
Data Lifecycle Management Series

IBM DB2

By Robin Schumacher
VP Product Management, Embarcadero Technologies, Inc.

July 2004

Corporate Headquarters
Embarcadero
Technologies
100 California Street 12th
Floor
San Francisco, CA 94111
www.embarcadero.com

EMEA Headquarters
Embarcadero Technologies,
Ltd.
Thames House
17 Marlow Road
Maidenhead
Berkshire
SL6 7AA
United Kingdom
www.embarcadero.co.uk

Table of Contents

Introduction	3
Overview of Data Lifecycle Management.....	3
Embarcadero Data Lifecycle Management Solutions.....	4
Models that Drive Data	4
Architecting Data	5
Moving Data with Models	7
Effective Data Management	8
Streamlining DB2 Development.....	8
Upgrading Administration Abilities	9
The Need for Proper Change Management.....	10
Guaranteeing High Performance.....	12
Testing for Performance Success	13
Exceeding Performance Expectations	14
Conclusion.....	15

Introduction

In a 2004 market share study by Gartner, IBM continued to be the database market leader for the second year in a row with a 33.8 percent share. IBM is forging ahead with its DB2 RDBMS offering on a variety of hardware and software platforms with significant new releases in 2004 for the z/OS and Windows/UNIX/Linux platforms. The DB2 platform also continues to maintain a high performance ranking for the various TPC benchmarks. All of the major application vendors such as SAP, Siebel, Peoplesoft, etc., are continuing with developing DB2 compatible solutions. All of these indications point to a thriving RDBMS platform that IBM is enhancing and promoting on an ongoing basis.

The success of the DB2 platform has also steadily increased the number and size of DB2 databases and the applications that work with these databases. This has significantly increased the workload of the database professionals that need to develop on and administer IBM's database.

This paper examines how to be successful on the IBM DB2 platform through the use of smart data lifecycle management techniques. It also showcases how software tools from Embarcadero Technologies automate and streamline the process of data lifecycle management, with the end result being dramatic savings in time and impressive reductions in overall implementation costs.

Overview of Data Lifecycle Management

Data lifecycle management is the process of cost-effectively managing and optimizing critical data and database infrastructures. It involves managing data from its logical conception down to its eventual archival or removal. Throughout the process, there are many individual disciplines that must be practiced by the database professional. These disciplines tend to fall under three broad headings:

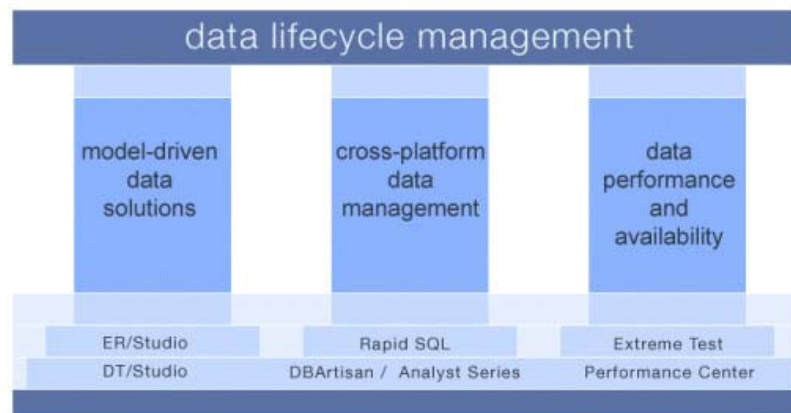
1. **Data Design and Movement** — this area concerns itself with defining individual data elements, their inter-relationships, and miscellaneous dependencies, apart from any actual physical representation. Once data has been physically created, the focus is then shifted to how that existing physical data is moved, migrated, and transformed into meaningful information. Models are the de-facto standard way to define logical data elements and their relationships, and models also tend to be the best way to define physical data movements between various source and target systems. The roles most active in this area include data modelers, data architects, database administrators, and data extract-transform-load (ETL) specialists.
2. **Data Management** — this category covers everything from the development of the business logic that acts upon data to the complex administration of physical data stores (databases). The database developer is typically the one who builds the business logic that is used to intelligently store and present data. The database administrator is the person who carries out the day-to-day management of actual databases, with their focus directed on things like storage, security, data objects, change control, and backup/recovery management.
3. **Data Availability and Performance** — for data to be of any use, it obviously has to be available to the decision makers who will use it. In addition, because of the accelerated pace of business, it is imperative that data be delivered in the fastest possible manner so important decisions are not delayed. Protecting the availability and performance of data

is the database administrator who is responsible for optimizing database systems for peak performance, as well as putting together availability mechanisms to keep data accessible. In addition to the DBA, system test specialists are often called upon to stress test applications and databases to ensure that they will be able to stand up under heavy transactional and query loads.

Embarcadero Data Lifecycle Management Solutions

Embarcadero Technologies offers solutions for every area of the data lifecycle. All Embarcadero lifecycle management products offer near-instant ROI and dramatic cost savings because of their ease-of-use and powerful, timesaving productivity aids.

Embarcadero's integrated product set consists of six core products that handle the bulk of data lifecycle management tasks, along with a number of supporting products that round out the complete data lifecycle suite. Embarcadero offers complete cross-platform coverage, so it does not matter if Oracle, Microsoft SQL Server,



DB2 UDB, DB2 z/OS, or Sybase is being used; everything looks and feels the same. Plus, all database platforms can be seamlessly managed from within the same core consoles, so a data professional does not have to jump in and out of separate products to manage their heterogeneous environment.

The rest of this paper will explore each area of the data lifecycle, with special focus being given to how data lifecycle management should be carried out on the IBM DB2 platform.

Models that Drive Data

Whether it's a new DB2 system that is being proposed, an existing DB2 database that needs to be understood, or a complex migration to or from DB2 that needs to be tackled, nothing accelerates the process of defining, understanding, and moving data than modeling. Through models, the fog of confusion lifts over how data is defined, interrelates, and exists.

Architecting Data

Because DB2 databases are seeing much larger data sizes than even 5 years ago, database professionals are paying much more attention to how a DB2 database is logically and physically designed. The database professionals are starting to realize that in order to obtain reasonable performance, availability, and scalability with DB2, it is crucial to model a DB2 database so that it continues to perform with increased workloads. A data architect or a team of data modelers carefully builds the definition of their database in a model so that data elements are properly defined, data standards are enforced, and a design is constructed that will meet the demanding needs of a busy system.

Data architects and modelers work at the logical level, meaning that a data model database agnostic at first. It's in the logical design that entities are defined along with their individual data elements, and the relationships between those entities are identified and cataloged. The logical layer is also where a data dictionary is first created (the component that holds reusable data elements).

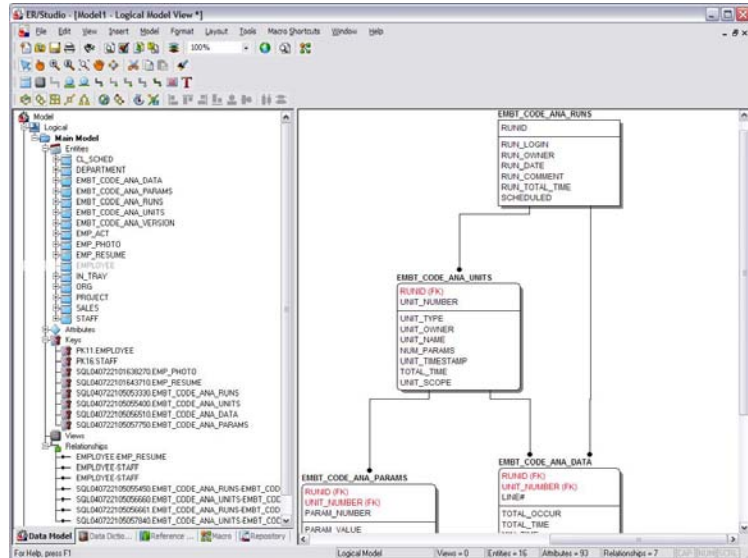
Once a logical model that accurately describes and documents the needs of a new database has been designed, a data architect will then turn their attention over to porting that design to a physical model that is tied to a specific database engine. The logical entities become tables, the relationships become foreign key constraints, and the individual data elements become columns. DB2 architects then begin making decisions about what tables will benefit from a clustered index, whether all foreign key columns should be indexed, and whether to split tables and indexes among different DB2 tablespaces. Other objects like views, procedures, triggers, and even stored database code will also find their way into physical designs. Architects and modelers should be careful when changing physical model object definitions so that the physical and logical models are kept in sync. Not doing so can negate the value found in data dictionaries and corrupt data standards.

Regarding physical design, it is important to realize that a well-constructed physical design contributes more to a database system's availability and performance than most people realize. In fact, a study done by Oracle Corporation several years ago found that nearly 36 percent of all downtime can be attributed to a database's faulty design. While DBAs work to monitor and tune a poorly performing database, most do not realize that when they monitor a database for performance, they are actually validating a physical model's implementation.

It's obvious that DB2 professionals need to practice formal logical and physical data modeling, however, such activities take expertise, time, and attention. Modeling and design are not functions to be carried out with pen and paper; instead, they should be facilitated through the use of strong modeling tools that assist in the proper creation of logical models as well as in the building of physical designs that target actual database engines. There is a vast array of features that data architects, modelers, and DBAs should consider when choosing a data modeling tool and such a complete list is beyond the scope of this document. However, a core set of must have features include:

- **Multi-level design environment** — the modeling environment should support both logical and physical modeling with intelligent mechanisms to sync either way.

- **Data dictionary support** — to support data standardization and data element reuse, data dictionary support should be present.
- **Reverse/Forward engineering** — the tool should be able to perform changes against an existing DB2 database through reverse engineering an existing design, as well as be able to build a completely new database through DDL generated by a physical model.
- **Documentation strengths** — to promote understanding and communication of logical and physical models, the modeling environment should be able to document designs in a variety of ways (HTML, Microsoft Word, etc.)
- **Team based collaborative abilities** — large models and designs are normally constructed through a team of architects or modelers, so the modeling environment should support the ability for team members to work on parts of a global model and then merge them back into a single, consolidated model.



Embarcadero's ER/Studio meets all the requirements listed above for a robust modeling solution plus offers much more. It's powerful combination of logical modeling and physical design capabilities make it the ultimate solution for serious DB2 data architects, modelers, and database administrators. Data architects and modelers will appreciate the complete support for graphical logical design and data dictionary support in ER/Studio, and DBAs will enjoy the near instantaneous transformation of a logical model to a physical DB2 design. Once a DBA or data architect has put the finishing touches on their DB2 design in ER/Studio, they are only a few mouse clicks away from generating all the DDL necessary to create a physical DB2 database along with all accompanying objects (tables, indexes, procedures, triggers, etc.) All along the way, ER/Studio can create full documentation sets for both logical and physical models in Microsoft Word format, as well a fully functional Intranet Web site that documents a design complete with the graphical model. And should the design effort require a group of modelers, ER/Studio's optional repository provides the mechanisms necessary for a team environment to operate with complete check in/check out and merge capabilities being present.

Once an architect or modeler builds a physical database through ER/Studio, the next step is to begin populating it with data. Oftentimes this consists of migrating data from an existing system to the newly created database. To assist with this process, models once again come to the rescue.

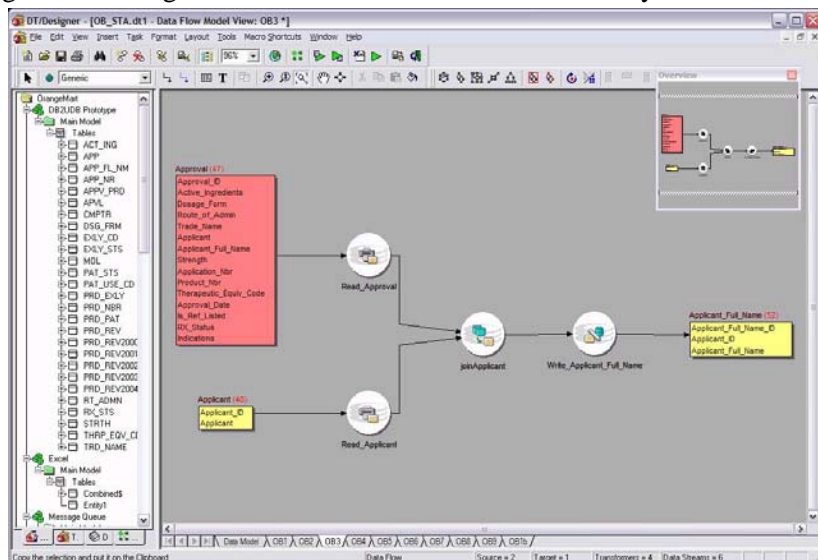
Moving Data with Models

Data movement and migration projects are rarely simple assignments. DB2 Developers, DBAs, and Extract, Transform, and Load (ETL) specialists oftentimes struggle to wrap their arms around complex migration efforts and when it comes to serious and ongoing data movement challenges, the need for more capabilities is often expressed. Large migration projects are especially difficult for the following reasons:

- **Data source and target understanding** — the need to deeply understand the source and target systems is oftentimes complicated by the sheer number of source or target destination objects (or both) and all their inter-relationships.
- **Large data volumes** — moving a few hundred megabytes is one thing, but moving half a terabyte is many times a different matter.
- **Complex data transformations** — it is rare that data is moved without modification from a source to a target database. More often than not, the data is sliced and diced in many different ways so it can become usable on the new target system.

To cope with complex and large DB2 data migration projects, DBAs, developers, and ETL specialists can turn to Embarcadero's DT/Studio. DT/Studio is a model-driven ETL solution that uses models to facilitate the understanding of source and target systems. The visual modeling interface of DT/Studio allows those involved in ETL efforts to represent their source and target systems in models found in data design tools like ER/Studio, which takes all the guesswork out of knowing how data is to be moved. Users of DT/Studio can visually design their dataflows between systems and not have to worry about the underlying complexities of how data is read or written, although they can control every aspect of such operations if they wish. DT/Studio supports all the major database engines as well as being able to deal with data that is housed in flat files, XML files, and more, so if another platform is involved in a data migration effort other than DB2, DT/Studio will have it covered.

DB2 professionals needing to move large amounts of data can do so confidently with DT/Studio because the solution is able to scale to meet the challenges of very large database installations. DT/Studio's data movement engine is written to run on many different operating system platforms (MS Windows, Linux, UNIX, etc.) and take advantage of multi-CPU machines so even large data movement tasks are completed in the time window desired.



When it comes to transforming DB2 data, it's hard to beat DT/Studio's extensible transformation framework. Common transformation tasks (horizontal, vertical partitioning, etc.) can be designed

through visual wizards. Detailed transformations can be carried out through the use of DT/Studio's over 1,000 built-in data transformers or through customized data alteration tasks that can be written by ETL specialists and saved for later use. Built-in scheduling abilities let ETL professionals create data migration jobs for recurring execution.

In addition, DT/Studio has a number of other power features. Migration tasks that are built in DT/Studio can be versioned, which allows rollbacks to previous migration tasks to be performed as well as comparisons between versions. Task reporting is also available, which makes the tracking of ETL operations from a business perspective clear and easy. These features and more leave no room for doubt that DT/Studio has what it takes for serious ETL projects to be successful.

For simple migration projects where data volume or complexity is not an issue, DBAs can use Embarcadero's DBArtisan Workbench or Change Manager products to move schema and/or data between like or disparate systems. For example, a DBA can easily move the schema from an existing Oracle system to a DB2 database and pull the data if desired as long as the volume is not excessive. For large or complicated data migration tasks, however, DT/Studio is the product of choice.

Effective Data Management

Once logical data structures have found a home in a physical database environment, the job of the database administrator and developer begin to come to life (although many DBAs and some developers are also involved in logical data modeling.) DB2 databases can be quite complex to manage, especially when they deal with large volumes of data, objects and partitions. Let's take a look at some of these difficulties and how they can be greatly reduced through Embarcadero's data lifecycle management solutions.

Streamlining DB2 Development

Many DB2 developers have been in a quandary over what development tool to use for serious DB2 work. Some have used simple text editors like vi and emacs, while others simply work within IBM supplied Command Center, and Control Center tools. These environments, while adequate for some tasks, often lack the punch that heavy-duty DB2 developers require.

What exactly is the short list for "must-have's" when it comes to DB2 development? While each developer will have his or her own specific needs, the following constitutes mandatory features that are necessary to carry out significant development work on DB2:

- **Rich SQL Programming IDE** — stating the obvious, a DB2 developer will need an environment that facilitates the rapid creation of SQL scripts, stored procedures, and other database code. The developer will need things like auto-code completion, formatting assistance, code generation, and graphical query building assistance. The ability to quickly extract and manipulate existing database code and dependent objects is also a must have.
- **Version Control Integration** — serious DB2 coding efforts should be linked to some sort of version control system so that, in the same way that application code is versioned, database code is versioned and protected.

- **Code Debugging, Optimization, and Profiling Support** — to locate coding mistakes, all developers must have the ability to debug stored database code objects. In addition, developers should have utilities that help them optimize slow running SQL code for better performance. Finally, developers need to be able to do response time analysis through profiling the execution of database code objects. Such ability allows a developer to understand why a procedure, or set of procedures, that used to execute in five minutes is now taking an hour to run.

Embarcadero Rapid SQL is a cross-platform development IDE that meets (and exceeds) all the criteria listed above. Its powerful SQL programming IDE supercedes anything found in IBM's command center utility. Developers will enjoy powerful code generation abilities, many coding aids (cross-database search utility, visual object creation wizards, data imports from Excel spreadsheets, etc.), along with built-in version/source code control integration with all the popular source code software vendors.

On the code optimization front, developers will find a strong SQL-PL code debugger, response time benchmarking aids, and a powerful stored code profiler (Code Analyst) that can be used to troubleshoot performance issues in stored procedures, triggers, and the like. With Code Analyst, a developer can quickly analyze where the time is being spent in executing database code. Developers can also benchmark stored procedures to obtain baselines and then use those historical baselines to compare to current code response times. When a developer receives a tap on the shoulder and is told that a stored procedure that used to take five minutes to run is now taking over an hour, they can quickly use Code Analyst to tell them the exact place(s) in the code that are causing the bottleneck.

Upgrading Administration Abilities

Most DB2 DBAs have traditionally relied on home-brewed DB2 command scripts to handle all their administration needs. While these scripts and IBM's UI tools are effective in handling basic administration tasks, today's DB2 DBA needs more when it comes to managing a database farm that's growing both in terms of size of database numbers (and partitions if you consider DPF environments). Administrators are increasingly looking for help with management issues like the following:

- **Proactive storage management** — DBAs need help understanding the total storage picture across their DB2 database and need assistance in identifying things like transaction logs that are nearing capacity and databases that need to be reorganized.
- **Capacity planning** — Administrators are in desperate need to know what databases are exhibiting the fastest growth, when they will need to buy more storage, and whether enough memory exists on their machines to serve an increasing user workload.
- **Enterprise job scheduling** — the DB2 task center is useful to DBAs who only have to schedule and manage jobs on a few DB2 servers, but administrators are increasingly asking for a way to centrally manage and run jobs across all their DB2 and other DBMS'.

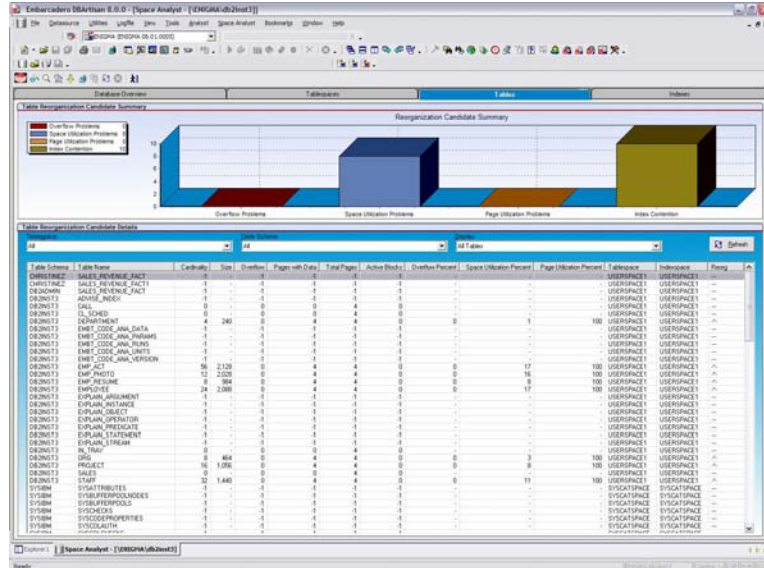
Embarcadero has recognized the need for more industrial strength administration tools and provides DBArtisan Workbench to DB2 DBAs who need more than what IBM's Control Center offers. DBArtisan Workbench offers an exceptional interface to accomplish all the tasks that IBM control Center does, and it picks up where IBM leaves off when it comes to managing

heavy-duty DB2 environments.

For example, on the storage management front, DBArtisan Workbench naturally can handle the creation of database partition groups (or node groups), and tablespaces.

It also contains global views of the complete DB2 storage picture plus deep drill-downs into fragmentation and other issues that threaten a DB2's availability and performance.

In addition, DBArtisan Workbench contains powerful space management wizards that can create smart reorganization jobs that not only reorganize database objects when needed (based on the DBA's customized thresholds) but also update stats and automatically rebind newly reorganized objects. A DBA can easily schedule these smart reorg jobs and put their space management program on autopilot.



DBArtisan Workbench also offers strong capacity planning capabilities to DBAs who need to proactively plan for their data center. DBArtisan Workbench provides visual capacity wizards that allow a DBA to track things like database and object growth. All information is collected in a central repository for historical analysis. A DBA can plot and forecast future growth/usage in a graphical way or through HTML reports, which can be created in ad-hoc fashion or scheduled to run on a recurring basis.

For DBAs who need to manage jobs across multiple servers, DBArtisan and Embarcadero Job Scheduler form the perfect combination to handle enterprise-wide scheduling needs. DBAs can schedule any SQL script or nearly any task in DBArtisan through integration with Job Scheduler. Job Scheduler's central console allows a DBA to view all jobs on all servers throughout the enterprise, or filter on only the jobs they want to manage. Of course, DBArtisan Workbench supports scheduling through cron, Windows task scheduler, etc., but for DBAs who need central control over all their DB2 jobs (as well as any other jobs database jobs), Job Scheduler has everything they require.

The Need for Proper Change Management

A DBA in a large IT shop manually made a rather complex change to a table in a production PeopleSoft database. She believed everything had gone well, however, she unknowingly lost several key indexes on the table she was changing as well as numerous security permissions in the

process. The next day, her phone was ringing with angry users complaining of slow performance and security errors. Because of the complex nature of the PeopleSoft database (thousands of tables, indexes, etc.), it took the DBA nearly all day to track down what had happened and make things right again. Such situations occur every day in data centers to DBAs who do not institute smart change management practices. Change management protects the DBA in ways that traditional backup and recovery plans do not. The basics of change management for DB2 revolve around the following four activities:

- **Database Archiving** — this is not to be confused with data archiving where lesser-used data is moved to other databases/servers so performance on the primary server is increased. Instead, this concerns taking ‘snapshots’ of database, security, and data/code object definitions to preserve what the database looked like at a particular point in time.
- **Database Comparison** — every DB2 DBA has asked or been asked the question “What Changed?” countless times. Change management allows the DBA to intelligently answer this question through the use of smart comparisons between live DB2 databases or archived definitions of DB2 datasources.
- **Database Migration** — copying or cloning all or parts of a DB2 database can be challenging given certain environments. Change management provides ways for intelligent full or partial copies of databases to be carried out without worry.
- **Database Synchronization** — synchronization is generally performed for one of two reasons: (1) when unwanted definition, configuration, or security changes occur to a database, a DBA will want to “roll back” to a particular point in time or (2) a DBA applies definition changes to one database that they want propagated to other databases.

To meet the change management needs of DB2 DBAs, Embarcadero offers Change Manager. Change Manager is a cross-platform change management tool that can perform all four functions listed above. Visual wizards guide the DB2 DBA through every task with ease, and built-in scheduling functions allow any task to be completely automated. Change Manager also offers notification capabilities so if any change occurs to a database without the DBAs knowledge, they can be notified of that fact.

Change Manager would have been able to quickly rescue the DBA discussed above who accidentally damaged her PeopleSoft database. If she had Change Manager, she would have been able to build scheduled archives of her database so it was completely and automatically protected. When the accidental loss of key indexes and security privileges occurred, the DBA could have invoked Change Manager’s powerful comparison facility, which would have compared her archived database copy with the live database and then pinpointed the missing indexes and security privileges. Finally, clicking one button in Change Manager to synchronize the live database with the missing indexes and security privileges would have easily restored everything. What took her nearly an entire day would have been reduced to minutes with Change Manager. Without question, Change Manager provides the insurance that no DB2 DBA should be without.

Guaranteeing High Performance

Nearly every database vendor is talking about how “self-managing” their database engine is these days. Not surprisingly, such claims are directed squarely at corporate management who have long since regarded databases as complicated black boxes that require management by trained and expensive personnel. The new message being sent to IT managers is that all complexity has been removed from the database and that it, like a new network card being introduced into a PC, has become plug-and-play.

DB2 v7-v8 introduced a new SMART (Self Managing and Resource Tuning) initiatives for administrative and performance characteristics of its database engine. But does this really work? Does DB2 handle performance in a way that requires no DBA tuning or intervention? Not really. Most of the smart performance management software requires you to specify performance thresholds and the actions that need to be taken once the thresholds are tripped. So the SMART solution in most cases is smart really only when you have sent correct thresholds and corrective actions. Even when the database engine can determine the thresholds and take self corrective actions, these are usually error prone since the only examine and correct a very small subset of all performance related parameters.

Clearly, performance analysis is still required for DB2 (and every other database engine). That being the case, how does one best approach the practical application of performance analysis? What techniques are needed to quickly find and remove performance inefficiencies that exist in a DB2 database, while keeping the server up and running at all times? A smart administrator can protect themselves and their critical databases in three ways:

1. **Institute performance testing** — rather than guess whether a Web-based DB2 system can handle one thousand active user connections and meet the need for ‘n’ number of transactions per second, it’s much better to actually simulate such a scenario beforehand and proceed confidently to Day One of production knowing a system will perform as expected.
2. **Establish unattended monitoring** — a DBA cannot personally watch hundreds of DB2 databases to catch performance problems, nor can they be present around the clock in the data center. Therefore, it’s imperative that a DBA set up a 24 x 7 unattended monitoring system to check key systems and proactively notify the proper personnel before downtime or system problems are experienced.
3. **Have a tactical performance methodology** — no matter how well a DBA tests or polices their systems with unattended monitoring, the phone will still always ring with users complaining of ‘slow’ performance or other problems. An administrator needs a SWAT-style plan that they can use to quickly interrogate a database to find bottlenecks and remedy the situation as quickly as possible.

Let’s explore each of these areas a little further to see how they can be practically implemented in a DB2 environment.

Testing for Performance Success

Proper performance testing catches the showstopper and performance vampire problems that inadequate user and quality assurance testing miss. In a nutshell, performance testing simulates what is expected from real world use. It stresses DB2 in ways that could otherwise only be accomplished by opening the floodgates of the production user community.

Smart performance testing uses the following elements to pull off a realistic simulation of what a database will experience during expected production usage:

- **Anticipated user presence** — it is critical that the test simulate the number of user connections that are expected during peak times and normal working hours. This is the major area where manual methods that pick a subset of users to test a database and application fail. The database server may run just fine with 15 or so user connections, but may fall over when 300 connect to the system.
- **Repetitive user activity** — once the anticipated user sessions have connected to the database, they obviously have to “do something” for the system to be stressed. And they cannot just “do something” once. Either all or a portion of the connected sessions need to repetitively perform tasks as they would occur during a normal workday. For an OLTP system, this may mean entering multiple orders. For a data warehouse, this may mean issuing long running analytical queries. The key is that the work is repetitive so repeated blows are dealt against the database.
- **Extended duration** — once a DBA sets a number of sessions performing repetitive work, they next need to ensure that the work continues for a period of time that makes the test meaningful. What a DBA is looking for is to unearth problems that take time to develop. For example, a DB2 table may not become fragmented after 30 minutes of OLTP work, but may surprisingly fragment in a dramatic fashion after 2 or more hours of repeated action.

So how is this practically accomplished? How can performance testing address both the hit-the-wall problems along with the silent killers of database performance? How should a DBA arrange tests to flush out all the major headaches that threaten a database server? Even database professionals who understand the need for real world testing of their database and know what areas need to be stressed are oftentimes at a loss for practically creating the scenarios they need for pulling off a true performance test. This is not surprising when one considers all the manual work that has to go into such an effort for it to yield the answers that will give the Go/No go signal to a database project team. It’s oftentimes much easier to perform a small test with a selected group of willing users and then cross your fingers on the first day of production or when a massive upgrade begins.

This does not have to be the case. Embarcadero’s performance testing tool, Extreme Test, is designed to bring peace of mind to a database staff by allowing them to know what areas of a DB2 and accompanying application/Web system will break prior to it going into production or experiencing a large upgrade. By simulating production or accelerated user and data volume, Extreme Test can make a database server unbreakable by letting a staff understand the weaknesses of a database server before the end users find them, with the end result being ironclad availability and performance for production systems.

Extreme Test allows a DBA to capture and simulate the activity of hundreds or even thousands of users, and then apply the resulting workload in varying amounts against a target database. While generating database load, Extreme Test monitors the underlying database components to identify the ones that are struggling to handle the stress. It captures and stores test results in a central repository that a DBA can query for further analysis and reporting. Extreme Test makes it exceptionally easy for an administrator and QA team to implement repeatable testing, to isolate bottlenecks, and to measure the impact of corrective actions on overall database performance. And for those wishing to test system components other than a database (application server, Web server, etc.), Extreme Test can build a single performance test against all defined components to produce a real world picture for an entire system.

So to ensure a DBA does not have an unwelcome surprise waiting for them in a database server, they need to apply proper performance testing to a system to uncover all problem situations before they impact a business. The best way to do this is to use intelligent and powerful testing solutions like Extreme Test that can provide the answers needed before a database server sees the first day of production or before a large upgrade occurs. Nothing serves as a better confidence builder than knowing what to expect.

Exceeding Performance Expectations

The goals of a successful DB2 performance management plan include the following:

- **Stop unwanted downtime** — naturally, key database servers must be kept available for use by the business systems that need access to data.
- **Reduce time to detect and resolve performance problems** — some businesses have the potential to lose tens of thousands of dollars per hour when a database is down or in trouble. Therefore it is critically important that a staff be able to restore a database to peak performance in the shortest amount of time possible.
- **Proactively remove threats to key databases** — needless to say, it is much better to detect that a DB2 transaction log is 70 percent full (and take action) than when it is 100 percent full and denying transactions from completing.
- **Increase staff knowledge and productivity** — database opinion polls state that a DBA spends anywhere from 27-44 percent of their time monitoring and troubleshooting database systems. Data center management need their staff to be fully educated and up-to-speed on the best database troubleshooting techniques so they can quickly resolve performance issues and move on to more strategic tasks.

To help troubleshoot and optimize the performance of today's complex DB2 systems, database professionals either build homegrown scripts or turn to software that is designed to monitor the activity of a busy system. The goal of a database monitor is to smartly present statistical data in a meaningful way so that the database administrator or developer can confirm acceptable availability and performance, or recognize a potential threat and take action. The choice as to which database monitor to use can be a complicated one as there are many third party solutions available. To make the correct choice, a database professional must understand both the needs of each database that requires monitoring as well as their own role in the performance optimization process.

There are at least three different categories of database monitors in use today, and DBAs will at times find themselves using one or all three depending on the monitoring need:

- **The cursory real-time monitor** — this monitor checks database server session activity, provides basic contention information, offers a general view of cumulative global performance measures, and presents basic SQL execution metrics.
- **The intelligent client-side monitor** — this monitor offers a better organized view of performance using popular performance methodologies, assists in locating and correcting detailed performance issues, provides several layers of drill down statistical views, and offers customizable options and thresholding abilities.
- **The enterprise 24 x 7 monitor** — this monitor provides constant, unattended monitoring of assigned databases, scales well for large database farms, supplies proactive notification and auto-corrective mechanisms to minimize downtime, and provides historical analysis of performance metrics.

Smart use of each type of monitor ensures complete protection for an enterprise-wide DB2 installation. Embarcadero Technologies offers each type of monitor listed above (24 x 7 monitor is under development and is expected around Q4 2004), where each is integrated with one another so a DBA can seamlessly switch from one monitor to another when the need arises.

DBArtisan Workbench serves as the main console for all three of Embarcadero's database monitors. For fast "Who's logged on and what are they doing?" type monitoring, the DBArtisan process monitor can quickly be invoked and checked for session activity and global performance metrics. For more detailed real-time monitoring, a DB2 DBA can drill down into a server by using DBArtisan Workbench's Performance Analyst monitor. Performance Analyst utilizes all the popular performance methodologies (ratio analysis, bottleneck/wait-based analysis, workload/SQL analysis) to provide the DBA with the most complete view of DB2 performance available. Powerful drill-downs into memory, I/O, space, sessions, and more lead the DBA to the exact problems plaguing a database. And, in addition to DB2 statistics, a DBA can also monitor the operating system DB2 is running on (AIX, Windows, Linux, HP-UX, Solaris) with Performance Analyst.

Finding problems is one thing, but fixing them is oftentimes a different story. Many problems that are found in the Embarcadero monitors can easily be fixed through integration with DBArtisan Workbench. For example, if Performance Analyst is showing a shortage of space in a DB2 tablespace, a DBA can highlight the tablespace and increase its size through DBArtisan Workbench editors. Everything can be identified and fixed in one place.

Using the combination of DBArtisan Workbench and Performance Analyst, DBAs can ensure that all their bases are covered with respect to proactive database and system performance monitoring. The end results are reduced downtime, faster problem diagnosis and resolution, the proactive removal of threats to DB2 performance, and increased staff productivity and knowledge.

Conclusion

Significant growth of database data volumes and the number of databases that a DB2 professional needs to use have brought growing pains to those who previously only used much smaller and fewer DB2 databases. The increased presence and scale has brought challenges to data

professionals who are charged with designing, migrating, managing, and protecting DB2 data assets.

Embarcadero Technologies offers the most complete line of serious DB2 software tools available, which extend beyond the free data management aids provided by IBM. The entire DB2 data lifecycle is covered by Embarcadero, so whether a data professional is designing a new database, creating complex data migrations, developing complex SQL-PL code, or managing the performance and availability of hundreds of DB2 applications, they can feel confident that tools from Embarcadero Technologies will save them both time and money, with the overall end result being the success of their critical IBM s

Embarcadero, the Embarcadero Technologies logos and all other Embarcadero Technologies product or service names are trademarks of Embarcadero Technologies, Inc. All other trademarks are property of their respective owners.