The Oracle Magician

Summer, 2006 OracleMagician.com Volume V, number 2

From the Editor

Welcome to the new issue of our online magazine, The *Oracle Magician*! This quarterly newsletter focuses on various "tricks of the trade" in the Oracle world--from DBAs, architects, developers, designers, and report writers.

Thank you to the many notes and comments from readers of my book, *The Art & Science of Oracle Performance Tuning*. Your notes are most appreciated!

In this issue we discuss the notion of a single process or tool that can solve all performance problems. Is there any such method?

Also in this issue we also discuss the interesting phenomenon of "implicit commit."

As always, we accept ideas or articles from reads that have interesting performance ideas.

Please send all ideas to Editor@OracleMagician.com

Chris Lawson

Editor

Is there a Flow Chart to Success?

By Chris Lawson

The Job of the Performance Analyst

It seems to me there really isn't just one approach or system that conclusively addresses all performance problems. In practice, the performance tuning business is too unpredictable—our job simply doesn't fit into a nice tidy box.

Oftentimes, the performance analyst is dropped into a messy mixture of people, technology, and business needs. Adding to the complexity of performance tuning, software engineers design programs in startling different ways. Sometimes, designers build amazingly clever code, with equally amazing results. More often, however, we encounter amazing designs with amazingly horrid performance.

A Flow Chart to Success?

Years ago, a large corporation

Continued on page 2

Table of Contents

Continued from page 1

issued a notebook to all the engineers. The purpose of the notebook was to illustrate sound approaches to problem solving. The first diagram was a flow chart illustrating the steps that an engineer should follow to solve problems. The first box said "TEST SYSTEM." The second box said "DOES THE SYSTEM MEET REQUIREMENTS?" A "Yes" response led to the box labeled "DONE." A "No" response led to the third box, labeled "REDESIGN SYSTEM."

As you might expect, we laughed at the absurdity of using such a flowchart.

This example is absolutely true. I mention it because it illustrates the absurdity of trying to reduce complex processes to a single "flow chart of success."

Performance Analysis

Making Sql run faster is indeed a big part of performance analysis, but it's sometimes (mis)represented as the *only* part. This notion is wrong because the process of tuning Sql *presupposes the form of the solution*. In other words, the act of tuning Sql is based on the idea that slow Sql is the *root cause* of the problem.

Of course, slow running Sql is not always the root cause of performance

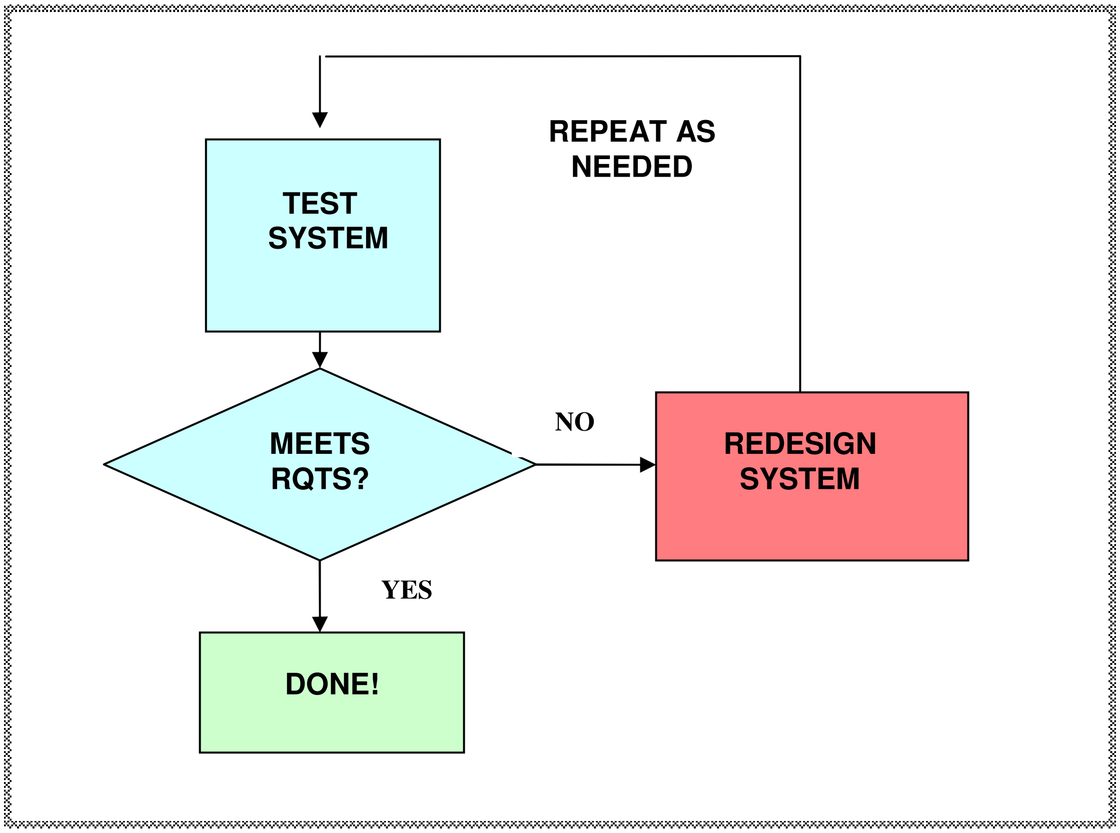


FIGURE 1. A GUIDE TO SOLVE ALL PROBLEMS

Continued from page 2

problems. For instance, we might need to ask *why* an application is running certain Sql. Did the designer make a poor design decision that requires the program to run millions of unnecessary transactions?

A Silliness Detector?

Several years ago, I assisted a group of report designers with a poorly running report. I noticed that the exact same long-running query was run multiple times. I pointed this out to the designer, suggesting that he remove the redundant queries.

Ignoring my advice, the designer instead installed a performance tool to tune the Sql. He slightly improved the efficiency of each redundant execution, but missed the whole point.

In the above example, the report designer did exactly what the tool told him, instead of stepping back and looking at the big picture. He presupposed the form of the solution, which led him to do something silly.

Here's another example--one that's much more subtle: The DBA team at a large insurance company isolated a problem query, which checked the last 12 months of payments. The key to resolving this issue was not how to make the query run

Continued on page 4

Implicit Commit

Did you know that Oracle will issue an *implicit commit* whenever you run a DDL command? In other words, every DDL that you run is really assigned a separate transaction.

This is not normally an issue—unless you expect to perform a rollback, or control the commits for some business reason. Note that when you execute the DDL transaction, issuing a subsequent *Rollback* or *Commit* command will do nothing.

For example, suppose you perform the following update:

Update Chris_Test
set object_name = 'TESTER';

Now, if you execute the following DDL command (on a completely unrelated object), your changes to Chris_Test will be implicitly committed:

Truncate Table Test-Table;

Here's another example of DDL that will cause your changes to commit:

Create Table Test_Table2 As
Select * From User_Objects;

Once again, note that the table affected by the DDL command can be completely unrelated to your initial changes.

The Oracle Magician

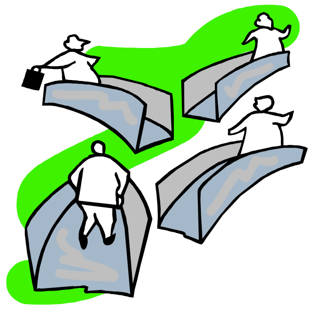
Editor: Chris Lawson
OracleMagician.com

Continued from page 3

faster; rather, it was *why* the program needed to check an entire year of payments.

In this case, the issue was not even a technical issue—it was the *business requirement* driving the query. The business need was overly broad, causing the database work to be much more extensive than necessary.

Complex Problems Require Flexible Approaches



Clearly, “one size fits all” algorithms or tools rarely work for complex tasks in huge organizations. Just like the silly flow chart that purported to solve all problems, there isn’t a “flowchart to success” for the performance analyst.

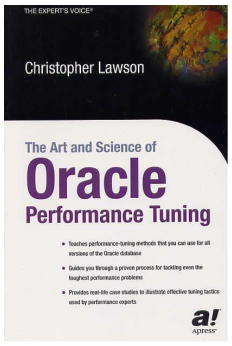
That’s not to say that using tools is always a bad idea. Indeed, tools can be helpful in checking issues such as missing indexes, bad statistics, etc. These type of tools can certainly solve some types of problems—but their method is not *universally applicable.* That is, the *scope* of their usefulness is limited.

One Approach Solves All?

I have found that few problems are efficiently resolved by relying *solely* on one approach. The best DBAs and performance specialists I know use a tool kit of different approaches. For any given assignment, these experts adapt their heuristics to fit the circumstances. If necessary, they develop new approaches or create different scripts.

Updates and Notes

If you find these tuning issues interesting, I discuss performance tuning in greater detail in my book, *The Art and Science of Oracle Performance Tuning*. It is available at most large bookstores, or online at Amazon.com



THE EXPERT'S VOICE®

Christopher Lawson

The Art and Science of
Oracle
Performance Tuning

- Teaches performance-tuning methods that you can use for all versions of the Oracle database
- Guides you through a proven process for tackling even the toughest performance problems
- Provides real-life case studies to illustrate effective tuning tactics used by performance experts

a! Apress®