Global Headquarters: 5 Speen Street Framingham, MA 01701 USA P.508.872.8200 F.508.935.4015 www.idc.com

WHITE PAPER

Microsoft SQL Server Versus Linux Enterprise RDBMS in Enterprise Computing: Comparing Costs in Development Cases

Sponsored by: Microsoft Corporation

Carl W. Olofson

October 2003

IDC OPINION

IDC interviewed 16 sites developing database applications for Microsoft SQL Server and 14 sites developing for a Linux relational database management system (RDBMS). We found that the most significant cost difference between these two sets of cases was that of staffing. IDC believes that this cost difference can be attributed to the following factors:

- Linux RDBMS products tend to require more expertise and effort than Microsoft SQL Server, resulting in higher salaries and more time spent on administrative tasks.
- The scarcity of Linux professionals, relative to the number of Microsoft Windows Server professionals, drives up salaries on the Linux side.
- Linux professionals, by and large, are repurposed Unix professionals, and therefore they are more highly paid than their Microsoft SQL Server counterparts because the proper management of Unix and Linux requires more technical expertise than do the Windows Server environments.
- The Linux RDBMS products showed significantly higher license and maintenance fees than did Microsoft SQL Server. The Linux OS itself had higher costs associated with it compared with Microsoft Windows 2000 Server, even though there is no license fee for Linux OS per se. Thus, the software costs, although a very small fraction of the total cost of ownership (TCO) of this sort of project, are higher for Linux, not lower, as is sometimes generally assumed.

EXECUTIVE SUMMARY

The purpose of this project was to discover what differences, if any, exist in the cost profiles of database development products for those organizations using Microsoft SQL Server compared with those organizations using a leading relational database management system (RDBMS) on the Linux operating environment. IDC surveyed 30 sites, 16 developing applications for Microsoft SQL Server on Windows and 14 developing applications for a leading RDBMS on Linux. In each of the Linux cases, the respondents were using one of the market-leading, high-end enterprise RDBMS products.

IDC found that the Linux RDBMS users encountered a 46.8% higher overall cost (estimated annual normalized costs per 100 tables) in developing their databases and database applications. By far, we found the most significant difference in the area of staffing-related costs. Reasons for these cost differences include factors endemic to the current state of the Linux platform (e.g., lack of qualified staff, which makes

personnel more expensive and drives a need for outsourced services) as well as strengths of the combination of SQL Server on Windows (e.g., ease of use and management, which lowers the experience and expertise requirement on staff; greater availability of qualified talent; and less need for highly technical staff).

This study explores these cost differences and the underlying reasons for them. It also explores the relative desirability and cost-effectiveness of these two configuration classes in satisfying various types of user requirements.

KEY FINDINGS

As previously stated, IDC normalized costs per 100 tables so that a reasonable comparison could be made based on equivalent workloads in each of the cases. Based on this analysis, we found that:

- Hardware and software costs, taken together, were roughly equivalent between the two sets of cases. Hardware costs were slightly higher on the Microsoft SQL Server side, perhaps because those cases tended to purchase their hardware from vendors that bundled extra software and services with their systems at a higher price. Software costs were slightly higher in the Linux RDBMS cases, perhaps because software bundled in the Microsoft SQL Server cases had to be purchased separately in addition to the RDBMS license and maintenance agreements themselves.

- Licensing and maintenance were significantly more expensive for the Linux RDBMS products than they were for Microsoft SQL Server, though this element was a very small fraction of the total cost in both cases.

- Staffing costs were significantly higher on the Linux RDBMS side than on the Microsoft SQL Server side. Because staffing costs constitute the lion's share of total costs, this area accounts for the overall difference in costs between the two sets of cases.

PROJECT SCOPE AND METHODOLOGY

SCOPE OF STUDY

This study presents the TCO analysis of 30 database environments drawn from 10 vertical industries running Linux RDBMS and Windows SQL on Intel platforms. Altogether, IDC surveyed 40 sites for this study, including 20 using Microsoft SQL Server as a production RDBMS and 20 using an RDBMS on Linux as a production RDBMS. Of these sites, 16 were developing database applications for Microsoft SQL Server and 14 were developing database applications for the Linux RDBMS. Our study involved these 30 cases.

METHODOLOGY

To obtain the TCO data used in this analysis, IDC conducted in-depth telephone interviews with database managers. The interviews consisted of a series of questions designed to elicit information regarding the database environment and growth, database-related IT operations and management, and costs associated with the acquisition and operations of the database servers. One-year historical data and current data were collected and then projected over five years.

Cases were selected on the basis of their conformance to certain general parameters regarding the usage of the technology in question, the size of organizations, and the types of RDBMSs employed. Obviously, half the cases were selected because they were using Microsoft SQL Server. The other half were selected because they were using products that were functionally similar, which specifically left out those employing very simple RDBMS products or open source products. During the period being analyzed, such products had not featured many of the capabilities necessary for business-oriented transaction processing and business intelligence. IDC also notes that a large percentage, if not a majority, of users of open source RDBMS products on Linux use them as simple data server farms. Such a use case would not represent an analogous equivalent to the Microsoft SQL Server cases from a cost analysis perspective.

MAJOR COST FACTORS

- Hardware
  - Purchase — acquisition of the hardware only
  - Installation — costs for initial setup of the server and for annual hardware upgrades
  - Maintenance — external and internal costs to support hardware
- Software — operating system (OS)
  - Purchase — costs of the OS stripped out from the total server costs
  - Installation — costs to initially deploy the OS and for annual upgrades
  - Training — external costs for initial training of IT staff, specifically on the OS
  - Maintenance — external and internal costs to support software
- Staffing
  - Annual loaded salary, which includes cost for overhead and bonuses
- Outsourced services
  - IT services to support and maintain servers
- Annual IT staff training
  - Costs that include fees for outside trainers as well as the productivity loss of staff for time spent in training

NORMALIZATION AND PRESENTATION

Ultimately, the value of any TCO analysis lies in its utility to the IT buyer. To be useful in the buying decision, the analyst must standardize information from very different environments so that IT buyers can compare their own environments to the standard.

To ensure that the two development environments are compared fairly, IDC normalized all costs per 100 tables. This means we set the tables at the same number, based on the average of the two environments (1,275 tables growing to 1,550). We then let the number of processors float in accordance with the average tables per processor that each of the environments experienced in practice.

Linux environments averaged 59.0 tables per processor, whereas Microsoft averaged 58.9. When comparing costs, we assumed the costs of 1.002 Microsoft processors for 1 Linux processor.

IDC presents the TCO findings per 100 tables so that companies of all sizes can relate the costs and benefits of the study to their environments.

A DETAILED REVIEW OF COSTS BY CATEGORY

The costs of the two cases are broken down into the following categories:

- **Hardware,** including the cost of equipment acquisition, installation, and configuration. Services provided by the hardware vendor are included in this category.
- **Software,** including the license fees for the OS and the RDBMS. Also included in this category are the costs of installation and configuration of the packaged software and development of the custom applications.
- **In-house staffing,** representing mainly the salaries of the database administration staff.
- **Outsourced staffing,** representing mainly the fees paid for the services of consultants contracted to provide database administration services.
- **Lost productivity costs,** due both to training (staff time and training fees) and downtime (recovery costs and lost user productivity).

Table 1 details the costs in each of these categories. Each category is examined in detail in the following sections. All cost values have been normalized based on an equal number of tables to be managed for each set of cases and presented as the average cost per 100 tables. This survey did not question study participants about development tools because database development tools are generally bundled with the RDBMS and because code development tools come at negligible cost. Most code development tools can be targeted at multiple environments, including Windows, Unix, and Linux, so deviations in this area would be a red herring in analyzing comparative costs.

SUMMARY OF COSTS

Costs were collected on the basis of two categories: the cost of getting the database server set up and running initially (the "initial costs") and the estimated (by the survey respondent) annual cost of developing databases and database applications.

INITIAL COSTS

IDC found that the initial costs of setting up database servers for development were 7.6% lower for the Linux RDBMS cases than for the Microsoft SQL Server cases. Table 1 shows the normalized costs per 100 tables.

TABLE 1

COMPARISON OF MICROSOFT SQL SERVER CASES VERSUS LINUX RDBMS CASES: INITIAL COSTS PER 100 TABLES

	Microsoft SQL Server	Leading Linux RDBMS	Linux RDBMS/ Microsoft SQL Server
Server hardware			
Purchase	$8,669.1	$7,162.1	-17.4%
Setup, upgrade	$2,613.0	$149.9	-94.3%
Training	$701.7	$306.6	-56.3%
Server subtotal	$11,983.8	$7,618.7	-36.4%
Software			
OS	$241.6	$391.9	62.2%
RDBMS	$6,871.7	$9,379.5	36.5%
Software subtotal	$7,113.3	$9,771.4	37.4%
Outsourced staff	$663.9	$874.8	31.8%
Total costs	$19,761.0	$18,264.9	-7.6%

Source: IDC, 2003

Most of the cost difference can be found in the purchase, setup, and upgrade of hardware on the Microsoft SQL Server side. This difference is probably due to the purchasing, on the Microsoft SQL Server side, of system and service bundles from premium hardware vendors versus more of a "white-box" approach taken by those setting up Linux systems.

IDC also notes that the software costs, both of the OS and of the RDBMS in terms of license, maintenance, and installation, were 37.4% higher on the Linux RDBMS side. This difference reflects higher license and maintenance fees for the RDBMSs used on Linux. Also, although a trivial element in the total cost profile, the Linux OS (including tools, managed configuration licenses, and maintenance fees) was actually 62.2% more expensive than Microsoft Windows 2000 Server. Of course, although the OS itself is open source and therefore free, the purchase of the other elements of the Linux package from a Linux provider are not.

ANNUAL COSTS

We asked respondents to estimate the annual cost of development against their RDBMS. Table 2 shows the results.

TABLE 2

COMPARISON OF MICROSOFT SQL SERVER CASES VERSUS LINUX RDBMS CASES: ESTIMATED ANNUAL COSTS PER 100 TABLES

	Microsoft SQL Server	Leading Linux RDBMS	Linux RDBMS/ Microsoft SQL Server
Server hardware			
Purchase	$1,007.3	$1,003.8	-0.3%
Maintenance	$1,069.4	$1,591.6	48.8%
Server subtotal	$2,629.8	$2,595.4	-1.3%
Software			
OS	$25.1	$326.2	1,198.7%
RDBMS	$2,162.5	$2,644.2	22.3%
Software subtotal	$2,187.6	$2,970.4	35.8%
In-house staff	$44,937.8	$68,483.8	52.4%
Outsourced staff	$45.2	$220.6	388.0%
IT training			
Fees	$1,359.2	$742.4	-45.4%
Productivity	$527.4	$839.9	59.2%
IT training subtotal	$1,886.7	$1,582.3	-16.1%
Total costs	$51,687.2	$75,852.5	46.8%

Source: IDC, 2003

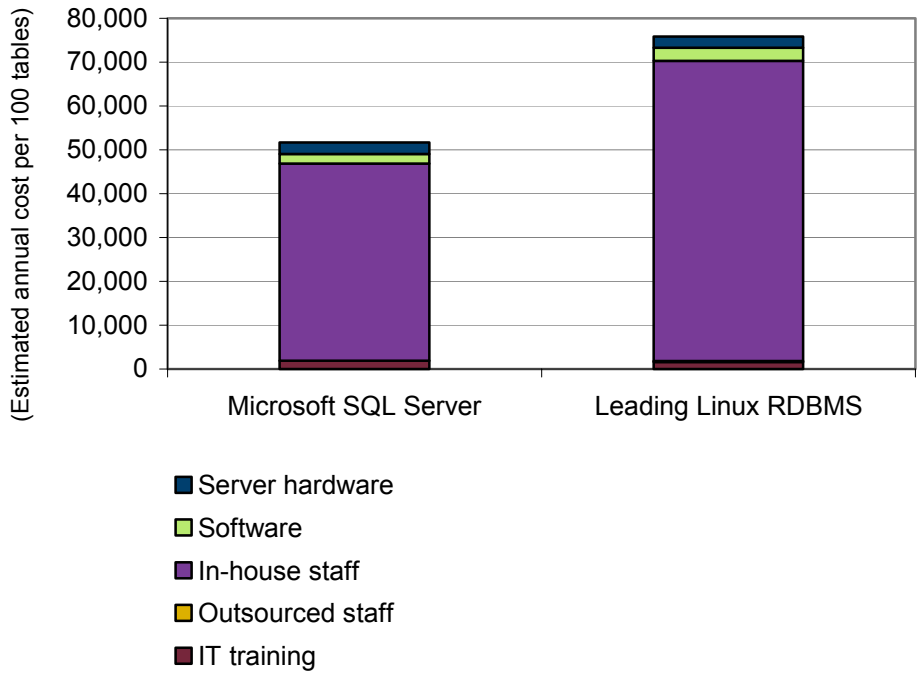
Server hardware costs were roughly equivalent between the two groups, but we found the annual cost of software to be about 35.8% higher in the Linux RDBMS cases, again owing mainly to the higher maintenance and support costs associated with the RDBMS products used on Linux. Figure 1 summarizes these results. IDC notes that the OS costs, though miniscule in comparison with the total costs, were nonetheless noticeably higher for Linux, probably because Linux respondents paid more, on a per-table basis, for support and upgrades from their OS provider than did Microsoft Windows respondents. Because these figures are very small, the chance of

statistical error is significant; therefore, it is hard to assert that this difference is indicative of a general rule. The difference is interesting anecdotally, nonetheless.

Figure 1 illustrates, quite dramatically, that the largest cost factor by far is that of in-house staffing, which dwarfs all other annual costs.

FIGURE 1

COMPARISON OF MICROSOFT SQL SERVER CASES VERSUS LINUX RDBMS CASES



Source: IDC, 2003

DATABASE MANAGEMENT COSTS BY CATEGORY

HARDWARE AND SOFTWARE COST COMPARISON

At one level, to separate hardware costs from OS and RDBMS software costs for dedicated database servers, especially in the initial cost category, is to indulge in a bit of a shell game. This is because some software costs represented by Microsoft products are bundled into the servers in some instances so that from the point of view of Windows 2000 Server, the associated other server software elements are included in the server at no apparent cost. In the Linux case, on the other hand, the costs of equivalent products are paid for explicitly and not bundled. Adding the initial costs of hardware and software, we find no significant difference between the two sets of cases.

As far as annual costs are concerned, the hardware differences are also insignificant, but the Linux RDBMS cases showed noticeably higher costs, probably due to higher upgrade and maintenance fees for the RDBMSs used on Linux. As we have noted, however, these costs are small relative to the staffing cost difference; therefore, they don't play a major role in the overall cost comparison. The 22.3% difference in

estimated annual costs, suggesting higher maintenance fees on the RDBMSs used for Linux, does point to the fact that in taking the total cost of software into account for comparison of these two environments, the higher fees of vendors providing critical capabilities over the top of the OS deserve consideration.

STAFFING AND LOST PRODUCTIVITY COST COMPARISON

IDC found that the Linux cases reported 52.4% higher in-house staffing costs than did the Microsoft SQL Server cases. This includes all staff involved in developing the database and performing database-related application development (e.g., SQL development) tasks. One reason for the higher cost in the Linux cases may be the difference in average salary, which was 15% higher in the Linux RDBMS cases, on average, than the Microsoft SQL Server cases. Compounding this difference was the fact that the Linux RDBMS cases reported using, on average, 32.5% more full-time equivalents (FTEs) for database development tasks than did their Microsoft SQL Server counterparts. These two facts, when added together, resulted in the overall cost difference reported here.

As Figure 2 illustrates, IDC found that a much larger percentage of the staff resources of Linux RDBMS cases were engaged in general tasks apart from design, development, testing, administration, installation, or tuning than those of their Microsoft SQL Server counterparts. What these other tasks were is not immediately clear, but they could account for the significantly large number of FTEs required by Linux RDBMS cases as opposed to Microsoft SQL Server cases.

Outsourcing and lost productivity costs were too small to serve a useful purpose in this study.

CHALLENGES/OPPORTUNITIES

IDC found that staffing played such a large role in the comparison between database development costs of Microsoft SQL Server and Linux RDBMS cases that all the other cost categories fade to insignificance. These cases show that Microsoft SQL Server required a smaller and less expensive staff than the Linux RDBMSs. IDC believes that this difference may not remain permanently, as other RDBMS vendors and Linux providers are improving their ease-of-use and expertise requirements.

Nonetheless, Microsoft enjoys this advantage for the present, and it can result in a kind of momentum, as the larger number of SQL Server database administrators (DBAs) and developers encourages more adoption of SQL Server, which encourages more IT personnel to train on SQL Server and so on. To maintain this advantage, however, Microsoft will need to continue to work hard and innovate to sustain its current lead in areas of ease of use and self-management over other RDBMS Linux vendors while adding capabilities that extend and expand the competitiveness of SQL Server.

CONCLUSION

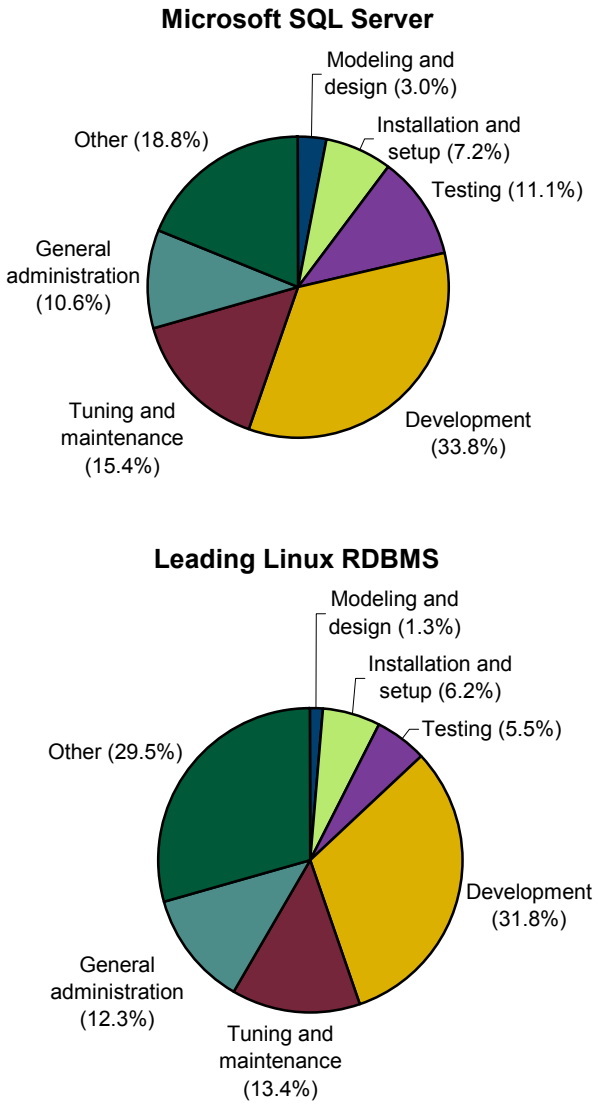
IDC had great difficulty finding enough Linux RDBMS cases to fulfill the requirements for this study. Although it is difficult to make any generalizations based on a study of just 20 cases, they certainly suggest that staffing for RDBMS development on Linux is more expensive than on Microsoft SQL Server, and Linux RDBMS staffs must devote more effort to the task of database application development than their Microsoft SQL Server counterparts.

IDC offers the following recommendations:

- Any IT manager considering a choice between a Linux RDBMS deployment and a Microsoft SQL Server deployment should take the relative staffing costs into account and make sure that, for the organization in question, this cost has been accurately estimated and factored into the decision-making process.
- Developers considering developing expertise in Microsoft Windows Server and SQL Server should be aware that the relatively larger number of sites using SQL Server and the larger developer community probably translate into more readily available training courses and materials and more opportunities for employment for Microsoft SQL Server–based positions than for Linux RDBMS positions.

FIGURE 2

COMPARISON OF MICROSOFT SQL SERVER CASES VERSUS LINUX RDBMS CASES: STAFFING COSTS BY WORKLOAD



Source: IDC, 2003

APPENDIX A: LINUX

Linux is available from multiple Linux "distributors," including Red Hat, SuSE, Caldera/SCO, Turbolinux, MandrakeSoft, and others. Thus, a single, underlying operating system kernel is shipped by multiple distributors around the world. Each adds software modules, including utilities and middleware, on top of the basic kernel. Linux was invented in 1991 by Linus Torvalds — then a Finnish graduate student — who sought help in completing the source code from a community of open source developers via the Internet in the 1990s. Today, Torvalds publishes updated versions of the Linux operating system, but new code from the open source community is added in each new release. According to the "rules" of Linux publication, developers can create new source code for Linux, but they must publish it back to the open source community for inclusion in later releases.

This IDC study includes data about servers at user sites that are running different distributions of the basic 32-bit Linux operating system. Thus, the Linux platform studied in this IDC white paper is a generic platform rather than one provided by any single Linux distributor.

Historically, adoption of the Linux operating system has come in "waves." In the first wave, Linux was added to existing, installed-base client or server machines that were shipped without operating systems, or it replaced existing operating systems. In the second wave, system vendors began shipping Linux on new server systems, starting in 1999. These Linux hardware platforms included appliance servers and general-purpose servers from Dell, HP, IBM, Sun, and others.

Linux was also being adopted by the high-performance technical computing (HPTC) community. For HPTC applications, Linux is often deployed on Linux workload-balancing clusters including dozens of individual servers running the "Beowulf" open source Linux clustering software. In the current wave of adoption, much "custom" software development is taking place at commercial Linux sites. It requires intense IT staffing/application development related to the Linux custom application creation, deployment, and maintenance. In coming years, there will also be a wave of Linux adoption for support of Web services, which are Web-enabled applications that can link and interoperate with other Web-enabled applications via the Internet. Web services for Linux server will likely be based on the Java development environment.

These "waves" of adoption have brought increasing reliability and support to the overall Linux environment. However, they have not yet succeeded in lowering the TCO for Linux servers, which, on average, require more custom software and hands-on management than do comparable Windows 2000 servers. This is the case because custom Linux applications require optimization and tuning, achieved at the expense of professional time from programmers/developers, system administrators, and operations personnel.

APPENDIX B: WINDOWS 2000

Adoption of Microsoft Windows 2000 has been ramping up since its introduction in February 2000. Windows 2000 is available in three versions: Standard Server, for hardware platforms with two to four processors; Advanced Server, for hardware platforms with four to eight processors; and Windows Data Center, for hardware platforms with eight or more processors.

This IDC study investigated RDBMS servers running Windows 2000; it did not consider servers running the Windows NT Server product. This aspect of the study's methodology ensured that respondents commented on the currently shipping product rather than on an older product. It also ensured that we gathered TCO metrics about the same Windows operating system platform.

Windows 2000 is available from one software vendor, Microsoft, rather than from multiple software vendors or software distributors, although it can be acquired indirectly through the purchase of OEM server systems. Microsoft Windows is widely available, and it is supported by Microsoft and its OEM partners, channel partners, and systems integration partners. The most scalable version of Windows 2000, Windows Data Center, is available on new servers from system vendors, which ensures that the total solution includes system configuration and support.

Windows 2000 is now the primary version of 32-bit Windows shipping on Intel-based servers. In many cases, Windows 2000 is a follow-on replacement for earlier versions of Windows, including the widely deployed Windows NT Server 4.0, which Microsoft began shipping in the summer of 1996 and stopped shipping as a generally available product in early 2002.

Windows 2000 is a mature operating system product; more than 1 million copies are shipped annually on a worldwide basis. Tens of thousands of packaged applications are available, including packaged databases that run on the Windows 2000 server operating system. While programmers can develop custom Windows 2000 programs, less custom development is typically associated with installing and deploying the Windows 2000 server operating environment, which serves as a platform on which to run those packaged applications.

The next wave of Windows adoption will be versions of the Windows 2000 operating system that include support for .NET, Microsoft's software technology for direct support of Web services. Microsoft expects to enhance the Windows 2000 server products with the addition of .NET versions of Windows 2000 Standard Server, Advanced Server, and Data Center Server, with additional built-in support for Web services, which are Web-enabled applications that can link and interoperate with other Web-enabled applications via the Internet, later this year.

APPENDIX C: LINUX RDBMS PRODUCTS

Most of the market-leading RDBMS vendors now support Linux with their full product lines, including those used by the Linux RDBMS cases in this study. These products tend to be identical to the Unix versions of their products. In fact, in most cases, these vendors offer products that have the same performance and usability characteristics on all supported platforms, including the Windows server environments. It should be noted that over the course of the past several years, all these vendors have made their products progressively easier to use and administer. Although these products have features that are designed to deliver high availability and scalability through clustered server support, these features do not seem to have come into play in the cases studied.

APPENDIX D: MICROSOFT SQL SERVER

Microsoft SQL Server is only available for the Windows operating environment; therefore, all cases in this study used the Windows server operating environments, specifically Windows 2000 Server. SQL Server has historically been known as easier to manage and use than the other RDBMSs involved in this study, though it may be argued that such differences have narrowed somewhat during the past three years. Because Microsoft server software products are developed as part of a coordinated product strategy, SQL Server also has certain affinities with the Windows operating environment that make administering SQL Server and its operating environment together easier — thus requiring less skill on the part of database IT staff — than is the case managing other RDBMS products on Linux. Note that all the cases studied were using SQL Server 2000 on Windows 2000 Server.