SQL PerformanceExpert® for Db2 z/OS

Improve dynamic and static SQL performance

Increase QA productivity and reliability by reducing retesting needs

Increase programmer productivity by automating SQL analysis

Improve SQL programming practices with an expert system with hundreds of predefined rules

Provide informative, automated reports on analysis performed

Explain trigger packages

Collect host variable values for static and dynamic SQL

Use TSO/ISPF as well as the Eclipse based world via Plug-ins: Rational for z (RDz) and Rational Software Architect (RSA)

Arrives prepackaged with rules for application programs and DBAs

Provides ability to build site-specific rules with an easy-to-use online dialog

SQL PerformanceExpert (SPX) automates analysis and tuning of SQL, making the effort easy and efficient.

Well-tuned and high-performing SQL statements equate to efficient CPU usage, improved response times, reduced I/O activity, and reduced locking. SPX provides automated and intelligent analysis and tuning of static and dynamic SQL statements, improving the overall performance of production applications.

SPX eliminates many of the performance problems typically found in a production environment by creating a simulated production environment for application testing. It detects poor performance early in the development process, saving valuable technical and CPU resources and associated cost. With SPX, your technical staff is empowered to work smarter — saving time, increasing productivity and maximizing the use of the shrinking batch window.

OPTIMIZE SQL
SPX is a complete toolbox that helps produce optimized SQL through a multi-tiered approach for the whole SQL lifecycle.

- Development — Perform tuning during development stage, identifying and intercepting potential problems before they are moved into production

- QA/Pre-production — Streamline the QA process

- Drop and create indexes to evaluate the resulting performance

- Post-production — Analyze production applications to address high-consumption processes

SPX exploits all Db2 13 features, allowing you to fully optimize your SQL workload. By taking advantage of features such as multi-row fetch and EXPLAIN tables, SPX provides a new milestone of speed and functionality.

Well-tuned and high-performing SQL statementsequate to efficient CPU usage, improved response times, reduced I/O activity, and reduced locking.
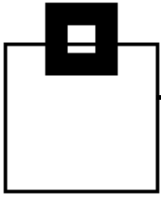
Simulation & Virtualization

- Software (ProductionSimulator)

- Hardware (VOX)

- Exploits DSN_VIRTUAL_INDEXES, by dropping/creating indexes you see the affect on SQL performance

---

Monitors production applications to identify performance problems

---

Flexibly integrates into existing QA and change control environments

---

Provides a single point of control via DRDA connectivity

Reduces QA tasks significantly reduced by SQL aggregation

Identifies performance relevant changes

Reduces DASD requirements in the test environment

Includes a global explain table pool

Supports XML/XPath analysis

Supports catalog history to create production baselines

SQL analysis and monitoring — Performs intelligent analysis and ongoing monitoring so you can easily identify SQL statements that negatively affect application performance. SQL performance is easily improved by implementing recommendations provided about poorly performing SQL statements.

By quickly identifying only new and changed SQL, the QA workload is reduced. Analysis includes the capturing and preparation of dynamic SQL statements that are needed for comprehensive analysis. SPX provides standards for SQL application development through an expert rule system that can be easily customized. Recursive viewing performs analysis of predicates within a view. A SQL aggregation feature further enhances the analysis process. SQL aggregation matches all like SQL so that you can easily see the accumulated CPU costs and row level statistics for each group of aggregated statements.

Production simulation — Enables development of high-performance applications, by generating reliable catalog statistics for use during testing and SQL analysis. This feature allows you to easily distribute Db2 statistics throughout multiple systems and creates rule production baselines to predict access path changes.

DBRM checking — Reduces I/O activity and locking problems with fast and efficient examination of DBRMs identifying potential performance problems. Use DBRM checking to determine compliance with current quality assurance rules by verifying new programs during the compile procedure and analyzing a single DBRM or whole DBRM libraries—without accessing the Db2 catalog. Easily generate predefined rules and recommendations that help improve quality and performance; these rules can be customized according to severity level.

Package management — Examines consistency of load modules, DBRMs, and the Db2 catalog to locate packages that are unnecessary or not bound, automatically cleaning up packages that are no longer needed.

SUMMARY

Properly tuned SQL and a well-tuned Db2 environment can yield noticeable performance improvements. These can mean faster response times for Db2 applications, improved user experience, and faster throughput. The key is a combination of programming practice, system optimization, and effective use of expert SQL tuning tools to automate simulation and code analysis, especially allowing you to take control of both static and dynamic SQL.