

Sizing Associative Databases

Lazysoft
technology

www.lazysoft.com

Copyright © 2003 Answerbrisk Ltd

Sizing Associative Databases

Associative vs. Relational Architectures

Sentences' unique architecture, the associative model of data, means that gigabyte for gigabyte comparisons with existing relational databases are not meaningful, for the following reasons:

- Relational databases store data items as values: multiple occurrences of the same value are represented and stored discretely, reusing nothing. Associative databases store each data item once only: subsequent references to the same item reuse the original occurrence.
- Relational databases store null marks in empty cells to represent missing data, and in some relational implementations, missing data may take up the full width of a cell. Associative databases record only the data that is present: missing data uses no space at all.
- Relational databases use primary and foreign keys, each of which may comprise several columns, to record relationships between rows, whereas Sentences uses a single association.
- Relational databases may store multimedia files etc inside the database as "blobs" (binary large objects). Sentences manages such data outside the database via URLs.
- All database management systems, including Sentences, use various techniques to manage spare space within the volumes assigned to the database to optimise disk page utilization and data distribution. By default, Sentences will initially write data to about half of each disk page.
- Sentences' page size is variable from 4 kbyte upwards, and this has a bearing on total size. The optimal page size will depend on the operating system: 64 kbytes gives good results with NT-based servers because it is the Windows NTFS cluster size.
- Using the standard Windows NT disk compression, Sentences databases operate with compression ratios of four or five to one without performance degradation.

Relative Size

The size of an associative database is a function of the number of associations that it contains. In broad terms, each occupied cell in a relational table corresponds to one association in an equivalent Sentences database. For example, if a relational table contains a million rows of 10 columns and every cell of every row is occupied (ie. not null), Sentences will use 10 million associations to store the same data.

In sparsely populated databases, the number of empty cells in the relational database should be subtracted from the total number of associations required. So in the example above, if five of the columns are null in half the rows, the number of associations drops to 7.5 million.

The number of associations should be reduced again if the relational database has a high incidence of tables with compound primary keys (ie. primary keys that comprise more than a single column.) A foreign key reference to a primary key the comprises three columns would itself occupy three columns, whereas an associative database would use only a single association to do the same job. So if three of the columns in our example together form a foreign key, the number of associations drops again to 6 million.

Absolute Size and Capacity

Sentences databases are made up of multiple chapters, each chapter being stored as a physical file. A single Sentences database may comprise anything from one up to twenty or more chapters. The capacity of a Sentences database is a function of the capacity of individual chapters. Our scalability testing of Sentences Version 3.0 (due for release during the last quarter of 2002) has already established that it delivers good performance with chapters containing over 100 million associations ("100 mass").

Taking into account transaction logging, multi-dimensional indexing and administrative overheads, a Sentences V3.0 database occupies around 160 bytes per association when first loaded, so a chapter of 100 mass occupies around 16 gigabytes. This is not significantly affected by the size of atomic data items unless they frequently exceed hundreds of bytes, which would be unusual.

At the time of writing, we are embarking on the next stage of our scalability testing, which is to construct a database comprising multiple 100 mass chapters. This will allow us to ascertain how many such chapters may partake in a single database with acceptable performance characteristics. We believe the result is likely to demonstrate that Sentences V3.0 can handle associative databases of over 500 mass (80 gigabytes) in size.

On-line Transaction Processing

The final stage of our performance testing is to develop an associative database on-line transaction processing (OLTP) performance benchmark that will be broadly comparable to the Transaction Processing Performance Council (TPC) Benchmark™ C, which is the most commonly referenced benchmark for relational database on-line transaction processing. The TPC-C benchmark uses a database design expressed in relational form, so is not directly relevant to associative databases. Nevertheless, we believe that such a benchmark will give prospective users reassurance that Sentences can meet their OLTP requirements.