



Alternative Technologies

**THE
DBMS SCALABILITY REPORT**

PERCEPTION VERSUS REALITY

EXECUTIVE OVERVIEW

David McGoveran
Alternative Technologies
6221A Graham Hill Road, Suite 8001
Felton, California 95018
Telephone: 831/338-4621 Fax: 831/338-3113
Email: mcgoveran@AlternativeTech.com
Website: www.AlternativeTech.com
Report Number 970703

Disclaimer

This report is produced and published by Alternative Technologies, Boulder Creek, CA. The information and opinions presented in this report are exclusively those of Alternative Technologies, except where explicitly quoted, acknowledged, and referenced. Although all opinions and information are reviewed for technical accuracy, the products discussed have not been subjected to formal tests and it is impossible to verify every statement made by sources. No guarantees or warranties of correctness are made, either express or implied. Readers are cautioned to evaluate the products in their own environments and to consider the information and opinions presented here with respect to their own requirements.

For information about this or other reports, or other products and services, including consulting and educational seminars, contact Alternative Technologies directly by telephone, mail, or via our Web site:

Alternative Technologies
6221A Graham Hill Road, Suite 8001
Felton, CA 95018
Telephone: 831/338-4621 FAX: 831/338-3113
Email: mcgoveran@AlternativeTech.com

Copyright 1997, All Rights Reserved – Alternative Technologies

I. Introduction

This report, and hopefully others to follow, is concerned with scalability, and more generally, high-end database requirements. It provides the results to-date of an on-going study pertaining to scalability including (1) an analysis of market requirements, and (2) case studies. The case studies were conducted as audits of applications which push some or all of the limits of their respective DBMS products. As of this report, only Oracle and Sybase sites have been included in the case studies. For this reason, we briefly discuss each of these companies and their approaches to scalability.

Scalability is becoming extremely important for today's enterprise DBMS. There is considerable reason for concern with the current state of analysts', press members', vendors', and consumers' understanding of DBMS scalability. Misleading advertisements and false claims have been accepted and repeated by key influencers. For example, it is not uncommon for database size "achievements" to be stated in terms of either allocated space or planned database size. Sometimes the expected user populations and planned database sizes are presented in briefings and new product announcements. More conscientious presenters may add an aside that the current pilot is significantly smaller and may

not even be in production. Analysts, the press, and consumers seem to accept the combination of laboratory benchmarks and user plans as proof of support for VLDB and processor scalability.

So far as we have been able to determine, DBMS scalability costs and benefits have never been investigated in detail, let alone published. Our study has already exposed some surprising myths concerning scalability and VLDB. This report, and those we expect that will follow, will expose numerous myths, fallacies, and flim-flam, while providing a source of unbiased information about DBMS scalability. We hope it will be a valuable aid to customers, vendors, press, and analysts in obtaining a better understanding of DBMS products, their capabilities, and the market.

II. Market Requirements

After careful study, it is clear that four marketing requirements (above all others) are considered important by both analysts and prospective DBMS users. The following requirements address the rapidly growing need for high-end business transaction systems:

- *tens of thousands of users online* - This is due to a combination of:
 - a. improved standards of customer service which are intended to permit online access to customer-related data
 - b. the rise of remote electronic access to databases
- *very large databases* - The increase in database size is due to:
 - a. increasingly high volume of data capture transactions, largely due to ever larger user communities as businesses improve their ability to reach geographically separated markets cheaply and easily
 - b. the rise of data warehousing
 - c. the increasing trend toward integration of timely operational, and historical informational data
- *very high transaction rates* - All the business trends that are driving larger user populations and larger databases are also driving higher transaction rates, whether for OLTP or informational systems.
- *electronic business transactions* - As the problems of secure electronic business transactions and high volume user loads on the net are solved, electronic commerce will become much more important, ultimately overwhelming the current volumes and growth rates.

Collectively, the first three of these market requirements demand open-ended scalability. Clearly, none of today's DBMS products can be expected to meet the anticipated need. When the load implied by the fourth requirement is taken into account, the requirements are impossible for today's DBMS products to meet.

III. The Myths of VLDB and Scalability

Among the key results of this study have been the identification of a number of beliefs about VLDB and scalability, which although widely held to be true, are actually misstatements of the facts. These *perceptions* are presented here along with a brief explanation of *reality*.

Perception 1: DBMSs can be produced and consumed as though they were commodities.

Reality 1: As ever larger database sizes, numbers of users, and workloads (such as transactions rates) must be supported by today's DBMS products, the DBMS features that lead to success or failure have become

increasingly more difficult to identify. Every vendor has chosen differentiating implementations of similar functionality. Customers use a variety of "workarounds" to circumvent the many unsolved DBMS scalability problems.

Perception 2: Workloads defined in terms of the number of physical transactions a DBMS processes are meaningful for scalability.

Reality 2: Each DBMS requires a different implementation of a business transaction if it is to give the best performance. The number of physical transactions that result often vary greatly from product to product and environment to environment.

Perception 3: A DBMS is either scalable or it is not.

Reality 3: There are many types of scalability: processor scalability, platform scalability, administrative scalability, etc. A DBMS can be said to be "X% scalable" only with respect to a particular resource or workload and over a certain range. As a qualitative property of DBMSs, scalability is not meaningful.

Perception 4: A demonstration of DBMS scaleup and speedup on any platform and application of the DBMS vendor's choosing is sufficient, irrespective of the intended platform and application, let alone transaction and database design considerations.

Reality 4: Practically any scalability or speedup is possible for any DBMS if a specific application is carefully selected. Scalability is, in fact, specific to the intended application's characteristics (for example, read-only versus read-write). Transaction and database design also have a powerful effect on scalability. Subsecond response times for a few users often go to tens of minutes or even hours when the load and resources were scaled up with some designs, but scale smoothly when redesigned.

Perception 5: The more scalable the system, the more efficient and cost effective the product.

Reality 5: It is possible to have two systems with identically the same percent scaleup or speedup, but with widely differing absolute throughput. The product with the poorer scalability may well provide better performance over the range of available resource. Most published scaleup or speedup percentages are derived from TPC (and related) benchmark numbers. Unfortunately, these tests do not properly measure scalability since more than one resource is allowed to vary.

Perception 6: The speed of administrative operations suffices to determine administrative scalability.

Reality 6: All the speed in the world will not lead to administrative scalability if administrative operations must be performed offline, the database is sufficiently large, and the available "window" is sufficiently short. Conversely, if administrative operations are performed continuously and without conflicting with user operations (that is, dynamically), then they need only keep up. The speed of a corresponding offline administrative utility is then of little importance.

Perception 7: Partial database operations provide administrative scalability.

Reality 7: Partial database operations improve availability and may offer some speedup due to parallelism.

Unfortunately, the complexity of their use does not scale. Partial database backup, restore, and recovery generally do not automatically maintain consistency, creating a serious manual operational load. Even with moderately sized databases, managing the restore and recovery operations on a database from a set of partial backups is tedious and uncertain.

Perception 8: A DBMS with good processor scalability can provide 100% speedup.

Reality Synopsis 8: Processor speedup is inherently non-linear. The maximum speedup T that can be expected in a system with N processors running in parallel $M\%$ of the time is given by the *non-linear* Amdahl's Law:

$$T = 1 / ((1 - M) + (M / N))$$

Perception 9: Parallelism is necessary for scalability.

Reality 9: The scaleup that parallelism can offer is strictly limited by the number of processors and processor scalability. The speedup that parallelism can offer is strictly limited by the inherent coarseness and serial character of the workload. Parallelism can help remove the cost of administrative functions as a barrier to availability by reducing the time required for loading, backup, restore, recovery, reorganization, and so on. Note, however, that at a reasonable forty (40) GB per hour, backup, load, or resynchronization of a terabyte requires 25 hours offline.

Perception 10: Many open systems databases are in production with a terabyte (or more) of data.

Reality 10: Most of the space associated with terabyte (and above) databases is due to overhead including storage inflation, indexing, temporary space, log or recovery related space, and mirroring. Many sites reported as having terabyte plus databases (often as reference sites) were simply planned, or purchased storage. Often multiple databases are reported as though they were a single database.

Perception 11: Storage addressability is an indication of DBMS scalability and value.

Reality 11: The full storage addressability of today's DBMSs is rarely tested by vendor QA due to the expense of building (over \$1 million per terabyte) and testing large databases. Practical considerations generally limit production database sizes long before the storage address limitations in the product are reached.

Perception 12: It doesn't matter how the space is used, as long as the DBMS can manage it.

Reality 12: For a simple (but real) database consisting of a single indexed table, the computed storage requirements for 2 billion rows was compared for Oracle and Sybase. The total mirrored space for Oracle was 1,052,447,153,398 bytes (over a terabyte) and that for Sybase was 511,337,680,618 (under half a terabyte). The products would not be so dramatically different for every database (but might be worse for some), however, the costs of storage (\$0.5 million), operation, administrative times, and maintenance costs are quite real and must be considered. The largest production databases are of similar size when compared in terms of data supported, regardless of DBMS used and despite large differences in total space consumed.

Perception 13: Data and index space support proves the ability of a DBMS to support large databases.

Reality 13: Temporary space (for sorting and reorganization), recovery or log space, redundancy for

performance, and redundancy for availability are part of the reality of a production database. Even if all the other issues involved in supporting a large and growing amount of user data can be managed, there are a number of operational issues that are aggravated in a non-linear fashion by the growth. These include:

- the difficulties of designing and controlling transaction isolation
- read-only transaction management overhead (if read-consistency is required)
- deadlock avoidance, detection, and resolution under increasing deadlock probabilities
- allocation errors and recovery
- space management and organizational complexity

Perception 14: Numbers of users supported is a measure of scalability.

Reality 14: Reported numbers of users at reference and survey sites vary greatly in meaning. These include concurrent transactions, indirect users (via multiplexing), concurrent users, connected users, size of the user community, identified users, and licensed users. Of these, concurrent users and concurrent transactions are probably the most useful, and rarely exceed a few thousand. Also, the costs associated exclusively with connection overhead need to be taken into account. With a small connection overhead of 75KB per user, ten thousand (10,000) users require 750 MB of memory. With a more common connection overhead of 150KB, it becomes 1.5 GB.

Perception 15: Databases are partitioned primarily in order to circumvent scalability limitations of particular products.

Reality 15: Case studies show that most *large* databases are partitioned, *regardless of the DBMS used*. Surprisingly, we found that scalability was among the least of their concerns, with storage addressability and performance being two scalability reasons actually cited. Among the more important reasons were:

- Platform limitations preclude DBMS scalability (e.g., 2 GB file limitations).
- Additional partitioned systems can be added online.
- The impact on the business is minimized in the event of failures (reliability).
- Individual partitions correspond to distinct aspects of business processing.
- Individual partitions correspond to distinct projects.
- Existing stovepipe applications dictate that partitions correspond to business and political divisions.

Perception 16: Replicated databases are more difficult to reorganize than those supporting table partitioning.

Reality 16: Table partitioning and shared nothing implementations have been treated as good, scalable solutions. In fact, difficulties with data reorganization in a table partitioned environment are one of the most significant reasons that shared nothing implementations fail to scale. Reorganization of table partitioning schemes is not only difficult and disrupts table access, but there are few guidelines for redesign. By contrast, the loose coupling afforded by asynchronous replication permits relatively independent and non-disruptive reorganization.

Perception 17: Asynchronous replication is used to counterbalance scalability limitations.

Reality 17: Asynchronous replication is used to provide cohesiveness and loose consistency among application systems that cannot be tightly integrated. Although some sites have mistakenly attempted to use replication to counterbalance scalability limitation of DBMS products, most sites learned very quickly that this is an inappropriate use of the technology and does not work.

Perception 18: Using multiple databases and/or servers and replication are workarounds for DBMS deficiencies.

Reality 18: Customers use multiple databases and/or servers and replications as particular methods to partition a database and maintain cohesiveness, most often because it fits the business model. They also use a number of other techniques to achieve the same end. For most customers pushing database limitations, a single, integrated database is impractical and would not meet business requirements even if the DBMS could support it.

Perception 19: Clustering is an important scalability solution.

Reality 19: DBMS clustering solutions primarily provide, and are used for, high availability rather than for scalability. Due to limitations in each of the various DBMS clustering solutions, designers must exercise great care to obtain even moderate scaleup or speedup from cross-node cluster resources. These considerations cause a clustered database to be designed more like a federation of loosely coupled physical databases than like a single integrated database.

IV. Conclusions

This study has uncovered a number of differences between the perception of DBMs scalability and reality. Given the importance of scalability in today's DBMS market, vendors, their customers, the press, and analysts need to exercise considerable care before accepting "obvious facts." Such "facts" have tainted our understanding of products and what can be achieved with them. Customers attempting to implement leading edge VLDB solutions all too often have found themselves on the bleeding edge, even though they were told others had already solved most of the potential problems. This report attempts to set the record straight and shed a more rational light on the subject of VLDB and scalability.

Acknowledgements

We wish to acknowledge the help of Winter Corporation (Boston, MA), Sybase, Oracle, and Strategic Partners (Richard Skrinde) in obtaining the data for this study.

APPENDIX

A COMPARITIVE EXAMPLE OF STORAGE EFFICIENCY

As an example of differences in computed storage efficiency we considered what would happen if Oracle and Sybase were each to manage 2 billion rows of data in a single table consisting of three columns: a date and time stamp, a numeric transaction identifier, and a numeric transaction amount. Both NUMERIC columns are precision 38. The table is indexed on the first two of these columns. This example was inspired by data managed in one of the case studies. We used the vendors recommended storage allocation computations to determine the amount of space required for the data and index. In each case we used 10% free space. We made certain extreme assumptions to optimize the storage allocation for an OLTP application:

- Assume maximum concurrency requirements for OLTP, leading to row locking for Oracle at some cost per page.
- Likewise, assume maximum overhead for Sybase for both data and index allocation management.

We then estimated the required log space by the following procedure:

1. Using vendor published TPC Benchmark C data, determine a ratio of the log space Oracle required to that Sybase required (13.37 kB per TPM versus 3.86 kB per TPM or a ratio of 3.46).
2. Compute the amount of log space recommended by Sybase as a percentage of data and index space (25% - Oracle made no general recommendation).
3. Compute the corresponding log space for Oracle by multiplying the published ratio found in step 1 times the value found in step 2 for Sybase.

We assumed the percentage of temporary space required by each product to be equivalent (25%). Finally, the entire database was mirrored for availability.

The results of this computation are summarized in Table III below:

Table I

Computation Step	Oracle 7.3** (< Oracle8*)	Sybase System 11
Row size (native format)	42	42
Row size (db format)	55	46
Rows per data block	22 (30)	39
Data rows per TB	11 (15) billion	19.5 billion
Data blocks	500,000,000	500,000,000
Data allocation overhead	0	7,844 – 250,000
Total data blocks (with maximum overhead)	500,000,000	500,250,000
Index entry size	38	30
Index entries per block	33 (45)	60
Index blocks per TB of data	333,333,333.33 (333,333,333.33)	330,508,476
Index allocation overhead	0	5,185 - 165,255
Total index size (maximum overhead)	333,333,333.33	330,673,731

Total index + data blocks	833,333,333.33	830,923,731
Scaling to 2 billion rows	151,515,151.5145 (111,111,111.1107)	85,222,946.76925
Total index + data (bytes)	303,030,303,029.1 (222,222,222,221.3)	170,445,893,539
Log space (bytes)	147,435,697,911	42,611,473,385
Temp space (25% in bytes)	75,757,575,758 (55,555,555,556)	42,611,473,385
Total before mirroring (bytes)	526,223,576,699 (425,213,475,689)	255,668,840,309
Total with mirroring (bytes)	1,052,447,153,398 (850,426,951,378)	511,337,680,618

** Note: We were unable to compute the Oracle8 allocation because it involves additional overhead, the computation of which is given in the Oracle8 documentation only as constants to be obtained from the installed server online. However, given the fact that an Oracle8 ROWID is larger than that used in Oracle 7.3, Oracle8 additional storage requirements may be significant. Certainly an Oracle 7.x database that is designed to eliminate block free space and migrates to Oracle8 will experience block overflow or chaining.*

***Note: Numbers in parentheses assume that page level locking is provided for in the initial block allocation, rather than row level locking.*