

Automatic Database Configuration for DB2 Universal Database: Compressing Years of Performance Expertise into Seconds of Execution

Eva Kwan
evakwan@ca.ibm.com

Sam Lightstone
light@ca.ibm.com

Berni Schiefer
schiefer@ca.ibm.com

Adam Storm
ajstorm@ca.ibm.com

Leanne Wu
leannewu@ca.ibm.com

IBM Canada, Toronto Laboratory

Abstract:

This paper describes the DB2[®] Configuration Advisor, an expert tool for the configuration of DB2 databases. This advisor has shown dramatic results for tuning and configuring DB2 servers on UNIX[®] and Windows[®] platforms. The recognition of the essential need for administration and design tools has spurred renewed interest among leading relational database management system (RDBMS) vendors. The DB2 Configuration Advisor is a key feature in DB2's Autonomic Computing self-managing technology portfolio. This paper discusses the purpose and features of this expert advisor. Experimental results are presented with a description of the advisor's interfaces.

1. Introduction and motivation

In this paper we present an overview and experimental results for a database configuration tool for DB2 Universal Database[™] for UNIX and Windows (DB2), known as the DB2 Configuration Advisor. This advisor makes recommendations on initial settings for configuration parameters and memory allotment within the database management system, which can be adopted by inexperienced administrators or fine-tuned by more experienced personnel. The performance of a tuned configuration versus an untuned configuration may be dramatic, with a measurable and significant performance improvement. The DB2 Configuration Advisor is one of a growing set of features for DB2 that reduces the human expertise required in database tuning and administration by combining automation, artificial intelligence, and expert advice. The experimental results shown illustrate how this technology can greatly simplify expert tuning work.

Understanding the priorities of administrators is critical in reducing manual administration. The increased complexity and volume of data seen in modern database systems has increased the burden of database administrators. The database administrator now faces a diverse combination of tasks, as shown in Figure 1.

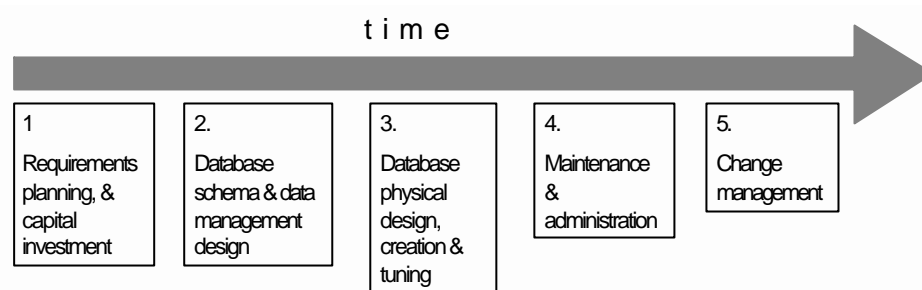


Figure 1. Tasks of a database administrator over the life cycle of a database

Often the design choices for databases are complex and nontrivial. Once a database has a physical and logical design, the operation of the database requires substantial human attention for numerous tasks including, but not limited to: table reorganization, data statistics collection, backup control, disaster recovery planning, performance tuning, and problem analysis.

As database management systems have grown in size and complexity, recognition of the importance of ease of administration and of better design tools has also increased, and this recognition has spurred renewed interest in research and development of software to reduce administration requirements [Be98] [Br94] [Ch99] [FST88]. IBM® has recently increased its focus on autonomic computing, which aims to produce self-managing systems that run with a minimum of human intervention. This research has resulted in IBM's Autonomic Computing initiative, which aims to enhance the IBM portfolio of clients, middleware, servers, and storage products through self-managing features, and which has the long term objective of producing true autonomic computing [IBM02a]. This initiative demonstrates IBM's commitment to providing solutions for increasingly complex systems. Within DB2, the Autonomic Computing initiative has as its objective the development of databases that will require minimal expertise and maintenance. As an automatic configuration feature, the Configuration Advisor falls under the aegis of IBM's corporate wide Autonomic Computing research and development effort, of which the DB2 Autonomic Computing effort is a part.

The goal is to create intelligent software tools that will reduce the burden on database administrators by providing expert design systems, performance tuning and configuration technology, with easier to use interfaces for administration and tools for automation. One area in which research has been focused has been in the development of internal tuning and configuring technology within the product deliverable. The DB2 UDB Configuration Advisor is one such product of this research. It allows large database servers to be configured for performance in seconds. While the utility has been available with DB2 for a few years, significant improvements have been performed for DB2 Universal Database version 8.1. IBM announced IBM DB2 Version 8 for Linux, UNIX and Windows on September 17, 2002, and the product was made generally available on Nov. 21, 2002.

2. Objectives and features

The Configuration Advisor is part of DB2 Universal Database for UNIX and Windows. IBM's DB2 Universal Database product provides a set of configuration parameters settings that can be used to configure and tune the DB2 environment. The objective of the Advisor is to define a set of initial database configuration parameters and memory assignments to optimize system performance without extensive monitoring of the system. These configuration parameters include ones that control memory distribution (sorting, locking, caching database pages and working heaps), parallelism, I/O optimization (asynchronous page readers and writers), many aspects of logging (file size, buffer size), and recovery. While there are default settings provided with each installation of DB2, it would be impossible to provide a single configuration that would perform equally well for the diverse workloads and systems that use DB2. The complexity of the memory topologies for enterprise database management systems makes these assignments a difficult task, even for skilled administrators deeply aware of their workload characteristics. The Configuration Advisor can be used both to recommend configuration settings, as well as to apply the settings.

The Configuration Advisor has been developed with three execution interfaces. The first is a graphical interface that is part of the DB2 Control Center, the graphical interface for DB2's administration. The second is through a DB2 system command, whereby the Configuration Advisor can be invoked through a text command [IBM02c]. The command level interface is a popular interface for DB2 administration, and also lends itself towards scripting. The third interface is through programmable APIs (DB2 provides both a C level API and a stored procedure interface to the Configuration Advisor) which allows database applications, including independent software vendors (ISVs), to programmatically invoke the Configuration advisor for seamless configuration of DB2 databases [IBM02b].

3. Design

The DB2 Configuration Advisor is designed on the principle that configuration choices can be made by modeling each database configuration setting as a mathematical expression which combines three sets of information regarding the system environment, the database characteristics and user priorities. The three sets of information are:

1. User specification of the database environment (designed as a small set of basic input, generally requiring minimal skill).
2. Autonomically sensed system characteristics (such as number of CPUs, disks, amount of RAM, number and size of relational tables, etc).
3. Expert heuristics for database configuration, as reported by experienced database administrators and performance tuning experts.

The first two of these information classes are essentially parametric, and can typically be represented as scalar values which are input by the user. The advisor is designed so that the responses collected from the user make up the minimal set of characteristics that are

required to effectively configure a database. Furthermore, the advisor is designed so that the questions require neither lengthy nor detailed analysis of the database in order to be answered. The second set of input information is obtained through the automatic detection of system resources, which includes the amount of physical memory, number of disks, storage available, and the number of CPUs. Included in this second set of information is information about the existing database's current characteristics. This information includes number of tables defined, the number and size of defined buffer pools, and the number of database containers. The first two classes of parametric information are then combined with a set of expert heuristics to define mathematical expressions of the estimated ideal settings, which define the configuration model. The expert heuristics were collected through an ad hoc process of surveying over a dozen of DB2's leading performance experts and architects, as well as through review of published information on DB2 performance tuning [SV99] [IBM02d].

The configuration model expresses the configuration settings as a mathematical expression. The model for the configuration settings is further divided into three distinct classes:

1. Independently modeled configuration settings, which can be modeled independent of other configuration settings.
2. Dependant configuration setting, where the value of one setting affects the model of another (or perhaps co-dependency), and
3. Zero sum game relationships (such as memory for sort, caching, locking etc) in which a fixed resource must be divided among a set of configuration parameters.

Figure 2 illustrates the design model for the Configuration Advisor. With one of the design goals of the advisor being to reduce human skill and involvement, the user specification is kept intentionally trivial, with a long term aim of reducing it further.

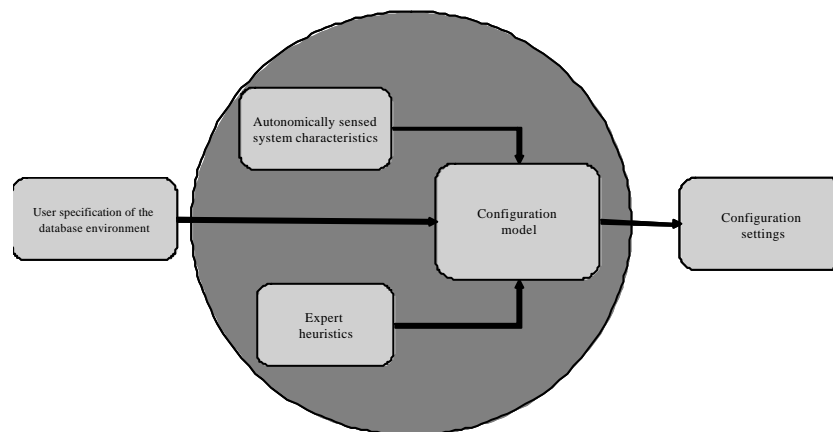


Figure 2: Design model for the DB2 Configuration Advisor combining user specified basic description, automatically sensed system characteristics and expert heuristics.

The configuration model includes algorithms that are based on documented guidelines, expert advice, and system behaviors generalized from analysis of workloads. For most configuration parameters, the advisor automates the calculation of settings based on known guidelines. Other configuration parameters are more complex, and this is where the advisor incorporates the expert heuristics of performance experts and architects at IBM. The developers have an in-depth insight into DB2's implementation, and performance experts have valuable experience in performance tuning.

The goal of the advisor is to be useful across a wide range of systems so its internal algorithms also include information gleaned from analysis of a wide range of workloads. The analysis allows the advisor to better classify systems and choose the key characteristics on which to base recommendations, a difficult task due to the complex interactions and uniqueness of each database. The result, as our results indicate, is a good set of initial settings, which can later be fine-tuned as administrators monitor system performance during run time.

To the authors' knowledge this approach is unique in the industry. Several other vendors provide database configuration tools; however these tools typically focus on a distinct subset of configuration settings (such as buffer pool configuration), are not generally based on parametric models of database configuration, and typically lack the extensive autonomic sensing capability that the DB2 Configuration Advisor embodies.

4. Experimental results

To examine the effectiveness of the Configuration Advisor, we performed four experiments on systems running distinct workloads and environments, comparing the performance of the database after tuning by the Configuration Advisor to the system performance achieved through tuning by an expert database administrator. These experiments include a degree of inaccuracy, since the tuning performed by human administrators has variable quality, and it is impossible to assess how close this tuning is to optimal. However, what these experiments do illustrate is the degree to which the Configuration Advisor is able to tune the database system relative to the human expert. While the human administrator typically spends a number of days tuning these configuration parameters, the advisor provides recommendations for the same set of parameters within seconds.

The four experiments included:

- An Online Transaction Processing (OLTP) industry standard workload, run once on a 32-bit implementation, and again on a 64-bit implementation. This benchmark simulates a population of terminal operators executing transactions against a relational database. The benchmark models the transaction environment of an order-entry environment.
- Two operational workloads tested on-site at two of the world's leading global investment banks.

While all tests in this set of experiments were run on AIX, this was more due to general availability of test systems rather than any platform-specific constraints on the advisor.

4.1 64-bit OLTP

The 64-bit OLTP experiment was run on an RS/6000[®] 44P Model 270 4-way 375 MHz Power3-II server, configured for 2-way processing. The system had 8 GB memory, and was running AIX 4.3.3 with DB2 UDB EE V7.2 (64-bit). The storage system on this server included 3 ServeRAID 4H (SCSI) adapters as follows: fifty-six 9 GB SCSI Disks (data), fifty-six 9 GB SCSI Disks (data), fourteen 9 GB SCSI Disks (log, backup, temp tablespaces). The hand-tuned system was configured by DB2 performance specialists at IBM. In this experiment the database performance after tuning with the Configuration Advisor was observed to be 93.58% of the hand-tuned system.

4.2 32-bit OLTP

The 32-bit OLTP experiment was run on an RS/6000 44P Model 270 4-way 375 MHz Power3-II server, with 8 GB of memory, running AIX 4.3.3 with DB2 UDB EE V7.2 (32-bit). The storage system on this server included 3 ServeRAID 4H (SCSI) adapters as follows: twenty-eight 9 GB SCSI Disks (data), forty-two 9 GB SCSI Disks (data), fourteen 9 GB SCSI Disks (log, backup, temp tablespaces). As in the 64-bit experiment, the hand-tuned system was configured by DB2 performance specialists. In this experiment the database performance after tuning with the Configuration Advisor was observed to be 91.52% of the hand-tuned system.

4.3 Global investment bank A

Two sets of tests were conducted on production workloads. The first of these tests was conducted on a high volume OLTP test system that stress tested the database using an authentication application. The application simulated authentication operations that would result from customer accesses of the bank's products.

The test environment consisted of a 6way RS/6000 Model F80 server with each processor running at 500 MHz. The server had 4 GB of memory and was running AIX 4.3.3. Its data was spread over 14 of its 16 drives (two 9 GB drives and twelve 18 GB drives). In this experiment the hand-tuned system was configured by the bank's database administrator. The results of the experiment show that after the Configuration Advisor was run, the workload performed 5.62% better than the hand-tuned system.

4.4 Global investment bank B

The second of the two production system tests was conducted on a system that allows customers to view their web pages and change their preferences. These preferences are sequenced into records in a DB2 UDB database. The transaction types are mixed between selects and inserts or deletes, with 60% of the workload being selects. To

provide for load balancing and for hot backup solutions, the system exploits two servers with peer-to-peer replication.

The two servers had identical hardware specifications and topologies. Both were IBM RS/6000 Model 270 4-way 375 MHz servers with 4 GB of memory running AIX 4.3.3. The storage system included sixteen 18 GB drives. At the time of the experiment, the research prototype of the Configuration Advisor did not include a model for systems that deploy peer-to-peer bulk replication. As a result, the advisor under configured memory for locking and logging, resulting in sub optimal system performance. The administrator at the customer site decided to leave the settings for locking and logging at their pre-advisor values and adopted all of the remaining Configuration Advisor recommendations. The result was a best-ever system performance for the database, with the new configuration performing at 224.72% of the original administrator-tuned throughput. It is interesting to note that the alteration in buffer pool size (set by the advisor) was an important change in the configuration. While the advisor was clearly inadequate without the planned extensions to support bulk replication, the recommendations of the advisor were effective in combination with two adjustments by the database administrator (for locking and logging) in achieving a massive improvement in system performance.

4.5 Experimental Summary

In all of these experiments (the result of which are in Figure 3) the Configuration Advisor was able to evaluate and compute a revised or proposed configuration in two or three seconds. The performance of the advisor is significant when compared to the amount of time typically spent configuring large database management systems, usually on the order of one to two weeks.

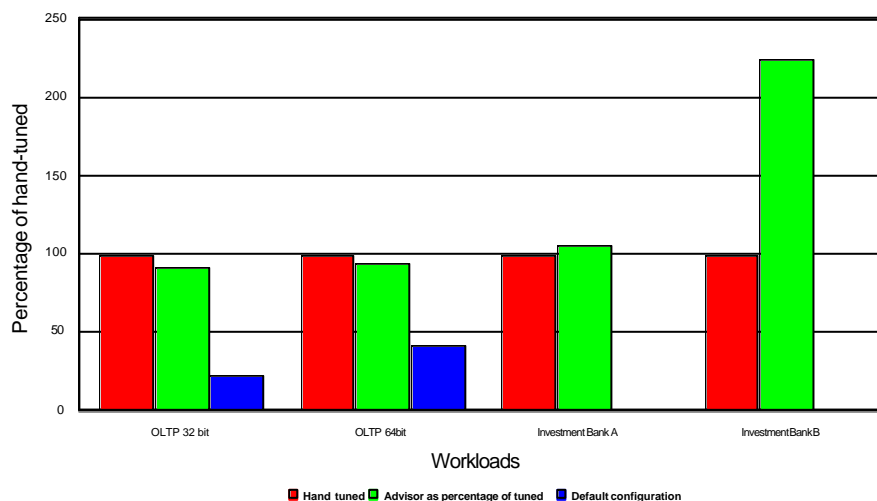


Figure 3. Database performance for hand-tuned parameters versus Configuration Advisor tuning

In the two cases in which the advisor was tested against systems configured by IBM performance experts, the resulting performance was within 10% of the hand-tuned result. As well, in these two cases when the Configuration Advisor selections were tested against the default configuration, the Advisor significantly outperformed the default settings.

In the two cases where the Configuration Advisor was tested against databases in the field, the Configuration Advisor exceeded the hand-tuned result (though in the case of “Global investment bank B” two parameters were left at the hand-tuned setting to account for the special needs of peer-to-peer replication).

5. Conclusions

The Configuration Advisor has the ability to reduce and possibly eliminate the tedious and time-consuming task of configuring a system for desired performance as is shown with the benchmarking results. In this paper we have studied the quality of the recommendations provided by the Configuration Advisor by comparing its performance to both industry standard benchmark systems tuned and configured by database specialists inside IBM, and systems in use by two major brokerage firms. Our results indicate that the advisor shows promise for providing quality recommendations on database configuration, and, in some cases, exceeding the performance quality of human-configured systems, as observed in the case of the two investment banking systems studied. If performance on the system is not an absolute priority, the Configuration Advisor eliminates the need for frequent manual tuning of performance related configuration parameters. If the system is required to be configured for near optimal performance, the advisor supplies a configuration that can serve as a springboard for further fine-tuning based on specific workload characteristics.

The simplicity and speed of the Configuration Advisor provide compelling arguments for deployment, and as part of both the Autonomic Computing initiative the advisor demonstrates some of the benefits that autonomic computing promises to bring. All these factors combine to make the Configuration Advisor a valuable tool for database administrators.

6. Future work

In addition to further refinement of the current modeling for database configuration, there are several avenues for future research. The first is incorporating automatic workload characterization schemes into the inputs that are used by the Configuration Advisor. Currently, the Advisor depends on user-specified workload characterizations. If the advisor could automatically detect these characteristics the result would be less work on the part of the user and increased Advisor accuracy. Another area in which further work could be performed is in investigating closed-loop performance feedback for the advisor. The advisor’s current architecture produces identical results regardless of the number of times it is run, if the same inputs are used on a given system. By providing a mechanism by which the advisor could learn about the effectiveness of its past

recommendations, it will be possible for the advisor to provide a better and more intelligent configuration with subsequent iterations. Additionally, there is ongoing research to refine the configuration model within the advisor. Improving this model will allow the advisor to perform well on an even wider range of database systems.

References

- [Be98] Bernstein, M. et. al.: The Asilomar Report on Database Research. ACM SIGMOD Record 27(4), September, 1998; pp. 74-80.
- [Br94] Brown, K. et. al.: Towards Automated Performance Tuning For Complex Workloads. Proceedings of 20th International Conference on Very Large Databases, Santiago de Chile, Chile, 1994. Morgan Kaufmann, 1994; pp 72-84.
- [Ch99] Chaudhuri, S. et. al.: Self-Tuning Technology in Microsoft SQL Server. IEEE Data Engineering Bulletin 22(2), June 1999; pp. 20-26.
- [FST88] Finkelstein, S.; Schkolnick, M.; Tiberio, P.: Physical Database Design for Relational Databases. ACM Transactions on Database Systems 13(1), March 1988; pp. 91-128.
- [IBM02a] IBM Research: Autonomic Computing. Online:
<http://www.research.ibm.com/autonomic/>
- [IBM02b] IBM Software: DB2 Information Center: db2AutoConfig – Autoconfigure. Online: <http://www-3.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v8infocenter.d2w/report?target=mainFrame&fn=r0003294.htm>
- [IBM02c] IBM Software: IBM DB2 Universal Database Version 8 Command Reference. Online: <ftp://ftp.software.ibm.com/ps/products/db2/info/vr8/pdf/letter/db2n0e80.pdf>
- [IBM02d] IBM Software: IBM DB2 Universal Database Version 8 Administration Guide: Performance. Online: <ftp://ftp.software.ibm.com/ps/products/db2/info/vr8/pdf/letter/db2d3e80.pdf>
- [SV99] Schiefer, B.; Valentin, G.: DB2 Universal Database Performance Tuning. IEEE Data Engineering Bulletin 22(2), June 1999; pp. 12-19.

Notices

The following are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AIX

DB2

IBM

Universal Database

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.