Tech Notes

New Tools for Faster SQL Tuning and Analysis

Embarcadero Technologies

November 2009

Corporate Headquarters
100 California Street, 12th Floor
San Francisco, California 94111

EMEA Headquarters
York House
18 York Road
Maidenhead, Berkshire
SL6 1SF, United Kingdom

Asia-Pacific Headquarters
L7. 313 La Trobe Street
Melbourne VIC 3000
Australia

INTRODUCTION

SQL tuning is complicated. The growing size and complexity of databases, innovations in application development tools that automatically generate SQL code that is often inefficient, and built in database optimizers that vary in their ability to pick the optimum execution path are all common challenges within the realm of performance optimization. Adding to the complexity, pressures to reduce hardware spending while meeting strict SLA and other performance requirements are constantly pushing the limits of SQL profiling and tuning tools as well as the expertise of the DBA. And now there is a growing trend to arm the developers themselves with SQL optimization tools and integrate SQL tuning into the development lifecycle.

To achieve the best results under these circumstances, organizations should standardize on one tool that serves the needs of the developer as well as the more advanced SQL tuning techniques required by the DBA. Traditionally, DBAs use hint injections and SQL rewrites, but now there are new innovations in SQL tuning like index analysis and, most recently, Visual SQL Tuning (VST). These new innovations serve two masters; they reduce the complexity of SQL tuning for the developer, while providing more detailed and advanced analysis for the DBA.

TRADITIONAL SQL TUNING

When you execute SQL on a particular database, the database uses its own internal optimizer to determine the quickest execution path for that SQL, based on the way it is written, and a 'cost' is reflected in the resultant explain plan. Because the database vendors are continually improving the performance of their respective optimizers, the database optimizer generally picks the optimum path. There are, however, always cases where the optimizer needs to be forced down a faster execution path in order to ensure the best possible performance. This method is delivered through a hint that is injected into the SQL to direct the optimizer as the SQL is executed.

Rather than injecting hints for the optimizer to follow, some DBAs prefer to rewrite the SQL to accelerate the query. Examples of common SQL coding mistakes that can be repaired by SQL rewrites include Cartesian join elimination, expression transformation, invalid outer join, transitivity, and push sub-query to name a few. The one thing to be wary of with some SQL rewrites is that the result set of the query could be affected, so it's important to take this into account when leveraging rewrites.

Embarcadero® DB Optimizer™ provides both hint injections and SQL rewrites for the DBA or developer to use as they wish. The DB Optimizer tuning feature automatically examines an SQL statement and suggests applicable hints as well as SQL rewrites in parallel, highlighting the SQL rewrite options to ensure you are aware that the result set may be altered (i.e. missing a valid join criteria).

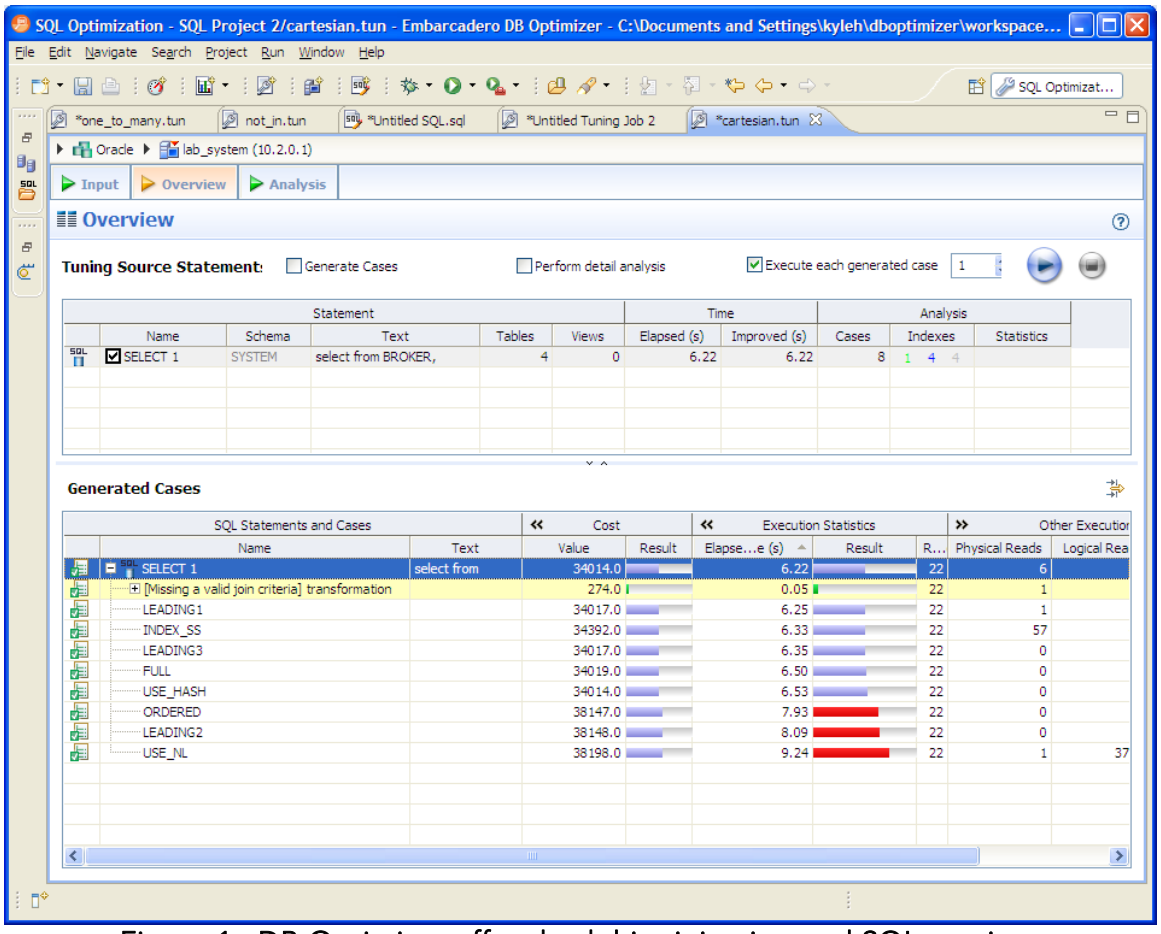

Figure 1: DB Optimizer offers both hint injection and SQL rewrites

INNOVATIONS IN SQL TUNING – INDEX ANALYSIS AND VISUAL SQL TUNING

Most SQL tuning tools stop at hint injection and SQL rewrites, but DB Optimizer takes analysis and tuning a step beyond with advances in index analysis and unparalleled innovation with Visual SQL Tuning (VST) diagrams. Index Analysis empowers DBAs and developers to fully examine SQL execution paths to better understand which indexes are used, not used, or missing. And if an index is missing, DB Optimizer offers indexing recommendations for optimum performance. The VST diagram displays indexes and constraints on tables and views, as well as the joins used in a SQL statement. This innovative visual format quickly reveals opportunities to tune the SQL or schema and enhance overall database performance.

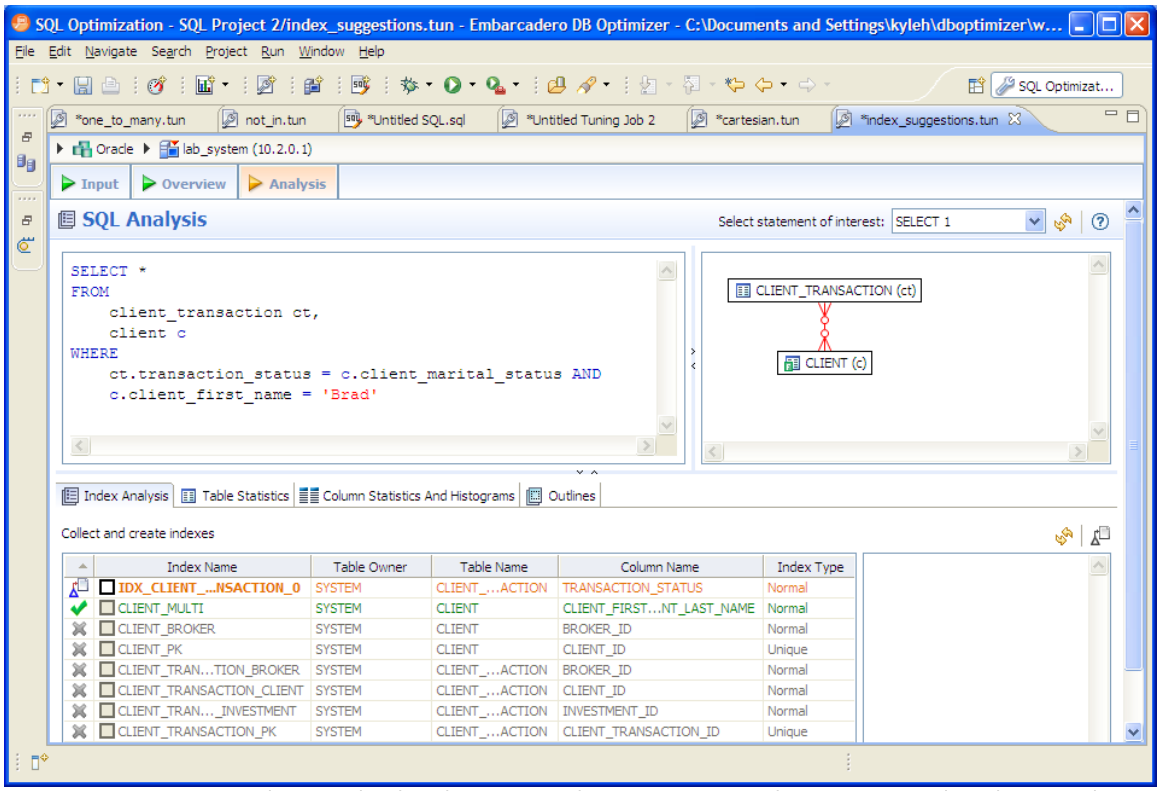

Figure 2: SQL Analysis tab displays Visual SQL Tuning diagrams and index analysis

ADVANCED INDEX ANALYSIS

Given a particular SQL statement and its variables (i.e., tables and columns used, join and filter criteria, ordering in the 'where clause', etc.), the database-specific optimizer returns an explain plan that charts the step-by-step execution path through the database, including the indexes leveraged along the way. DB Optimizer takes this information as well as information about existing indexes and charts them in four different ways: indexes that are being used (green); indexes that exist but are not used by the database-specific optimizer (blue); indexes that exist on the table but are not leveraged based on the existing 'where clause' (grey); and indexes that do not exist but are suggested by DB Optimizer (orange).

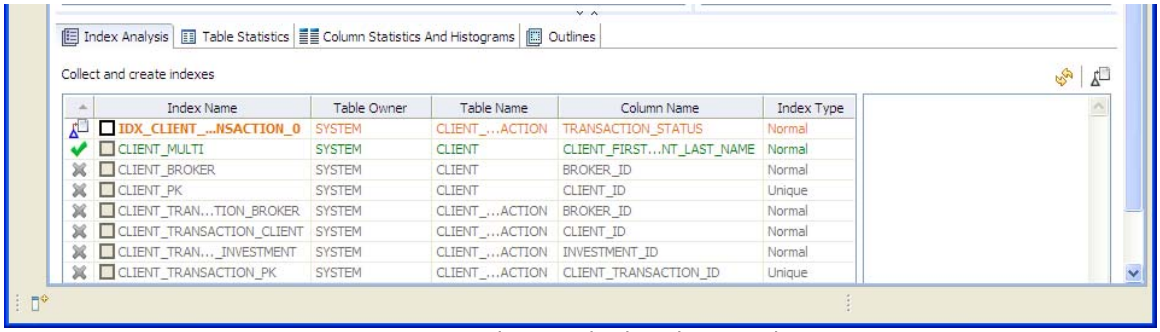

Figure 3: Color-coded index analysis

In the case where the index does not exist, DB Optimizer allows you to generate the recommended index at the click of a button. It is also very useful to understand when an index is not being used. A developer might write an index in an attempt to speed up the query, but then learn the optimizer is not using it for that query. This information gives the developer the opportunity to rewrite the index to try to get the optimizer to use it or decide to remove the index entirely if it is superfluous.

VISUAL SQL TUNING (VST) DIAGRAMS

Delivering a new level of innovation in SQL tuning, DB Optimizer can parse an SQL query and analyze the indexes and constraints on the tables and views in the query and display the query in graphical format. The resultant VST diagram, which can be displayed in either Summary Mode or Detail Mode, helps developers and DBAs see flaws in the schema design such as Cartesian joins, implied Cartesian joins, and many-to-many relationships. The VST diagram helps you quickly understand the components of an SQL query, thus accelerating trouble-shooting and analysis. This is a very useful feature for the DBA to initiate informed design conversations with Data Architects for cross-organizational collaboration.

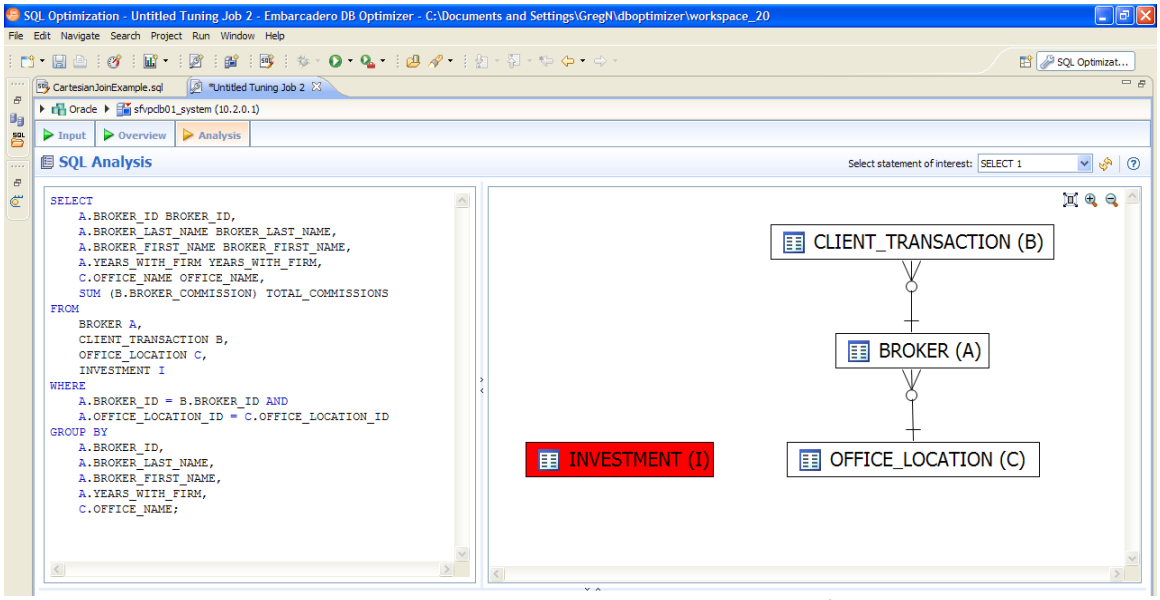

Figure 4: Visual SQL Tuning diagram provides at-a-glance information about query or schema design – in this case a Cartesian join

In the case above, where a Cartesian join is discovered, DB Optimizer resolves the performance issue based on the results of analysis and rewrites the query.

CONCLUSION - BEST PRACTICES IN SQL TUNING

When faced with a number of options, it is always a good idea to take a step back and look at your existing workflow and insert best practices in each step of the process. The best way to identify database bottlenecks caused by SQL is to start with a profiling session. Once DB Optimizer has filtered out the worst-performing SQL code, you can import that SQL code

directly into the tuner and into the Input Tab. The Overview Tab displays case generation using the SQL rewrites and hint injections. DB Optimizer's Load Editor, a SQL stress testing feature, lets you simulate parallel sessions and number of executions to measure any performance gains side-by-side with pre-tuned and tuned SQL while profiling the database. When classic tuning methods will not suffice or service levels are still out of reach after retesting, the Analysis Tab reveals opportunities for advanced tuning methods.

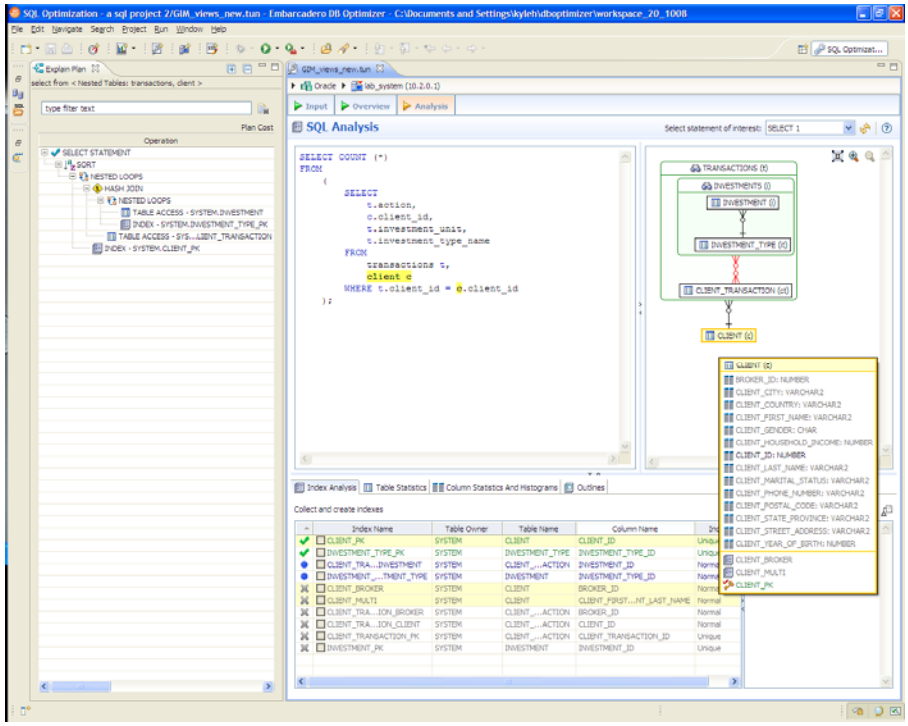

Figure 5: DB Optimizer Analysis Tab offers both developers and DBAs the tool they need to satisfy their respective SQL optimization needs.

Embarcadero Technologies, Inc. is a leading provider of award-winning tools for application developers and database professionals so they can design systems right, build them faster and run them better, regardless of their platform or programming language. Ninety of the Fortune 100 and an active community of more than three million users worldwide rely on Embarcadero products to increase productivity, reduce costs, simplify change management and compliance and accelerate innovation. The company's flagship tools include: Embarcadero® Change Manager™, Embarcadero® RAD Studio, DBArtisan®, Delphi®, ER/Studio®, JBuilder® and Rapid SQL®. Founded in 1993, Embarcadero is headquartered in San Francisco, with offices located around the world. Embarcadero is online at www.embarcadero.com.