# The Oracle Data Warehouse


**By Sandra Lee**

This article describes how parallel processing software such as Oracle7, in combination with multiprocessing hardware such as the IBM RISC System/6000 Symmetric Multiprocessor (SMP), provides a Decision-Support Systems (DSS) solution that will help companies predict future trends and growth opportunities that will give them a competitive edge in the marketplace.

In an ever-changing marketplace, collected data about a corporation becomes a very valuable asset. The effect of decisions based on this data can have a significant effect on the organization and can be a key to gaining and keeping a competitive edge in the marketplace. In recent years, a market for Decision Support Systems (DSS) has emerged to help companies use their stored data.

Most companies have an Information Systems (IS) organization made up of Operational Systems and DSS that deals with corporate data. An *Operational System* performs Online Transaction Processing (OLTP) functions such as order entry, work scheduling, and financial systems. OLTP gathers and stores organizational data from the day-to-day operations part of running the business. *DSS applications*, on the other hand, are responsible for providing consolidated data for making decisions about the day-to-day and long-term workings of an organization. Although such decisions influence the business, they are not directly involved with the actual running of the business.

In the past, many DSS analysts inefficiently gathered data. For example, a marketing director from a retail clothing company who wanted to do a targeted marketing campaign would ask the DSS analyst for the results of a complex query: "Who are all the people who spent more than $300 using credit cards after the interest rate increase in the four geographical regions of the United States?"

The DSS analyst would then turn to four regional Operational IS groups (or machines) to get a snapshot of the requested data. The snapshots obtained from the Operational IS groups would invariably be taken at different points in time, and sometimes from different data models. For example, the Western Region might track customers who used credit but not dates of purchases, while the Northern Region may track customer purchase dates but not method of payment. Additional research would be necessary to provide requested information not stored in that region's database. In short, this process of getting results from multiple sources with redundant and overlapping extracts would be very time-consuming with a high rate of errors.

Extracting inconsistent data from multiple databases is only one problem faced by DSS. The intensive CPU usage required by DSS to do complex read-only transactions of massive amounts of data is another challenge. Furthermore, OLTP and DSS use logically separate data. To address these issues, some companies began providing expensive proprietary query database computers, pioneering the technique of separating OLTP from DSS. Data from the OLTP computers was separated and consolidated on a query database computer. All DSS transactions would be run against the DSS consolidated query database. The objective of separating OLTP and DSS onto separate computers was to speed transaction processing as well as ease the painful searching for data from several different sources.

Nevertheless, OLTP and DSS remain intimately linked. Consolidating data from company-wide OLTP systems, and perhaps from external sources, constitutes the core of DSS. The term *data warehousing* describes the collection, transformation, and distribution of data useful to

decision-makers. With the right tools, company decision-makers can exploit massive amounts of diverse data to respond quickly and flexibly to rapidly changing customer needs and competitive environments.

In the retail clothing company example above, it would have been more efficient for the DSS analyst to run a DSS application against a consolidated data warehouse that contained information from the four geographical regions plus credit card information loaded from external sources. The resulting data would have been integrated and organized into a consistent form, providing a much faster query result with a high degree of accuracy.

**Consolidating data from company-wide OLTP systems, and perhaps from external sources, constitutes the core of DSS.**

## Parallel Processing in the Data Warehouse

Hardware and software for a data warehouse must support several, often conflicting, requirements.

Hardware requirements for a data warehouse vary depending on the size of the warehouse that needs to be created. Considerations include the volume of data to be kept online in primary storage, the size of the storage subsystem supported by the hardware architecture, and the memory and processor requirements for running batch jobs or long-running complex queries. Access to external storage media or machines may also be important in considering the time needed to load data into the warehouse.

The data warehouse software must support both complex preplanned queries and small ad hoc requests. Easy information addition and updating, and fast query performance are essential.

Both hardware and software technology developed in recent years are beginning to address these DSS requirements.

### Server Architectures

Until recently, data warehouse applications were limited to only those projects that could justify the high cost of implementing a data warehouse. Existing server technology generally did not provide cost-effective access to enterprise data because of architectural limitations.

Lack of client-server functionality for users to perform ad hoc data access and analysis has limited the usefulness of mainframes for data warehousing. In addition, mainframe performance has not kept up with the processing power needed to mine enterprise data.

Specialized approaches, such as proprietary query-processing systems, have provided improved query performance, but at a price/performance ratio that is often expensive. For example, the cost per gigabyte of disk storage on a proprietary query processor can be up to four times that of an open system. Lack of openness also deterred many companies from implementing these systems.

Open systems have provided dramatic improvements in price/performance for OLTP applications and offer the best support for client-server data access. However, with uniprocessors, response time for queries has been limited by the speed of a single CPU, and the performance of complex queries in an online or batch environment cannot meet application requirements.

With parallel processing machines, the limitations of uniprocessors have been significantly overcome for data warehousing applications. By providing multiple processors, performance and response times for complex queries, data loads, and index creation against large data sets are reduced. The price/performance of open systems has improved by 400% during the last ten years alone. Open multiprocessor systems are the only server platforms that can keep pace with users' processing demands and provide adequate interoperability, data access tools, and data distribution.

IBM's RISC System/6000 Symmetric Multiprocessors (RS/6000 SMPs) fill a large market need in DSS applications. RS/6000 SMPs, initially offered with four CPUs and scaling up to eight, provide multiple processors at a reasonable price and respectable performance on high-end machines. When IBM incorporates the PowerPC™ 604 and the 620 chipsets into the RS/6000 family, performance should increase even more. If the data warehousing capacity provided by these RS/6000 SMPs becomes insufficient, IBM will also provide HACMP/6000™ SMPs—loosely clustered SMPs similar to the HACMP/6000 uniprocessors—and massively parallel SP/2s.

### Parallel Software

Although open systems, particularly multiprocessors, have made excellent gains in price/performance, the primary architectural limitation of the DBMS software—until recently—was the inability to take advantage of multiple processors in performing typical DSS operations. Another weakness of traditional Database Management Systems (DBMSs) was an inability to use multiple memories not shared by CPUs.

In recent years, DBMSs have made great strides in parallel processing. Parallel processing allows a task to be broken into many smaller

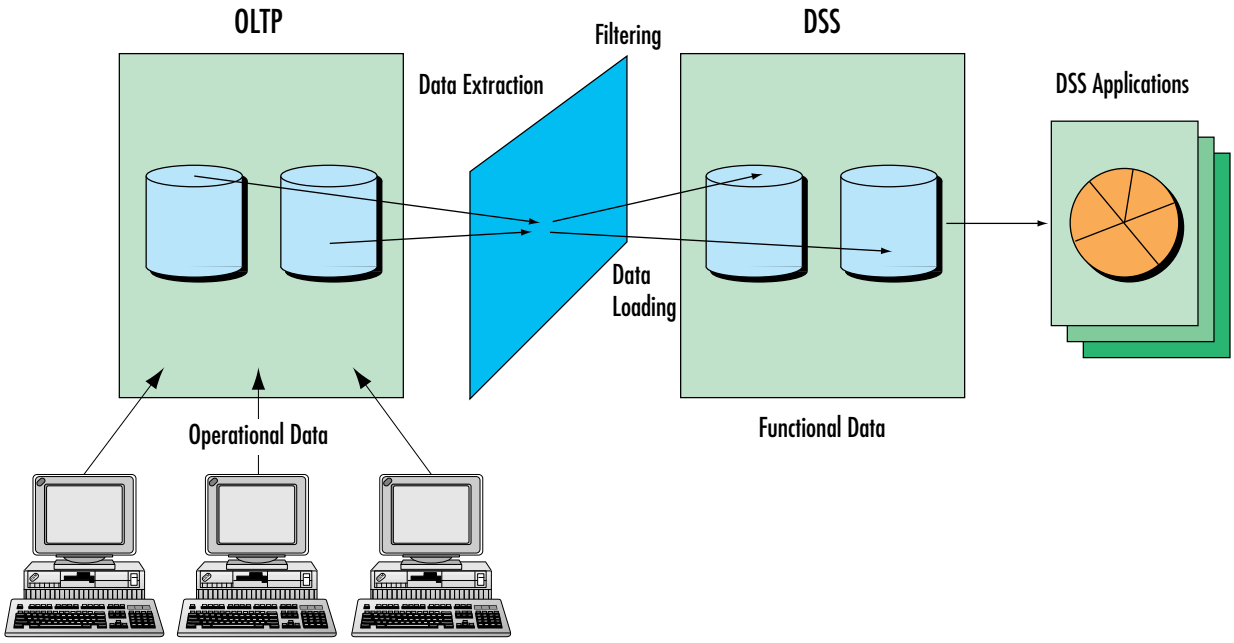


**Figure 1. The data warehousing process**

tasks that are performed simultaneously. This is unlike traditional serial processing in which a single CPU executes instructions from a single shared memory, one task at a time. Because tasks are smaller and shared among several CPUs in parallel processing, a particular task is finished much more quickly, improving response time and increasing the amount of processing that can be done in a given amount of time.

Although parallel processing has been used for CPU-intensive scientific and analytic applications, it has not been widely used in commercial applications because of the difficulty in parallelizing applications. The difficulty lies in structuring tasks so that certain steps can be executed in parallel, while ensuring that the sequencing is preserved for steps that must be executed serially.

### Oracle7

Traditional DBMSs, such as Oracle7, now provide parallel processing capabilities that are appropriate in a commercial DSS environment with SMP and Massively Parallel Processing (MPP) architectures.

The Oracle7 database has a successful history on SMP systems. Oracle7's Parallel Query Option (PQO), Parallel Server, and interoperability enable Oracle7 to support application integration, distributed operations, and mixed application workloads—all part of DSS.

Since it offers excellent price/performance on open systems hardware and is architected to serve a wide variety of enterprise computing needs, Oracle7 PQO is an attractive alternative to specialized DBMSs for data warehousing. Oracle7's PQO processes database requests in parallel, reducing response time for complex queries and for queries against large databases with data warehouses.

The Oracle® Parallel Server enables the Oracle7 DBMS to use multiple nodes on a uniprocessor such as the RS/6000, as well as clusters of multiprocessors such as IBM's upcoming HACMP SMP cluster. Cluster support provides high availability, thereby allowing access to data as needed with reduced database downtime. IBM's HACMP clustering technology permits nodes to be taken down and serviced without affecting the entire system.

Oracle7 runs on many platforms, including the full range of IBM AIX offerings. With Oracle's Gateway technology, Oracle can also access legacy data from other databases and flat files on a variety of different platforms while supporting OLTP and DSS on hardware best suited for a company's particular needs.

## Creating the Oracle Data Warehouse

There are several steps in creating a data warehouse, which will be more than just a collection

of OLTP data. Careful design, maintenance, and good data retrieval tools are essential for a data warehouse to be a valuable component of DSS.

Figure 1 provides an overview of the processing of operational data into functional data for decision support. Day-to-day operational data entered into a traditional OLTP database is extracted, filtered, and then loaded into a DSS system. From there, the data is extracted by applications for decision-support use.

### Designing the Warehouse

Although good DSS software will improve the performance and usefulness of a data warehouse, careful database design is crucial to its effectiveness. A good data warehouse must support both preplanned decision-support applications and ad hoc queries relating items that may have little in common, such as linking payment methods with a particular zip code.

**In one test on a 16-processor system, scaling of more than 95% was measured for the Oracle7 Direct Path Loader.**

The data warehouse model must also address the conflicting goals of system flexibility and data delivery performance. Flexibility is required to be able to add tables and attributes of primary data and summaries. Quick query response is also required since it determines the number of analyses that can be performed and the amount of data that can be fed to a presentation package for dynamic data representations.

Although these issues are common to all relational databases, the primarily read-only nature of a data warehouse permits some specialized optimization techniques, such as generating summarized data tables for frequent query paths. Information in data warehouses can be summarized within the database as well. Such summarized data can be used for simple, ad hoc queries while the more detailed information can be used in complex, prewritten queries.

An IS organization designing a data warehouse must decide what type of data should go into the warehouse based on the type of information most often used, the type of queries that will be run, what information should be summarized, and so on.

### Loading and Maintaining the Warehouse

Data that will eventually be stored in the data warehouse must first be extracted from the OLTP database. The data is then filtered to remove unwanted information and possibly combined with data from other databases or files. At times, additional data such as a date or location may be added, or the data may be converted to allow comparisons with similar data, such as converting all revenues to U.S. dollars.

The example retail clothing company's data warehouse would contain information from the four IS regions that has been massaged into the same form and contains the same information. For example, the data warehouse would include purchase date and method of payment for all customers in all geographic regions. Additional credit information from external sources, such as a credit card company, may also be added to the data warehouse.

Once OLTP data is extracted and filtered, it must be loaded into the DSS database. The time required to load data into a data warehouse can be a critical factor in the success of a data warehouse system, especially for maintenance. Maintenance can require periodic incremental updates of the data in the warehouse: adding or changing data that has been changed since the last update of the warehouse. Periodically, a complete refresh of the data in the warehouse may also be needed.

Oracle7's parallel data-loading capability speeds the creation and maintenance of the data warehouse, particularly when adding or updating information. Oracle7's PQO provides a parallel load function that uses the already efficient direct path of SQL*Loader. Oracle7 Direct Path Loader bypasses SQL processing to load data directly into database tables. Rows of a single table can be loaded simultaneously by different processors. Multiple sessions running SQL*Loader use the direct path capability to load data simultaneously into the same table. Each SQL*Loader session acts independently, producing near-linear scaling of performance and reducing the time required for data loading. In one test on a 16-processor system, scaling of more than 95% was measured for the Oracle7 Direct Path Loader. While this result is spectacular, it must be remembered that highly parallel systems are highly tunable. The data warehouse must be carefully designed and tuned for best results.

Parallel indexes, which can be created either during the data loading process or just after loading, are often needed in a large database such as a data warehouse. Indexing speeds creation and reconfiguration of a data warehouse. Oracle7's PQO includes a parallel index feature. It automatically executes a `CREATE INDEX` command in parallel across multiple CPUs by parallelizing the table scan and sort associated with index creation. Oracle7's parallel index function provides good scaling, but it is important to remember that

index creation performance is heavily dependent on the amount of sort area available for each of the index build processes.

Data in the data warehouse constantly changes. New information must be added, old data updated, and the data warehouse itself may have to be rearranged for better performance on changed query needs. To do this, *data replication*—the maintenance of several consistent copies of data across different databases—is necessary. The data warehouse must be synchronized with data in the OLTP database, and summarized data within the data warehouse must be kept consistent.

Oracle7's data replication capability helps data warehouse maintenance. Oracle7 can transparently replicate data to and from all databases used to store data, including OLTP databases, DSS databases, and combined OLTP/DSS databases. Using the Oracle Gateway product, Oracle7 can access data stored in non-Oracle databases or flat files.

### Using the Warehouse

Retrieval and presentation of contents by DSS applications will determine the worth of the data warehouse. The design of the data warehouse is key to efficiently using the indexes, summary data, and other tuning techniques that ensure rapid retrieval of desired data. The data is more valuable if both anticipated and unanticipated data requests can be rapidly completed. As non-operational data is added to the data warehouse, there will be many unanticipated queries.

Oracle7's PQO is essential to obtaining full value from the data warehouse. It speeds queries by splitting a complex query into smaller parts that can be processed in serial, as shown in Figure 2. Oracle7 also ensures contention-free queries whether or not the queries are parallel, so users attempting to write data do not block users attempting to read data, and vice versa. For more information about Oracle7's PQO, see the May 1994 *AIXpert* ("Oracle Parallel Technology Empowers AIX Systems," page 37).

Data retrieval also presents query results from the data warehouse. To create customized DSS applications for retrieving data, Oracle provides CASE and the Cooperative Development Environment (CDE) tools. Tools such as Oracle Forms and Oracle Reports enable application developers to create interfaces to retrieve the results of prewritten complex queries into a particular format. Oracle also provides end-user query tools, such as Oracle Data Browser or Oracle Data Query, for ad hoc queries without the need for users to know underlying database structures. Oracle supports hundreds of third-party tools, giving companies a wide range of tools choices to fit their requirements.

Today, computer technology is fueling the emergence of data warehousing as an essential component of successful businesses. Parallel processing software such as Oracle7 in combination with multiprocessing hardware like the RS/6000 SMP, provide a DSS solution that will help companies predict future trends and growth opportunities, and give them a competitive edge in the marketplace.

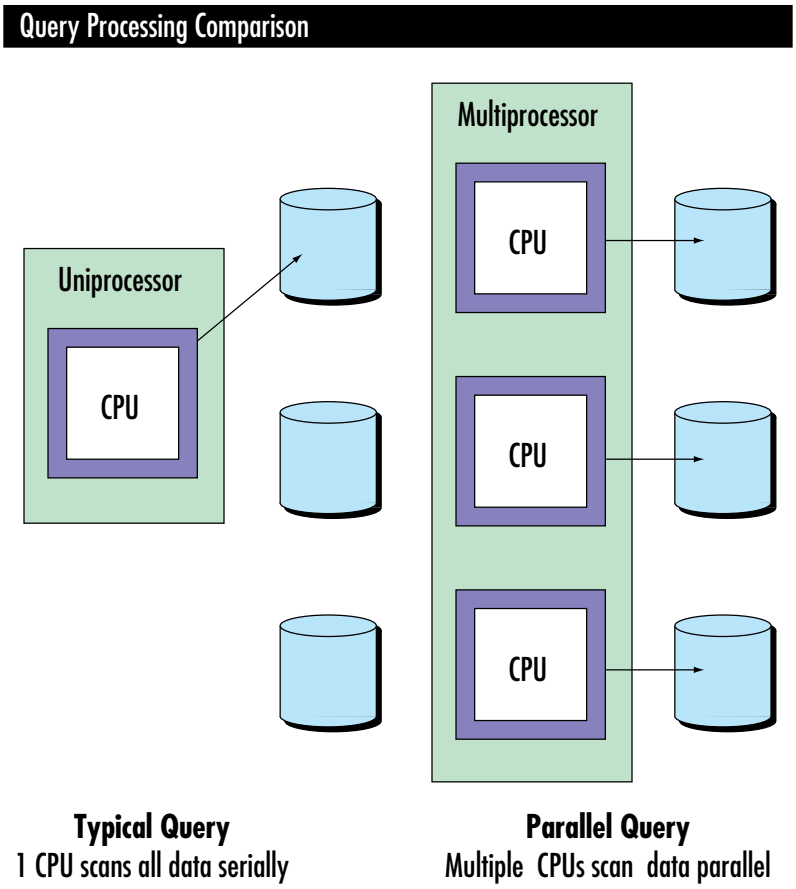


**Figure 2. Query processing on uniprocessor versus multiprocessor**

**Sandra Lee**, Oracle Corporation, 500 Oracle Parkway, Box 659406, Redwood Shores, CA 94065. Internet: shlee@us.oracle.com. Ms. Lee is the product line marketing manager for the AIX platform at Oracle. She has BA degrees in Computer Science and English from the University of California at Berkeley and a graduate degree from Boston University.