2005 RDBMS Software Cost of Ownership Study

Providing the financial basis for choosing a strategic database supplier

Abstract   In this study we provide a detailed cost of ownership analysis of the three market leading Relational Database Management System (RDBMS) software packages available today. We evaluate the database software offered by the industry's premiere vendors (IBM, Microsoft, and Oracle) and demonstrate that acting on a buying decision, based on list price of the license alone, can result in a significantly higher overall cost of ownership in a relatively short period of time. We define three primary, cost of ownership metrics; *Comparative Cost of Ownership (CCO), Total Cost of Ownership (TCO),* and *Probable Cost of Ownership (PCO).* Using our tools, methods, and study results, it is possible to compare software costs to help cut through the fog of the different pricing and licensing schemes offered by each of the vendors. You will be able to make intelligent, financial based decisions on not only your next specific RDBMS license purchase but also the strategic decision of who should be your preferred RDBMS supplier.

August 2005

Price US$250.00

2005 RDBMS Software Total Cost of Ownership Study

Providing the financial basis for choosing a strategic database supplier

Executive Summary

The purchase of software licenses and support can be a daunting and confusing experience due to the complexity of the different pricing models and the different terms and conditions offered by every software vendors. To aid our clients in cutting through the fog of software terms and conditions, Market Magic Research (MMR) has developed a methodology and a number of metrics for comparing the costs associated with each of today's top three Relational Database Management Systems (RDBMS) software vendors: IBM, Microsoft and Oracle.

We developed three cost comparison metrics to aid in decision making. We feel that these three key metrics are of great help in comparing true software costs. Each metric has strengths and weaknesses but combined they provide a complete picture. The three metrics we developed were:

- *Comparative Cost of Ownership (CCO)* compares the cost of software license and software support services which you will purchase from a vendor over a fixed period of time. CCO focuses on the list price for software licenses and services. CCO is helpful in negotiating better discounts for a specific project from your potential RDBMS vendors. Our CCO tool automatically generates cost data for each RDBMS configuration that you input.

- *Total Cost of Ownership (TCO)* builds upon CCO and to give a more complete idea of what it will cost to live with different RDBMS options that you face. Like CCO, TCO looks at a individual RDBMS server configuration. But, TCO includes other key costs and credits such as DBA costs, training costs, vendor specific discount credits, and other factors which you might wish to include. Our CCO tool has extra fields in it to aid you in calculating TCO.

- *Probable Cost of Ownership (PCO)* builds upon TCO and is designed to help you with selecting a strategic, preferred RDBMS vendor. For any given configuration, a particular RDBMS vendor might look great. But, the vendor that looks great for that first configuration might be the wrong vendor when all of the possible configurations are viewed. The PCO results presented in this study was based upon survey data collected from IT professionals across different enterprises. For the complete range of enterprise

RDBMS servers we surveyed, we calculated TCO data and compiled it as a Probable Cost of Ownership or (PCO).

As described in the definitions above, our CCO tool will help you find the best value for a specific project, it will do little to help you with making the strategic decision of who to select for your preferred RDBMS supplier. To aid with strategic decision making, we have developed our PCO methodology and have published the results in this study. To calculate PCO, we surveyed IT managers from enterprises with annual revenues greater than $100 million to find out the requirements from both their most important and their most recent RDBMS powered applications. We used the survey to get a sense of what the range of real-world RDBMS servers look like since it is system size that will drive the cost of software licensing and support.

Finally, we combined the survey data with our CCO tool to calculate the Total Cost of Ownership (TCO) for each of the range of surveyed configurations and cost. We combined the TCO data as a statistical distribution such that we could determine a Probable Cost of Ownership™ (PCO) for each of our three target vendors. We believe that PCO is the best indication available of what that next, unknown and arbitrary project will cost in terms of the RDBMS's server software. Further, we believe that PCO is the ideal tool to help with the strategic decisions of selecting preferred RDBMS vendors since it looks at such a wide range of configurations.

When we calculated PCO for our three target RDBMS vendors, we encountered a major problem with software availability from Microsoft. Specifically, in 58.1% of the real-world configurations that came from our survey data, Microsoft was unable to satisfy the requirements of the RDBMS server configuration. Microsoft SQL Server's inability to meet the requirements of the system was due to two factors: Microsoft does not support any operating system other than their own Windows and they do not support enterprise level disaster recovery[1]. It is not possible to calculate meaningful PCO statistics when one of the comparative RDBMS packages is excluded. To accommodate the shortcomings of Microsoft SQL Server yet still provide meaningful statistics, we restricted our final PCO analysis to configurations which are based only upon Microsoft Windows and that did not require disaster recovery such that Microsoft SQL Server would always be included. Thus, the final set of PCO statistics must be viewed in light of this major concession.

Throughout the study it was discovered that, although each of the three vendors had both strengths and weaknesses, some consistent trends emerged. IBM, with their DB2 offering, consistently offered the best overall pricing for a combination of both licenses and support. Microsoft's SQL Server offered the best list licensing prices but fell short in

---

[1] It should be noted that our evaluation was based upon Microsoft SQL Server 2000. At the time of this study, Microsoft had announced, but had not released, SQL Server 2005. Microsoft has stated that the new version of the SQL Server 2005 will address customer disaster recovery needs. Further, preliminary pricing indicates that there will be an increase in SQL Server license prices but this has not been finalized as of our publication date. The release of Microsoft SQL Server 2005 will be the subject of a future study.

the ability to support all required configurations and in the area of telephone support. As a result, the Microsoft total package is competitive, but on average, it is slightly more expensive than IBM's DB2. Finally, Oracle simply proved to be not price competitive with their Oracle 10g offering even though it is competitive on availability, features, and functions.

Although this current study goes into considerable greater detail than our previous study of 2003, the results of both studies are very consistent. We recommend that you use the CCO tool to help you decide on the RDBMS supplier for your next database powered application. We recommend that you use the PCO analysis to help you decide on your strategic, preferred RDBMS suppliers particularly if you plan upon rationalizing the number of your RDBMS suppliers. Finally, regardless of which RDBMS supplier that you choose, we recommend that you use this study to help negotiate the best possible discounts and terms in your next RDBMS software purchase.

Introduction

All database software vendors claim to offer the best set of features, best performance, best overall value, the most responsive user support, and the highest productivity available for the money spent. The most confusing aspect of these claims is that each company can find data to support their claims, so reading the literature, or checking a website, will provide lots of information and very little in the way of discriminatory data. In the world of sales, this sort of 'attribute stacking' is to be expected, and buyers would be surprised to find that a database vendor was not putting their best foot forward. But such selling does not benefit either the buyer or seller beyond a certain point of the selling process. With large amounts of money at risk, facts, not opinions, become critical to the decision making process, and when facts are not available, the decision process can break down. It is imperative that a prospective buyer of an enterprise RDBMS server get facts and data that can be analyzed and parsed based on the buyer's needs. Third party analysis is always a useful tool for the discriminating buyer. If done properly, an objective third party can take disparate data points and compare them in a standard fashion.

In order to fairly and objectively compare the attributes of database software licensing and support, Market Magic Research (MMR) compared and analyzed the three major Relational Database Management System (RDBMS) software providers; IBM, Microsoft and Oracle[2]. Specifically, the analysis compared the software licensing and support costs for IBM's DB2 Universal Database v8.2, Microsoft's SQL Server 2000, and Oracle's 10g.

There were two goals for this study. The first goal was to build a tactical Comparative Cost of Ownership (CCO) tool that could be used to compare the cost of deploying and supporting your next RDBMS server from the top three database vendors on the market. The CCO metric, as opposed to a Total Cost of Ownership (TCO) metric, was used because analysis of CCO focuses on limited, quantifiable data points, i.e., price of a license, characteristics of licenses, limitations of licenses, number of users, etc. A TCO analysis, by comparison, has a much broader, less quantifiable focus, which is to say a TCO analysis includes data that may have more of a subjective, rather than objective, basis. For example, all software vendors make claims on productivity and hardware efficiency that are used to show their software in a better light. Some vendors argue that, while their software license is more expensive initially, it will be cheaper in the long run since less staff will be needed, less expensive hardware can be used, or even that less hardware will be needed to successfully run the application. Database software is becoming less differentiated over time, and though vendors are regularly adding new features, most applications that require an RDBMS do not readily exploit features that are not standard. In fact, the greatest areas of quantifiable difference for assessing the best database for an application might just be both the CCO and TCO of the RDBMS software. However, without factoring elements such as staffing and hardware costs, this analysis cannot accurately be deemed a TCO study. To this end, our CCO tool also makes

---

[2] Throughout this whitepaper we will list all three database vendors in alphabetical order. This order is completely arbitrary.

provision for the inclusion of these other factors. Thus we are able to include the more subjective elements of TCO along side the CCO tools. The data used in the CCO tool was culled from the publicly available web sites for each RDBMS vendor. We combined each vendor's licensing and support pricing and additional information to calculate the costs used to determine both CCO and TCO in a single tool. The CCO tool that we have developed is a powerful tool that you can use to help guide you in selecting the vendor that will provide the best value in RDBMS for your project.

The second goal of the study was much more strategic. We wanted to aid readers in making a strategic decision on a preferred RDBMS supplier. When we surveyed IT managers from enterprises with annual revenues greater than $100 million, they overwhelmingly told us that they wanted to reduce the number of RDBMS suppliers that they deal with. Ideally, they would like to reduce the number of RDBMS suppliers to one but they do not feel that it is possible. Most enterprises have not yet decided upon which RDBMS suppliers they will keep, but the short list is generally down to IBM, Microsoft, and/or Oracle. When we saw this in our survey data, we knew that we needed to help with this decision making process. Although our CCO tool is great for making a specific tactical decision on which RDBMS will give you the best value on your next database powered application, it does little to help with making the strategic decision of a preferred supplier. To help with the strategic decision, we focused on our Probable Cost of Ownership™ (PCO) methodology. The concept behind PCO is to look at the widest range of possible enterprise RDBMS server configurations using our CCO tool which calculates both CCO and TCO. We then compile the data to give an idea of the probability distribution over this very wide range of real-world configurations. Using the PCO technique, we can tell you which vendor will most likely be the best value on your next, arbitrary database powered application. Thus, PCO is a much better method than single project CCO or TCO for helping make the decision on which RDBMS suppliers to keep in your portfolio of preferred suppliers.

For the purposes of this study, IBM DB2, Microsoft SQL Server, and Oracle were the only databases examined. This decision was made because our survey told us that these three vendors have the most number of systems installed in enterprise situations and that these three vendors are expected to continue to lead the market. There are other database application providers, but MMR believes that none will challenge these three leaders, and their inclusion in this study would provide little additional value for those looking for strategic RDBMS vendors.

Methodology

The key to this, or any, comparative analysis is to ensure that like elements of the database software are compared. To ensure that fair and accurate comparisons were made, we used a multi-step approach:

1. For each target RDBMS software vendor, we collected their U.S. dollar price list (and standard licensing terms) as published on their web sites as of the 9th of June,

2005. We only considered products that were available for shipment and for which a definitive price was published. For instance, Microsoft had announced that SQL Server 2005 would soon be released and had announced tentative prices; however, we did not consider this since it could not be ordered and the pricing had not yet been finalized.

2. We sampled and surveyed 120 IT managers from enterprises with annual revenues greater than $100 million. We asked them a number of questions about their RDBMS experiences and current systems. We also asked them about their RDBMS strategy moving forward; specifically, if they had any plans to reduce their number of RDBMS suppliers. Of those sampled that also had experience with all three of our target vendors, we allowed them to answer additional questions which then compared their experiences on a wide range of subjects such as the configuration of their last RDBMS project, their perceptions on ease-of-use, typical discounts that they receive from each of the three target vendors, and their support needs. The second part of the survey was used to determine how our three target RDBMS systems compare on price, ease-of-use, and vendor discounting.

3. Using the data from our web survey of vendor prices, we built our Comparative Cost of Ownership (CCO) tool[3]. This Excel spreadsheet consists of an input questionnaire which is used to describe the requirements for a specific RDBMS server. Once the data is input, the tool uses the prices from the vendor's web site along with our interpretation of the licensing terms and conditions to calculate the list price that would be offered from each of the vendors. The tool also supports the input of any additional vendor discounts, DBA manpower requirements, and spaces for any other line item costs or credits that you would like to consider in your analysis. Thus, by using the additional line items, a Total Cost of Ownership (TCO) analysis can be performed.

4. Finally, we used an advanced Monte Carlo simulation tool to re-sample data from our survey and input it into our CCO tool; thus, we were able to combine the data from all three previous steps to get a total picture of what we call Probable Cost of Ownership (PCO). In our simulation, we ran 10,000 iterations of real-world configurations and calculated the TCO for each simulated project for each of our three target vendors. In our simulations we considered list price, standard discounts, DBA costs, and additional vendor discounts to give a complete TCO picture. It should be noted that to calculate PCO in a meaningful fashion, we needed to exclude several scenarios which Microsoft SQL Server simply could not support. Specifically, Microsoft SQL Server only runs on Microsoft's own Windows Operating System; thus we needed to exclude the 46.0% of the configurations in which either UNIX or Linux was used. Additionally, SQL Server

---

[3] A licensed copy of the Microsoft Excel based CCO model is available for use in your own decision making scenarios with the purchase of this MMR Research Note.

2000 does not support one type of high availability (which we called disaster recovery[4]) and thus was excluded for an additional 12.1% of the configurations. In total, SQL Server 2000 did not support 58.1% of the required configurations. Removing Microsoft SQL Server from so many configurations created a problem: how to calculate meaningful statistics when one of the target vendors is excluded. To deal with this problem, we forced the Monte Carlo Simulation to only consider configurations which ran on Microsoft Windows and did not require disaster recovery. Thus, we were able to calculate a meaningful PCO statistics but it must be considered in light of the restrictions created to include SQL Server 2000.

Vendor Licensing, Packaging, and Pricing

As stated earlier, the three database software packages studied were IBM Universal Database v8.2, Microsoft SQL Server 2000, and Oracle 10g. Figure 1, *Vendor Licensing and Support Options*, illustrates different options that each vendor offers in the licensing, upgrade services, and support services for their database software. The table also captures the licensing metrics and restrictions that each vendor has for each edition of the RDBMS software which they provide. The CCO model that we developed considers all of these differences in calculating the price of the database option from each vendor.

---

[4] Microsoft has announced that SQL Server 2005 will support disaster recovery but it was not available at the time of our study. This will be re-evaluated in a future study.

	IBM	MICROSOFT	ORACLE
Database Name	DB2 Universal Database v8.2	SQL Server 2000	Oracle 10g
Entry Edition	Express Edition	Workgroup Edition	Standard Edition One
Entry Edition Restrictions	Max 2 CPU's, 4GB memory, and no clustering	Max 4 CPU's, 2 GB memory, no clustering, no log shipping	Max 2 CPU's, no scale out clustering, no clustering, no disaster recovery, no Internet use
Mid-Range Edition	Workgroup Edition	Standard Edition	Standard Edition
Mid-Range Edition Restrictions	Max 4 CPU's and no clustering	Max 4 CPU's, 2GB memory, no clustering, and no log shipping	Max 4 CPU's and not disaster recovery
High-end Edition	Enterprise Server Edition	Enterprise Edition	Enterprise Edition
High-end Edition Restrictions	None	No Disaster Recovery	None
OS Availability[5]	Linux, UNIX and MS Windows	MS Windows Only	UNIX, Linux and MS Windows
User Licensing Model	Named User for Express, Concurrent User for Workgroup	Client Access License (CAL) required per client device	Named User
User License Minimums Restrictions	None	5 or 10 CAL Packs for Standard Edition, 25 CAL Packs for Enterprise Edition	5 Named Users per CPU for Standard Edition, 25 Named Users per CPU for Enterprise Edition
CPU Licensing	Yes	Yes	Yes
Multi-Core CPU Policy	Complicated, but generally a dual-core chip counts as one CPU	A dual-core CPU counts as a single CPU	A dual-core CPU counts as two CPU's[6]
Tuning Software	Included	Included	Tuning Pack
Diagnostics Software	Included	Included	Diagnostics Pack
Passive Failover	Single CPU License or Single Initial User Required if production system is User Based Licensing	No License Required	Full Licensing Required
Update Service	Upgrade Protection Subscription with first year included in the license fee	Software Assurance Subscription	Update Service Subscription
Telephone Support	Unlimited 7x24 included with Upgrade Protection	Per Incident supplied by Microsoft or Microsoft Authorized Premier Support (MAPS)	Unlimited 7x24 Telephone Support Subscription

Figure 1 – Vendor Licensing and Support Options

---

[5] The scenarios used in the analysis performed by MMR used Microsoft Windows to ensure that SQL Server was included in all comparisons. However, if the reader chooses to use the model for their own needs, they are free to choose operating systems other than Microsoft Windows.

[6] After the data for the study was collected on 9 June 2005, Oracle announced a change in multi-core policy; specifically that a dual-core CPU's would be counted as 1.75 of a CPU for licensing purposes We did not include this in our study but will include this in our next update. Because of the limited availability of dual-core systems, we do not see this as a major issue today.

Figure 2, *Basic Software Licensing Costs*, illustrates the different raw price points that each vendor uses when they license their software. The table includes three different licensing metrics; named user, concurrent user, and CPU.

Named User is an individual employee within the enterprise; that is, each unique employee who will use the RDBMS powered application will require a named user license. In the case of Microsoft, they use a Client Access License (CAL) instead of a named user. For Microsoft, a CAL is required for each computer used to access the RDBMS server. For example, if an employee accesses an application using a Desktop PC and a PDA, then two CAL's will be required; one for each client device.

Concurrent user metric requires a license for each user that is actively logged on and using the system at any one time. Thus, if only a quarter of the 100 named users are active at any one time, then only 25 concurrent user licenses will be required.

The CPU metric requires a license for each CPU that is installed into the server that is hosting the RDBMS software. Special rules apply may apply when the CPU is a multi-core CPU. A Multi-core is a single chip that has more than one physical CPU located inside. Multi-core CPU's are just hitting the market and are only starting to make a big impact on servers and their use will only increase over time. Each vendor has taken a different approach to multi-core CPU licensing policy. Microsoft has announced that a multi-core Chip will be counted as a single CPU. Oracle has announced that each core in the CPU will be counted as an individual CPU but is in the process of refining this. IBM has a very confusing policy that depends upon the make of the chip as well as the OS that is running on the chip, but for the most part, they count each chip as a CPU.

Our CCO model and tool considers each pricing and metric options available when calculating the price of the database license and support from each vendor. The tool then selects the licensing metrics that results in the lowest possible price from each vendor.

License Metric	IBM[7]	Microsoft	Oracle
Per Named User	Express $124	Workgroup $148	Standard ONE $149
(or Client Access License – CAL in the case of Microsoft)		Standard $179	Standard $300
		Enterprise $262	Enterprise $800
			NOTE: Minimums apply
Per Concurrent User	Workgroup $311	Not available	Not available
Per CPU (or Core depending upon the Vendor's terms)	Express $4,874	Workgroup $3,899	Standard ONE $3,899
	Workgroup $9,375	Standard $5,999	Standard $15,000
	Enterprise $33,125	Enterprise $24,999	Enterprise $40,000

Figure 2 – Basic Software Licensing Costs

---

[7] IBM license prices include the first year of software maintenance and telephone support and thus they include more than just license.

Each of the three vendors offers a different software support package. Software support consists of two services; software update service or maintenance and telephone support service. The method of pricing these services clearly distinguishes one vendor from another.

In the case of Oracle, the *Update Subscription Service* package is offered to all Oracle customers. The price of the update service is 15% of the software license list price per year and the service provides license to use any software corrections and/or upgrades that become available during the subscription term. However, Oracle's *Update Subscription Service* does not include telephone support – this is a separate service. Oracle's 7x24, unlimited *Telephone Support Service* is an additional 7% per year of the software license list price. Thus, a total of 22% of the software license list price per year buys the Oracle customer software update and telephone support services. Although each service is purchased separately from Oracle there are some restrictions; first, you can not purchase *Telephone Support Service* without first purchasing *Update Subscription Service*, and second, we were not able to find Update Subscription Service in the price list by itself.

Like Oracle, IBM offers its DB2 UDB customers an update service called the *Upgrade Protection Subscription*. IBM sells this support service in a rather complicated and confusing way. First, they do not present it as a percentage of the license price even though that is clearly the model used for the pricing. Second, they include the price for the first year in the license list price. The effective percentage charged by IBM is 20% of the software license list price per year with the first year included in the software license price. This IBM service provides access to software upgrades as they become available. Unlike Oracle, IBM provides access to software corrections over the web to anyone who has licensed DB2 even if they did not purchase *Upgrade Protection Subscription*. Also, unlike Oracle, IBM's 7x24, unlimited telephone support is included in the *Upgrade Protection Subscription* package. Telephone support, because it is part of the IBM support package, cannot be purchased separately.

Microsoft also offers a software upgrade service for its SQL Server 2000 users. This service is called *Software Assurance* and is priced at 26% of the software license list price per year. This package provides a license to use any new versions of SQL Server that are released while on contract, but the service does not provide telephone support. Like IBM, Microsoft makes software corrections freely available on their web site for use by any licensed customer. Microsoft offers telephone support on a per incident basis or via its *Microsoft Authorized Premier Support (MAPS)* Partners.

Microsoft's telephone support is different from the telephone support provided by IBM and Oracle. Microsoft telephone support not offered on an unlimited basis but rather is provided on a 'per-incident' basis, with customers purchasing support in 'incident packs.' When purchased directly from Microsoft, the customer pays $1,950 per year which includes 12 non-defect incidents. A non-defect incident is when you place a support call for a situation that was not caused by a previously reported defect. When the incident is deemed a defect incident, the incident is not consumed from the number of incidents

allotted in the 'incident pack'; that is, if the problem is a new, previously unreported software defect. After you consume your allotted 12 incidents per year, you must purchase additional incidents. These are priced at $99 for an email incident, $245 per telephone incident. Telephone incidents can be purchased in 5 telephone incident packs for $1,225. You can also purchase Microsoft SQL Server telephone support contracts from third parties called *Microsoft Authorized Premier Support (MAPS) Partners*. The two largest MAPS Partners that provide international support coverage are HP and IBM.

We have heard from several third parties that Microsoft is willing and able to enter into custom contracts to provide unlimited, 7x24 telephone support; however, we have been unable to find any pricing for such an option. If you require this option from Microsoft, you should ask your Microsoft representative if they would consider this.

Because Microsoft's telephone support options are so different than Oracle and IBM, we needed to make some provisions in the CCO tool. To do this, we provided space to detail the number of support calls that you believe that you will make each month. We then used the standard Microsoft pricing detailed above to calculate the cost for this support. For the purpose of our PCO study, we asked in the survey the number of calls made per month to each of the vendors. We found the result to be an exponential distribution and used the statistics of this distribution to create a random variable that drove the simulation.

Figure 3, *Vendor Support Terms*, provides a quick summary of the software and telephone support offerings of the three vendors.

DATABASE	SOFTWARE SUPPORT	TELEPHONE SUPPORT	COMMENTS
IBM DB2 UDB v8.2	20% per year first year included in the license	Included	Based on annual license fees
Microsoft SQL 2000	25% per year	$1,950 per year for first 12 incidents $99 per email incident $245 per telephone incident $1,225 per telephone incident 5 pack	Software support based on annual license fee
Oracle 10g	15% per year	7% per year	Based on annual license fees

Figure 3 – Vendor Support Terms

2005 RDBMS Survey

In order to help understand the issues facing IT managers about RDBMS decisions and to fuel our PCO calculations, we needed data. To get this data, we sampled 120 IT managers from U.S. enterprises with annual revenues greater than $100 million. We did not limit our survey to geography nor did we limit it to an industry.

Our double-blind[8] 2005 RDBMS survey was conducted using a survey web site and was done in three steps.

The first step was screening. During screening we validated that our subject was both an IT professional and from an enterprise of qualifying size.

The next step asked each of the 120 qualified subjects about their experience and strategy with all RDBMS suppliers that they deal with including our three targets. We wanted to understand from them which RDBMS vendors they use, which they saw as strategic, whether they want to reduce the number of vendors, and what features they look for from their RDBMS software.

The third part of the survey was limited to the 63 surveyed who had experience with all three of the target RDBMS packages: IBM DB2, Microsoft SQL Server, and Oracle. In

---

[8] Double-blind refers to the fact that person completing the survey did not know who the data was being collected for and that we did not given access to the name or the company of the person completing the survey. The name and company of the person completing the survey was managed by a third-party survey company.

this part of the survey we wanted to understand the relative experiences and perceptions about all three vendors including the quantity and level of telephone support required, the type and size of RDBMS servers being required, the relative ease-of-use for the different software packages, and the level of additional discounts that they have negotiated on their purchases from each of the three vendors.

Figures 4, 5, and 6 illustrate the size of the enterprises in terms of annual revenue, number of employees, and number of IT employees that the 120 IT professionals surveyed worked for. We did not allow IT professionals from smaller enterprises (with revenue less than $100 million) to complete the survey.

Company Size (Revenue)
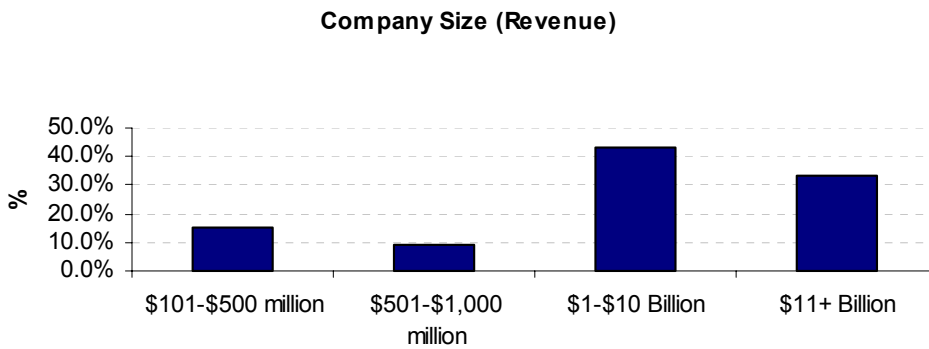
Figure 4 – Company Size Based Upon Annual Revenue

Company Size (Employees)
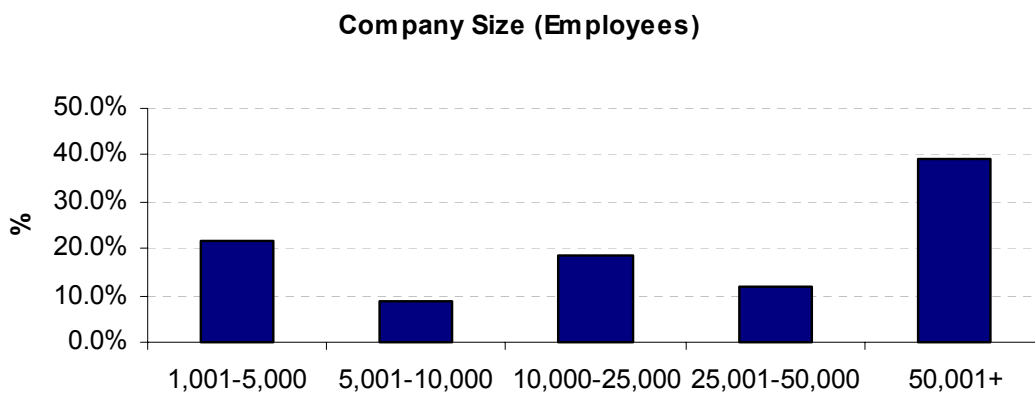
Figure 5 – Company Size based upon Number of Employees

Company Size (IT Employees)
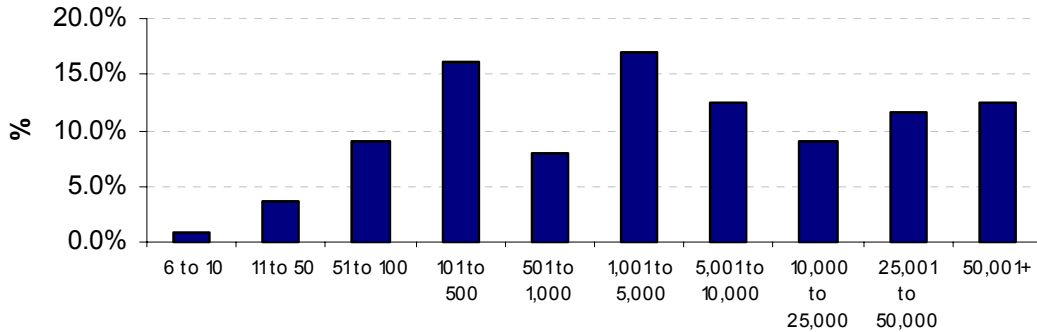
Figure 6 – Company Size Based upon Number of IT Employees

The question in our survey focused on the overall experiences and perceptions that IT professionals have of all RDBMS systems in the market. Figure 7 shows the different makes of RDBMS's installed with our sampled audience. The three most used RDBMS's were Oracle (95.5%), Microsoft SQL Server (92.0%), and IBM DB2 UDB (59.8%). The next most commonly used RDBMS was Sybase with 31.3%. This finding validated our choice of the three market leaders being IBM DB2 UDB, Microsoft SQL Server, and Oracle.

Databases in Use within Companies
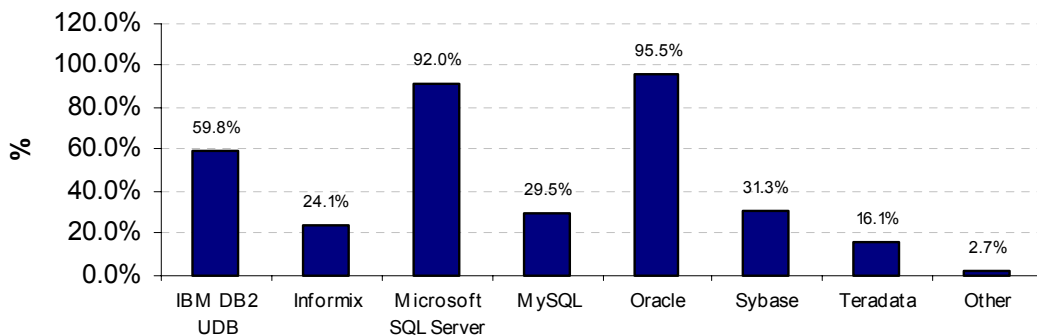
Figure 7 – Which RDBMS's Are in Use

Fifty-three percent (53%) of IT professionals surveyed and screened told us that they had a strategy to reduce the number of RDBMS vendors they purchase from. They also told us that they would ideally like to rationalize to a single preferred RDBMS vendor; but, they do not believe that this is achievable. We asked which RDBMS's they have decided

to keep. Although 18% have not yet made this decision, most are planning to keep Oracle (30%), IBM DB2 (21%), or Microsoft SQL Server (19%) The result is in Figure 8.
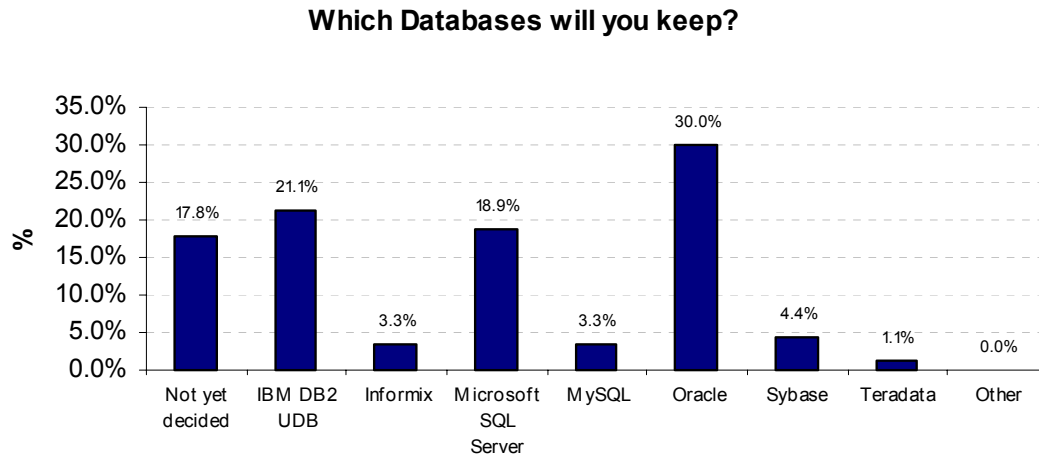
Which Databases will you keep?

Figure 8 – Which RDBMS's Will You Keep

The rest of our survey questions were limited to those IT professional in the sample that had experience (*i.e.,* have implemented) all three of our target RDBMS's: IBM DB2 UDB, Microsoft SQL Server, and Oracle. We asked questions in three forms. First, we asked about their overall experience with each of the target RDBMS's. Next, we asked what their experience was with each target RDBMS in their most important application. Finally, we asked what their experience was with each target RDBMS in their most recently implemented application.

Our survey told us that 54% of all RDBMS systems are being implemented on Microsoft Windows, 41% on UNIX, and 5% on Linux. Further, Linux is the fastest growing Operating System of choice for RDBMS's and will be a platform that we believe that many enterprises will exploit in the future[9].

When asked about the level of support required from the three target RDBMS vendors, we were told that 64.5% of the time they require 7x24 hour support from their RDBMS vendor. Figure 9 illustrates the support requirement for all three RDBMS's while Figure 10 illustrates the complete support requirement for each of our three target vendors.

---

[9] Other independent studies have suggested that the growth in the number of Linux RDBMS servers is growing at a rate in excess of 110% per annum.
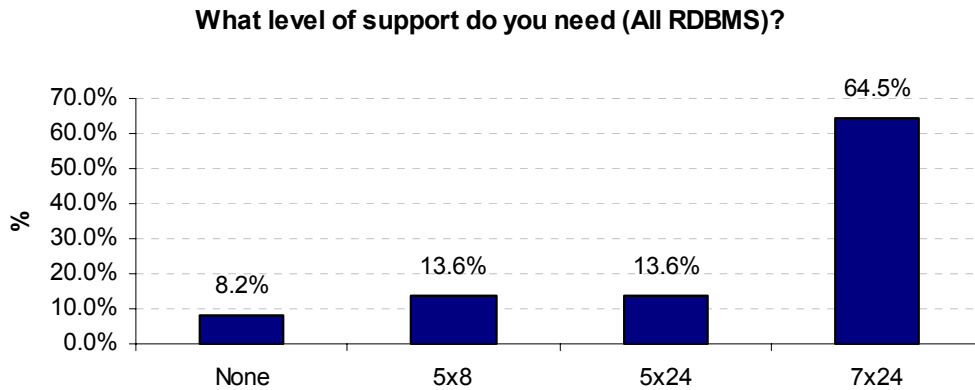
What level of support do you need (All RDBMS)?

Figure 9 – RDBMS Vendor Support Requirements
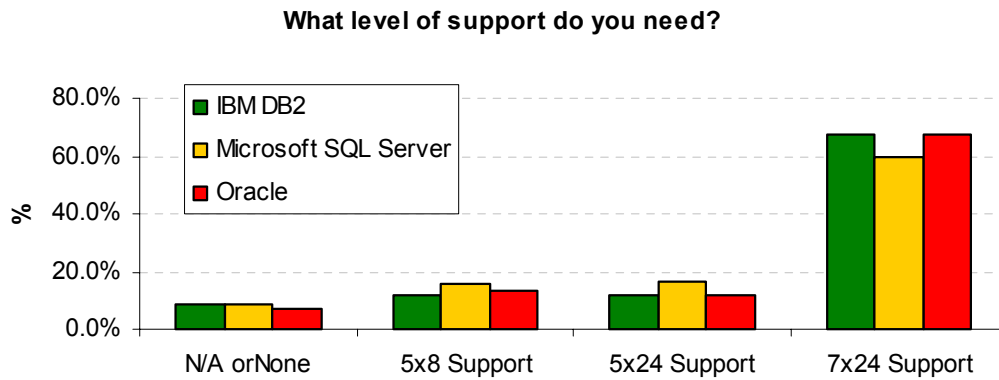
What level of support do you need?

Figure 10 – RDBMS Vendor Support Requirements by Vendor

Because Microsoft does not provide a published unlimited telephone support option while IBM and Oracle do, we felt that it was important to understand the number of telephone support incidents that our sampled users would require. As expected the survey data appears to be exponentially distributed with an average value of 5.7 calls per month. This data was then used to model the Microsoft support calls in our Probable Cost of Ownership (PCO) calculations. Figure 11 shows the distribution for all RDBMS's while Figure 12 shows the data by RDBMS vendors.
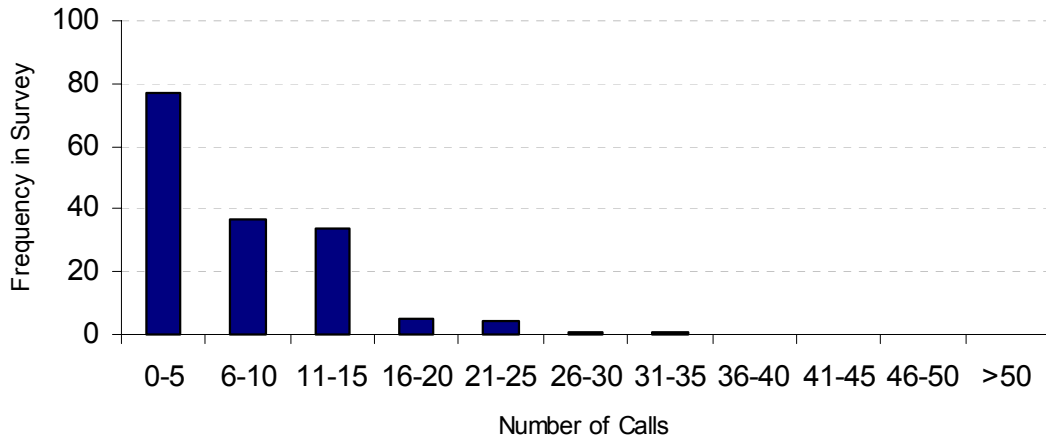
Support Calls per Month

Figure 11 – Telephone Support Incidents per Month (All Vendors)

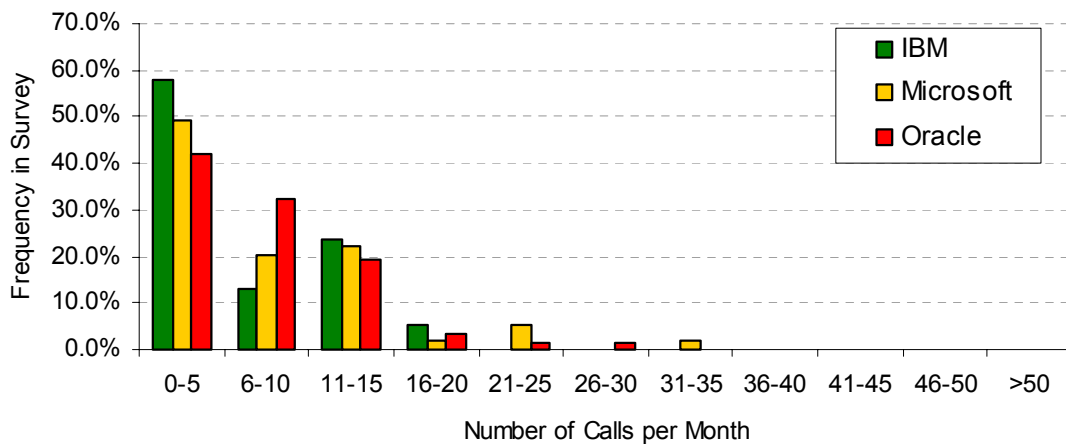## Distribution of Support Calls per Month (by Vendor)

Figure 12 – Telephone Support Incidents per Month by Vendor

We asked about the type of applications that are being built upon RDBMS servers. We broke this into four categories; *On-Line Transaction Processing (OLTP)*, *Analytics* or Business Intelligence, *Web* or Portal, and finally a catch-all of *Mixed*. We wanted to understand the probability that any given application would fall into this type of application category. We felt that this is important because the type of application heavily influences the RDBMS system sizing and requirements. We found that the largest number of applications

currently being built are Web (39%), followed by Analytic (22%), OLTP (21%), and Mixed (18%).

Figure 13 shows the High Availability requirements by application type. We found that 40% of the time this meant having a cluster for failover, 30% of the time this meant having a backup system that relied on log shipping, and 30% of the time this meant highly robust and rapid disaster recovery systems such as IBM's *High Availability Data Recovery (HADR)* or Oracle's *Data Guard.* Microsoft does not currently have a compatible offering for SQL Server but claims that it will be available in SQL Server 2005.
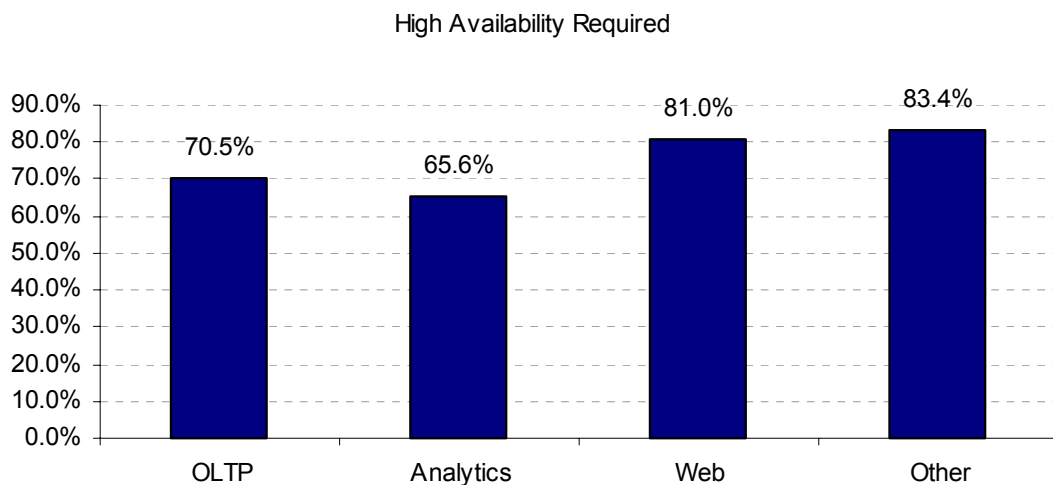
High Availability Required

Figure 13 – High Availability Requirement by Application Type

We also wanted to understand what percentages of applications are built in-house as custom applications rather than being purchased as *Commercial-off-the-Shelf (COTS)* applications. Figure 14 shows the percentage of in-house applications by type of application. We see that most Web applications are being built in-house but that the majority of other applications are being purchased. We found that the custom built applications required a higher level of effort to manage the RDBMS servers; this was not a surprise.
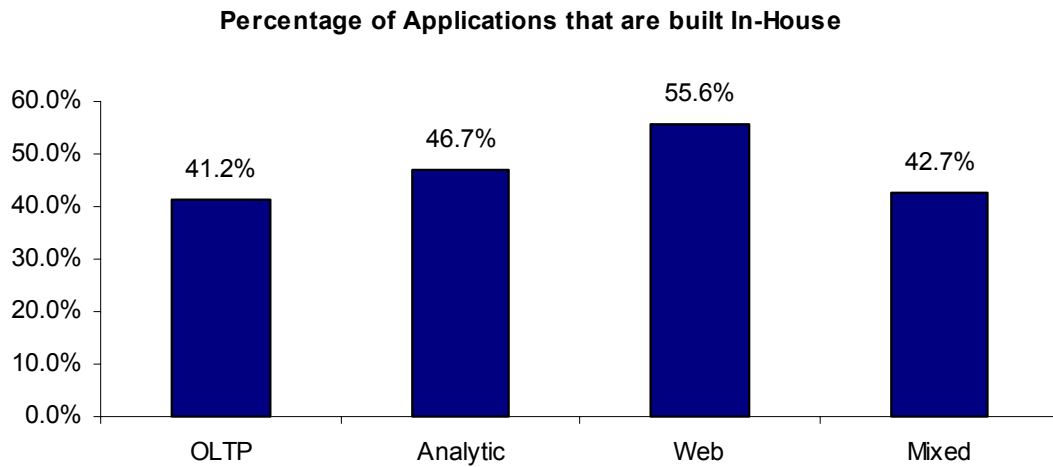
Percentage of Applications that are built In-House

Figure 14 – Percentage of Applications that are built In-House

One of the claims that software vendors always seem to make is that their software is easier to use than their competitor's. We have always been skeptical of such claims and decided that we would validate our skepticism based upon survey data. We asked our sampled IT professionals about their experience on the relative level of effort to perform common DBA tasks on each of the three target RDBMS packages. Although there is a very slight bias towards IBM DB2 UDB and Microsoft SQL Server in the survey data, we found that it is not as significant as the vendors' claims. Figure 15 shows the results of the ease of use survey for which we asked for all of the DBA tasks listed below:

- Logical database design

- Physical database design

- Database installation

- Database configuration

- Database upgrades

- Schema changes and management

- System performance management and tuning

- SQL (i.e., query) tuning

- Data replication

- Database re-organization

- Data Extract, Transformation, and Load (ETL)

- End user database support

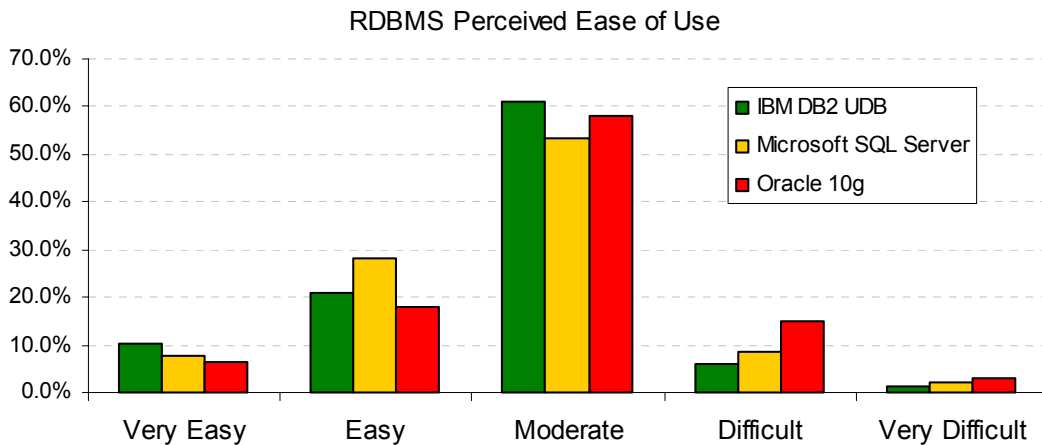- Application tuning

- Application support

RDBMS Perceived Ease of Use

Figure 15 – RDBMS Software Perceived Ease of Use

Finally, the last part of the survey was used to understand the sizing of the RDBMS systems that are currently being built for each of the different application types. Figure 16 shows the averages that resulted from this part of the survey. It is interesting to note that the largest systems in terms of the numbers of CPUs were OLTP (this surprised us, we thought that it would be Analytics) and that the largest in terms of Database Size are mixed environments (we believe that this is because many of the data intensive document management and workflow applications fall into this category.) This data was used in its raw state to drive the Monte Carlo simulations used to calculate Probable Cost of Ownership.

	OLTP	Analytic	Web	Mixed
Named Users	2095	2012	8040	2935
Concurrent Users	865	817	1656	1207
CPU's	12	8	10	10
Memory	5.0 GBytes	4.3 GBytes	4.1 GBytes	5.7 GBytes
Database Size	1.4 TBytes	0.92 TBytes	0.98 TBytes	2.6 TBytes

Figure 16 – Typical System Sizes and Configurations

2005 RDBMS CCO Tool

The 2005 RDBMS CCO Tool, designed and implemented by MMR, is a spreadsheet tool that will aid the prospective RDBMS purchasers in assessing which of the three target RDBMS vendors will provide the best overall value for a specific IT project. The model considers many factors in assessing and evaluating the both the Comparative Cost of Ownership (CCO) and the Total Cost of Ownership (TCO) of each RDBMS vendor.

The CCO tool automatically calculates and compares the prices associated with licensing of the database software along with the costs associated with software maintenance (*i.e.,* access to software corrections), software update services (*i.e.,* access to new updates and versions of the software), and support services (*i.e.,* access to telephone support) for each of the target vendors. The rules for license restrictions, metrics, and pricing for each RDBMS package were taken from each vendor's website and were used in the building the CCO tool. The input questionnaire is simple and straightforward. Below is the list of the key questions that are asked in the CCO input questionnaire to drive the RDBMS configurations and, thus, cost for each vendor. These questions were chosen since they are non-vendor specific and because they will drive the licensing and support costs.

- What operating system is required? (Linux, Unix, or Windows)

- What type of application are you going to build? (OLTP, Analytic, Web, or Other)

- Will the application be built in-house (versus a purchased application)? (Yes or No)

- Do you need to provide high availability clustering (e.g., passive failover provided by OS)? (Yes or No)[10]

- Do you need to provide basic disaster recovery (e.g., log shipping)? (Yes or No)

- Do you need to provide extreme high availability (e.g., from system or site failure)? (Yes or No)

- Is this an Internet application (i.e., users outside your company)? (Yes or No)

- How many named users will access the application? (Subscribers)

---

[10] Each RDBMS vendor uses different techniques for providing high availability. With our three high availability questions we were trying to come up with a generic concept that could then translate into vendor specific products. High availability clustering is defined as using the clustering available with the Operating System to create a redundant, passive system that can take over in the event of a failure in the primary system. Basic disaster recovery is defined as having a remote, passive failover system which periodically receives transaction logs from the primary system and is applying them to a separate instance of the database; when a failure occurs the remote system takes over. Extreme high availability is defined as systems that automatically record all transactions on both a primary and a failover system such that the failover system can take over from the primary in the event of a disaster within seconds.

- What is the largest percentage of users who will access the system at any one time?

- What is the total number of CPU's (i.e., chips) required in the active RDBMS server (all nodes)?

- How many "Cores" are in the type of CPU chip which will be used in the RDBMS server?

- What is the total amount of memory in the RDBMS server (in Gigabytes)?

- What is the maximum total size required for the database (in Gigabytes)?

- Will you require software corrections (i.e., bug fixes)? (Yes or No)

- Will you require software updates (i.e., license to use new versions)? (Yes or No)

- What level of telephone support do you require? (None, 5x8, 5x24, or 7x24)

- Do you require unlimited telephone support? (Yes or No)

- How many support calls will you make per month? (Number of calls)

In addition to the questions above, the tool also accommodates the input of information about additional discounts that specific vendors might be providing, the cost of DBAs, as well as any other additional costs or credits which we did not consider but that you feel is important. The additional costs and credits are used to transform a CCO value into a TCO value.

The CCO tool uses the answers to the questions to calculate the lowest-priced configuration from each vendor that will satisfy the needs specified. The cost of licensing and support are presented to the user in a clear, easy-to-understand format that succinctly shows a one- to five-year CCO of the three RDBMS vendors.

As mentioned earlier, the CCO tool calculates the best price available from each vendor for each input scenario. For example, each vendor offers an entry-level package that provides a lower cost for RDBMS applications that use a limited number of CPUs, limited amount of memory, and/or do not require all of the features. The CCO tool will use the lower cost entry-level edition if it meets all of the scenario needs and thus provides a cost advantage to the customer. Each vendor also offers user based licensing that can offer advantageous pricing in some scenarios. Again, the CCO tool will use whichever licensing model provided the best overall price to the customer for each vendor.

It is important to point out a special aspect of user based pricing that makes analysis difficult, but not impossible. All three vendors offer different forms of user based licenses. IBM's *Concurrent User License*, defined as one license required for each active connection to the database server, is very different than the Oracle and Microsoft user based license

offerings. Oracle has a *Named User License* (one license required for each person who is authorized to access the application that uses the database server) and Microsoft offers a *Client Access License* (CAL) (one license required for each client device, *i.e.,* client CPU that is authorized to connect to the database server). Since IBM's *Concurrent User License* requires no minimum order other than a single user, it is difficult to level the field with Oracle (requires a minimum of five users per CPU for its Standard Edition and 25 users per CPU for its Enterprise Edition) and Microsoft (requires the purchase of CAL's in five or ten packs for its Standard Edition and 25 packs for its Enterprise Edition) to any meaningful degree. To highlight the differences, consider two examples.

For the first example, consider an application that is used by personnel over three shifts. Further, suppose that there are 100 users in the first shift, 60 users in the second shift, and 30 users in the third. In this example, the customer would need 190 Oracle *Named User Licenses* (since there are a total of 190 workers who require access to the application) but only 100 IBM *Concurrent User licenses* (since there are only a maximum of 100 workers who will be logged onto the application at any one time). If the workers in each shift share desks and thus PCs, the customer would need 100 Microsoft CAL's (since the customer would need to have at least 100 PCs for the largest shift to log onto the application). However, if these workers each used different PCs, the client would need 190 CAL's.

In the second example, consider a customer who has 5,000 employees, all of whom have access to an Intranet based, self-help Human Resource application from their desktop or notebook PC. Most employees are not paid to spend their entire day logged into the HR system but, rather, only log in when they need to perform a specific task that requires the use of the application. Suppose that only 2% of the client's employees are using the HR application at any one time. Under the Oracle licensing scheme the client would be required to purchase a license for each user that has access the system, a total of 5,000 Oracle *Named User Licenses.* Under the Microsoft scheme, the client would be required to purchase a CAL for each desktop and notebook client that has need to access the HR system, which would be one CAL for all 5,000 employees. Finally, under the IBM scheme, the client would be required to purchase an IBM *Concurrent User License* for the peak number of users that access the system at any one time, a total of 100 users, i.e., 5000 employees with 2% access during peak would be 100 concurrent users.

In some scenarios it is either not possible or desirable to count the number of CAL's or Named Users; for example, when the application is connected directly to the Internet using Web technology. In these scenarios, the customer can either opt or will be forced to purchase a CPU license. All three vendors force CPU licensing for Internet application in which it is not possible to count users since a CPU license allows an unlimited number of users to connect and use the system.

It becomes clear that trying to compare a named user (Oracle) to a designated client device (Microsoft) to a number of concurrent users (IBM) would not have produced information that would aid buyers in their decision making process. Even so, the model is designed to look at the particular client scenario and, for each package, select the licensing

options that will minimize client costs. All businesses, particularly smaller ones, should carefully evaluate their needs and estimate the ratio of concurrent users to named users that they expect. At some point, it will become clear that it might be more cost effective to move from a number of user licenses to the less restrictive CPU licenses.

By using the CCO tool it is possible to calculate and assess the database CCO for a particular project comparing the projected cost for using all three target vendors; IBM, Microsoft, and Oracle. But the CCO tool is designed for a single, specific customer project. To make a meaningful comparison of the cost over the full range of RDBMS configuration scenarios requires a different approach. A couple of years ago we developed a new technique called *Probable Cost of Ownership™ (PCO)* which used our CCO tool to look at the broadest possible range of real-world customer scenarios and thus give a composite, or probable, cost of ownership.

Probable Cost of Ownership (PCO) Results

As previously mentioned, to make this study truly useful for strategic decision making, we needed to come to some form of conclusion about the full range of RDBMS server configurations. Because the complexity of licensing restrictions (as expressed in terms and conditions) and the complexity of pricing options, it is difficult to say that any given vendor offers the best value unless you look at a single, specific configuration (which is what our CCO tool does). We needed a new technique that would look at all of the possible and likely RDBMS server configurations that IT departments build such that we could calculate the TCO of all of these configurations and then combine for comparison all at the same time.

We used an advanced statistical technique called Monte Carlo analysis to calculate PCO. Monte Carlo analysis, or Monte Carlo simulation, allows us to treat each of the inputs to the CCO questionnaire as a random variable driven by a set of statistics. We used the results from our 2005 RDBMS survey to drive each of the random variables since the survey provided us with the key statistics needed to model each of the input probability distributions. The Monte Carlo simulation software automatically varied each of the input random variables for each simulated trial. For each of the simulated trial, the CCO tool calculated the optimal cost for each vendor's software. The Monte Carlo simulation software collected and stored the results for each simulated trial for subsequent analysis. Using this technique we simulated 10,000 real world RDBMS server configurations. The result is a set of meaningful statistics that show us the Probable Cost of Ownership (PCO) for a very large number of configurations.

We encountered some problems with calculating PCO due to limitations in Microsoft SQL Server software. As previously noted, 58.1% of the time, Microsoft is not able to support the required configuration due to lack of OS availability and lack of disaster recovery features. To deal with this problem, we ran two sets of simulations, in the first simulation we allowed configurations which would exclude Microsoft; and, in the second simulation, we deliberately restricted the simulation to eliminate the configurations that

would result in Microsoft being excluded. When looking at the final PCO statistics, it is important to keep in mind that Microsoft was excluded in 58.1% of the needed, real world configurations.

Figure 17 shows the cost rankings for the three industry leading RDBMS vendors based upon a TCO driven PCO. A cost ranking of "best" means that the vendor was the best priced option, "middle" means the vendor was the middle priced option, and "worst" means that the vendor was the worst priced option. Finally, a cost ranking of "failed to compete" means that the vendor was unable to support the configuration and it was impossible to be ranked. We found was that IBM DB2 UDB was ranked the best priced option in over 68% of the configurations while Microsoft and Oracle were ranked best only 19.0% and 12.6% of the time, respectively. Notice, that this is where we find that Microsoft is excluded in 58.1% of the configurations.
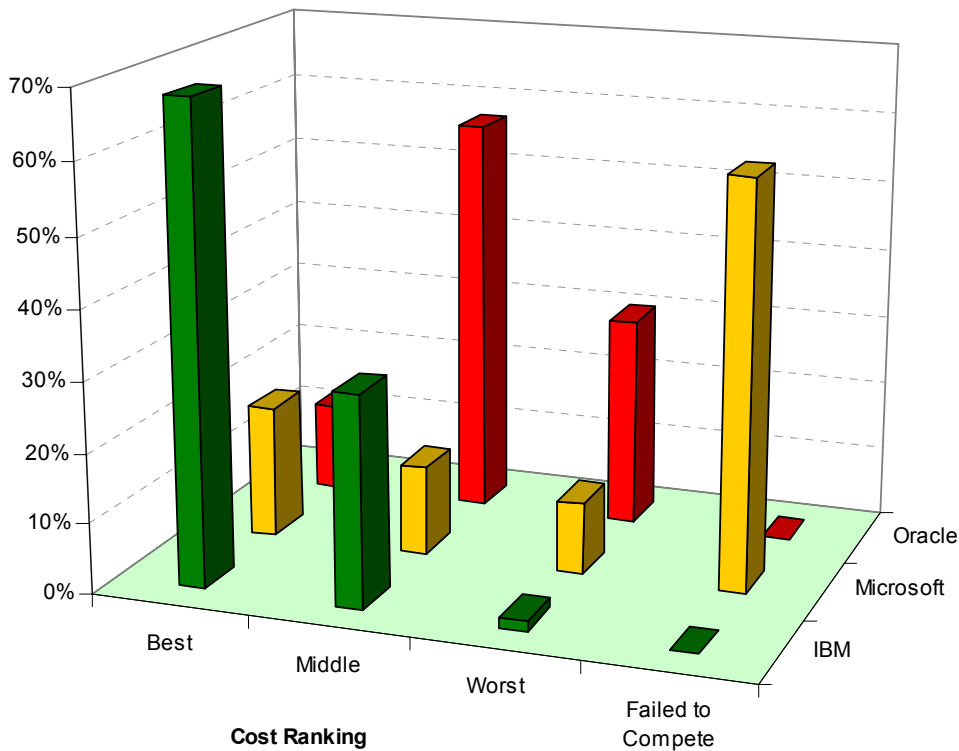
	Best	Middle	Worst	Failed to Compete
■ IBM	68.4%	30.1%	1.5%	0.0%
■ Microsoft	19.0%	12.8%	10.2%	58.1%
■ Oracle	12.6%	57.1%	30.2%	0.0%

Figure 18 – Frequency of Cost Ranking (All Possible Configurations Included)

As we mentioned earlier, it is not possible to calculate meaningful, comparative cost statistics when one of the vendor's RDBMS is excluded from the comparison. The reason

for this is simple; what do you use for a cost for the excluded RDBMS? If you treat it as zero, you will pull down the average; if you treat it as an arbitrarily high number you will inflate the averages. The only safe thing to do is to force the excluded vendor to be included in all of the configurations by limiting the simulated configurations and then reporting that the resulting statistics have a big caveat to them. With this big caveat in mind, the following statistics have been collected by forcing Microsoft to be included in all of the configurations.

We needed to force all of the configurations to Microsoft Windows and to exclude the requirement for extreme high availability (i.e., disaster recovery); thus, we have exclude 58.1% of the real world configurations to ensure that Microsoft SQL Server would have meaningful data for use in the statistics.

Figure 19 shows the statistics that result from the simulations in which Microsoft SQL Server is forced to be included. Notice that IBM DB2 offers the best average and the best highest cost of ownership even when Microsoft SQL Server is forced to be included. It is interesting to note that although Oracle offers the best lowest price, Oracle offers the highest average and the highest maximum cost of ownership. We believe that this is because Oracle has a strategy to offer a very attractive low end configuration to attract customers while earning profits from those customers who need to expand the size of their configurations.

	IBM DB2 v8.2	Microsoft SQL Server 2000	Oracle 10g
Average	$759,096	$763,477	$1,952,454
Std Dev	$754,137	$761,658	$2,347,914
Std Err	$7,541	$7,617	$23,479
Max	$4,778,773	$6,489,733	$17,967,600
Min	$4,419	$8,968	$2,413

Figure 19 – Average Cost of Ownership Statistics for RDBMS Server by Vendor (Windows Configurations Only)

Figure 20 is very similar to the Figure 18 in that it shows the frequency for each vendor's cost ranking from best to worst; but, it is also different because in this case we have forced Microsoft SQL Server to be included by eliminating all requirements for UNIX, Linux, and disaster recovery. Notice that when we force Microsoft SQL Server to be included, that Microsoft has the highest frequency of best rankings, followed by IBM and Oracle; but you must keep in mind, that this means no Unix, no Linux, and limited disaster recovery.
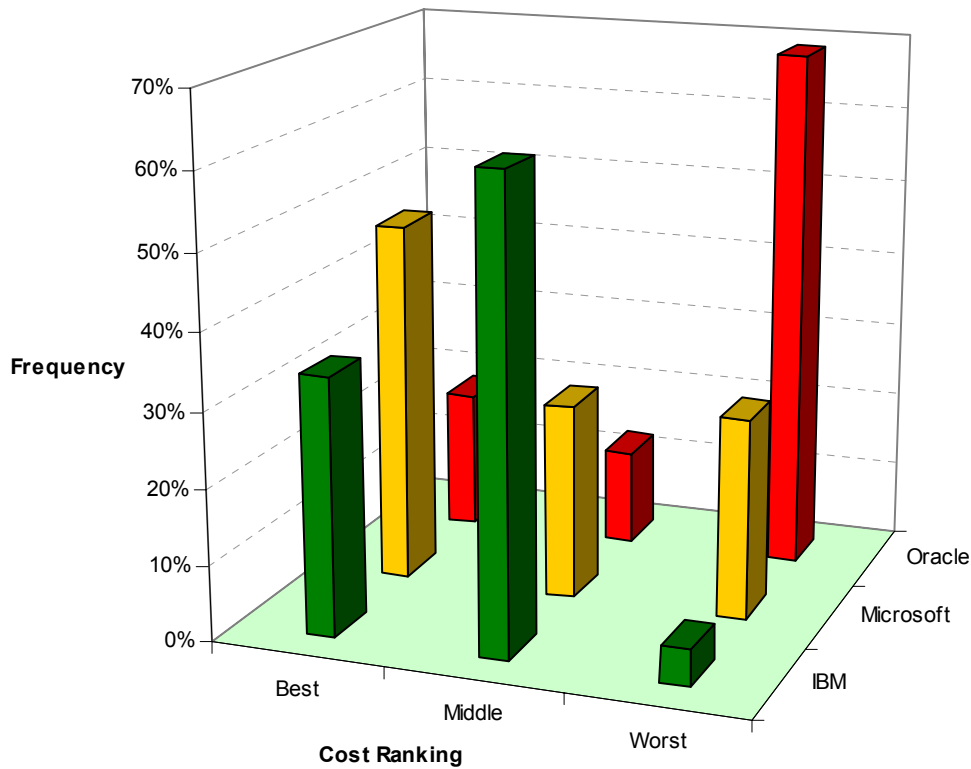
	Best	Middle	Worst
■ IBM	33.9%	61.5%	4.6%
□ Microsoft	47.6%	25.9%	26.5%
■ Oracle	18.5%	12.7%	68.8%

Figure 20 – Frequency of Cost Ranking (Windows Configurations Only)

Finally, Figure 21 shows the cumulative frequency distribution for the 5-Year TCO for all of the scenarios when we forced Microsoft SQL Server to be included in the analysis. This chart shows two very important facts. First, when Microsoft SQL Server meets customer requirements (i.e., Windows is the only required operating system and disaster recovery is not required), it is competitive with IBM DB2 from a cost perspective. Second, Oracle is competitive for the entry-level, lower cost systems but becomes very non-competitive from a cost perspective once the size of the RDBMS server starts to grow.
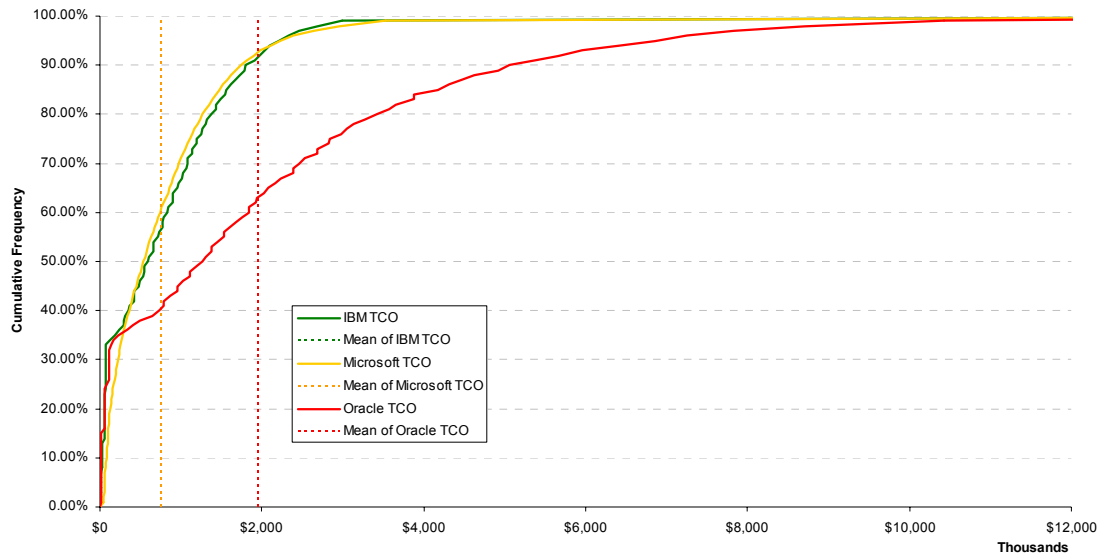
Figure 21 – Cumulative Frequency Distribution for 5 Year TCO (Windows Configurations Only)

Conclusions

Generally, it was observed that each vendor had both strong selling points and weaknesses. IBM was strong in raw price, lower cost of support, and flexibility of operating system. Microsoft was strong in raw price but weak in terms of operating system support. When it came to software updates, Oracle has a lower overall percentage of annual costs that surpassed Microsoft – but you must keep in mind that it is a lower percentage of a higher list price.

IBM –IBM's DB2 UDB v8.2 emerged with the superior Probable Cost of Ownership (PCO). IBM DB2 is clearly more cost effective than Oracle and, although IBM DB2 appears at first glance to have a slightly higher database license cost than Microsoft, IBM more than compensates for this by bundling the first year of software maintenance into the price of the license. Followed, by a low cost of software maintenance in subsequent years, IBM's overall Probable Cost of Ownership (PCO) was found superior to the alternatives. IBM does, however, have a hole in their pricing. While offering a vastly superior user based licensing option due to the concurrent user licensing metric, this concurrent user licensing is only available for the entry and mid-range editions. Additionally, although it is a minor point today, we also found IBM's policy on multi-core CPUs very confusing.

Microsoft – Though SQL Server 2000 offers what appears to be a less expensive database license than the other competitors, the Microsoft software maintenance options can be very expensive. Further, restrictions on the SQL Server 2000 entry and mid-range product offerings will often force the buyer to purchase the more expensive Enterprise Edition. It is disappointing that Microsoft SQL Server does not support over half of the required

enterprise configurations due to lack of support for UNIX/Linux and the lack of high-end disaster recovery capabilities. And, although high-end disaster recovery is expected to be addressed in SQL Server 2005, we do not believe that Microsoft will get through the emotional issues around supporting UNIX and Linux. That is, we do not expect Microsoft to offer SQL Server support for these platforms anytime in the near future. Because of OS and product limitations, we needed to bend our PCO simulation to include Microsoft SQL Server to get meaningful statistics. After bending the simulation to force the inclusion of Microsoft SQL Server into all of the configurations, we found that Microsoft was most often the lowest cost offering but not the lowest *average* cost offering; we believe that this is because Microsoft SQL Server becomes more expensive at the higher end of the configurations.

Oracle – Within the parameters of the scenarios used in the Comparative Cost of Ownership tool, Oracle's *a la carte* pricing scheme fares very poorly against both IBM and Microsoft. Oracle's strategy of pricing everything, and bundling little, allows the customer absolute flexibility and discretion in how to purchase database licenses and support; however, the price for such flexibility is very high in most instances. Given the data presented here, one could argue that Oracle's prices are prohibitively high, especially given the alternatives. There is however, one exception to this. Oracle does very well in small, entry-level configurations.

Summary

Cost of ownership is a difficult assessment to make at the best of times, especially when database application vendors make such comparisons difficult to perform. By using our CCO tool and the results of our PCO methodology it is possible to level the field such that complex terms, package characteristics, and price elements can be objectively analyzed such that the vendor with the best CCO and TCO for a particular scenario can be found. Further, PCO gives a picture of what the Probable Cost of Ownership will look like across the broadest range of real world configurations to help in choosing your strategic, preferred RDBMS vendor.

The results of this study are clear:

IBM DB2 UDB v8.2
- IBM licensing rules and restrictions can be confusing.

- License price is attractive and competitive.

- Software maintenance and support pricing is more attractive than Microsoft or Oracle.

- DB2 runs on all major platforms, thus your can exploit the great value offered by Linux.

- Very attractive licensing terms and availability for extreme high availability (*i.e.,* disaster recovery) requirements.

Microsoft SQL Server 2000

- Runs only on Microsoft Windows and thus does not meet the requirements in nearly half of the RDBMS systems that are being built today. Further, this gap will increase as more systems are deployed on Linux.

- Current offer does not support disaster recovery although this is expected to be addressed in SQL Server 2005.

- License prices are competitive but not as inexpensive as first perceived due to the high cost of software maintenance.

- Software telephone support is pay-as-you-go only.

- The recently announced SQL Server 2005 should do better at filling some holes but it has brand new packaging as well as early indications that license price will be higher, and thus SQL Server 2005 may not be as competitive as the current offering.

Oracle 10g

- Runs on all major platforms and thus you can exploit the great value offered by Linux.

- Consistently the most expensive of the RDBMS when it comes to license price.

- Can be the lowest cost for entry level solutions with low number of users and low number of CPU's.

- Most often ranked as the highest cost alternative.

Each vendor has strengths and weaknesses, and their individual strengths and weaknesses will be perceived differently by prospective buyers. At the very least, the data contained in this report, or the use of the CCO Tool, can help you clearly understand your needs, your options, and provide you with the leverage necessary to negotiate additional discounts that can, based on other considerations, help them to further improve your actual cost of ownership.

A Final Note

As stated early in this paper, IBM and Oracle were forced to run on Windows for the purposes of comparison. Since Microsoft's products run only on their own operating system, MS-Windows, no other Operating Systems were considered in our analysis; to do so would have excluded SQL Server from our analysis. From a buyer's perspective, this circumstance is problematic.

Both IBM DB2 UDB v8.2 and Oracle 10g run on operating systems other than MS-Windows, such as IBM AIX, HP-UX, Linux and Sun Solaris. Because we are interested in lowering prices for our clients, the ability of IBM and Oracle to operate effectively on Linux, for example, is very attractive. Though Linux has not achieved widespread popularity as a desktop environment, it is rapidly gaining popularity as a server operating system and is being widely deployed on enterprise servers as each day passes. We think it is interesting to consider Linux in server environments that require a commercial RDBMS. Linux has some distinct advantageous as a RDBMS server. The first and most obvious advantage is that Linux will provide additional savings on license cost because the Linux Operating System licenses are essentially royalty free. Linux also has a proven track record of running with considerable performance on low-cost hardware because the Linux software can be recompiled at install time to be optimized for the hardware that it is running on – as a result you save on hardware costs. Finally, Linux has proven itself as a robust and secure operating system – exactly what most database servers need to thrive. A question that Windows users need to ask, then, is there further savings that can be realized by changing to a different operating system such as Linux?

As a final, final note, the above "final note" about the power and cost effectiveness of Linux has been virtually unchanged since our 2003 report on database cost of ownership was published.

Additional copies of this white-paper can be purchased by contacting Market Magic Ltd below.