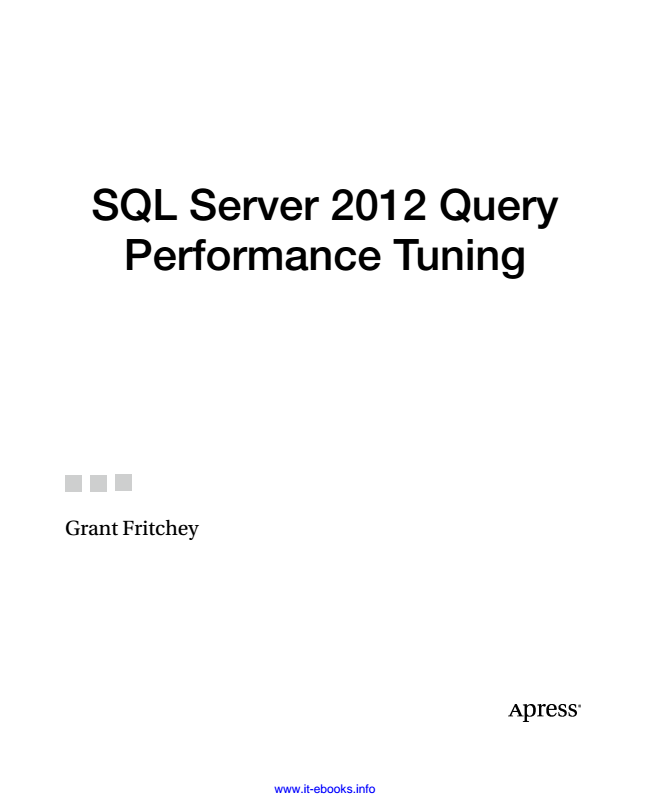


www.it-ebooks.info



*For your convenience Apress has placed some of the front matter material after the index. Please use the Bookmarks and Contents at a Glance links to access them.*

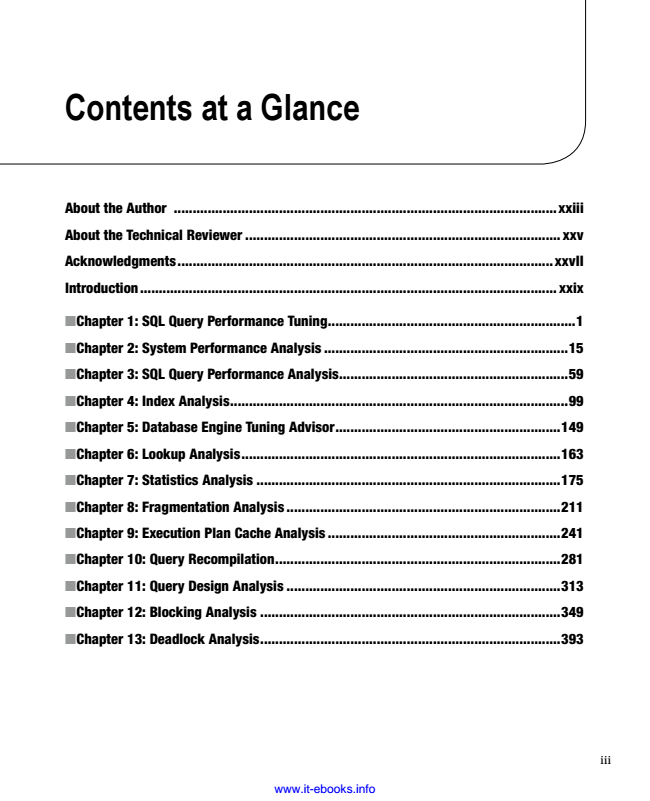
www.it-ebooks.info



SQL Server 2012 Query Performance Tuning

Grant Fritchey

www.it-ebooks.info



**Contents at a Glance**

About the Author ......................................................................................................xxiii

About the Technical Reviewer .................................................................................... xxv

Acknowledgments ....................................................................................................xxvII

Introduction ............................................................................................................... xxix

Chapter 1: SQL Query Performance Tuning ■ ..................................................................1

Chapter 2: System Performance Analysis ■ .................................................................15

Chapter 3: SQL Query Performance Analysis ■ .............................................................59

Chapter 4: Index Analysis ■ ..........................................................................................99

Chapter 5: Database Engine Tuning Advisor ■ ............................................................149

Chapter 6: Lookup Analysis ■ .....................................................................................163

Chapter 7: Statistics Analysis ■ .................................................................................175

Chapter 8: Fragmentation Analysis ■ .........................................................................211

Chapter 9: Execution Plan Cache Analysis ■ ..............................................................241

Chapter 10: Query Recompilation ■ ............................................................................281

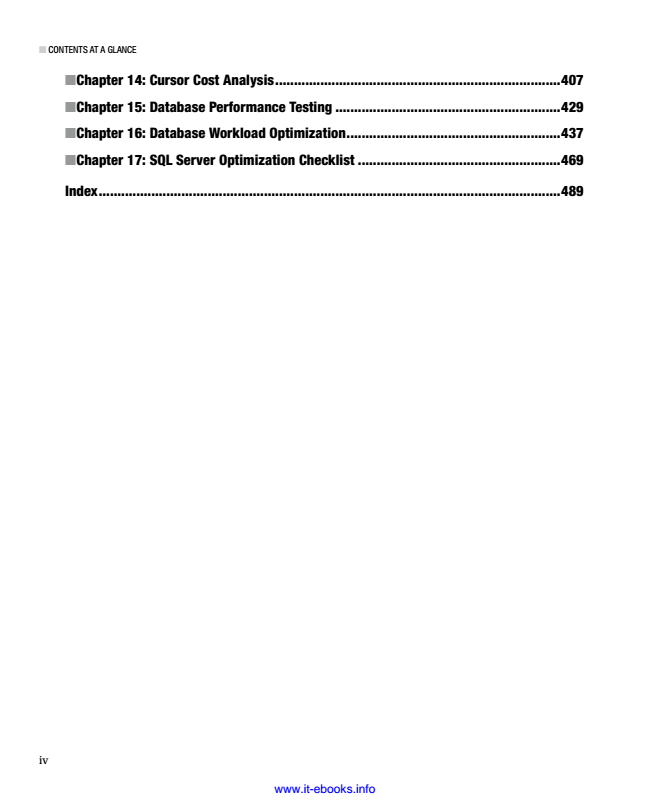
Chapter 11: Query Design Analysis ■ .........................................................................313

Chapter 12: Blocking Analysis ■ ................................................................................349

Chapter 13: Deadlock Analysis ■ ................................................................................393

www.it-ebooks.info

iii



■ CONTENTS AT A GLANCE

iv

■ Chapter 14: Cursor Cost Analysis ............................................................................407

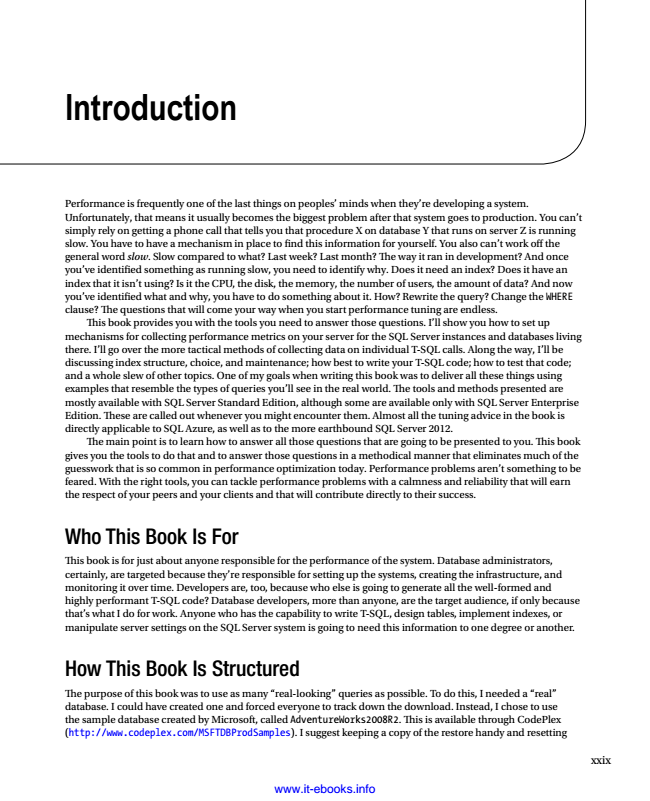
■ Chapter 15: Database Performance Testing ............................................................429

■ Chapter 16: Database Workload Optimization .........................................................437

■ Chapter 17: SQL Server Optimization Checklist ......................................................469

Index ...........................................................................................................................489

www.it-ebooks.info



**Introduction**

Performance is frequently one of the last things on peoples’ minds when they’re developing a system. Unfortunately, that means it usually becomes the biggest problem after that system goes to production. You can’t simply rely on getting a phone call that tells you that procedure X on database Y that runs on server Z is running slow. You have to have a mechanism in place to find this information for yourself. You also can’t work off the general word slow. Slow compared to what? Last week? Last month? The way it ran in development? And once you’ve identified something as running slow, you need to identify why. Does it need an index? Does it have an index that it isn’t using? Is it the CPU, the disk, the memory, the number of users, the amount of data? And now you’ve identified what and why, you have to do something about it. How? Rewrite the query? Change the WHERE clause? The questions that will come your way when you start performance tuning are endless.

This book provides you with the tools you need to answer those questions. I’ll show you how to set up mechanisms for collecting performance metrics on your server for the SQL Server instances and databases living there. I’ll go over the more tactical methods of collecting data on individual T-SQL calls. Along the way, I’ll be discussing index structure, choice, and maintenance; how best to write your T-SQL code; how to test that code; and a whole slew of other topics. One of my goals when writing this book was to deliver all these things using examples that resemble the types of queries you’ll see in the real world. The tools and methods presented are mostly available with SQL Server Standard Edition, although some are available only with SQL Server Enterprise Edition. These are called out whenever you might encounter them. Almost all the tuning advice in the book is directly applicable to SQL Azure, as well as to the more earthbound SQL Server 2012.

The main point is to learn how to answer all those questions that are going to be presented to you. This book gives you the tools to do that and to answer those questions in a methodical manner that eliminates much of the guesswork that is so common in performance optimization today. Performance problems aren’t something to be feared. With the right tools, you can tackle performance problems with a calmness and reliability that will earn the respect of your peers and your clients and that will contribute directly to their success.

Who This Book Is For

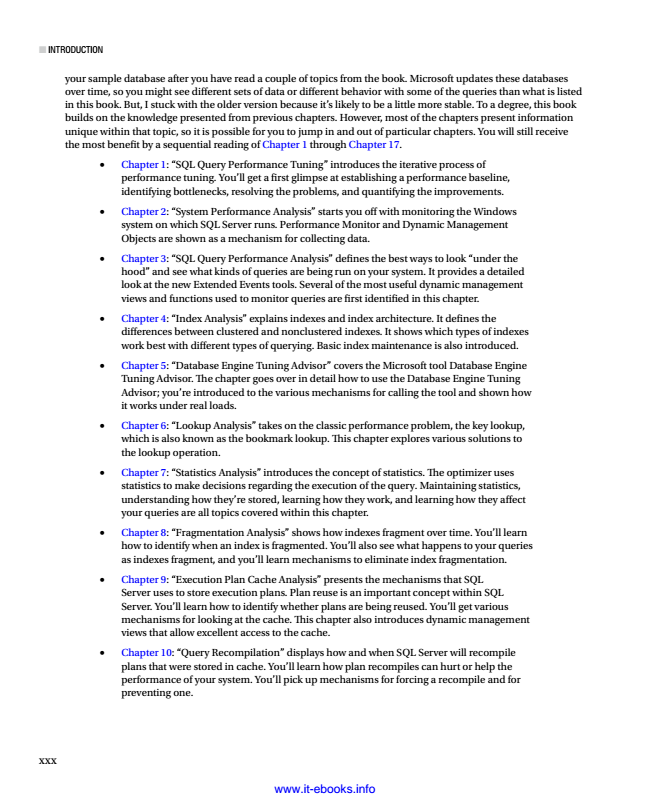
This book is for just about anyone responsible for the performance of the system. Database administrators, certainly, are targeted because they’re responsible for setting up the systems, creating the infrastructure, and monitoring it over time. Developers are, too, because who else is going to generate all the well-formed and highly performant T-SQL code? Database developers, more than anyone, are the target audience, if only because that’s what I do for work. Anyone who has the capability to write T-SQL, design tables, implement indexes, or manipulate server settings on the SQL Server system is going to need this information to one degree or another.

How This Book Is Structured

The purpose of this book was to use as many “real-looking” queries as possible. To do this, I needed a “real” database. I could have created one and forced everyone to track down the download. Instead, I chose to use the sample database created by Microsoft, called AdventureWorks2008R2. This is available through CodePlex (http://www.codeplex.com/MSFTDBProdSamples). I suggest keeping a copy of the restore handy and resetting

www.it-ebooks.info

xxix



■ INTRODUCTION

xxx

your sample database after you have read a couple of topics from the book. Microsoft updates these databases over time, so you might see different sets of data or different behavior with some of the queries than what is listed in this book. But, I stuck with the older version because it’s likely to be a little more stable. To a degree, this book builds on the knowledge presented from previous chapters. However, most of the chapters present information unique within that topic, so it is possible for you to jump in and out of particular chapters. You will still receive the most benefit by a sequential reading of Chapter 1 through Chapter 17.

• Chapter 1: “SQL Query Performance Tuning” introduces the iterative process of performance tuning. You’ll get a first glimpse at establishing a performance baseline, identifying bottlenecks, resolving the problems, and quantifying the improvements.

• Chapter 2: “System Performance Analysis” starts you off with monitoring the Windows system on which SQL Server runs. Performance Monitor and Dynamic Management Objects are shown as a mechanism for collecting data.

• Chapter 3: “SQL Query Performance Analysis” defines the best ways to look “under the hood” and see what kinds of queries are being run on your system. It provides a detailed look at the new Extended Events tools. Several of the most useful dynamic management views and functions used to monitor queries are first identified in this chapter.

• Chapter 4: “Index Analysis” explains indexes and index architecture. It defines the differences between clustered and nonclustered indexes. It shows which types of indexes work best with different types of querying. Basic index maintenance is also introduced.

• Chapter 5: “Database Engine Tuning Advisor” covers the Microsoft tool Database Engine Tuning Advisor. The chapter goes over in detail how to use the Database Engine Tuning Advisor; you’re introduced to the various mechanisms for calling the tool and shown how it works under real loads.

• Chapter 6: “Lookup Analysis” takes on the classic performance problem, the key lookup, which is also known as the bookmark lookup. This chapter explores various solutions to the lookup operation.

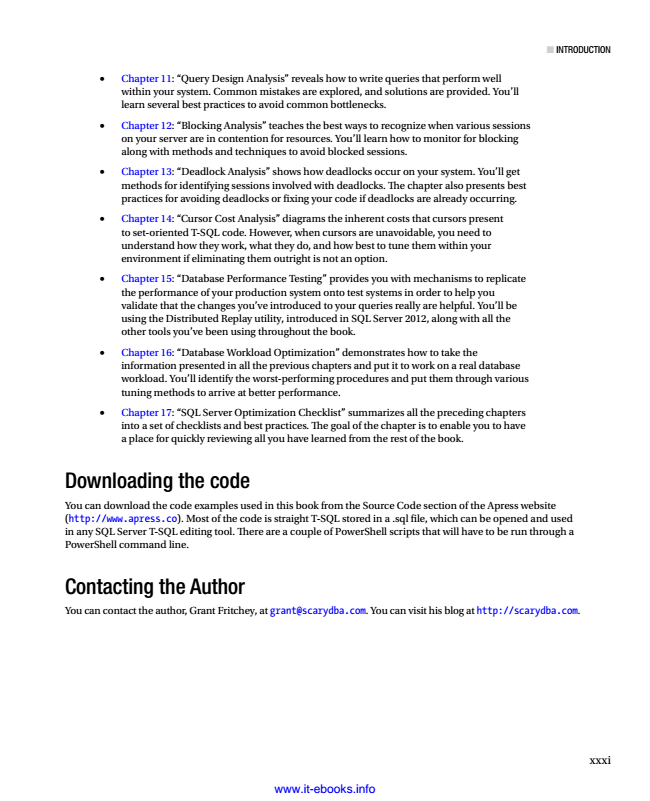
• Chapter 7: “Statistics Analysis” introduces the concept of statistics. The optimizer uses statistics to make decisions regarding the execution of the query. Maintaining statistics, understanding how they’re stored, learning how they work, and learning how they affect your queries are all topics covered within this chapter.

• Chapter 8: “Fragmentation Analysis” shows how indexes fragment over time. You’ll learn how to identify when an index is fragmented. You’ll also see what happens to your queries as indexes fragment, and you’ll learn mechanisms to eliminate index fragmentation.

• Chapter 9: “Execution Plan Cache Analysis” presents the mechanisms that SQL Server uses to store execution plans. Plan reuse is an important concept within SQL Server. You’ll learn how to identify whether plans are being reused. You’ll get various mechanisms for looking at the cache. This chapter also introduces dynamic management views that allow excellent access to the cache.

• Chapter 10: “Query Recompilation” displays how and when SQL Server will recompile plans that were stored in cache. You’ll learn how plan recompiles can hurt or help the performance of your system. You’ll pick up mechanisms for forcing a recompile and for preventing one.

www.it-ebooks.info



■ INTRODUCTION

• Chapter 11: “Query Design Analysis” reveals how to write queries that perform well within your system. Common mistakes are explored, and solutions are provided. You’ll learn several best practices to avoid common bottlenecks.

• Chapter 12: “Blocking Analysis” teaches the best ways to recognize when various sessions on your server are in contention for resources. You’ll learn how to monitor for blocking along with methods and techniques to avoid blocked sessions.

• Chapter 13: “Deadlock Analysis” shows how deadlocks occur on your system. You’ll get methods for identifying sessions involved with deadlocks. The chapter also presents best practices for avoiding deadlocks or fixing your code if deadlocks are already occurring.

• Chapter 14: “Cursor Cost Analysis” diagrams the inherent costs that cursors present to set-oriented T-SQL code. However, when cursors are unavoidable, you need to understand how they work, what they do, and how best to tune them within your environment if eliminating them outright is not an option.

• Chapter 15: “Database Performance Testing” provides you with mechanisms to replicate the performance of your production system onto test systems in order to help you validate that the changes you’ve introduced to your queries really are helpful. You’ll be using the Distributed Replay utility, introduced in SQL Server 2012, along with all the other tools you’ve been using throughout the book.

• Chapter 16: “Database Workload Optimization” demonstrates how to take the information presented in all the previous chapters and put it to work on a real database workload. You’ll identify the worst-performing procedures and put them through various tuning methods to arrive at better performance.

• Chapter 17: “SQL Server Optimization Checklist” summarizes all the preceding chapters into a set of checklists and best practices. The goal of the chapter is to enable you to have a place for quickly reviewing all you have learned from the rest of the book.

Downloading the code

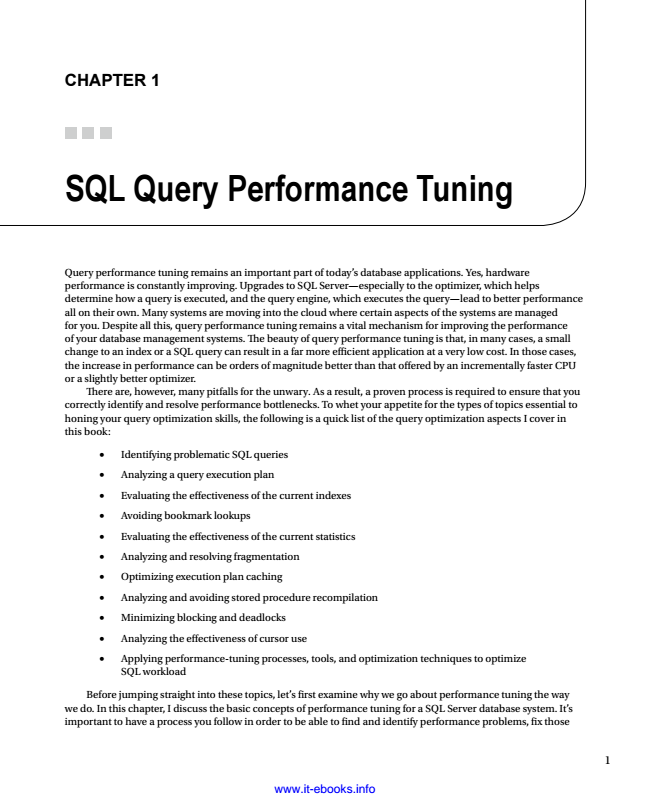
You can download the code examples used in this book from the Source Code section of the Apress website (http://www.apress.co). Most of the code is straight T-SQL stored in a .sql file, which can be opened and used in any SQL Server T-SQL editing tool. There are a couple of PowerShell scripts that will have to be run through a PowerShell command line.

Contacting the Author

You can contact the author, Grant Fritchey, at grant@scarydba.com. You can visit his blog at http://scarydba.com.

xxxi

www.it-ebooks.info



**CHAPTER 1**

**SQL Query Performance Tuning**

Query performance tuning remains an important part of today’s database applications. Yes, hardware performance is constantly improving. Upgrades to SQL Server—especially to the optimizer, which helps determine how a query is executed, and the query engine, which executes the query—lead to better performance all on their own. Many systems are moving into the cloud where certain aspects of the systems are managed for you. Despite all this, query performance tuning remains a vital mechanism for improving the performance of your database management systems. The beauty of query performance tuning is that, in many cases, a small change to an index or a SQL query can result in a far more efficient application at a very low cost. In those cases, the increase in performance can be orders of magnitude better than that offered by an incrementally faster CPU or a slightly better optimizer.

There are, however, many pitfalls for the unwary. As a result, a proven process is required to ensure that you correctly identify and resolve performance bottlenecks. To whet your appetite for the types of topics essential to honing your query optimization skills, the following is a quick list of the query optimization aspects I cover in this book:

Identifying problematic SQL queries •

Analyzing a query execution plan •

Evaluating the effectiveness of the current indexes •

Avoiding bookmark lookups •

Evaluating the effectiveness of the current statistics •

Analyzing and resolving fragmentation •

Optimizing execution plan caching •

Analyzing and avoiding stored procedure recompilation •

Minimizing blocking and deadlocks •

Analyzing the effectiveness of cursor use •

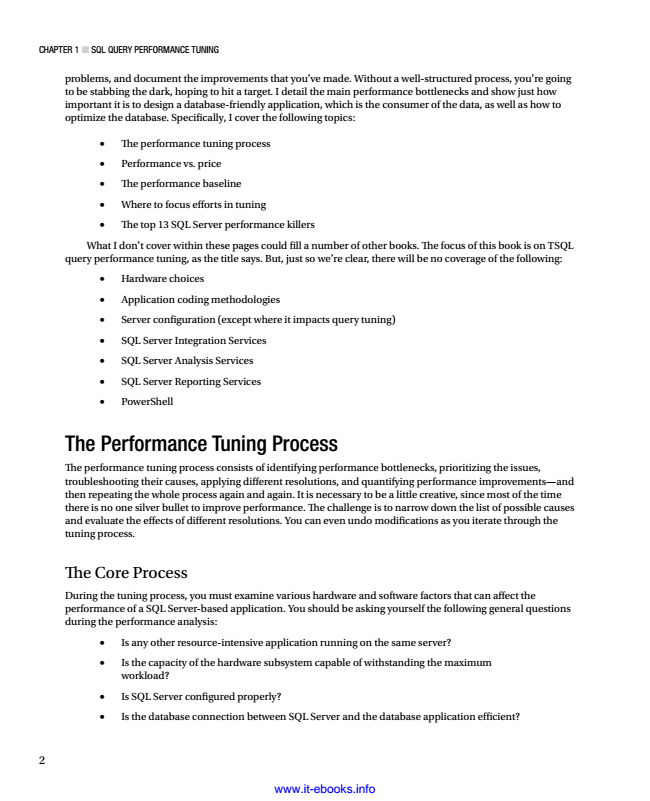
Applying performance-tuning processes, tools, and optimization techniques to optimize •

SQL workload

Before jumping straight into these topics, let’s first examine why we go about performance tuning the way we do. In this chapter, I discuss the basic concepts of performance tuning for a SQL Server database system. It’s important to have a process you follow in order to be able to find and identify performance problems, fix those

www.it-ebooks.info

1



CHAPTER 1 ■ SQL QuERy PERfoRmAnCE Tuning

2

problems, and document the improvements that you’ve made. Without a well-structured process, you’re going to be stabbing the dark, hoping to hit a target. I detail the main performance bottlenecks and show just how important it is to design a database-friendly application, which is the consumer of the data, as well as how to optimize the database. Specifically, I cover the following topics:

• The performance tuning process • Performance vs. price • The performance baseline • Where to focus efforts in tuning • The top 13 SQL Server performance killers What I don’t cover within these pages could fill a number of other books. The focus of this book is on TSQL query performance tuning, as the title says. But, just so we’re clear, there will be no coverage of the following:

• Hardware choices • Application coding methodologies • Server configuration (except where it impacts query tuning) • SQL Server Integration Services • SQL Server Analysis Services • SQL Server Reporting Services • PowerShell The Performance Tuning Process

The performance tuning process consists of identifying performance bottlenecks, prioritizing the issues, troubleshooting their causes, applying different resolutions, and quantifying performance improvements—and then repeating the whole process again and again. It is necessary to be a little creative, since most of the time there is no one silver bullet to improve performance. The challenge is to narrow down the list of possible causes and evaluate the effects of different resolutions. You can even undo modifications as you iterate through the tuning process.

The Core Process

During the tuning process, you must examine various hardware and software factors that can affect the performance of a SQL Server-based application. You should be asking yourself the following general questions during the performance analysis:

• Is any other resource-intensive application running on the same server? •

Is the capacity of the hardware subsystem capable of withstanding the maximum workload?

• Is SQL Server configured properly? •

Is the database connection between SQL Server and the database application efficient? www.it-ebooks.info