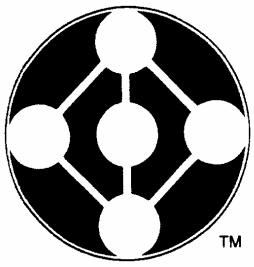

THE
DBMS SCALABILITY REPORT

- BRIEF -

A COMPARITIVE EXAMPLE OF STORAGE EFFICIENCY

David McGoveran
Alternative Technologies
13150 Highway 9, Suite 123
Boulder Creek, California 95006
Telephone: 408/338-4621 Fax: 408/338-3113
Internet: mcgoveran@AlternativeTech.com
Website: www.AlternativeTech.com
Report Number 970718C

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 warrantees 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
13150 Highway 9, Suite 123
Boulder Creek, CA 95006
Telephone: 408/338-4621 FAX: 408/338-3113
Internet: mcgoveran@AlternativeTech.com

Copyright 1997, All Rights Reserved – Alternative Technologies

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 1times 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 III

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.

****Note:* These calculations appear to remain the same for Adaptive Server Enterprise 11.5.