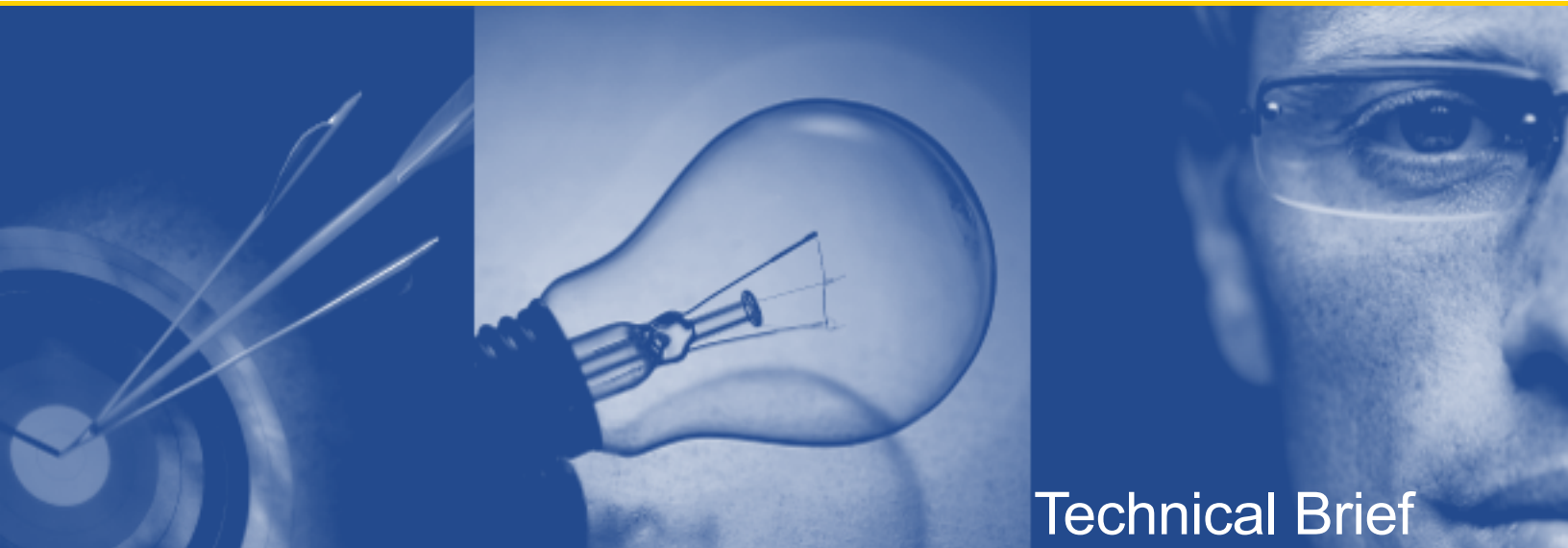


Foglight® Performance Analysis for SQL Server: Value for DBAs and Developers Alike

*Written by
Thomas LaRock*



Technical Brief

**© 2009 Quest Software, Inc.
ALL RIGHTS RESERVED.**

This document contains proprietary information, protected by copyright. No part of this document may be reproduced or transmitted for any purpose other than the reader's personal use without the written permission of Quest Software, Inc.

WARRANTY

The information contained in this document is subject to change without notice. Quest Software makes no warranty of any kind with respect to this information. QUEST SOFTWARE SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTY OF THE MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Quest Software shall not be liable for any direct, indirect, incidental, consequential, or other damage alleged in connection with the furnishing or use of this information.

TRADEMARKS

All trademarks and registered trademarks used in this guide are property of their respective owners.

World Headquarters:
5 Polaris Way
Aliso Viejo, CA 92656
e-mail: info@quest.com

Please refer to our Web site (www.quest.com) for regional and international office information.

Updated—May 6, 2009

CONTENTS

- INTRODUCTION 1**
- CHALLENGES 2**
 - GETTING A CLEAR PICTURE: THE LIMITS OF NATIVE TOOLS 2
 - WRESTLING WITH IN-HOUSE SOLUTIONS 2
 - CHOOSING A THIRD-PARTY SOLUTION 3
- USING FOGLIGHT PERFORMANCE ANALYSIS FOR SQL SERVER 4**
 - GETTING AROUND 4
 - VIEWING OVERALL HEALTH OF YOUR INSTANCE AT A GLANCE 4
 - GETTING EXPERT ADVICE: PERFORMANCE ADVISORIES 5
 - VIEWING REAL-TIME ACTIVITY 6
 - FOCUSING IN 7
- VALUE FOR DEVELOPERS AS WELL AS DBAS 8**
- CONCLUSION 9**
- ABOUT THE AUTHOR 10**
- ABOUT QUEST SOFTWARE, INC. 11**
 - CONTACTING QUEST SOFTWARE..... 11
 - CONTACTING QUEST SUPPORT..... 11

INTRODUCTION

Foglight® Performance Analysis for SQL Server provides workload analysis and performance tuning for the SQL Server database environment. When performance degradation occurs, Performance Analysis helps you find the most efficient methods of resolving the root issues. Moreover, Performance Analysis enables both DBAs and developers to work more efficiently.

This technical brief explains how my company came to adopt Performance Analysis and the benefits we realized: increased DBA and developer efficiency, reduced troubleshooting time, and a 25% drop in development costs.

CHALLENGES

Getting a Clear Picture: The Limits of Native Tools

“What happened last night?” I was often asked this question in my role as a production database administrator (DBA). For example, if a batch run took five minutes too long, people would naturally want to ask me if I knew why, and how I would make certain everything would run smoothly tonight. But my typical response to the question was, “I don’t know; I was asleep.” Still, there remained a simple truth: I should have had an idea of what happened last night, right?

Microsoft SQL Server has a handful of rough tools that can give you some information about past events. But you do not always get the entire picture. Some tools are very good at getting one piece of the puzzle and some tools are good at getting other pieces, so you spend a lot of time trying to put the whole puzzle together. And in the end, your best response to the question of what happened last night is often something like, “Looks like the network was slow—I think.” Not only is this troubleshooting process incredibly inefficient, but it does not inspire confidence in your customers, who look to you for details and facts, not opinions or assumptions.

Wrestling with In-house Solutions

To supplement or even replace native tools, many organizations develop some monitoring tools of their own. However, in-house solutions always seem to be incomplete, and continually working on the tools reduces the time DBAs can spend on their primary functions.

For example, I started building my own set of monitoring tools, and many of them worked fairly well. But they always seemed to be lacking the feature I needed to solve the problem at hand. I would often hear myself explaining to someone that, “Yes, of course we have some monitoring tools in place, but they just were not built to capture ...” and then mention the one or two things that were missing. And there was always something missing. Moreover, performance issues don’t generally recur on demand, so I never knew whether all the time I spent enhancing my tools would actually enable me to capture the details I needed when the issue surfaced again.

Because I seemed to always be modifying the tools, I found that I was spending more of my time as an application developer and less time as a database administrator, so I wasn’t as available to help people with design questions or to performance-tune their queries. There are only so many hours in the day, right? But when your customer wants—nay, *demand*s—an answer to a performance issue, you feel you need to work on that particular issue, and you miss the bigger picture, even though as a DBA, you should always see the big picture first.

Choosing a Third-Party Solution

When my organization reached a breaking point, the decision was made to bring in a third-party tool to help us to answer that simple question, “What happened last night?” We had a handful of requirements in mind. Specifically, the tools needed to:

- Be non-intrusive
- Capture all activity, at all times of the day
- Provide real-time and historical analysis
- Determine baseline metrics based upon individual server activity
- Alert or notify us for important issues pertaining to the server baselines
- Provide some reporting capability

With those requirements in mind, we started evaluating some products. After a few months of searching and testing, we chose Quest Software’s Foglight Performance Analysis, which satisfied all of our requirements. The ability to slice and dice both real-time and historical data was a wonderful benefit for our team, but more important was the significant time it saved us when we needed to troubleshoot issues, both past and present.

USING FOGLIGHT PERFORMANCE ANALYSIS FOR SQL SERVER

Getting Around

When you connect to your instance with Performance Analysis, you are greeted with the home page by default. At the top of this page you will see five tabs, which enable you to quickly navigate through the tool. For example, to examine real-time data, click the Real-Time tab; for historical information, click the History tab.



Figure 1. Tabbed interface enables easy navigation

Viewing Overall Health of Your Instance at a Glance

The home page displays the overall health of your instance including workload, performance and system utilization:

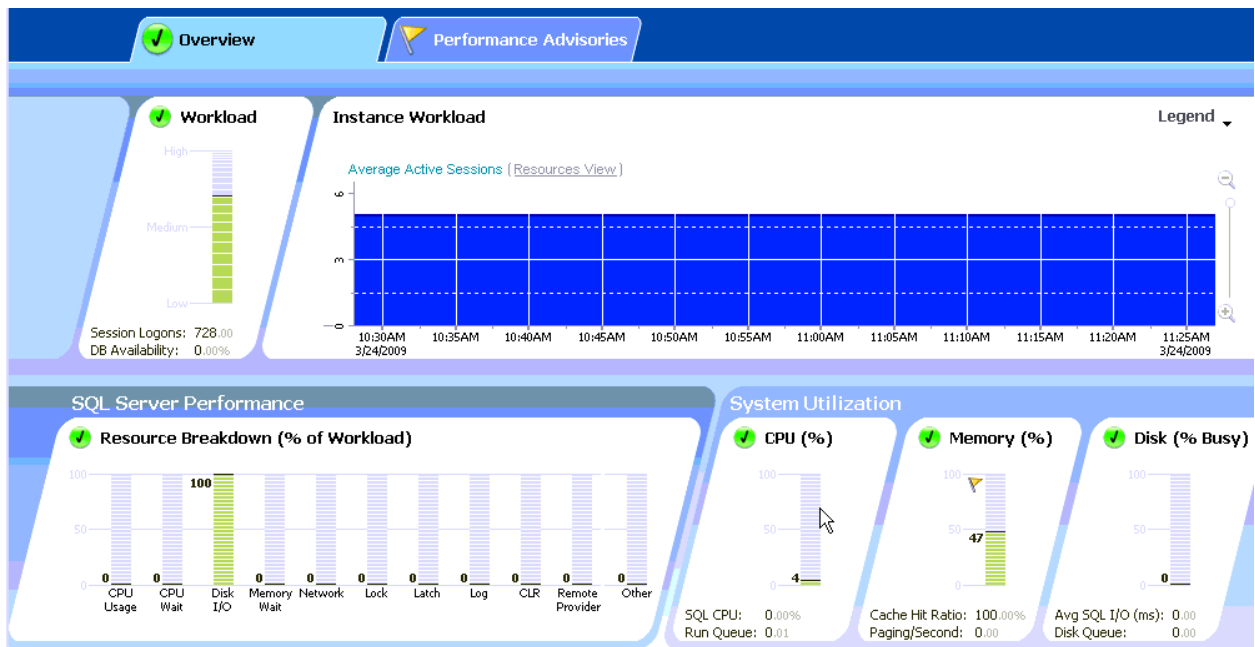


Figure 2. Instance health can be viewed at a glance.

Getting Expert Advice: Performance Advisories

Performance Analysis also offers Performance Advisories that include not only a detailed report about what the advisory is for, but also a detailed action plan for resolving the issue at hand:

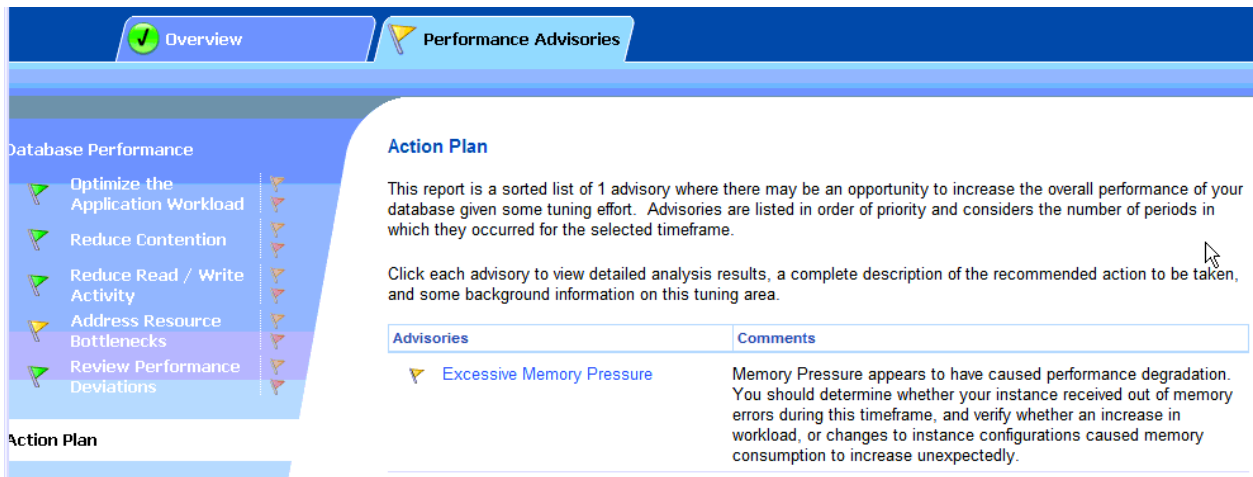


Figure 3. Step-by-step action plans help you resolve performance issues.

Viewing Real-Time Activity

After examining the overview of the instance and reviewing the advisories, you can click the Real-Time tab to see the activity hitting the instance:



Figure 4. The graphic representation of real-time performance is easy to understand.

Focusing In

Performance Analysis allows you to slice and dice the information in several ways. First, you can drill down into a particular resource, such as CPU or I/O, or you can drill into a particular session. Alternatively, you can change the time range displayed (the last hour is displayed by default). For example, you could narrow the focus to just the past fifteen minutes.

To go back further in time, simply click on the History tab for additional time range options. In addition, the menu on the left enables you to quickly filter on a particular session, SQL statement, users, or databases, as shown below:

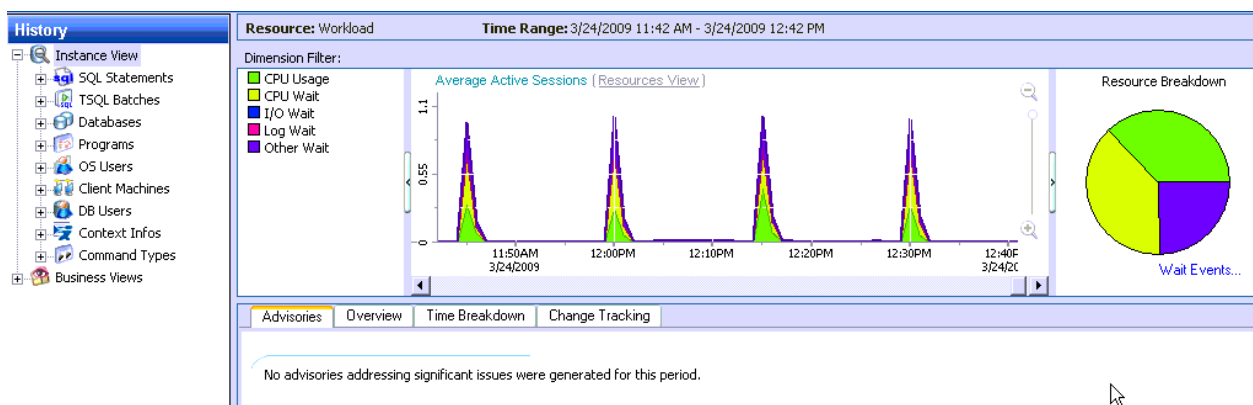


Figure 5. Use historical filters to view performance over any time range.

VALUE FOR DEVELOPERS AS WELL AS DBAS

The DBAs at my organization continued to play with Performance Analysis, slicing and dicing our way through a wonderful amount of detailed information. Then it happened: everyone in the room had the same thought at the exact same time. We looked at each other with amazement and everyone tried to get the words out first: “Wouldn’t it be great if our developers had this tool so they could see the effect their code has on the box as they build their apps?”

For years, our developers had struggled with tuning their code. With native tools, they often focused on the wrong things, such as the queries with the longest durations, instead of the actual performance of the application. They were often unable to really view application activity, saying things like, “We don’t know what the code is doing.” And they were dependent on the DBAs for access to the tools and for troubleshooting; for example, we’d often hear, “The server is slow; please fix.”

Performance Analysis solves all these issues. It enables our developers to see all the activity on a server in ways that the native SQL tools never could. They can slice and dice the data in real time, including drilling down into the performance bottlenecks at will. Instead of focusing just on the queries with the longest duration, they can now focus on any inefficient queries, regardless of duration. We estimated Performance Analysis saved roughly 25 percent of development time, and expected that figure to increase as the developers became more familiar with the tool.

Our developers were no longer guessing as to where to focus their efforts; Performance Analysis allowed them to work efficiently and quickly. With the addition of the integrated SQL optimization component, they were able to get recommendations on how to write better code, learning along the way so that in the future they would be able to write better T-SQL right from the start. Moreover, they no longer needed to call a DBA every time the smallest thing interrupted their work; they could look for themselves, improving both their efficiency and ours, since we now spent less time in a support role.

Another benefit to this efficiency was that we no longer needed to call for “all hands on deck” to resolve issues; they could now be resolved quickly by just a few people. For example, instead of needing to gather four people together for an hour to track down an issue, two people could resolve it together in 15 minutes—a huge cost savings. After about six months we really started to see the gains in productivity, without any increase in headcount. Since we were spend less time resolving issues, we had more time to spend administering the environment, which in turn reduced the number of issues that came up. No doubt about it: this was a win-win situation for everyone.

CONCLUSION

Without question, the number one tool in our shop is Foglight Performance Analysis for SQL Server. It is more important to us than any other product, except of course Microsoft SQL Server itself.

ABOUT THE AUTHOR

Thomas LaRock is a seasoned IT professional with more than a decade of technical and management experience. Currently serving as a DBA for an international investment company, Thomas has progressed through several roles at the company, including programmer, analyst, and DBA. Prior to his current position, he worked at several software and consulting companies, spending a lot of time at customer sites in the United States and abroad.

Thomas holds a master's degree in mathematics from Washington State University. He is a member of the Usability Professional's Association and Quest's Association of SQL Server Experts. He also currently serves on the board of directors for the Professional Association for SQL Server (PASS) and is a SQL Server MVP.

ABOUT QUEST SOFTWARE, INC.

Quest Software, Inc., a leading enterprise systems management vendor, delivers innovative products that help organizations get more performance and productivity from their applications, databases, Windows infrastructure and virtual environments. Quest also provides customers with client management through its ScriptLogic subsidiary and server virtualization management through its Vizioncore subsidiary. Through a deep expertise in IT operations and a continued focus on what works best, Quest helps more than 100,000 customers worldwide meet higher expectations for enterprise IT.

Quest's System Center solutions enable the Microsoft System Center platform to serve as the comprehensive systems management platform for organizations managing heterogeneous environments. Together, an integrated platform improves IT productivity, increases return on IT investments, and ensures compliance and service levels to help drive business profitability. Visit Quest's System Center solutions at www.quest.com/system-center. Specifically for QMX for Configuration Manager, visit: <http://www.quest.com/quest-management-xtensions-configuration-manager/>.

Quest Software can be found in offices around the globe and at www.quest.com.

Contacting Quest Software

Phone:	949.754.8000 (United States and Canada)
Email:	info@quest.com
Mail:	Quest Software, Inc. World Headquarters 5 Polaris Way Aliso Viejo, CA 92656 USA
Web site:	www.quest.com

Please refer to our Web site for regional and international office information.

Contacting Quest Support

Quest Support is available to customers who have a trial version of a Quest product or who have purchased a commercial version and have a valid maintenance contract. Quest Support provides around the clock coverage with SupportLink, our web self-service. Visit SupportLink at <http://support.quest.com>

From SupportLink, you can do the following:

- Quickly find thousands of solutions (Knowledgebase articles/documents).
- Download patches and upgrades.
- Seek help from a Support engineer.
- Log and update your case, and check its status.

View the **Global Support Guide** for a detailed explanation of support programs, online services, contact information, and policy and procedures. The guide is available at: http://support.quest.com/pdfs/Global_Support_Guide.pdf