have the answers to questions about performance at the drop of a hat?

To help you answer these questions, consider the following:

What is monitored and how are alerts provided?

Ideally, the tool should provide monitoring and alerts for the most important operational and performance issues, and preferably have the ability to send multiple alerts on a threshold/escalation mechanism. Some of the more important alerts include those covering query performance, deadlocks and blocking processes, database and job health, SQL Server issues, database file growth, and machine-level concerns.

The data and alerts provided must be meaningful, reliable, and accurate, so that immediate corrective action is achievable, but also so that your DBAs are not plagued by over-alerting or false positives.

Although preconfigured alerts are important, alerts must also be customizable for issues of particular importance in your environment.

Is it extensible with custom code or events?

Every DBA in every company or organization needs different things from their monitoring tool. Monitoring tools should come with a library of alerts about I/O, disk, and memory issues, as well as wait stats and performance metrics. The more customizable the better, especially with larger estates, to reduce alert "noise" and make the tool even more useful.

Balance – too many alerts versus the important alerts

While a long list of what can be monitored is impressive, overload can actually lead to problems. It's easy to get into an "alert overload", caused by a combination of failsafe alerts and duplicate alerts from different tools. If multiple monitoring tools are needed to cover every required feature, it becomes quite a task to monitor the monitoring tools and ensure that all the mechanisms are working appropriately. This can also lead to duplicate alerts and an overly long list of alerts, making it difficult to prioritize and act on the information.

How are the alerts and dashboards presented and shared?

The better the data is presented and contextualized, the easier it is to identify the issue and how that has affected performance. Does the tool have real-time visibility into problems, at a glance? This is especially important when looking after a large estate.

Alerts should be displayed by common themes and issues, so they are easier to see, prioritize, and act on. Dashboards and visualization should make it easy to diagnose and determine the root cause by correlating metrics and comparing them to their baseline activity.

Does your monitoring tool provide trend and historical data?

Providing ongoing, effective, and informative alerts about issues that come up is only half of the job of a monitoring tool. The other half is ascertaining the trends revealed by those alerts – the issues that occur at certain times of the month, or quarter, or year that may uncover bottlenecks or shortcomings in capacity planning.

With the ability to see and analyze those trends, the monitoring tool provides a window to problems in the future, as well as those that are occurring now. In addition, your monitoring tool should help you understand unusual patterns of behavior.

Performance impact

Put simply, we want a web application that anyone can get to from anywhere. Having a thick client application is always a burden because it requires a remote desktop session to a central server, or an install on the DBA's laptop. These are generally slow to connect and pull down data, especially for a WAN/VPN connection.

Another question to ask of any monitoring tool is whether or not an agent is installed on the SQL Server instance. If so, why, and what is it doing?

Most likely, it is just communicating with the mothership on the central storage server, which houses the monitoring database. However, in some cases, the actions of a monitoring tool can affect the performance of the normal database workload.

Reporting

The monitoring tool should be capable of creating customized reports that can be scheduled, exported, and automatically emailed. Out-of-the-box reporting should combine available performance metrics and analysis with summary information on things like server uptime, the numbers of alerts being raised, or the disks which are filling up fastest.

Scalability

Real time visibility into problems at a glance is a must, especially when looking after a large estate. Distributed or multi-base monitoring is important if your environment is spread across multiple geographies or networks. The data collected should be consolidated into a single UI for monitoring and reporting, without compromising security or bandwidth.

Evaluating the vendor as well as the tool

When evaluating a vendor, you should take into consideration:

- *How is support provided and what is included in the price of the tool?*
- *Do they provide access to subject matter experts?*
- *Do they have a roadmap? What does the innovation and investment in their tools looks like?*
- *Can they demonstrate that they collect and incorporate customer feedback, for example through UserVoice?*
- *Are any customer references and reviews available?*
- *Do they provide evaluations and proofs of concept?*
- *Do they offer integration with Continuous Integration, Continuous Deploymet, and other automation tools?*

Total cost of ownership and return on investment

In order to determine the return on your investment, it is important to first understand the total cost of ownership of any in-house or third-party monitoring tool. We recommend obtaining a multi-year quote (for at least three years) so that you're better able to estimate the annual cost for the tool over a longer time period. Some vendors heavily discount their pricing in the first year only to levy significant price increases in subsequent years.

Redgate has a nifty tool to help you determine the return on your investment, depending on the size of your SQL Server estate and the cost of your DBAs. A good monitoring tool can save your organization several man-hours each day and allow DBAs time to focus on value-add, strategic work, such as migrating to the cloud, optimizing performance, and deploying changes faster by becoming more agile.

Scoring your options

In the end there a lot of factors to consider, generally too many to outline a clear choice. We've found that breaking these factors down and assigning values to them can better allow you to objectively weigh their importance to your company, and so help you pick the most suitable tool. Here's a downloadable example of a table that you can use to score your options in exactly that way.

About Redgate

Redgate is the leading provider of software for professionals working on the Microsoft Data Platform.

Our SQL Toolbelt helps users implement DevOps best practice for their databases, easily monitor database performance, and improve team productivity.

We do all this by investing heavily in making our software intuitive to use even though we're solving complex customer problems – something we call ingeniously simple.

230
Redgaters and counting

18
years old

202,000
customers

100,000
cups of coffee each year

91%
of the Fortune 100 use our tools

4m
website visits each year

139
product release so far this year

103
User Groups sponsored so far this year