

SOLITAIRE

INTERGLOBAL

Whitepaper: Comparing Real World Database Performance: IBM® DB2® versus Oracle® Database and Microsoft SQL Server®

Introduction

One of the most difficult architectural component decisions in today's information management environment is that which addresses the choice of database management technology. By far the largest market share in this type of software is held by Oracle Database, DB2 and Microsoft SQL Server products. In an April 2011 release of an ongoing study by the Solitaire Interglobal Ltd. (SIL) research team, the production behaviors of Oracle Database and DB2 on the IBM Power Systems and similar behaviors of Oracle Database, SQL Server and DB2 on the IBM System x platforms were analyzed and compared. This release is a scheduled part of the continuing Operational Characterization Master Study (OPMS) that has been conducted over the last 23 years by the SIL research team. OPMS is used in support of industry standards and performance certification worldwide.

During this study, the main behavioral characteristics of both hardware and DBMS were examined closely. These traits affect the overt capacity and reliability of the combined architecture. The resultant behavior has to be examined within this conceptual framework to be clearly understood and to prepare that understanding for projection to other situations.

Although the raw performance of a subject system is an important metric, the translation of that performance into business terms is more germane to today's market. In this venue, the issue of relative costs can be a more significant metric than base performance. This measure encompasses a myriad of other factors, including reliability, staffing levels, vendor service responsiveness and time-to-market effects.

In addition to raw performance and business metrics, the accelerating pace of business in today's marketplace means that certain strategic metrics are also becoming important. In this arena, metrics like technology leadership days, research patent activity and other measures of thought leadership provide some insight into the strength of an organization's position as an ally in the quest of organizational growth and viability.

Study Scope

In order to understand the impact of the synergy of DB2 on the Power Systems and System x, a large number of database deployments were examined for this study. These deployments included situations where the DBMS implemented was DB2, as well as competitive offerings. The relative degree of difference in operating behavior for each implementation was then compared to understand the net affect of the respective combinations. The effects were observed in general performance and capacity consumption, as well as reliability and customer satisfaction.

This study examined several congruent areas of activity for the DBMSs. The performance evaluation looked at the production behavior of installed systems, including costs, reliability, throughput and other objective measurements. These systems all captured specific metrics for SIL-identified tests for the performance side. A congruent area of investigation also captured metrics on thought leadership and research metrics, to contribute to a clearer understanding of the value of strategic vendor partnerships.

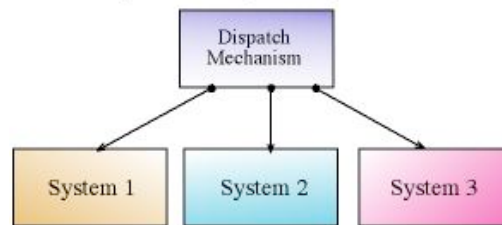
All of the evaluation systems included platforms from IBM and other vendors. This broad scope was used in order to crystallize an understanding of the impact of DBMS and platform synergy. Once the domain was collected, the relative degree of difference in performance on each platform set was compared to provide the net affect of the respective combinations. Due to the comprehensive nature of the desired analysis, over 36,600 systems were used for input to this study. All of the systems were active within the SIL heuristic database, although some of the organizations reported only partial data. Where performance data was reported, the performance comparison options were run in one of two general topologies – side-by-side and correlated. The side-by-side process methodology allows the system options to have separately dispatched activities. In this architecture, none of the executing platforms is dependant on the execution or completion of any of the others. Due to this lack of dependency, no allowance was made for serial latency in the analyzed timings.

The second architecture (referred to as “correlated” in this document) identifies one executing system as the parent or original operation. Each transaction or execution is first sent to that system. Then, depending on the type of spawning mechanism, the same request can be sent on immediately to the next system, and the next and so on. In the test cases, only the DSS system used the correlated form of the parallel testing mechanism. Therefore, the DSS performance was adjusted to eliminate the dependency figures and latencies.

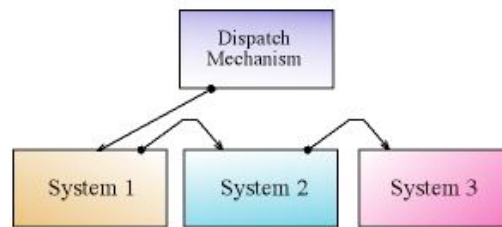
The following diagram illustrates the difference in execution architecture between the two methods.

Dispatching Architectures

Side-by-Side Dispatching Mechanism



Correlated Dispatching Mechanism



The details of the different test comparisons were intriguing and provide a good insight into the relative efficiencies of the possible system deployments.

Methodology

The approach taken by SIL uses a compilation and correlation of operational production behavior, using real systems and real business activities. For the purposes of this investigation, over 9,200 closely watched production environments were observed, recorded and analyzed to substantiate the findings, and supplemented by additional data from a total of over 36,600 systems. Some of these systems were used for partial comparison, providing key metrics, but lacking detailed, daily tracking. Nine thousand two hundred fourteen of the systems were closely watched. This means that SIL receives data showing daily granularity for these systems. Overall, over 3,467,000 individual data points were used from the 36,600+ systems to build the analysis.

Using a large mass of experiential data, a more accurate understanding of real-world behavior can be achieved. The data from these production systems was used to construct a meaningful perspective on current operational challenges and benefits. The reported behavior of the systems was analyzed to isolate characteristics of that software architecture from both a raw performance and a net business effect perspective. All input was restricted to those organizations operating DBMS-centric systems in versions that were current in either 2010 or 2011. Since many of the components for DBMS implementations have releases at staggered points in time, only those application components that were either the current version or a -1 version based on those calendar restrictions were included in the study. Additional information on the methodology and study diversity can be found in additional methodology notes at the end of this document.

The SIL research activity focuses on the capture of actual customer production information, rather than the construction and observation of benchmark tests. One

of the main reasons for this strategy is that the capture of real world data represents a far greater variation of behaviors than any constructed test.

DBMS Characteristics

The basic memory and I/O handling methods are considerably different among the DBMS products, with unique patterns of memory usage and reclamation significantly affecting the required memory levels, refresh patterns and overall workload. The treatment of I/O is also substantially different, with radically different workloads associated with each I/O methodology. Each area of differentiation can be clearly seen in the details of the comparative tests. While each test is unique, a substantive trend can be seen in the distinctive handling of memory and I/O for each DBMS.

Hardware Characteristics

The IBM equipment has definitive characteristics of its own to contribute to the overall behavioral mixture. These traits are actually a combination of the operating system and the physical hardware that the operating system addresses. While looking at the Power Systems running AIX, the salient characteristics are:

- Width of the I/O channels
- High level of start I/Os that the AIX operating system supports
- Efficiency of the memory-handling algorithms
- Operating system efficiencies
- Network port addressing efficiencies

The complex synergies that occur with different architectural configurations mean that the same hardware running a different operating system will demonstrate significant differences in performance. These characteristics of the hardware and operating system provide an interesting amalgamation with DBMS activity.

The I/O trait is a dual-dimensional factor. The first dimension is the number of bytes that can be sent along the necessary platform pathways, while the second dimension defines the sheer number of I/O activities that can be initiated in a single second on one platform processor. The conduit restriction of the I/O channels is a function of both of these factors.

A similar synergy can be seen in the combination of the System x platform and the DBMS product, although the differences from other Intel-based servers are not as pronounced. Here the contributing hardware characteristics are the breadth of the I/O channel on the System x and the efficiency of the memory algorithms.

Power Systems Comparisons

The projects (and their method of comparison) that were examined for the Power Systems platforms are:

- Credit card processes – side-by-side
- Customer support (CRMS) activities - side-by-side
- Transactions – side-by-side
- Decision Support System (DSS) – correlated

Many of the systems were run either in full or partial parallel for comparison purposes, while operating in full production. All of the systems tested used the Power System architecture. The number of separate organizations (referred to as “core” systems), along with the number of instances that were compared at those locations is shown in the following table.

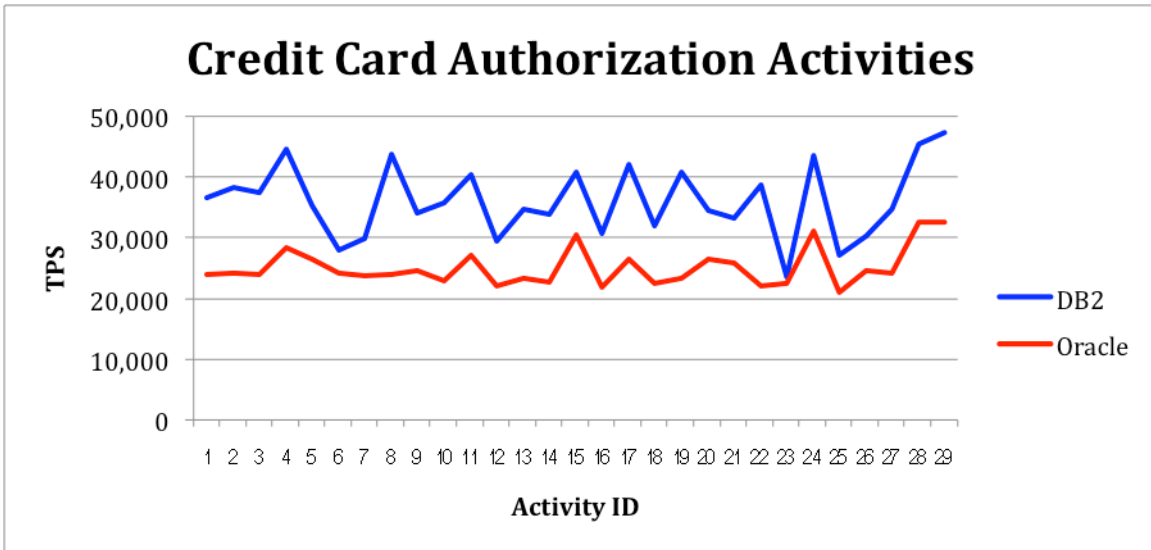
Study Structure – Power Systems Comparison Components

System Type	Core Systems	Comparison Option Count
Credit Card	82	113
CRMS	54	126
Transaction	242	687
DSS	103	504
Total	481	1,430

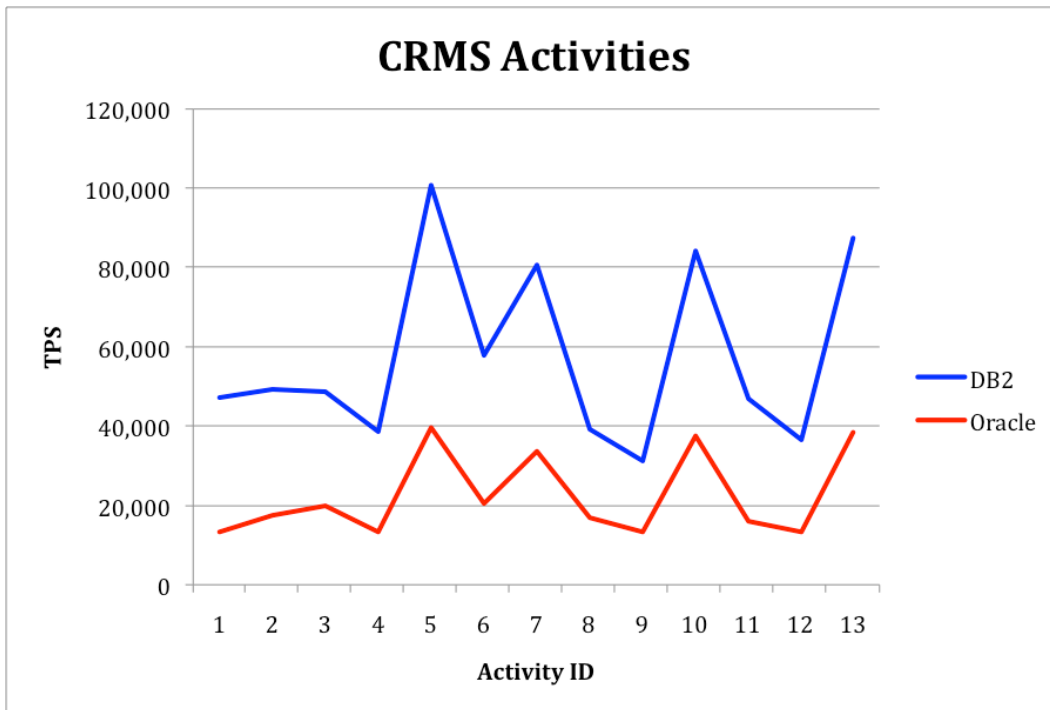
Note that the comparison figure equates to application deployments across the listed number of core systems, with more than one server per core system as standard. In many situations, there are application servers, database servers and web servers deployed for a single core system. Therefore, the total UNIX®/AIX comparison environment was comprised of 1,430 separate comparison options of 481 core systems. The individual projects were comprised of normal business activity in the subject areas. The activity was tracked by the contributing business action, including the number of expected rows per query, etc. A short list of the individual actions can be seen in Appendix A for each general project type.

Performance

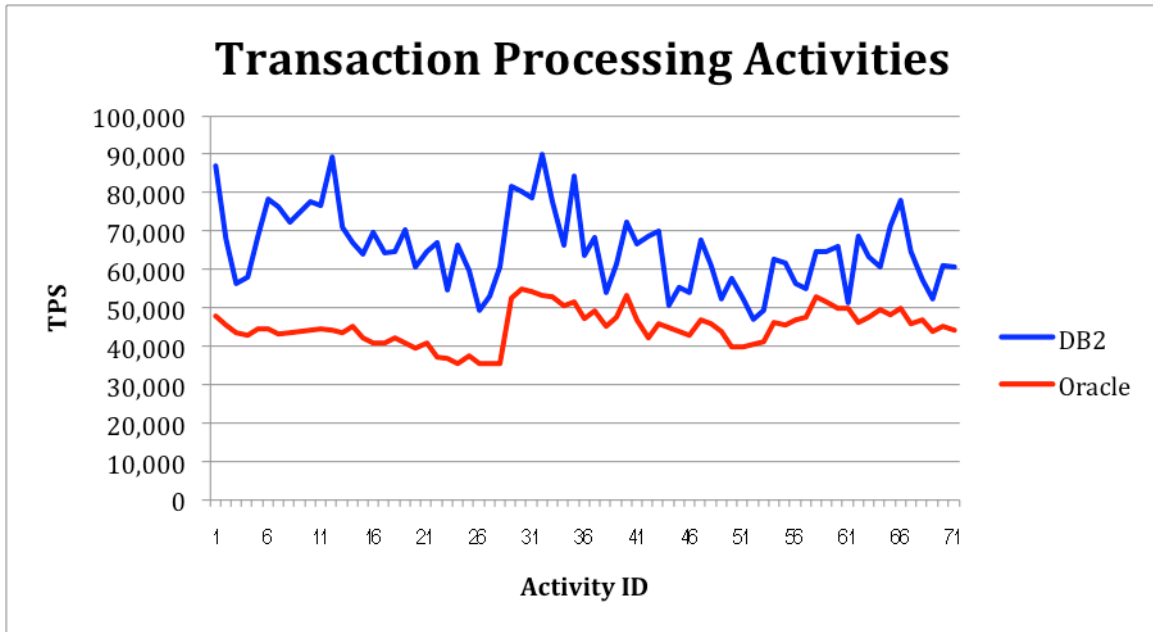
The first area of examination is the base performance of the relative systems. A detailed comparison of the client tests, showing Oracle Database and DB2 performance characteristics are represented by the system-specific graphs.



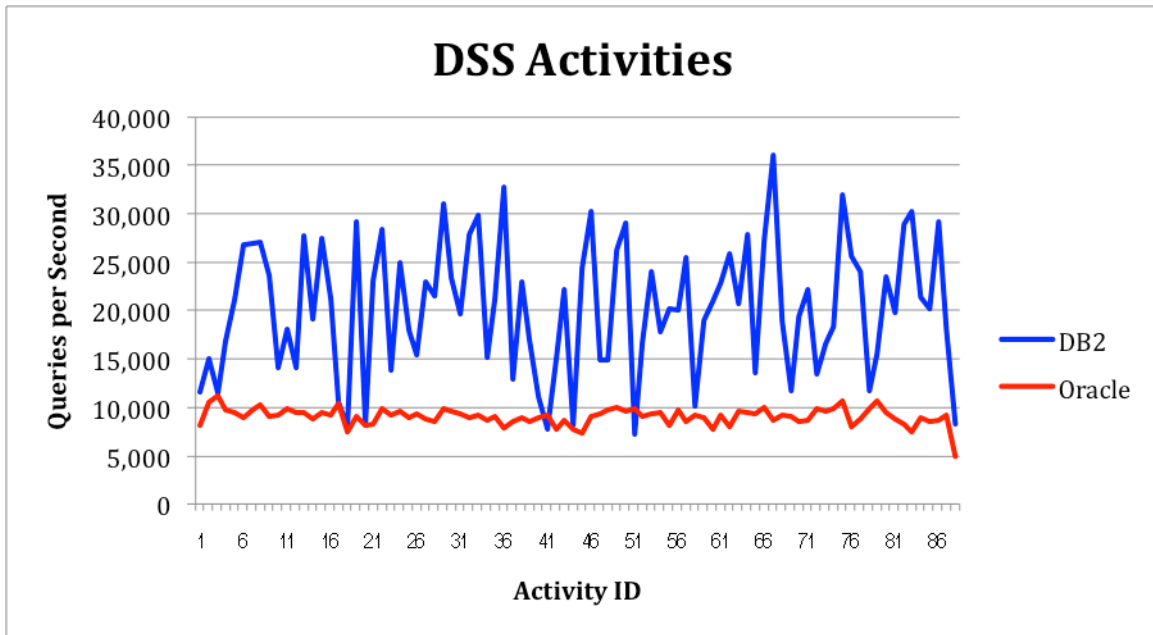
Overall, the credit card authorization activities show a clear throughput advantage for DB2. The primary sources of this efficiency have been reported as the overall I/O and memory handling and are present across the large majority of the reported experience.



CRMS activity tends to be more complex than credit card processing, and as such displays more variation in the reported performance behavior. It is notable that the more complex and larger transactions show an increased differentiation for the DB2 implementations.

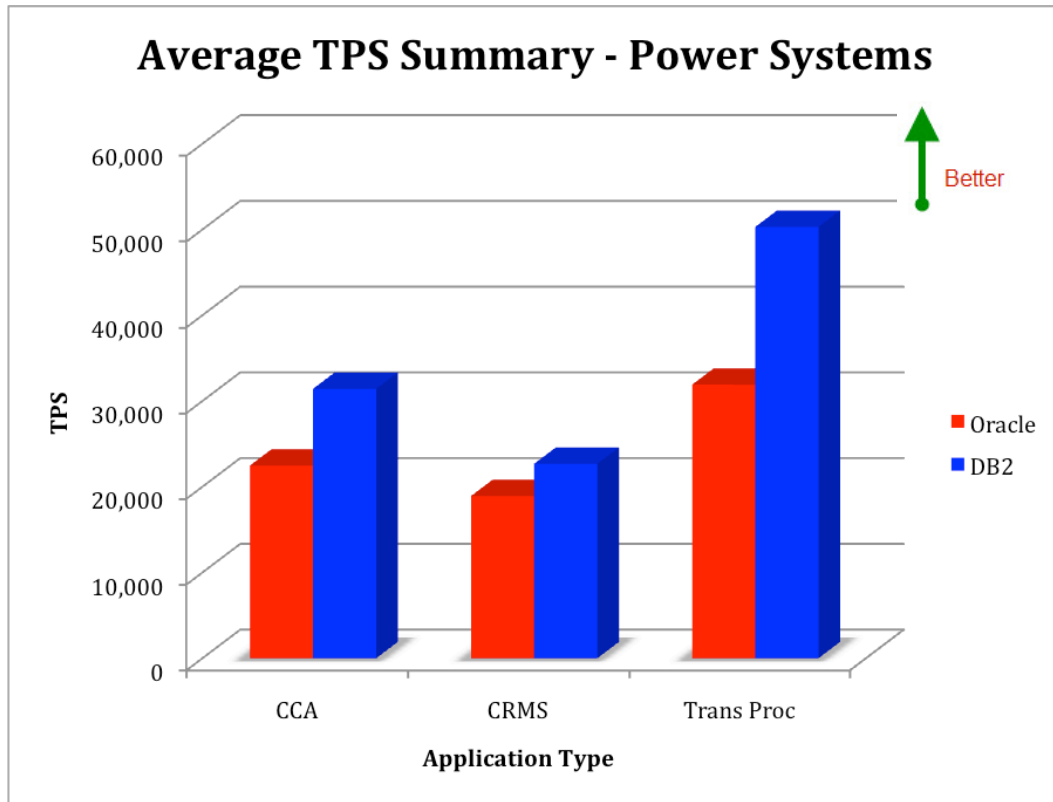


The transaction processing activities encompass an extremely large range of activities. It is obvious where the differences in transaction latency, memory handling and I/O are more significant, since the DB2 performance is notably better than the other DBMS experience.



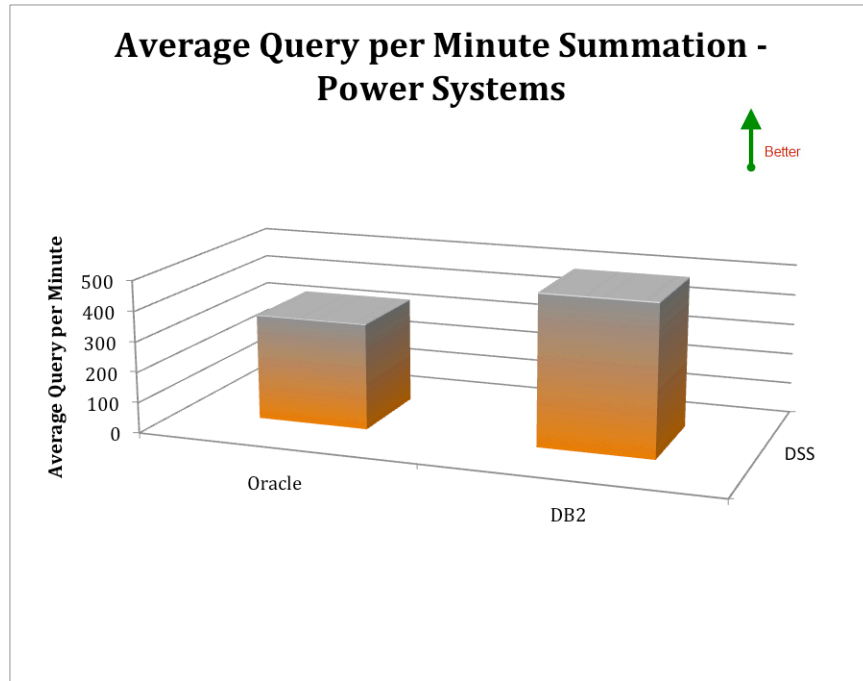
The operational characteristics and efficiencies of the DB2 and Oracle Database comparisons show that the DB2 DBMS demonstrates a good range of differentiation, when hosted on IBM platforms. The amount of divergence varies widely, based on application, platform, etc. and cannot be closely correlated for a clear and single answer to which DBMS is most efficient on those platforms. When DB2 is hosted on IBM Power equipment, the range of improved performance is heavily clustered. This

provides a more sharply defined demarcation to answering the DBMS uncertainty. The summation of the differences in performance can be seen on the following chart.



The effects of the increased throughput, more efficient memory and other optimizations within DB2 are reflected in the TPS summary. The more complex the activity results in a larger differentiation between DB2 and the other DBMS options.

The DSS comparison does not lend itself to the transaction per second metric, and so its comparison was done on a different basis and is summarized in the chart below.



Once again, DB2 production data shows an increased throughput and efficiency.

The improved efficiency of the DB2 product on the Power architecture is significantly higher than on the overall equipment population domain, ranging from 19.9% through 57.7% on the transaction-based metrics and averaging 42.2% better on the query-based metric. Even more significant is the small mean variance for each of the application type comparisons. The data values representing DBMS execution on all platform types varied by approximately 31.4%. For the Power system and DB2 combination, variance was less than 5.1%. This is significant in that it increases the predictability of performance behavior. Since the dependability of the projections is indicated by how closely they are clustered among the observations, this tight correlation is important.

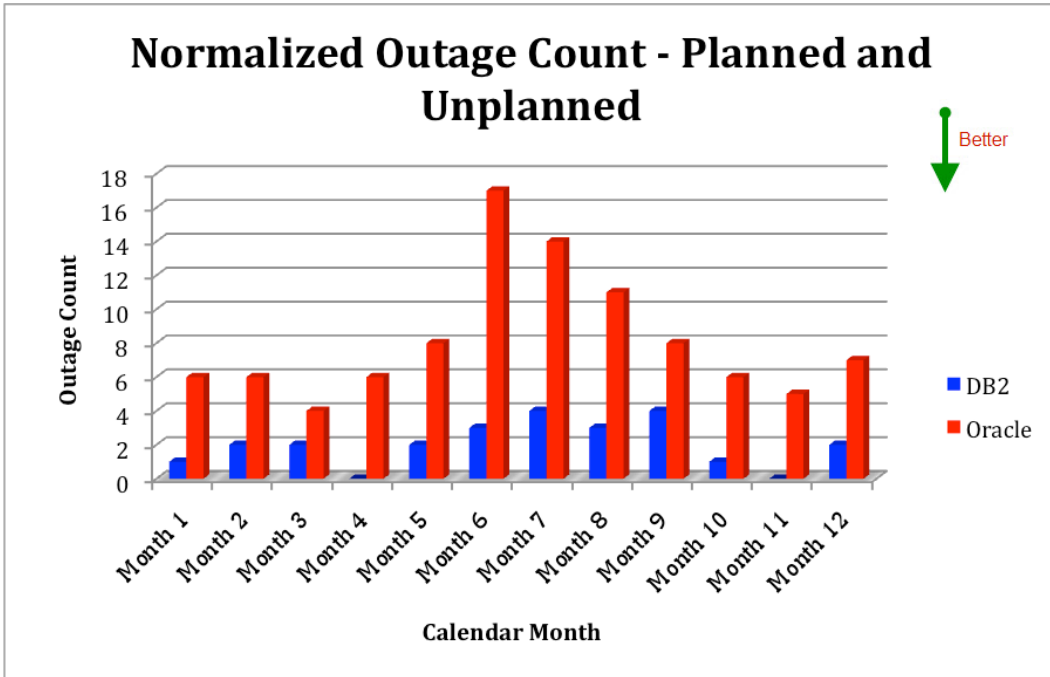
Business Measures

Many businesses consider the total cost and risk of ownership to be key metrics when evaluating purchase options. Some of the contributory metrics that go into building this picture are reliability, staffing, total costs, overall resource utilization and time-to-market effects. These factors were measured during the same time period that the performance metrics were gathered for all of the subject tests. A summary of these findings is presented here, so that a more comprehensive understanding of the real benefits of the platform choice can be obtained.

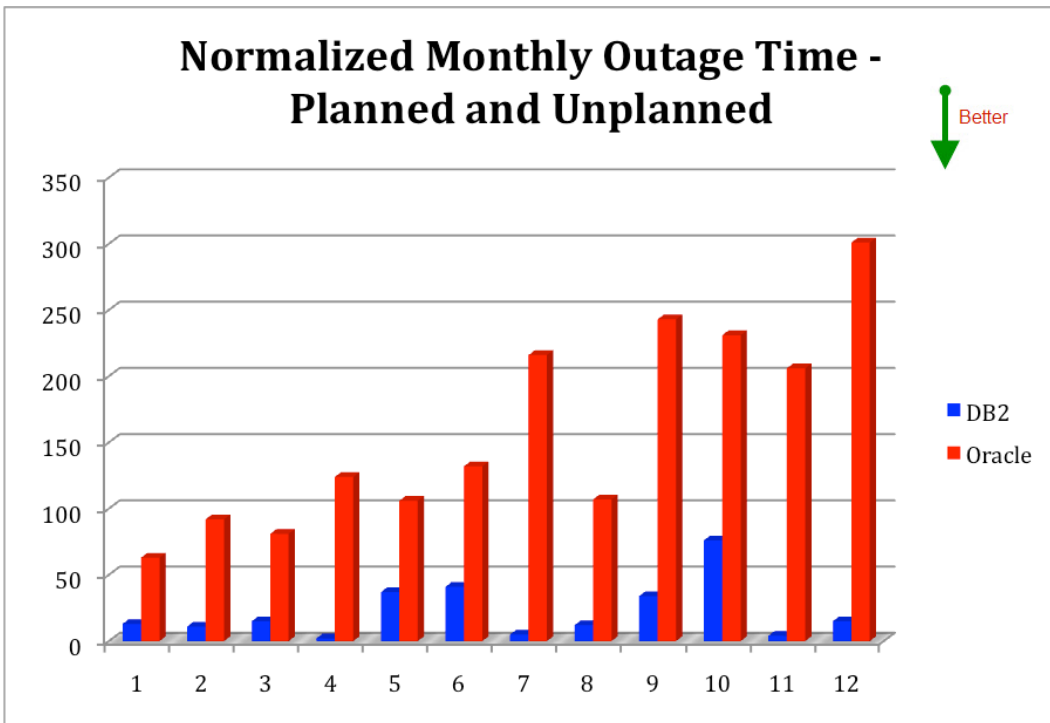
Reliability

The dependability of the implementation is a combination of the individual reliability of each component, along with the quality and effectiveness of the actual implementation. As such, both the planned and unplanned outages affect the overall usability of the total system. The charts below show the number of outages that were

recorded during the testing period, as well as, the total number of minutes that those outages consumed.



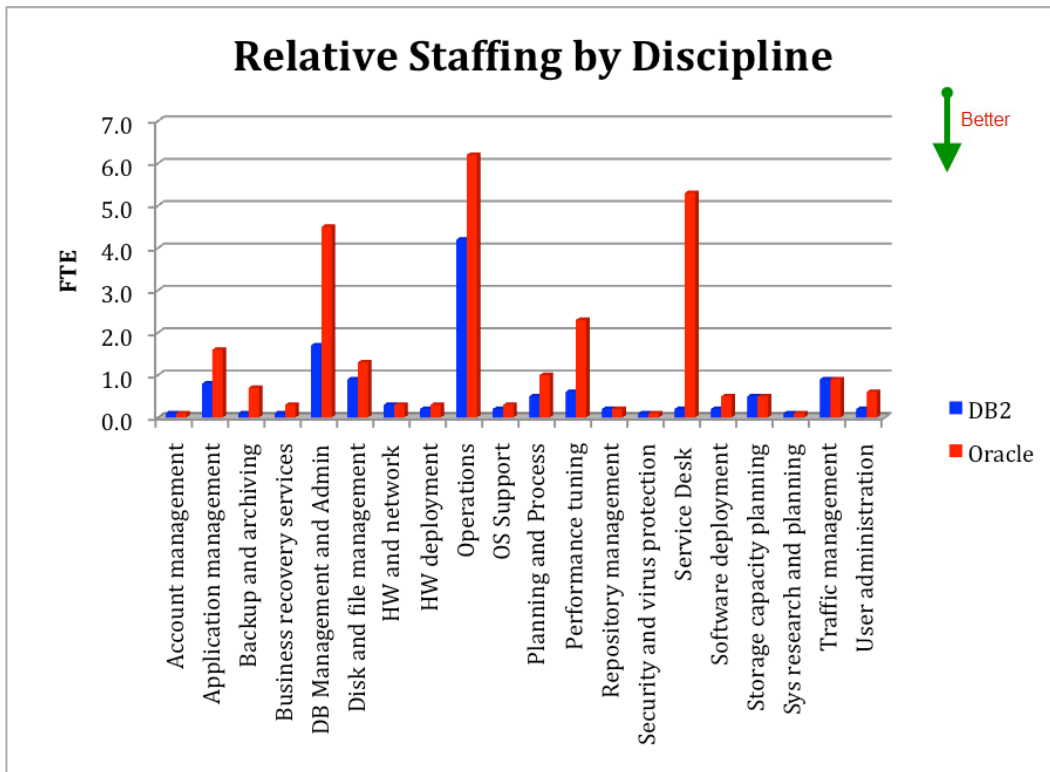
The number of outages has been normalized for a 100-platform operation, with both planned and unplanned outages included. One of the most contributory factors is the need for reorganization and redeployment. Each of these outages takes valuable access time away from corporate resources. The following chart shows the total number of minutes of unavailability that those outages represent.



When the root causes of reliability are examined, the biggest reported differences can be found in the amount and count of planned downtime occurrences. DB2 was heavily favored in this regard, with customers consistently reporting easier movement and allocation of resources without any interruption of availability. The secondary contribution to the reliability was the need for fewer upgrades.

Staffing

The number and skill category of the staff required to support the various implementations is another way of looking at the business issues for a deployment. The chart below shows the relative staffing for a normalized operations group. This comparison can be used to understand the challenges of staffing and overall budget costs for the DBMS implementations on IBM equipment. These staffing figures were collected from the actual operation groups measured in the other metric collection efforts, and cover organizations in North and South America, Asia, Europe, Antarctica and Australia. These organizations have reported staffing for 24x7 coverage, rather than single shift.



The source chart that was used to construct this graph has also been included.

Staff Discipline	DB2	Oracle Database
Account management	0.1	0.1
Application management	0.8	1.6
Backup and archiving	0.1	0.7
Business recovery services	0.1	0.3
Database management and administration	1.7	4.5

Staff Discipline	DB2	Oracle Database
Disk and file management	0.9	1.3
HW and network configuration / re-configuration	0.3	0.3
HW deployment	0.2	0.3
Operations	4.2	6.2
OS support	0.2	0.3
Planning and process management	0.5	1.0
Performance tuning	0.6	2.3
Repository management	0.2	0.2
Security and virus protection	0.1	0.1
Service desk	0.2	5.3
Software deployment	0.2	0.5
Storage capacity planning	0.5	0.5
Systems research, planning and product management	0.1	0.1
Traffic management & planning	0.9	0.9
User administration	0.2	0.6
Total staffing level	12.1	27.1

Note that even though the number of staff to support this small section of machines has a small variation, given FTE rates that run in excess of \$97,500 in most operations, the FTE difference accumulates quickly.

This cost and risk component is extremely sensitive to the degree and quality of integration and any autonomic features. The substantial strides made in this area in recent releases have significantly affected the necessary staffing levels.

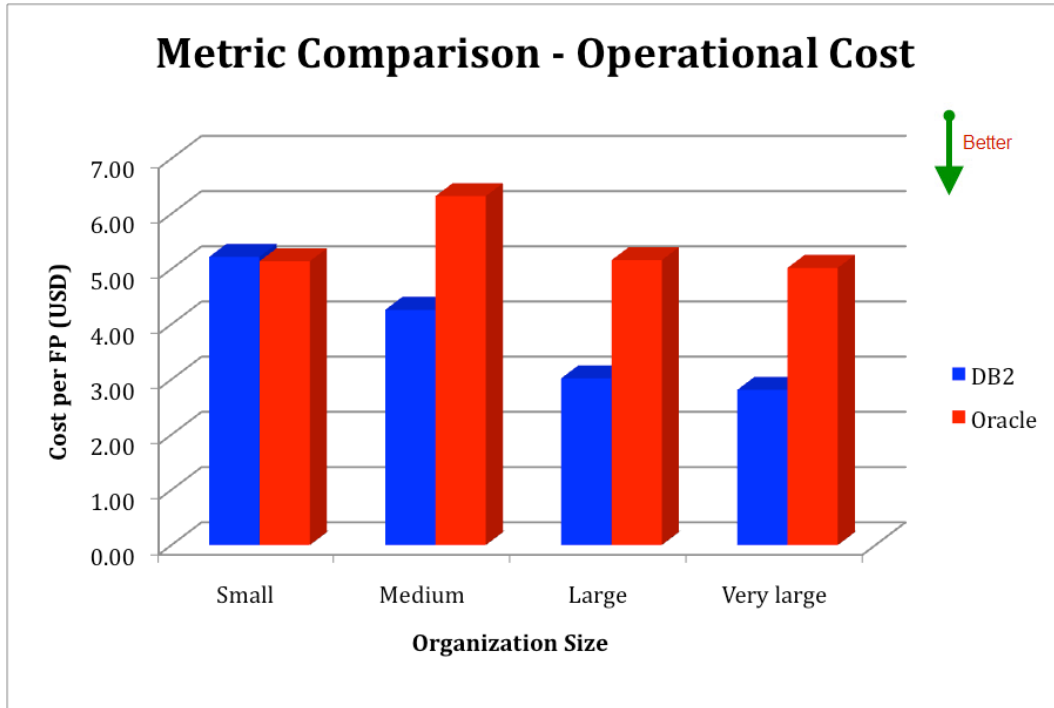
One of the staffing areas that shows a large amount of difference is the service desk, or help center. This difference can be traced to a substantial variance in overall calls, and an equivalent reduction in the amount of time that is required to settle each call. Since many of the calls to the reported help desks were due to latency issues, uneven performance results and resource unavailability, the differences in the DBMS reliability translates into significant support cost savings. Not only are fewer calls initiated (31.2% less), but a higher percentage of calls are resolved in Tier I analysis (52.4%). This results in a shorter handling time per call that ranges from 19.8-49.1%. All of these factors help to form a clear differentiation in staffing.

Costs and Expenses

Operational Cost is a metric that is commonly used to evaluate IT investment from an overall business perspective. It refers to the total cost for computing within the organization, including software and hardware. These costs are primarily IT-related and are based strongly on infrastructure and staffing, including services. However, they do not necessarily include some of the overhead costs, which are more organizational in nature. Examples of excluded costs would be those that relate to new headquarters, property acquisition, initial deployment, etc. Nonetheless, this yardstick for IT spending is a valid evaluation point for business success or failure. It should be noted that the lower the metric, the better customer value is present in this business metric domain. In order to isolate acquisition costs, which are captured in

the Overall Cost metric, this comparison examines the aggregate cost for the first three years of analytics system production. These aggregated costs are then averaged against the workload units that are generated by the deployed system, for a consistent cost base.

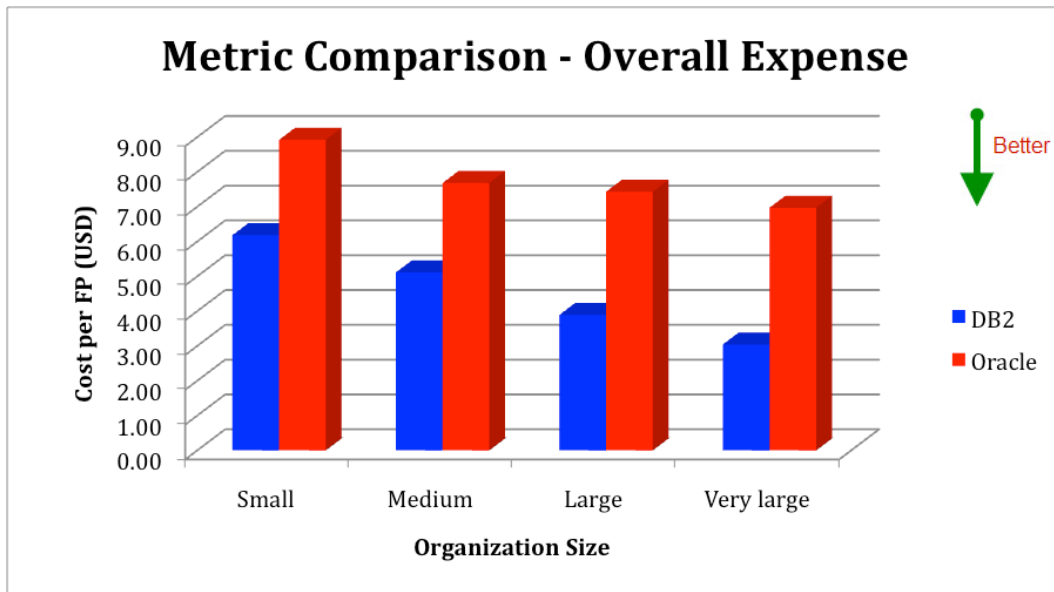
Operational Cost Metric Summary



As shown in this graph, it is apparent the use of DB2 provides lower operational costs in a wide range of organization size. For example, when examining medium-sized organization behavior, the experience of the DBMS customers shows an improvement in operational costs of up to 21% over those costs experienced by other DBMS customers. The lower cost is driven by a higher utilization of the individual platform, in which DB2 has a demonstrated strength.

This expense perspective looks at the total cost to the corporation during a specific time period. This is normalized on three bases: employee, sales revenue and legal entity count, and contains expenses associated with a 6-month deployment preparation phase, if pertinent. These expenses span all of those included in the operational cost metric and are supplemented by expense contributions for physical plant, corporate overhead, long-term investments, etc. This financial metric is more comprehensive than straight operational expense but should not be viewed in isolation, since extraordinary expense patterns for individual organizations may introduce some scattering of data correlation. However, with the large number of contributing organizations, the data is sufficiently comprehensive that combined with the other business metrics, this comparison helps to set an appropriate perspective.

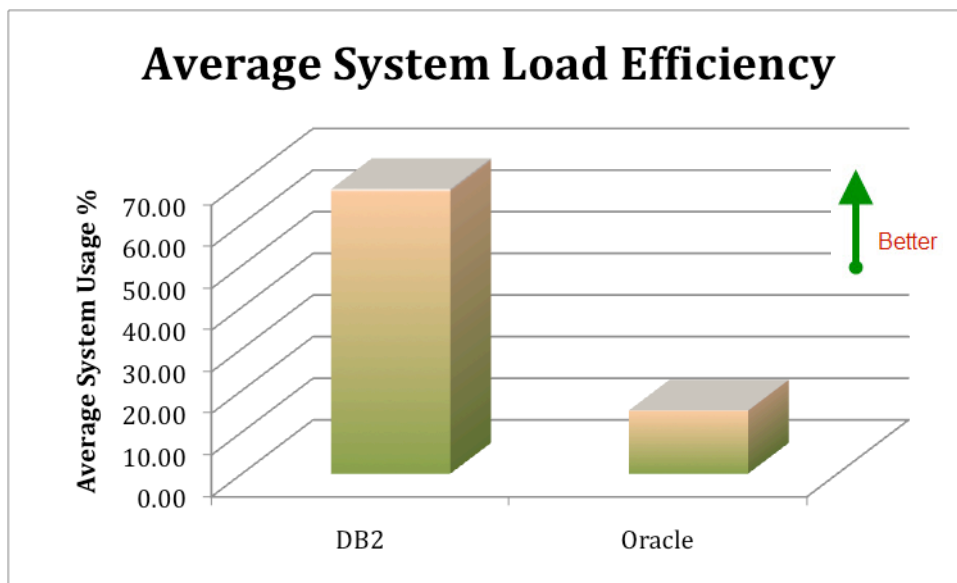
Overall Expense Metric Summary



Again, the DB2 software deployment shows lower overall expenses over a wide range of organization size, ranging up to 41.2% better than the other DBMS choices. This is based largely on the lower costs for the efficient deployment and the lower overall acquisition cost of the solution, including licensing, capital expenses and staffing.

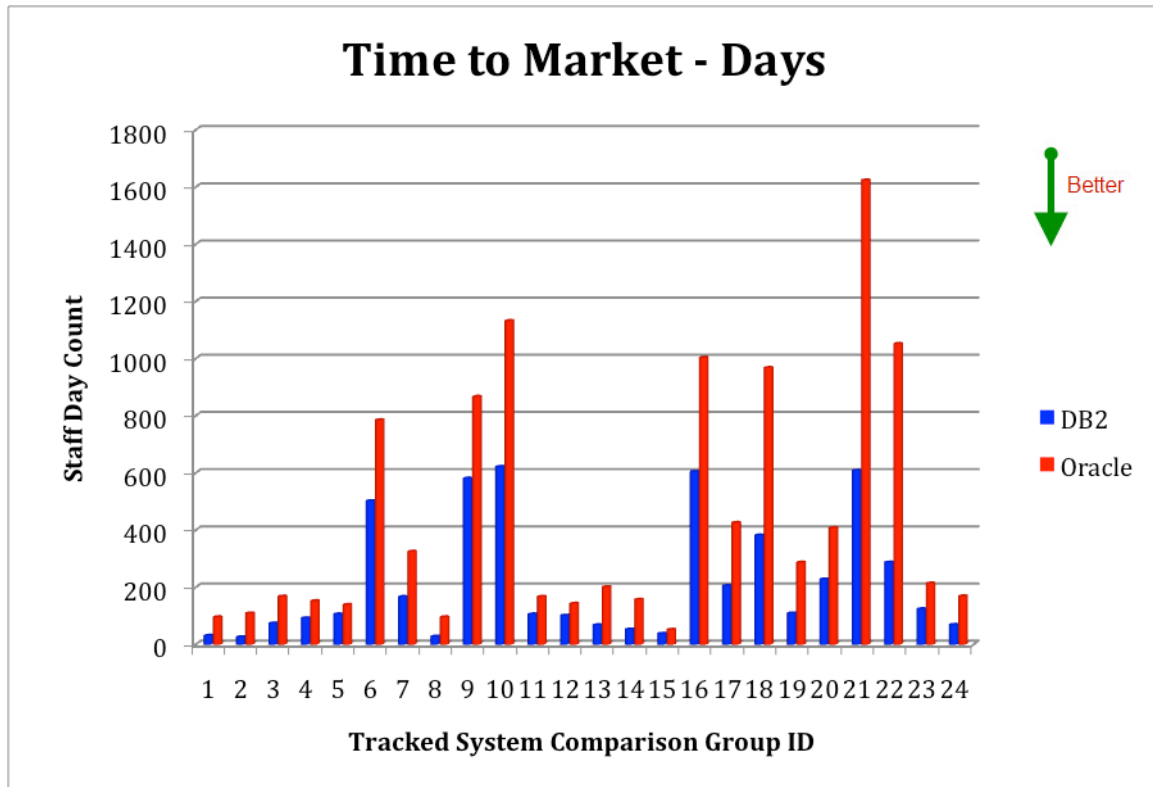
Architectural Efficiency

The level to which systems can be loaded is a general metric for efficiency. Used as an indicative metric, this measurement shows the level that operations can leverage the resources present at an organization.



Time-to-Market Effects

All of the contributory factors, such as staffing, reliability, etc. radically affect the speed in which a company can move a business concept from inception to market. This nimbleness is a key element of increasing market share and continued corporate viability. While the performance metrics were gathered on the production systems, additional measurements were also collected to track the amount of time that the systems took to move from the initial conception to the full production implementation. The results of this effort are summarized on the following chart.



The 24 systems tracked for this portion of the study were paired based on either simultaneous comparative development, or function point equivalents and application type. The comparison is intended to be evocative and not quantitative, since other critical success factors can enter into this picture. However, there are some interesting contributors to the difference in the time-to-market interval. These are primarily a substantially lower staff time to provision (up to 76.15%) and fewer schedule delays. The reasons for schedule delays appear to be layered, but anytime additional funding has to be obtained, the schedules tend to expand. Additional data points show that the tools available within the DBMS play a big role in the speed of implementation. Hence, any reliance on the speed of implementation should take this source of efficiency into account.

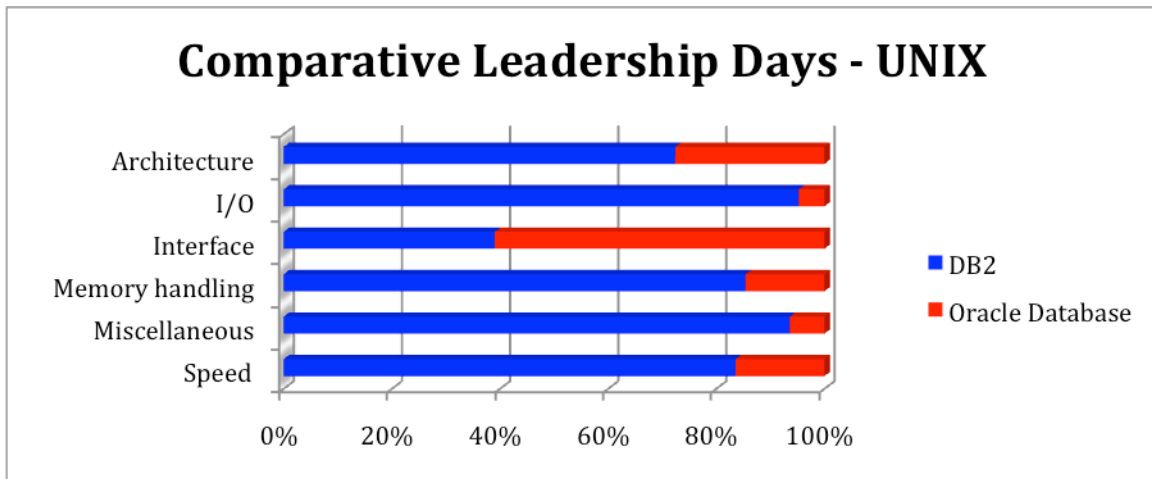
Strategic Thought Leadership

The need for trusted advisors in the complex business environment is becoming more and more intense. The advisor position requires both trust and knowledge. While the trust factor is something that each organization must develop for itself, the level of

expertise and knowledge can be examined from the perspective of industry progress. From this viewpoint, when an organization makes a new feature or function available to its customers that has no matching competition, it can be said to be in a position of leadership. The number of days that such a feature is available before a similar feature or function is made available by any of the competing organizations is an indicative metric for the vision, innovation and expertise of an organization. This metric is reflective of the level of intellectual capacity and research dedication that is part of the organization.

In order to develop a metric base, the new announcement and enhancement information from the DBMS vendors was analyzed to identify new feature availability and enhancement positioning. The analysis was limited to the last 18 calendar months. Only the most advanced position on each feature was credited to the specific vendor. That leadership position was then considered neutralized by any competitor’s release of a similar function. The results of the leadership analysis for the DBMS for Power Systems are shown below.

Overall Technology Leadership Summary – UNIX



The chart above shows the leadership position days as a percentage of the overall number of leadership days in that category. It is clear from the data that DB2 has a strong position in this regard, with constant technological advances that position it advantageously against its competition in the UNIX market, and on POWER platforms.

Customer Satisfaction

The ultimate metric for the success of any product is customer satisfaction. The satisfaction rating is an aggregate result of how well the customer feels that the DBMS provides benefit for the expense, reliability and thought leadership. As such, this rating can be seen as the final accolade of how well a product is doing. The customer satisfaction information has been split into two distinct groups – executive and operational. The difference in expectations between the groups that manage the overall organization and those that deal intimately with the day-to-day operations of the line of business and IT infrastructure make the delineation germane to really

understanding how the different DBMS options are perceived by the production customers.

Customer Satisfaction – Power Systems



As the chart above shows, the overall satisfaction with DB2 is high, with the balance of feature, cost and reliability that the customer expects.

System x Comparisons

Similarly to the examination of the Power platforms, the System x platforms were investigated for raw performance and a myriad of other factors, including reliability, time-to-market effects, staffing levels and total costs. Microsoft SQL Server was also included in this venue, due to its substantial market position in the Intel-based platform arena.

Performance

The examination of performance on the System x was restricted to the models normally used for general workload, such as System x Models x3950 and Blades such as the HSxx. This examination was conducted within the available production environments and centered on those application workloads that typically are designated for System x, such as client ERP activity, customer support presentation layers, order entry, etc.

The delineation between the DBMS products is not as clearly defined on the System x platforms as on the Power System architecture. Primarily due to the smaller range of platform characteristics in the production domain for Intel-based systems, the System x shows a smaller, but still definitive, advantage with the DB2 product over

Oracle Database and MS SQL Server. The projects examined under this category include platforms utilizing Microsoft Windows or Linux operating systems. The projects (and their method of comparison) that were examined for the System x platforms are:

- ERP – side-by-side
- CRMS – side-by-side
- Order entry – side-by-side
- Retail and distribution - side-by-side
- Transactions – correlated

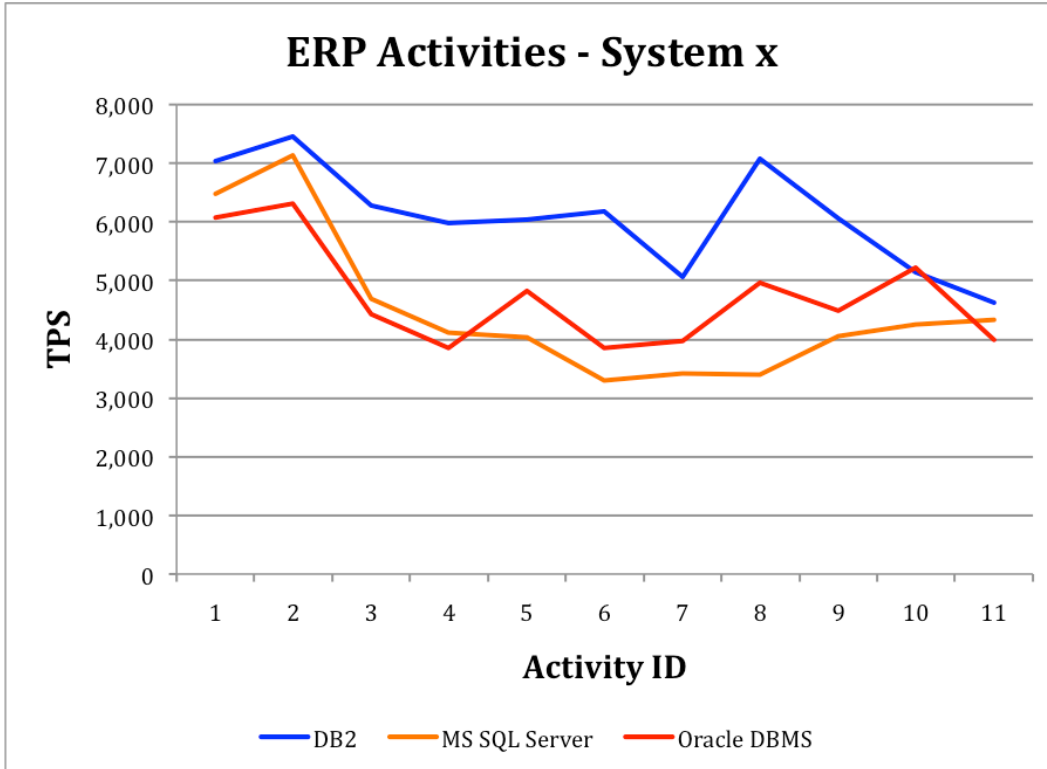
Many of the systems were run either in full or partial parallel for comparison purposes, in full production. The number of separate organizations (referred to as “core” systems), along with the number of instances that were compared at those locations is shown in the following table.

Study Structure – System x Comparison Components

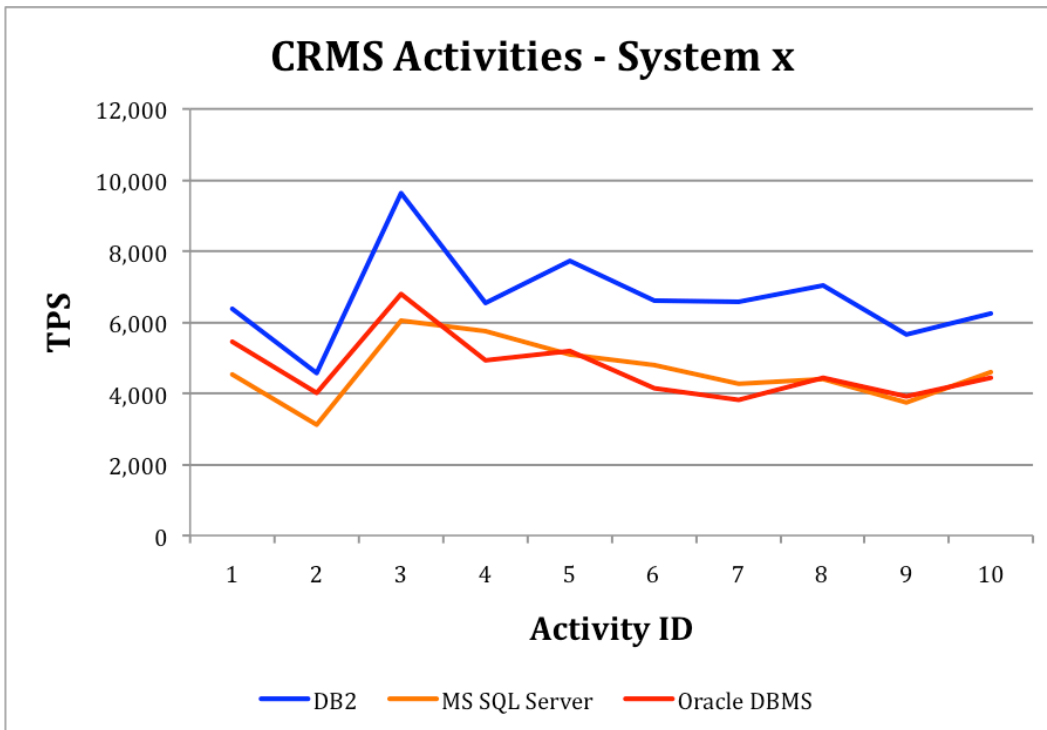
System Type	Core Systems	Comparison Option Count
ERP	2,394	3,162
CRMS	4,622	5,198
Order Entry	631	1,358
Retail and Distribution	126	367
Transaction	967	3,842
Total	8,740	13,927

Note that the comparison figure equates to application deployments across the listed number of core systems, with more than one server per core system as standard. In many situations, there are application servers, database servers and web servers deployed for a single core system. Therefore, this total comparison environment was comprised of 13,927 separate comparison options of 8,740 core systems. The individual projects were comprised of normal activity in the subject areas. The activity was tracked by the contributing business action, including the number of expected rows per query, etc. A short list of the individual actions can be seen in Appendix A.

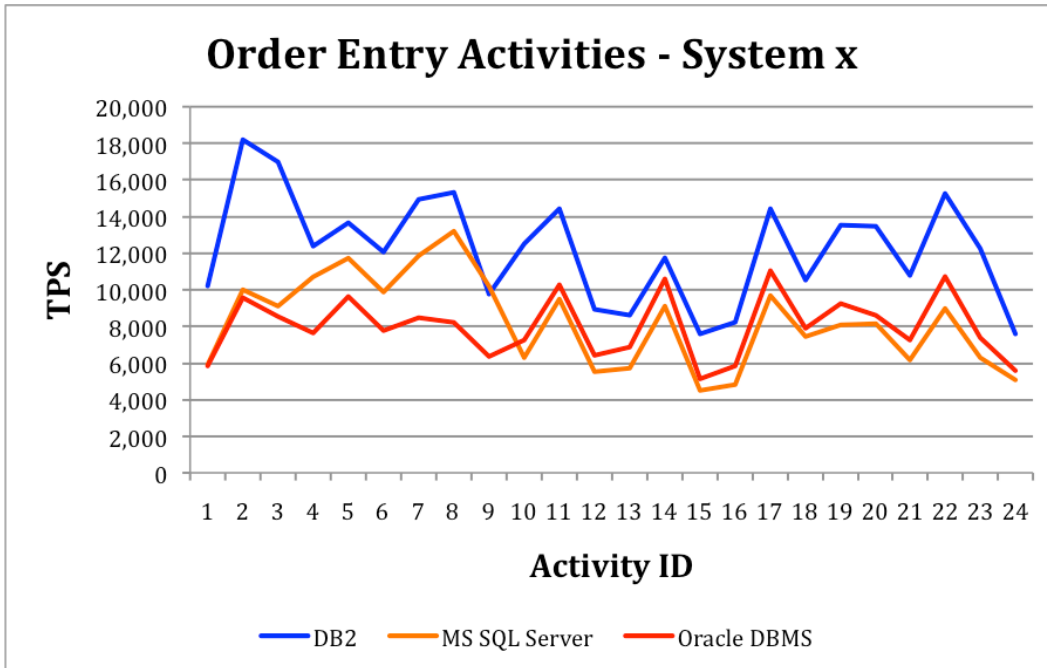
A detailed comparison of the client tests, showing Oracle Database, MS SQL Server and DB2 performance characteristics, is represented by the following system-specific graphs.



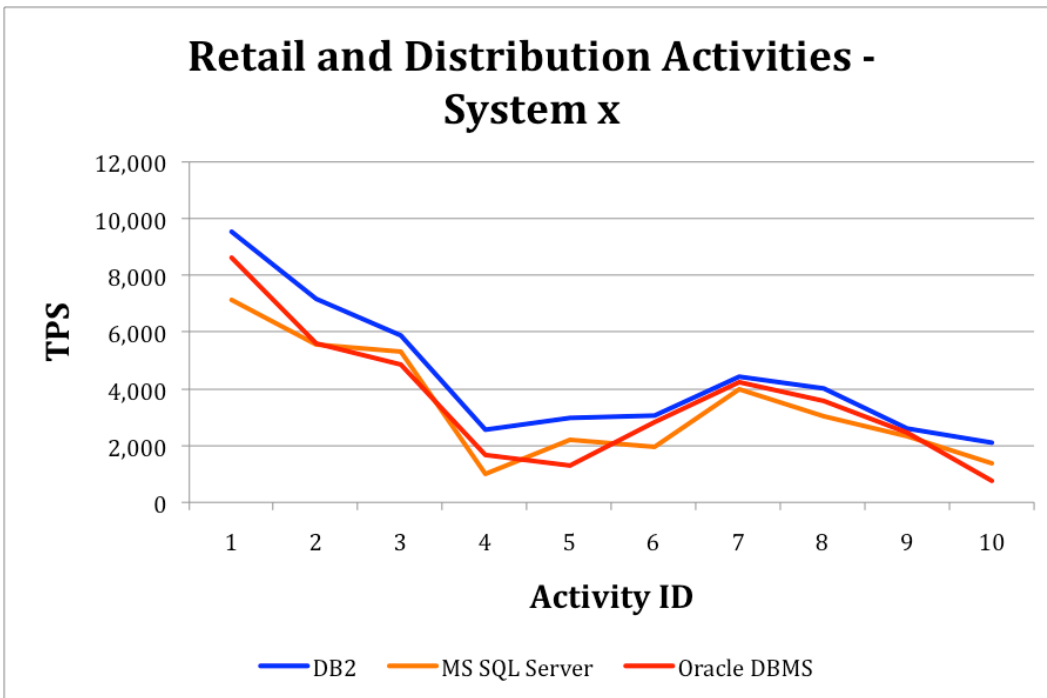
The mixed workload from the ERP activity shows the results of the DB2 optimization on the System x platform. Much of the difference in performance can be tracked to the more efficient database calls that result in less overall data manipulation resource demand.



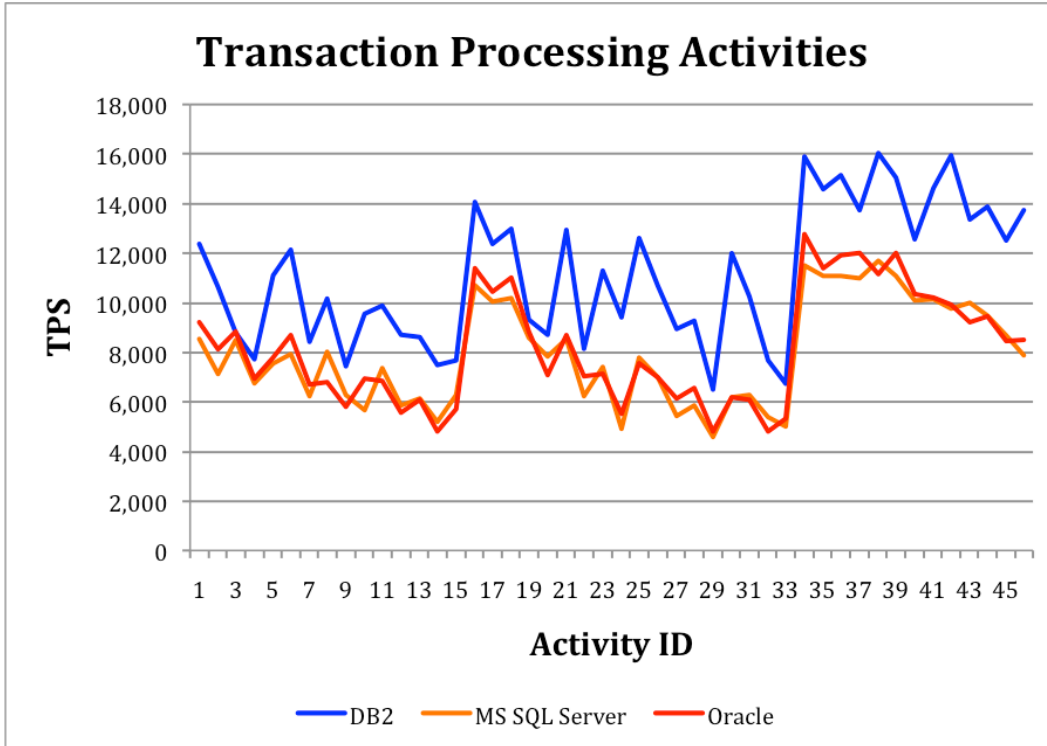
The clear advantage to the DB2 database calls can be seen in the higher throughput for both short and longer latency CRMS activities.



The volatile nature of order entry activities causes a wide variation of throughput for the overall demand. However, DB2 again shows a differentiation in the ability of the system to push more workload through.



The high volume of most retail and distribution systems means that even a small difference in throughput results in significant savings. The higher throughput shown with DB2 was reported to be as high as 22% more than the competitive installations.



The transaction processing activities on the System x platforms are significantly different for the different DBMS offerings. This is primarily due to the strong reliance on memory and I/O integration in the DBMS, but also on the ease of tuning the system to fit a varied workload.

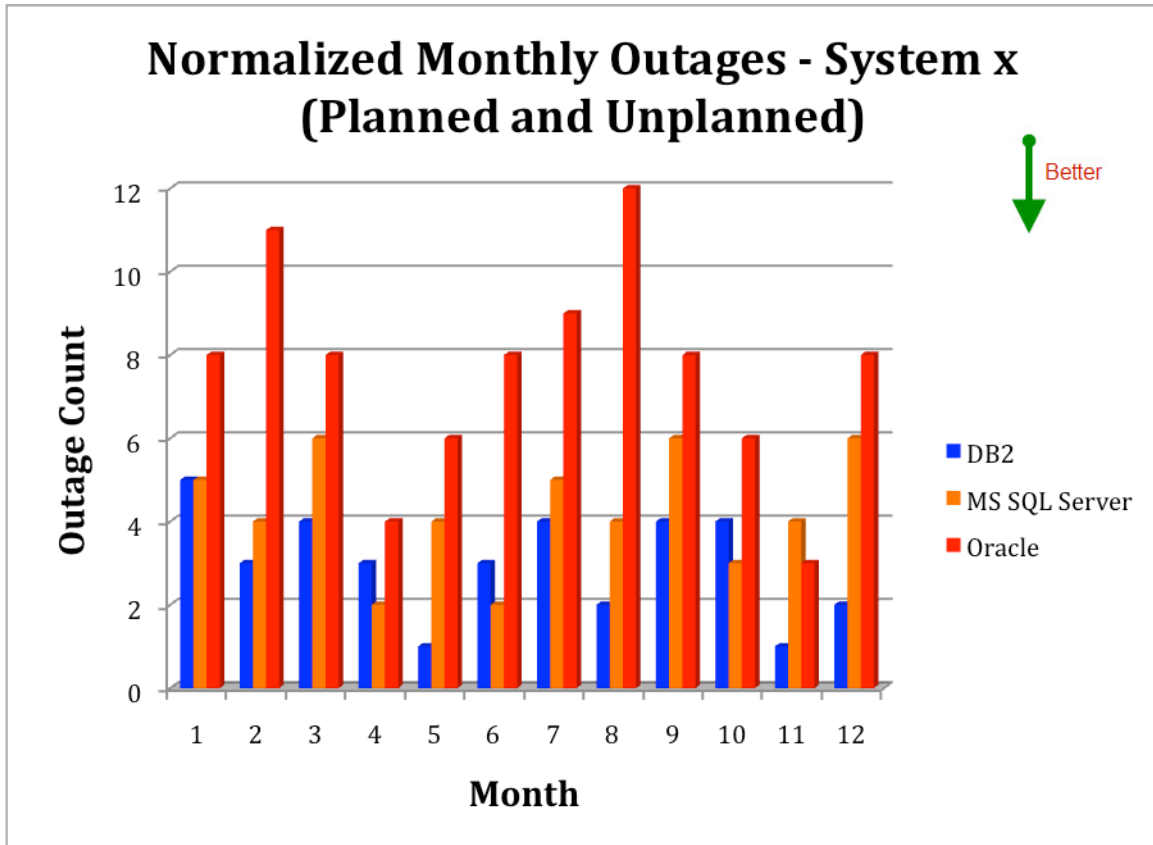
The improved efficiency of the DB2 product on the System x is notably higher than on the overall equipment population domain, ranging up to an additional 49% on the transaction-based metrics, compared to Oracle Database and MS SQL Server.

Business Measures

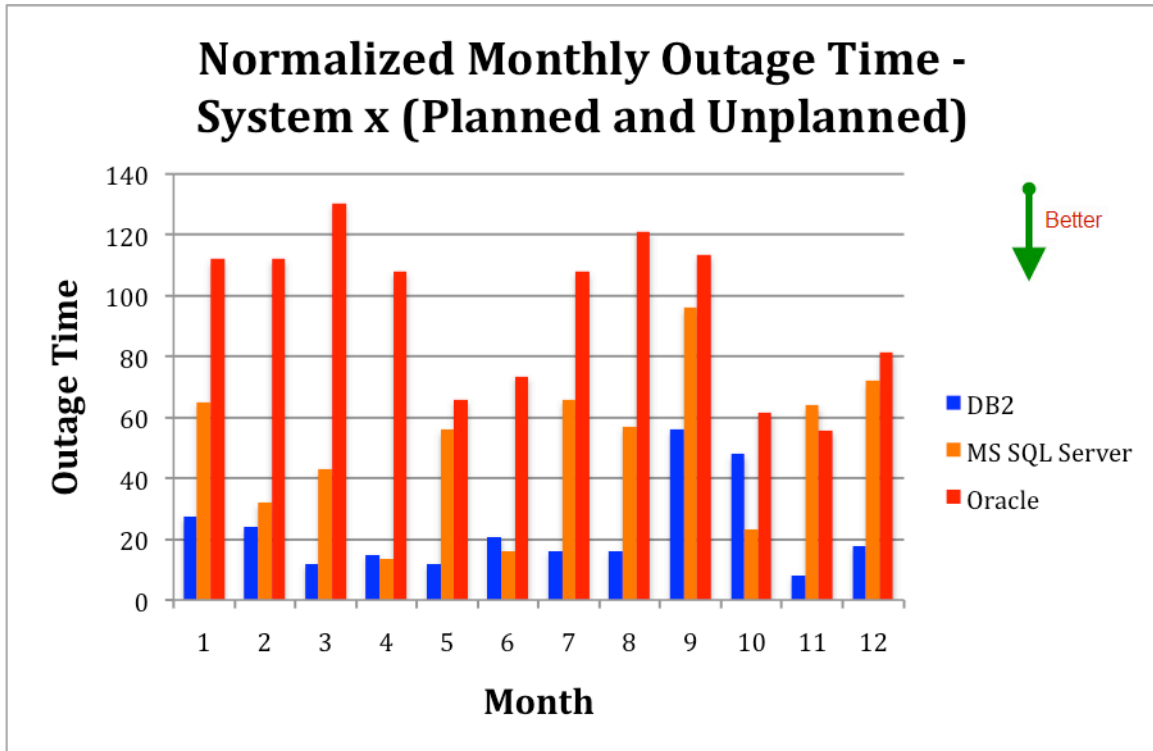
Similar to the UNIX environment, the total cost and risk of ownership remain the primary focus when evaluating purchase options. The contributory metrics that go into building this picture are still reliability, staffing, total costs, and time-to-market effects. These factors were measured during the same time period that the performance metrics were gathered for all of the subject tests for the Intel-based test group.

Reliability

The dependability of the implementation is a combination of the individual reliability of each component, along with the quality and effectiveness of the actual implementation. As such, both the planned and unplanned outages affect the overall usability of the total system. The charts below show the number of outages that were recorded during the reported period, as well as, the total number of minutes that those outages consumed.



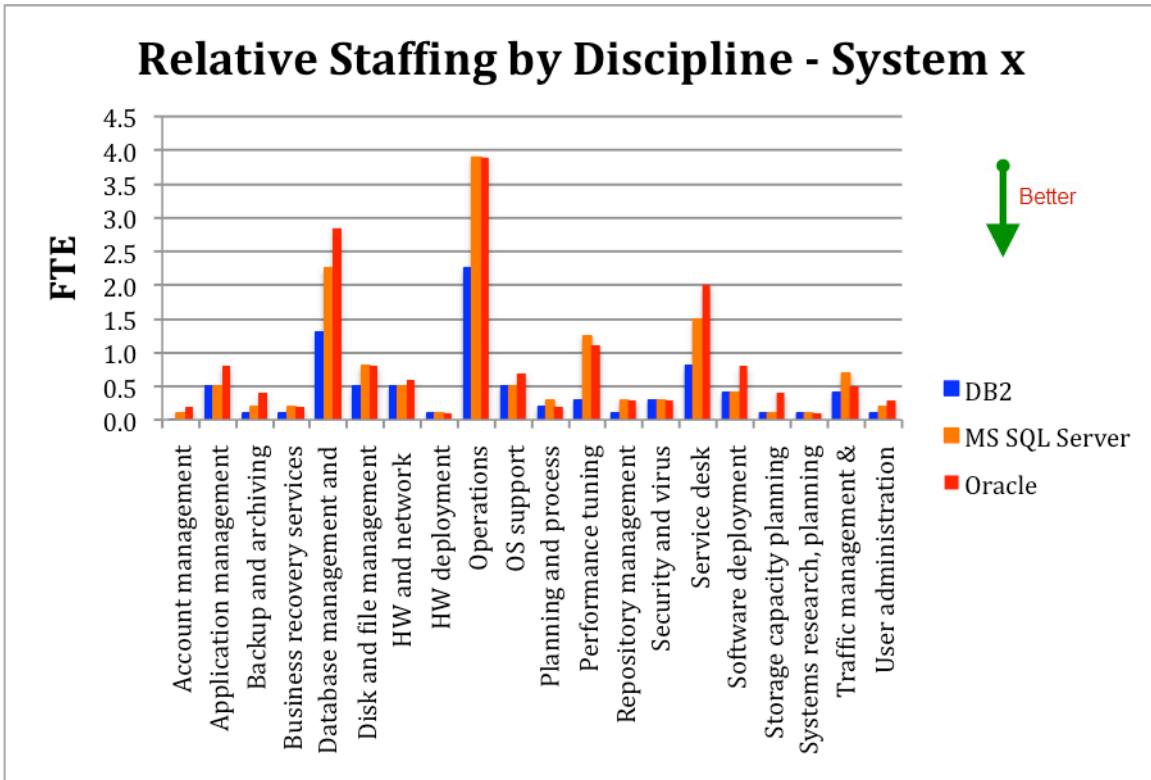
The number of outages has been normalized for a 75-platform operation, with both planned and unplanned outages included. Since normal learning curve may skew the results, the first three months of any project incorporating a previously-unknown operational architecture have also been excluded. Of the remaining factors, one of the most prevalent is the need for database reorganization and redeployment. Each of these outages takes valuable access time away from the corporate resources. The chart above clearly shows the lower number of outages that accompany a DB2 deployment. The following chart shows the total number of minutes of unavailability that those outages represent.



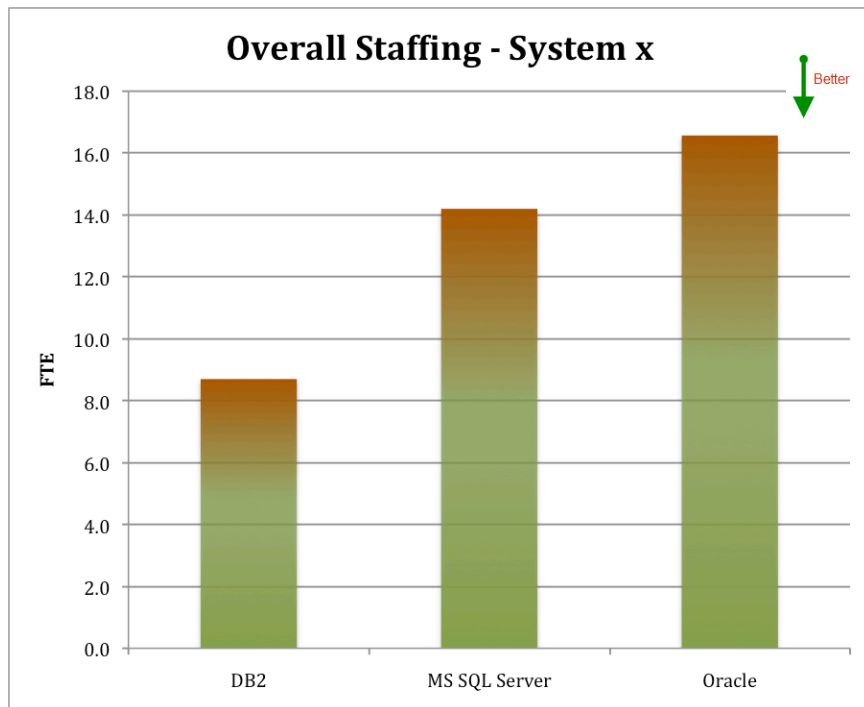
The differences in deploying DB2 are even more evident when the total time is taken into account.

Staffing

Another important perspective of business issues for any deployment is the number and skills of the supporting staff. These factors represent a significant cost contribution and are indicative of the complexity and risk of the environment. The chart below shows the relative staffing for a normalized operations group. This comparison can be used to understand the challenges of staffing and overall budget costs for the three DBMS implementations on IBM equipment. These staffing figures were collected from the actual operation groups measured in the other metric collection efforts, and cover organizations in North and South America, Asia, Europe, Antarctica and Australia.



The differences in individual discipline are more apparent when seen in aggregate.



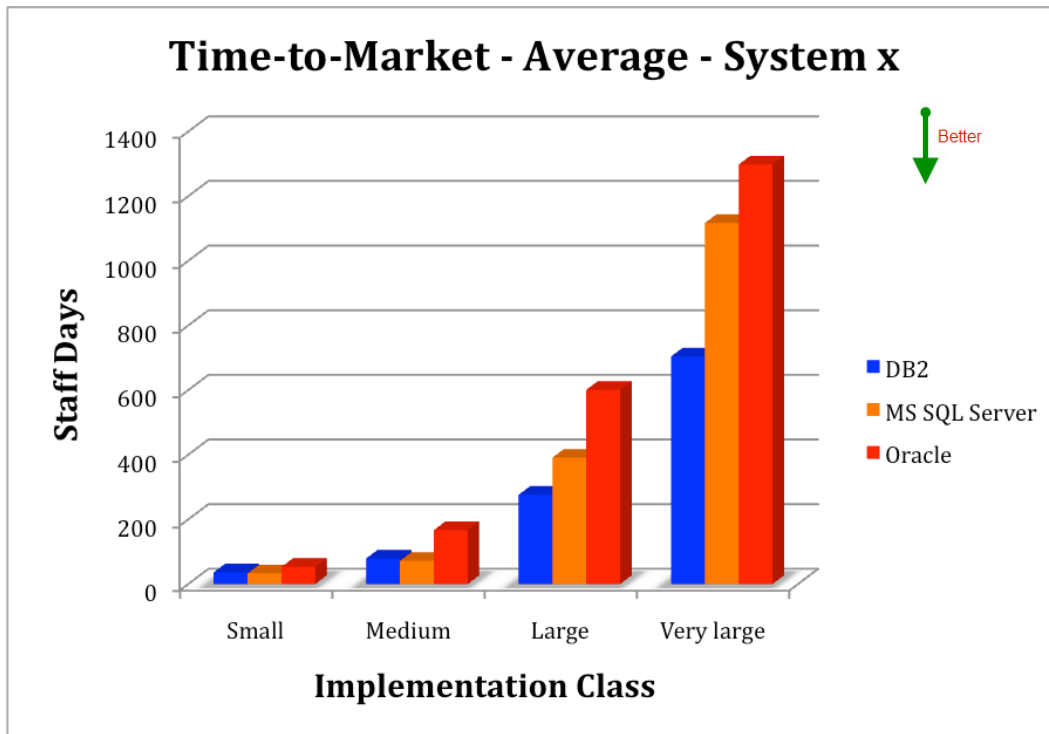
This summary highlights the substantial percentage difference in staffing for the different DBMS solutions, with DB2 reporting a significantly lower staff requirement. Even though the number of staff to support this small section of machines has a small variation, given FTE rates that run in excess of \$69,300 in most operations, the FTE

difference accumulates quickly. Once again, integration and any autonomic features affect the necessary staffing levels.

One of the staffing areas that shows a difference is the service desk, or help center. This difference can be traced to a substantial variance in overall calls, and an equivalent reduction in the amount of time that is required to settle each call. Since many of the calls to the reported help desks were due to latency issues, uneven performance results and resource unavailability, the differences in the DBMS reliability translates into significant support cost savings. Not only are fewer calls initiated (12.3% less), but a higher percentage of calls are resolved in Tier I analysis (15.4%). This results in a shorter handling time per call that ranges from 6.2-9.8%.

Time-to-Market Effects

All of the contributory factors, such as staffing, reliability, etc. radically affect the speed in which a company can move a business concept from inception to market. This nimbleness is a key element of increasing market share and continued corporate viability. In addition to the production performance metrics, data was also collected on various aspects of the project lifecycle. Relevant portions of the examination are summarized on the following chart.



Note that the implementation interval is broken down by the relative size of the deployment. This has been done to avoid any obliteration of important details. The total number of staff days is significantly affected by the DBMS, with the DB2 advantage saving as much as 52% of the overall staffing load for projects.

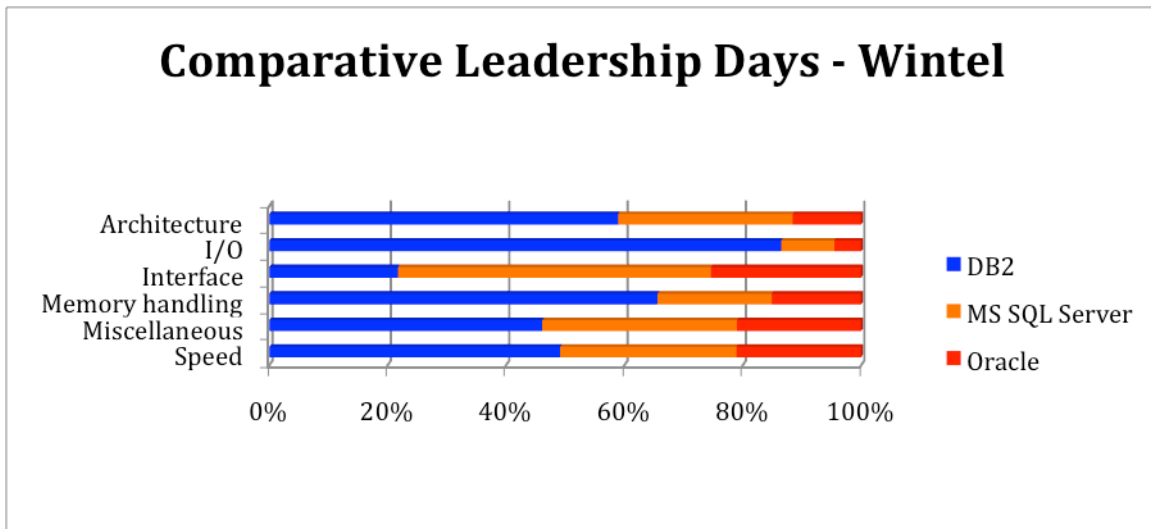
The systems tracked for this portion of the study were associated based on either simultaneous comparative development, or function point equivalents and application type. The comparison is intended to be evocative and not quantitative.

The average development days to market are significant indications, within the application classification.

Strategic Thought Leadership

The rationale for strategic thought leadership for the System x side of this analysis is substantively the same as that for Power Systems. The leadership in feature and function for this study segment is shown below and is reflective of the level of intellectual capacity and research dedication that is part of the organization. Once again, the analysis was limited to the last 18 calendar months. Only the most advanced position on each feature was credited to the specific vendor. That leadership position was then considered neutralized by any competitor’s release of a similar function. The results of the leadership analysis for the DBMS for System x are shown below.

Overall Technology Leadership Summary – System x

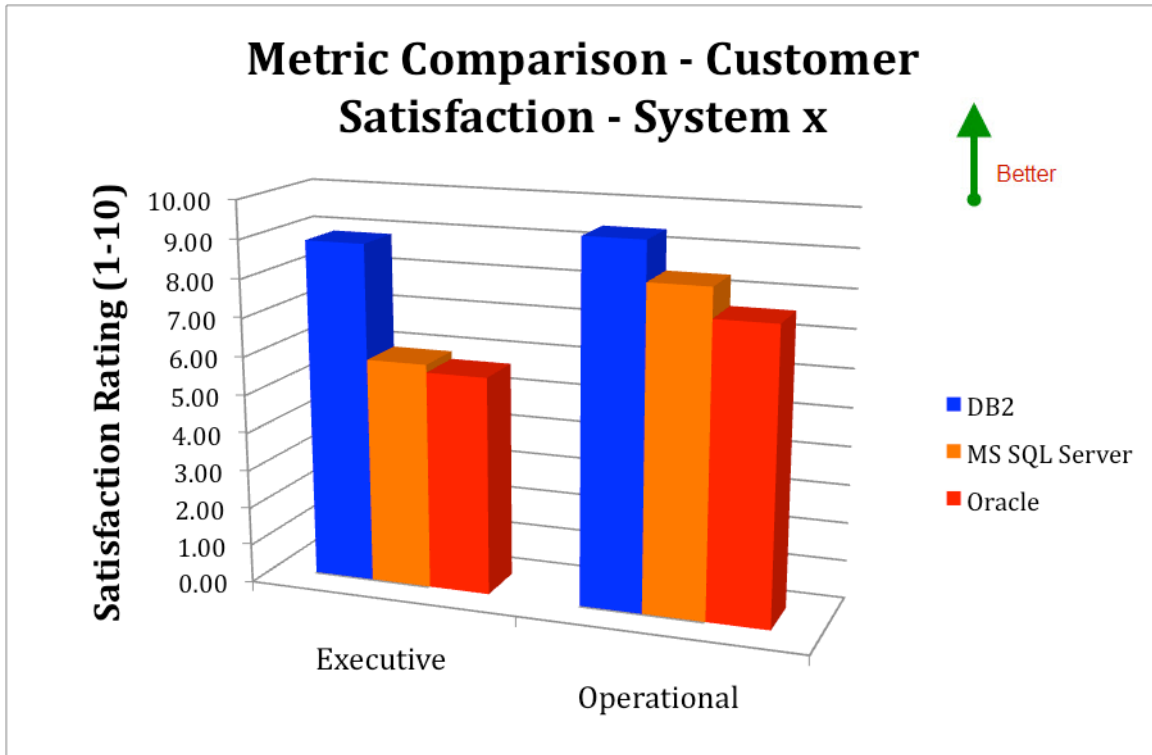


The chart above shows the leadership position days as a percentage of the overall number of leadership days in that category. It is clear from the data that DB2 has a strong position in areas associated with architecture, I/O and memory handling, representing technological advances that position it advantageously against its competition in the Wintel market, and on System x platforms.

Customer Satisfaction

The customer satisfaction for the DBMS performance on System x architecture is similar to that of the ratings for the Power systems. Once again the customer satisfaction information has been split into two distinct groups – executive and operational.

Customer Satisfaction – System x



The difference in executive satisfaction is considerable, with DB2 consistently ranking higher by as much as 35%. The operational satisfaction is also higher for DB2, although the differential averages closer to 20% for the line of business and IT personnel.

Conclusions

The advantages of running DB2 on IBM Power and System x equipment are strongly supported by more than 3,467,000 data points covering over 9,200 closely watched production comparisons. These advantages translate into hard cost savings for each customer than can substantially affect the bottom line cost of ownership. The real world affect on business is considerable. This benefit is most achievable when a good fit between the DBMS and hardware platform is employed. Allowing the strengths of the System x and Power System platforms to mingle with the strengths of the DB2 product is one of the best ways to support the customer in their continuing quest for lower costs and improved user satisfaction.

Attributions

AIX, DB2, IBM, Power System, and System x are trademarks or registered trademarks of International Business Machines Corporation in the United States of America and other countries.

Microsoft, Windows, SQL Server and the Windows logo are trademarks of Microsoft Corporation in the United States of America.

UNIX is a registered trademark in the United States of America and other countries, licensed exclusively through The Open Group.

Oracle and Oracle Database are registered trademarks in the United States of America and other countries of ORACLE Corporation.

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

Further Methodology Notes

In order to support the comprehensive nature of this analysis, information from diverse DBMS deployments, industries, geographies, and vendors were obtained. In any collection of this type, there is some overlap that occurs, such as when multiple vendors are present at an organization. In such cases, the total of the discrete percentages may exceed 100%. Those organizations with a multi-layered deployment, such as multiple geographical locations or industrial classifications, have been analyzed with discrete breakouts of their feedback for all metrics.

The main data perspective examines DBMS deployments for sheer performance, without or with virtualization. Virtualization is present in 31.4% of the surveyed organizations, while non-virtualized deployments are present in 68.6%.

The industry representation covers manufacturing (31.20%), distribution (18.62%), healthcare (9.26%), retail (5.92%), financial (17.20%), communications (12.68%) and a miscellaneous group (5.12%) that includes government and other areas.

The geographies are also well represented with North and South America providing 41.62% of the reporting organizations, Europe 30.87%, Pacific Rim and Asia 18.62%, Africa 6.14%, and those organizations that do not fit into those geographic divisions reporting 2.75% of the information.

SIL takes the experiential data that is reported from these tests and derives average and median results from the pools. The average derived from this is a simple average, where the number of reports divides the total of all the values. The median is also a simple median, which marks the value where half of the reports are above that number and half below.

Since strategies and benefits tend to vary by organization size, SIL further groups the organizations by the categories of small, medium, large and extra large where appropriate. These categories combine the number of employees and the gross annual revenue of the organization. This staff count multiplied by gross revenue creates a metric for definition that is used throughout the analysis. In this definition, a small organization could be expected to have fewer than 100 employees and gross less than \$20 million, or a value of 2,000, e.g., 100 (employees) X 20 (million dollars of gross revenue). An organization with 50 employees and gross revenue of \$40 million would have the same size rating, and would be grouped in the analysis with the first company.

Appendix A: Test Cases

UNIX Test Cases

A short list of the individual actions can be seen below for each general project type. In all of the situations, the components of the system type were carefully recorded, with overall performance and capacity characteristics analyzed closely. The findings from this analysis were then scrutinized to show root cause and the unique performance footprint that each DBMS and platform creates.

Tracked System Activity: Credit Card Processes

1. card verification
2. card approval
3. card rejection
4. merchant verification
5. merchant approval
6. merchant rejection
7. purchase amount approval
8. purchase amount rejection
9. posting purchase to cardholder account
10. posting purchase to merchant account
11. posting returns to cardholder account
12. posting returns to merchant account
13. posting cardholder payments
14. posting merchant payments
15. card limit adjustment
16. over limit notification to cardholder
17. merchant limit adjustment
18. periodic billing to cardholder
19. billing adjustments to cardholder
20. special request bill printing
21. periodic accounting to merchant
22. accounting adjustments to merchant
23. fraud analysis
24. clearinghouse batch transmission
25. cash advance posting
26. interest recalculation
27. retroactive posting calculations
28. balance inquiry
29. customer-applied payment

A total of 82 core systems were tracked with this type of large-scale application. The subject organizations covered a mixture of full financial service companies, others that are solely focused on this market sector and retail organizations that also offer this type of customer product. Seventeen of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Customer Support

1. log new support call
2. update support call data
3. log completion of support call
4. lookup customer name
5. lookup customer address
6. lookup customer program information
7. lookup customer bill or invoice
8. update customer name

9. update customer address
10. update customer program information
11. update support staff dashboard
12. update supervisor dashboard
13. escalate call priority

A total of 54 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Twenty-one of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: DSS

1. receive data
2. clean data
3. resolve data discrepancies
4. verify user
5. insert one row of data into single small table
6. insert one row of data into single medium table
7. insert one row of data into single large table
8. insert one row of data into single very large table
9. insert one row of data into each of 3-5 small tables
10. insert one row of data into each of 3-5 medium tables
11. insert one row of data into each of 3-5 large tables
12. insert one row of data into each of 3-5 very large tables
13. insert one row of data into each of 3-5 combination tables
14. insert one row of data into each of 6-10 small tables
15. insert one row of data into each of 6-10 medium tables
16. insert one row of data into each of 6-10 large tables
17. insert one row of data into each of 6-10 very large tables
18. insert one row of data into each of 6-10 combination tables
19. insert ten rows of data into single small table
20. insert ten rows of data into single medium table
21. insert ten rows of data into single large table
22. insert ten rows of data into single very large table
23. insert ten rows of data into each of 3-5 small tables
24. insert ten rows of data into each of 3-5 medium tables
25. insert ten rows of data into each of 3-5 large tables
26. insert ten rows of data into each of 3-5 very large tables
27. insert ten rows of data into each of 3-5 combination tables
28. insert ten rows of data into each of 6-10 small tables
29. insert ten rows of data into each of 6-10 medium tables
30. insert ten rows of data into each of 6-10 large tables
31. insert ten rows of data into each of 6-10 very large tables
32. insert ten rows of data into each of 6-10 combination tables
33. insert 100 rows of data into single small table
34. insert 100 rows of data into single medium table

35. insert 100 rows of data into single large table
36. insert 100 rows of data into single very large table
37. insert 100 rows of data into each of 3-5 small tables
38. insert 100 rows of data into each of 3-5 medium tables
39. insert 100 rows of data into each of 3-5 large tables
40. insert 100 rows of data into each of 3-5 very large tables
41. insert 100 rows of data into each of 3-5 combination tables
42. insert 100 rows of data into each of 6-10 small tables
43. insert 100 rows of data into each of 6-10 medium tables
44. insert 100 rows of data into each of 6-10 large tables
45. insert 100 rows of data into each of 6-10 very large tables
46. insert 100 rows of data into each of 6-10 combination tables
47. insert 1000 rows of data into single small table
48. insert 1000 rows of data into single medium table
49. insert 1000 rows of data into single large table
50. insert 1000 rows of data into single very large table
51. insert 1000 rows of data into each of 3-5 small tables
52. insert 1000 rows of data into each of 3-5 medium tables
53. insert 1000 rows of data into each of 3-5 large tables
54. insert 1000 rows of data into each of 3-5 very large tables
55. insert 1000 rows of data into each of 3-5 combination tables
56. insert 1000 rows of data into each of 6-10 small tables
57. insert 1000 rows of data into each of 6-10 medium tables
58. insert 1000 rows of data into each of 6-10 large tables
59. insert 1000 rows of data into each of 6-10 very large tables
60. insert 1000 rows of data into each of 6-10 combination tables
61. roll-up data for 3-5 small tables from daily to weekly
62. roll-up data for 3-5 small tables from daily to calendar month
63. roll-up data for 3-5 small tables from daily to fiscal month
64. roll-up data for 3-5 small tables from daily to calendar quarter
65. roll-up data for 3-5 small tables from daily to fiscal quarter
66. roll-up data for 3-5 small tables from daily to calendar year
67. roll-up data for 3-5 small tables from daily to fiscal year
68. roll-up data for 3-5 medium tables from daily to weekly
69. roll-up data for 3-5 medium tables from daily to calendar month
70. roll-up data for 3-5 medium tables from daily to fiscal month
71. roll-up data for 3-5 medium tables from daily to calendar quarter
72. roll-up data for 3-5 medium tables from daily to fiscal quarter
73. roll-up data for 3-5 medium tables from daily to calendar year
74. roll-up data for 3-5 medium tables from daily to fiscal year
75. roll-up data for 3-5 large tables from daily to weekly
76. roll-up data for 3-5 large tables from daily to calendar month
77. roll-up data for 3-5 large tables from daily to fiscal month
78. roll-up data for 3-5 large tables from daily to calendar quarter
79. roll-up data for 3-5 large tables from daily to fiscal quarter
80. roll-up data for 3-5 large tables from daily to calendar year
81. roll-up data for 3-5 large tables from daily to fiscal year

82. roll-up data for 3-5 very large tables from daily to weekly
83. roll-up data for 3-5 very large tables from daily to calendar month
84. roll-up data for 3-5 very large tables from daily to fiscal month
85. roll-up data for 3-5 very large tables from daily to calendar quarter
86. roll-up data for 3-5 very large tables from daily to fiscal quarter
87. roll-up data for 3-5 very large tables from daily to calendar year
88. roll-up data for 3-5 very large tables from daily to fiscal year
89. roll-up data for 3-5 combination tables from daily to weekly
90. roll-up data for 3-5 combination tables from daily to calendar month
91. roll-up data for 3-5 combination tables from daily to fiscal month
92. roll-up data for 3-5 combination tables from daily to calendar quarter
93. roll-up data for 3-5 combination tables from daily to fiscal quarter
94. roll-up data for 3-5 combination tables from daily to calendar year
95. roll-up data for 3-5 combination tables from daily to fiscal year
96. roll-up data for 6-10 small tables from daily to weekly
97. roll-up data for 6-10 small tables from daily to calendar month
98. roll-up data for 6-10 small tables from daily to fiscal month
99. roll-up data for 6-10 small tables from daily to calendar quarter
100. roll-up data for 6-10 small tables from daily to fiscal quarter
101. roll-up data for 6-10 small tables from daily to calendar year
102. roll-up data for 6-10 small tables from daily to fiscal year
103. roll-up data for 6-10 medium tables from daily to weekly
104. roll-up data for 6-10 medium tables from daily to calendar month
105. roll-up data for 6-10 medium tables from daily to fiscal month
106. roll-up data for 6-10 medium tables from daily to calendar quarter
107. roll-up data for 6-10 medium tables from daily to fiscal quarter
108. roll-up data for 6-10 medium tables from daily to calendar year
109. roll-up data for 6-10 medium tables from daily to fiscal year
110. roll-up data for 6-10 large tables from daily to weekly
111. roll-up data for 6-10 large tables from daily to calendar month
112. roll-up data for 6-10 large tables from daily to fiscal month
113. roll-up data for 6-10 large tables from daily to calendar quarter
114. roll-up data for 6-10 large tables from daily to fiscal quarter
115. roll-up data for 6-10 large tables from daily to calendar year
116. roll-up data for 6-10 large tables from daily to fiscal year
117. roll-up data for 6-10 very large tables from daily to weekly
118. roll-up data for 6-10 very large tables from daily to calendar month
119. roll-up data for 6-10 very large tables from daily to fiscal month
120. roll-up data for 6-10 very large tables from daily to calendar quarter
121. roll-up data for 6-10 very large tables from daily to fiscal quarter
122. roll-up data for 6-10 very large tables from daily to calendar year
123. roll-up data for 6-10 very large tables from daily to fiscal year
124. roll-up data for 6-10 combination tables from daily to weekly
125. roll-up data for 6-10 combination tables from daily to calendar month
126. roll-up data for 6-10 combination tables from daily to fiscal month
127. roll-up data for 6-10 combination tables from daily to calendar quarter
128. roll-up data for 6-10 combination tables from daily to fiscal quarter

129. roll-up data for 6-10 combination tables from daily to calendar year
130. roll-up data for 6-10 combination tables from daily to fiscal year
131. retrieve one row of data from single small table
132. retrieve one row of data from single medium table
133. retrieve one row of data from single large table
134. retrieve one row of data from single very large table
135. retrieve one row of data from each of 3-5 small tables
136. retrieve one row of data from each of 3-5 medium tables
137. retrieve one row of data from each of 3-5 large tables
138. retrieve one row of data from each of 3-5 very large tables
139. retrieve one row of data from each of 3-5 combination tables
140. retrieve one row of data from each of 6-10 small tables
141. retrieve one row of data from each of 6-10 medium tables
142. retrieve one row of data from each of 6-10 large tables
143. retrieve one row of data from each of 6-10 very large tables
144. retrieve one row of data from each of 6-10 combination tables
145. retrieve ten rows of data from single small table
146. retrieve ten rows of data from single medium table
147. retrieve ten rows of data from single large table
148. retrieve ten rows of data from single very large table
149. retrieve ten rows of data from each of 3-5 small tables
150. retrieve ten rows of data from each of 3-5 medium tables
151. retrieve ten rows of data from each of 3-5 large tables
152. retrieve ten rows of data from each of 3-5 very large tables
153. retrieve ten rows of data from each of 3-5 combination tables
154. retrieve ten rows of data from each of 6-10 small tables
155. retrieve ten rows of data from each of 6-10 medium tables
156. retrieve ten rows of data from each of 6-10 large tables
157. retrieve ten rows of data from each of 6-10 very large tables
158. retrieve ten rows of data from each of 6-10 combination tables
159. retrieve 100 rows of data from single small table
160. retrieve 100 rows of data from single medium table
161. retrieve 100 rows of data from single large table
162. retrieve 100 rows of data from single very large table
163. retrieve 100 rows of data from each of 3-5 small tables
164. retrieve 100 rows of data from each of 3-5 medium tables
165. retrieve 100 rows of data from each of 3-5 large tables
166. retrieve 100 rows of data from each of 3-5 very large tables
167. retrieve 100 rows of data from each of 3-5 combination tables
168. retrieve 100 rows of data from each of 6-10 small tables
169. retrieve 100 rows of data from each of 6-10 medium tables
170. retrieve 100 rows of data from each of 6-10 large tables
171. retrieve 100 rows of data from each of 6-10 very large tables
172. retrieve 100 rows of data from each of 6-10 combination tables
173. retrieve 1000 rows of data from single small table
174. retrieve 1000 rows of data from single medium table
175. retrieve 1000 rows of data from single large table

176. retrieve 1000 rows of data from single very large table
177. retrieve 1000 rows of data from each of 3-5 small tables
178. retrieve 1000 rows of data from each of 3-5 medium tables
179. retrieve 1000 rows of data from each of 3-5 large tables
180. retrieve 1000 rows of data from each of 3-5 very large tables
181. retrieve 1000 rows of data from each of 3-5 combination tables
182. retrieve 1000 rows of data from each of 6-10 small tables
183. retrieve 1000 rows of data from each of 6-10 medium tables
184. retrieve 1000 rows of data from each of 6-10 large tables
185. retrieve 1000 rows of data from each of 6-10 very large tables
186. retrieve 1000 rows of data from each of 6-10 combination tables

A total of 103 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Twenty-seven of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Transactions

1. insert one row of data into single small table
2. insert one row of data into single medium table
3. insert one row of data into single large table
4. insert one row of data into single very large table
5. insert one row of data into each of 3-5 small tables
6. insert one row of data into each of 3-5 medium tables
7. insert one row of data into each of 3-5 large tables
8. insert one row of data into each of 3-5 very large tables
9. insert one row of data into each of 3-5 combination tables
10. insert one row of data into each of 6-10 small tables
11. insert one row of data into each of 6-10 medium tables
12. insert one row of data into each of 6-10 large tables
13. insert one row of data into each of 6-10 very large tables
14. insert one row of data into each of 6-10 combination tables
15. insert ten rows of data into single small table
16. insert ten rows of data into single medium table
17. insert ten rows of data into single large table
18. insert ten rows of data into single very large table
19. insert ten rows of data into each of 3-5 small tables
20. insert ten rows of data into each of 3-5 medium tables
21. insert ten rows of data into each of 3-5 large tables
22. insert ten rows of data into each of 3-5 very large tables
23. insert ten rows of data into each of 3-5 combination tables
24. insert ten rows of data into each of 6-10 small tables
25. insert ten rows of data into each of 6-10 medium tables
26. insert ten rows of data into each of 6-10 large tables
27. insert ten rows of data into each of 6-10 very large tables
28. insert ten rows of data into each of 6-10 combination tables

29. insert 100 rows of data into single small table
30. insert 100 rows of data into single medium table
31. insert 100 rows of data into single large table
32. insert 100 rows of data into single very large table
33. insert 100 rows of data into each of 3-5 small tables
34. insert 100 rows of data into each of 3-5 medium tables
35. insert 100 rows of data into each of 3-5 large tables
36. insert 100 rows of data into each of 3-5 very large tables
37. insert 100 rows of data into each of 3-5 combination tables
38. insert 100 rows of data into each of 6-10 small tables
39. insert 100 rows of data into each of 6-10 medium tables
40. insert 100 rows of data into each of 6-10 large tables
41. insert 100 rows of data into each of 6-10 very large tables
42. insert 100 rows of data into each of 6-10 combination tables
43. read data in single small table
44. read data in single medium table
45. read data in single large table
46. read data in single very large table
47. read data in 3-5 small tables
48. read data in 3-5 medium tables
49. read data in 3-5 large tables
50. read data in 3-5 very large tables
51. read data in 3-5 combination tables
52. read data in 6-10 small tables
53. read data in 6-10 medium tables
54. read data in 6-10 large tables
55. read data in 6-10 very large tables
56. read data in 6-10 combination tables
57. update one row of data in each of 3-5 small tables
58. update one row of data in each of 3-5 medium tables
59. update one row of data in each of 3-5 large tables
60. update one row of data in each of 3-5 very large tables
61. update one row of data in each of 3-5 combination tables
62. update one row of data in each of 6-10 small tables
63. update one row of data in each of 6-10 medium tables
64. update one row of data in each of 6-10 large tables
65. update one row of data in each of 6-10 very large tables
66. update one row of data in each of 6-10 combination tables
67. update ten rows of data in single small table
68. update ten rows of data in single medium table
69. update ten rows of data in single large table
70. update ten rows of data in single very large table
71. update ten rows of data in each of 3-5 small tables
72. update ten rows of data in each of 3-5 medium tables
73. update ten rows of data in each of 3-5 large tables
74. update ten rows of data in each of 3-5 very large tables
75. update ten rows of data in each of 3-5 combination tables

76. update ten rows of data in each of 6-10 small tables
77. update ten rows of data in each of 6-10 medium tables
78. update ten rows of data in each of 6-10 large tables
79. update ten rows of data in each of 6-10 very large tables
80. update ten rows of data in each of 6-10 combination tables
81. update 100 rows of data in single small table
82. update 100 rows of data in single medium table
83. update 100 rows of data in single large table
84. update 100 rows of data in single very large table
85. update 100 rows of data in each of 3-5 small tables
86. update 100 rows of data in each of 3-5 medium tables
87. update 100 rows of data in each of 3-5 large tables
88. update 100 rows of data in each of 3-5 very large tables
89. update 100 rows of data in each of 3-5 combination tables
90. update 100 rows of data in each of 6-10 small tables
91. update 100 rows of data in each of 6-10 medium tables
92. update 100 rows of data in each of 6-10 large tables
93. update 100 rows of data in each of 6-10 very large tables
94. update 100 rows of data in each of 6-10 combination tables
95. delete one row of data from each of 3-5 small tables
96. delete one row of data from each of 3-5 medium tables
97. delete one row of data from each of 3-5 large tables
98. delete one row of data from each of 3-5 very large tables
99. delete one row of data from each of 3-5 combination tables
100. delete one row of data from each of 6-10 small tables
101. delete one row of data from each of 6-10 medium tables
102. delete one row of data from each of 6-10 large tables
103. delete one row of data from each of 6-10 very large tables
104. delete one row of data from each of 6-10 combination tables
105. delete ten rows of data from single small table
106. delete ten rows of data from single medium table
107. delete ten rows of data from single large table
108. delete ten rows of data from single very large table
109. delete ten rows of data from each of 3-5 small tables
110. delete ten rows of data from each of 3-5 medium tables
111. delete ten rows of data from each of 3-5 large tables
112. delete ten rows of data from each of 3-5 very large tables
113. delete ten rows of data from each of 3-5 combination tables
114. delete ten rows of data from each of 6-10 small tables
115. delete ten rows of data from each of 6-10 medium tables
116. delete ten rows of data from each of 6-10 large tables
117. delete ten rows of data from each of 6-10 very large tables
118. delete ten rows of data from each of 6-10 combination tables
119. delete 100 rows of data from single small table
120. delete 100 rows of data from single medium table
121. delete 100 rows of data from single large table
122. delete 100 rows of data from single very large table

123. delete 100 rows of data from each of 3-5 small tables
124. delete 100 rows of data from each of 3-5 medium tables
125. delete 100 rows of data from each of 3-5 large tables
126. delete 100 rows of data from each of 3-5 very large tables
127. delete 100 rows of data from each of 3-5 combination tables
128. delete 100 rows of data from each of 6-10 small tables
129. delete 100 rows of data from each of 6-10 medium tables
130. delete 100 rows of data from each of 6-10 large tables
131. delete 100 rows of data from each of 6-10 very large tables
132. delete 100 rows of data from each of 6-10 combination tables

A total of 242 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Seventy-two of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

System x-based Test Cases

Tracked System Activity: ERP

1. refresh screen
2. move from one screen to another
3. add a customer
4. update customer data
5. terminate a customer
6. add a vendor
7. update vendor data
8. terminate a vendor
9. add an employee
10. update employee data
11. terminate an employee

A total of 2,394 core systems were tracked with this type of large-scale application. The subject organizations covered a mixture of full financial service companies, others that are solely focused on this market sector and retail organizations that also offer this type of customer product. Two hundred four of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Customer Support

1. log new support call
2. update support call data
3. log completion of support call
4. lookup customer name
5. lookup customer address
6. lookup customer program information
7. lookup customer bill or invoice
8. update customer name

9. update customer address
10. update customer program information
11. update support staff dashboard
12. update supervisor dashboard
13. escalate call priority

A total of 4,622 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Two hundred sixteen of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Retail and Distribution

1. check inventory
2. apply payment
3. calculate discount
4. consolidate shipments
5. create backorder
6. delete suspended transaction
7. search for alternative inventory location
8. search for product by keyword
9. update buy list
10. validate credit

A total of 126 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Forty of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Order Entry

1. retrieve customer data
2. verify customer credit standing
3. add ship-to address
4. retrieve product data
5. insert one detail order line
6. insert ten detail order lines
7. insert 100 detail order lines
8. update one detail order line
9. update ten detail order lines
10. update 100 detail order lines
11. insert one detail return line
12. insert ten detail return lines
13. insert 100 detail return lines
14. update one detail return line
15. update ten detail return lines
16. update 100 detail return lines
17. verify in-stock products

18. lookup product location
19. purge one row of order header information
20. purge ten rows of order header information
21. purge 100 rows of order header information
22. purge one row of order detail information
23. purge ten rows of order detail information
24. purge 100 rows of order detail information

A total of 631 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. Ninety-four of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.

Tracked System Activity: Transactions

1. insert one row of data into single small table
2. insert one row of data into single medium table
3. insert one row of data into single large table
4. insert one row of data into each of 3-5 small tables
5. insert one row of data into each of 3-5 medium tables
6. insert one row of data into each of 3-5 large tables
7. insert one row of data into each of 3-5 combination tables
8. insert one row of data into each of 6-10 small tables
9. insert one row of data into each of 6-10 medium tables
10. insert one row of data into each of 6-10 large tables
11. insert one row of data into each of 6-10 combination tables
12. insert ten rows of data into single small table
13. insert ten rows of data into single medium table
14. insert ten rows of data into single large table
15. insert ten rows of data into each of 3-5 small tables
16. insert ten rows of data into each of 3-5 medium tables
17. insert ten rows of data into each of 3-5 large tables
18. insert ten rows of data into each of 3-5 combination tables
19. insert ten rows of data into each of 6-10 small tables
20. insert ten rows of data into each of 6-10 medium tables
21. insert ten rows of data into each of 6-10 large tables
22. insert ten rows of data into each of 6-10 combination tables
23. insert 100 rows of data into single small table
24. insert 100 rows of data into single medium table
25. insert 100 rows of data into single large table
26. insert 100 rows of data into each of 3-5 small tables
27. insert 100 rows of data into each of 3-5 medium tables
28. insert 100 rows of data into each of 3-5 large tables
29. insert 100 rows of data into each of 3-5 combination tables
30. insert 100 rows of data into each of 6-10 small tables
31. insert 100 rows of data into each of 6-10 medium tables
32. insert 100 rows of data into each of 6-10 large tables

33. insert 100 rows of data into each of 6-10 combination tables
34. insert 1000 rows of data into single small table
35. insert 1000 rows of data into single medium table
36. insert 1000 rows of data into single large table
37. insert 1000 rows of data into each of 3-5 small tables
38. insert 1000 rows of data into each of 3-5 medium tables
39. insert 1000 rows of data into each of 3-5 large tables
40. insert 1000 rows of data into each of 3-5 combination tables
41. insert 1000 rows of data into each of 6-10 small tables
42. insert 1000 rows of data into each of 6-10 medium tables
43. insert 1000 rows of data into each of 6-10 large tables
44. insert 1000 rows of data into each of 6-10 combination tables
45. read data in single small table
46. read data in single medium table
47. read data in single large table
48. read data in 3-5 small tables
49. read data in 3-5 medium tables
50. read data in 3-5 large tables
51. read data in 3-5 combination tables
52. read data in 6-10 small tables
53. read data in 6-10 medium tables
54. read data in 6-10 large tables
55. read data in 6-10 combination tables
56. update one row of data in each of 3-5 small tables
57. update one row of data in each of 3-5 medium tables
58. update one row of data in each of 3-5 large tables
59. update one row of data in each of 3-5 combination tables
60. update one row of data in each of 6-10 small tables
61. update one row of data in each of 6-10 medium tables
62. update one row of data in each of 6-10 large tables
63. update one row of data in each of 6-10 combination tables
64. update ten rows of data in single small table
65. update ten rows of data in single medium table
66. update ten rows of data in single large table
67. update ten rows of data in each of 3-5 small tables
68. update ten rows of data in each of 3-5 medium tables
69. update ten rows of data in each of 3-5 large tables
70. update ten rows of data in each of 3-5 combination tables
71. update ten rows of data in each of 6-10 small tables
72. update ten rows of data in each of 6-10 medium tables
73. update ten rows of data in each of 6-10 large tables
74. update ten rows of data in each of 6-10 combination tables
75. update 100 rows of data in single small table
76. update 100 rows of data in single medium table
77. update 100 rows of data in single large table
78. update 100 rows of data in each of 3-5 small tables
79. update 100 rows of data in each of 3-5 medium tables

80. update 100 rows of data in each of 3-5 large tables
81. update 100 rows of data in each of 3-5 combination tables
82. update 100 rows of data in each of 6-10 small tables
83. update 100 rows of data in each of 6-10 medium tables
84. update 100 rows of data in each of 6-10 large tables
85. update 100 rows of data in each of 6-10 combination tables
86. update 1000 rows of data in single small table
87. update 1000 rows of data in single medium table
88. update 1000 rows of data in single large table
89. update 1000 rows of data in each of 3-5 small tables
90. update 1000 rows of data in each of 3-5 medium tables
91. update 1000 rows of data in each of 3-5 large tables
92. update 1000 rows of data in each of 3-5 combination tables
93. update 1000 rows of data in each of 6-10 small tables
94. update 1000 rows of data in each of 6-10 medium tables
95. update 1000 rows of data in each of 6-10 large tables
96. update 1000 rows of data in each of 6-10 combination tables
97. delete one row of data from each of 3-5 small tables
98. delete one row of data from each of 3-5 medium tables
99. delete one row of data from each of 3-5 large tables
100. delete one row of data from each of 3-5 combination tables
101. delete one row of data from each of 6-10 small tables
102. delete one row of data from each of 6-10 medium tables
103. delete one row of data from each of 6-10 large tables
104. delete one row of data from each of 6-10 combination tables
105. delete ten rows of data from single small table
106. delete ten rows of data from single medium table
107. delete ten rows of data from single large table
108. delete ten rows of data from each of 3-5 small tables
109. delete ten rows of data from each of 3-5 medium tables
110. delete ten rows of data from each of 3-5 large tables
111. delete ten rows of data from each of 3-5 combination tables
112. delete ten rows of data from each of 6-10 small tables
113. delete ten rows of data from each of 6-10 medium tables
114. delete ten rows of data from each of 6-10 large tables
115. delete ten rows of data from each of 6-10 combination tables
116. delete 100 rows of data from single small table
117. delete 100 rows of data from single medium table
118. delete 100 rows of data from single large table
119. delete 100 rows of data from each of 3-5 small tables
120. delete 100 rows of data from each of 3-5 medium tables
121. delete 100 rows of data from each of 3-5 large tables
122. delete 100 rows of data from each of 3-5 large tables
123. delete 100 rows of data from each of 3-5 combination tables
124. delete 100 rows of data from each of 6-10 small tables
125. delete 100 rows of data from each of 6-10 medium tables
126. delete 100 rows of data from each of 6-10 large tables

127. delete 100 rows of data from each of 6-10 large tables
128. delete 100 rows of data from each of 6-10 combination tables
129. delete 1000 rows of data from single small table
130. delete 1000 rows of data from single medium table
131. delete 1000 rows of data from single large table
132. delete 1000 rows of data from each of 3-5 small tables
133. delete 1000 rows of data from each of 3-5 medium tables
134. delete 1000 rows of data from each of 3-5 large tables
135. delete 1000 rows of data from each of 3-5 very large tables
136. delete 1000 rows of data from each of 3-5 combination tables
137. delete 1000 rows of data from each of 6-10 small tables
138. delete 1000 rows of data from each of 6-10 medium tables
139. delete 1000 rows of data from each of 6-10 large tables
140. delete 1000 rows of data from each of 6-10 very large tables
141. delete 1000 rows of data from each of 6-10 combination tables

A total of 967 core systems were tracked with this type of application. The subject organizations covered a mixture of industries and market sectors. One hundred thirty-five of these organizations ran full parallel efforts on different equipment and DBMS that allowed close detailed comparisons. The remainder ran partial comparisons that provided significant input.