Infostructure Associates

2005 Database Report: OnLine Analytical Processing and the Virtual Operational Store

Sixth Edition

June 2005

Infostructure Associates
an affiliate of Valley View Ventures
24 Muzzey St.
Lexington, MA 02421
Telephone: 508-265-1558
www.valleyviewventures.com

TABLE OF CONTENTS

24 Muzzey St.
Lexington, MA 02421
Telephone: 508-265-1558
Fax: 781-862-6236
www.valleyviewventures.com

TABLES

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Database Report

Preface

Over the last one and one-half years since our last report, the pace of innovation in databases themselves has slowed, but the pace of innovation in related areas such as OLAP (online analytical processing), grid computing, RFID, and EII (Enterprise Information Integration) has been swift. As a result, the focus of IT buyers seeking competitive advantage via databases should properly be on "database plus":

- Database plus integrated OLAP,
- Database architected for grid computing,
- Embedded database plus vertical application,
- BI (business intelligence) database extended by EII,
- Database plus EII for the "virtual operational store" to achieve the "virtual real-time enterprise,"
- Database plus EII and data-movement tools for strategic information management or "information integration,"
- Database plus business-process modeling and development tools for business process integration, and
- Database plus business-process modeling tools and RFID software for RFID infrastructure software.

In other words, there are many exciting new developments in the database market; only, they do not involve databases alone.

This Infostructure Associates report aims to provide advice for the IT buyer in how to cut through vendor hype and focus on the business value-add that "database plus" technology and database products can bring — and sometimes do not. We take a balanced view: new database implementations should bring not only competitive advantage but also cost savings — and yet IT does not achieve cost savings simply by choosing the product with the lowest license cost. For that reason, we consider Infostructure Associates research in database and database-plus TCO (total cost of ownership), especially in describing vendor products.

It seems to some IT buyers that the buying decision is straightforward: pick a corporate-standard database (e.g., Oracle) and use it for all corporate needs. Unfortunately, our research shows that this aim not only is extremely difficult to achieve, but that it can in fact lead to *higher* costs and *less* competitive advantage. For one thing, our research shows that the most scalable and comprehensive products often involve the highest administrative costs. Typically, the administrative simplicity achieved by forcing all data sources into one database is more than counterbalanced by the additional cost of that database's administration compared to a more customized and more "near-lights-out" database. For another thing, most major organizations have multiple existing databases and data sources from multiple vendors, many hidden from view as "embedded" databases supporting an application. Practically speaking, it has proved impossible to remove all of these.

We recommend, instead, that IT buyers pick the right database for the job, with due attention to architectural complexity. We would add two other factors in information-architecture design and the buying decision that, in our experience, are too often overlooked:

1. *Attention to TCO and ROI*, or Return on Investment (where the database is used as part of a development), and not merely license costs. While TCO and ROI calculators are not always the best guide to the buying decision in some circumstances, it is possible to draw some general conclusions about a *type* of database (e.g., one designed for data warehousing and large OLTP versus one designed for "embedded" use beneath a medium-scale vertical application). Our research shows that TCO can vary by as much as an order of magnitude, and ROI over three years by 200 or 300 %, depending on the type of database chosen — even when license costs are approximately equivalent.

2. *Attention to proprietary information as a source of competitive advantage*. The typical organization has information about its customers and suppliers that no one else has. It is increasingly proving easier to leverage that information by harvesting it from existing databases and thereby achieving competitive advantage, than to build a new competitive-advantage application that is increasingly easy for a competitor to duplicate rapidly.

This report includes:

- A summary of *key report findings*.
- *Key criteria* for choosing a database, and the technologies or features that indicate vendor superiority in meeting those criteria.
- *New-technology overview*. This pays particular attention to new "database plus" technologies and approaches that are beginning to prove their value or should prove valuable over the next one to two years.
- *Market size and growth projections.*
- *Supplier Analyses.*

This report is intended for IT buyers, IT strategists, CIOs, and CEOs who seek an understanding of databases in order to implement corporate strategies more effectively. Infostructure Associates believes that this report will aid users in buying, architecting, implementing, and carrying out "best practices" for their data-oriented solutions.

Chapter One: Key Report Findings

- The typical transaction stream in today's market continues to shift away from the traditional OLTP stream-of-updates and data warehousing stream-of-queries towards "mixed" transactions more evenly divided between updates, random reads, and queries. Key applications driving this trend are Web and Web-service solutions, packaged applications, and portals.
- The amount and proportion of non-relational data in the typical organization continues to increase, to the point where the total amount of storage devoted to text, spreadsheet, and "rich-media" (audio, video, graphics) data is now a major proportion of overall storage. Key technologies to handle this are content managers, EII (Enterprise Information Integration) tools, "object inside" upgrades to major relational databases, and XML support.
- The so-called mid-tier ODS (operational data store), sometimes combined with an EII tool, is offering the querying power of data warehouses plus the "real-time" data freshness of OLTP systems, as well as the ability to handle content, for a vital subset of enterprise data. A mid-tier ODS combined with an EII tool can form a "virtual operational store" (VOS) which can deliver most of the benefits of the much-touted real-time enterprise.
- RFID has been implemented widely, with most implementations being "slap and ship", with small databases to handle local-warehouse processing of moderate volumes of data. However, present RFID software infrastructure is not well suited to the large volumes of data that should occur in some cases over the next five years, with small databases unable to handle a projected 4 terabytes of data per day, and large systems not well prepared to integrate with RFID's business-process code and metadata ("metadata in motion").
- Database administration costs, not license costs, continue to dominate five-year TCO for systems with less than 1000 end users.
- The Web services market, although slow to develop, is causing a slow improvement in the database market. Web-service implementation success demands a "Web-service-friendly" databases that allows programmers to use existing data cost-effectively rather than "reinvent the wheel," while the Web service makes that data available to a wide audience.
- The right database as the core of a development and production platform (e.g., the embedded database used by an ISV) can improve TCO and ROI for an application by 20-30%, according to recent Infostructure Associates studies.
- Moreover, databases are an under-utilized source of competitive advantage, as proprietary information increasingly becomes key to longer-term competitive advantage. However, this information is typically not all locked in one relational database, so an EII tool that can query across databases complements existing databases.
- Lower-end "niche" databases continue to flourish. IBM's recent push into the medium-sized business market with DB2 UDB appears to have expanded the market but not significantly impacted existing niches.

- While retarded by the recent slump in the telecommunications market, mobile/small-footprint databases that support handhelds and cell/smart phones continue to be a wise investment as a complement to enterprise databases.
- As predicted, EII is driving a significant proportion of database sales for new implementations.
- Key database assessment criteria for IT buyers should include performance/scalability, robustness, flexibility/extensibility, and programmer productivity.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Chapter Two: Key Criteria and Technologies for Today's Databases

Databases are now everywhere, and most readers are familiar with their basic concepts and uses. Therefore, this chapter will spend relatively little time on terminology and on the typical benefits of the enterprise relational database. Rather, we focus on what differentiates today's databases — the technologies that make a particular database especially well suited for a particular use.

However, readers should carefully note one bit of terminology — or rather, one unfortunate change in terminology. Until the last seven years, the term for software that focused on carrying out operations on large quantities of data was a database management system, or DBMS; and the data on which those operations were carried out was called a "database." About seven years ago, vendors popularized the term "database" to mean "DBMS." Because many, including vendors, use the term "database" to mean both the software and the data, it is important for the reader to clearly distinguish between the two.

We will follow that newer convention in this report — a *database* is software that carries out operations on large quantities of data, and the data on those operations are carried out is called a *data store*. Parenthetically, we may note that when an EII tool performs queries on multiple data stores of multiple types from multiple vendors, each data store is called a *data source*, to indicate that between the tool and the data store lies a database that "intermediates" the transaction.

Overall Key Database Criteria

Table 1 below lists the key criteria that we recommend that IT buyers use in choosing a database for a new implementation, as well as the key technologies that contribute to some databases' ability to meet the criteria better than others.

Table 1: Key Database Criteria and Technologies

Criteria	Technologies
Scalability	OLTP scalability technologies: SMP support Clustering support Multithreading Decision-support scalability technologies: Query optimization Replication Cost optimization Bit-mapped indexing "Mixed" scalability technologies: Stored procedures Distributed-database synchronization (2PC) Running the application "inside the database" Load balancing
Robustness/administrative costs	Online, parallel load and backup/recovery, monitoring, and metadata management "Zero" and "near-lights-out" administration" tools and automated reorganization Cross-database tools
Flexibility	Standards support SQL Java/EJBs XML Complex-data-type support rich-media support file support- object and object-relational technology content management technology Integration with infrastructure software application servers infrastructure APIs and component libraries
Programmer-productivity support	High-level development environment support Java programming support Frameworks
Specialized technologies	Main-memory technology Small-footprint mobile/wireless technology Non-relational and "post-relational" technology OLAP technology

Source: Infostructure Associates, June 2005

Database Scalability Criteria

Over the past ten years, database performance and scalability has improved incrementally and for particular transaction patterns, rather than dramatically as a result of theoretical breakthroughs. Nevertheless, the result of these incremental and localized improvements has been an overall performance improvement of 30-40% from release to release, for a total improvement of more than an order of magnitude — overall, databases are more than 10 times faster than a decade years ago.

Unfortunately, as quickly as database scalability technology has improved, user needs plus database sizes and the speed of hardware and networking have increased even faster. The globalized Web has brought massive increases in the number of potential end users of a database; database sizes have moved from 100 GB to the tens of terabytes at the high end; Moore's Law has made processor speed more than 20 times as fast as a decade ago; and networking for the common user has gone from 14K modems to multi-Mbps cable and DSL modems. Now more than ever, optimizing database performance and scalability is key to application success.

Infostructure Associates defines three overall types of transaction stream handled by the typical database:

1. *OLTP (online transaction processing).* This involves mostly updates to single small data items, scattered randomly throughout a data store. Typical applications include airline reservations and order entry. In fact, major enterprise databases such as Oracle and IBM DB2 UDB were originally designed for OLTP.

2. *Decision support.* This involves queries that access ("read") a large number, or all, of the (usually small) data items in a data store. Typical applications involve data mining, business intelligence, enterprise reporting, and OLAP (online analytical processing). Decision-support data is typically stored in a data warehouse that may or may not be "refreshed" (fed data) by smaller data marts, OLTP data stores, and applications. A variant of decision support is "batching", in which the database is taken offline so that a large stream of inserts, updates, and deletes is applied to it — this approach is still used in some mainframe applications.

3. *"Mixed".* This combines the updates of OLTP and the queries of decision support, and adds single-data-item reads — the proportions of these transactions, and the volume of the stream, typically vary rapidly and widely. The major applications involving "mixed" transaction streams are enterprise and vertical applications such as ERP; Web-site support (which involves large "rich-media" data items such as graphics and video); and the operational data store, or ODS (which provides "near-real-time" data access to end users at the price of being able to handle a much smaller data store than the data warehouse).

Major enterprise databases (such as Oracle, IBM DB2 UDB, Microsoft SQL Server, and Sybase Adaptive Server Enterprise) boast the ability to scale the highest of all database products in handling OLTP and decision support transaction streams. However, other databases that were designed to handle larger data items and "mixed" streams may do better in providing embedded databases for vertical applications, supporting Web sites, and acting as an ODS.

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

Universally accepted performance and scalability benchmarks are rare. The Transaction Processing Council offers TPC-C to assess OLTP database (and server) performance (now rarely performed), TPC-D to measure decision-support performance, and TPC-H to measure "mixed"/Web performance. Usually, only the large database suppliers participate. Infostructure Associates believes that when users take care to avoid comparing this year's benchmark from one supplier with last year's from another, TPC benchmarks can be of significant help in determining not only relative database performance and price/performance, but also improvements in performance and price/performance from year to year. At the same time, your mileage may vary, and IT buyers should consider doing their own benchmarking where the transaction stream involved is unusual.

OLTP Scalability Technologies

Scaling a database for an OLTP transaction stream is a matter of parallelism — coding the database process so as to maximize the ability of the server on which the database runs to parallelize transactions. Generally speaking, a server (or servers) can parallelize the processes (including transactions) that run on it in one of three ways:

1. *Shared-memory*. In this case, transactions operate in parallel on shared main memory and shared disk. Symmetric multiprocessing, or SMP, is the usual form of shared-memory parallelism.

2. *Shared-disk*. In this case, disk is shared, but each processor in the server (or across multiple servers) has its own memory. When multiple servers are involved, this is referred to as *clustering*. In single-system-image clustering, the operating systems coordinate to make several servers appear as one to the user, administrator, and developer.

3. *Shared-nothing*. In this case, neither main memory nor disk is shared. Effectively separate systems coordinate by high-speed, high-bandwidth links, as in fault tolerant systems such as HP Tandem.

Each of these does best with a different number of processors. Loosely speaking, shared-memory performs best with 2-32 processors, shared-disk with 32-100 processors, and shared-nothing with 100 processors and up; but there are many exceptions to this rule. Most databases support SMP optimization, but relatively few support clustering and/or shared-nothing optimization — because most of today's implementations scale beyond 32 processors *horizontally* (by adding servers and database copies) rather than *vertically* (by adding processors to a single server).

Multithreading and SMP support

The most effective way for a database to take advantage of SMP servers is to write the database code so that it is easy for the server to allocate transactions between servers with the maximum of parallelism. The way to do this is to write the code so that multiple processors can be accessing the same code at once — an idea called *multithreading*. A database should also be "SMP-aware" — that is, able to proactively increase parallelism by telling the processors how to schedule its transactions. Today, all enterprise databases and most databases are multi-

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

threaded and SMP-aware; however, IT buyers seeking highly scalable OLTP systems should verify SMP performance, either by benchmarking on 4- and 8-processor systems to see if the database "scales up linearly", or by referring to TPC benchmarks done on SMP systems.

Clustering

Today, most clustering is at the operating system level, and is not integrated with database transaction processing. It is also true that most clustering today is used to improve system robustness, as clusters can rapidly "fail over" from a failure of one server, and recover quickly when that server is brought back up.

The major exception to both of these trends today is Oracle RAC (Real Application Clusters). This solution was originally derived from innovative Digital Equipment technology that Infostructure Associates personnel assessed in its original form eight years ago. We found that typically clustering could not achieve better scalability than 70% performance improvement per one processor added — not enough to make clustering scalability cost-effective. By contrast, the Digital Equipment solution — and now Oracle RAC — can achieve 80% scalability — at least comparable to SMP scalability. The major reason for this improvement is that the clustering software effectively "load balances" between servers and between processors. Moreover, Oracle has tuned the technology to work at the transaction, not the process, level, meaning that RAC works much more effectively to scale transaction streams than other clustering solutions.

Infostructure Associates interviews with Oracle users suggest that many have not opted to acquire RAC because they don't need the additional scalability. For those who do, however, this is an effective way to scale beyond SMP.

Other OLTP Scalability Considerations

The IT buyer should note that relational databases are usually especially appropriate for OLTP and for scaling OLTP. Relational databases were originally designed for OLTP, and relational databases excel at "normalization" — the ability to store data on disk so that data items do not need to be duplicated. This minimizes the storage needed, thus indirectly improving performance (because less space used on disk means that the disk head need not seek as far).

The IT buyer should also keep in mind that Internet OLTP has changed the OLTP transaction stream significantly. Non-Web OLTP has meant fairly steady and predictable transaction streams during business hours (e.g., bank deposits and withdrawals). Web order entry takes place globally 7x24, and its transaction stream may be unpredictable and "bursty" (subject to sudden surges).

One idea that has been widely touted to handle this new OLTP transaction stream is the "information utility," or "data on demand." The basic approach of the "information utility" is to make backup processing power available automatically when transactions surge, much as an electrical utility will make additional power available at times of peak demand. Oracle and IBM, in particular, have been in the forefront of redesigning their software to achieve the "information utility" — e.g., by incorporating "flexible capacity" features into storage management and operating system software, or offering "dynamic partitioning" on high-end systems.

However, IT buyers should be aware that because in many solutions the storage management software, operating system, and database come from different vendors, the "information utility" is some distance from realization in the real world.

Decision Support Scalability Technologies

The usual decision support transaction stream is dramatically different from the OLTP stream. Where the OLTP transaction is an update, applied to a single data item located anywhere in the database, requiring "write locks" that prevent access to the data item by any other transaction, the decision support transaction is a query involving reads of a large number of data items, accessed in some order that will usually reflect physical data-item clustering on disk, and requiring only "read locks" that prevent only OLTP-type updates from accessing each data item queried. In fact, in a data warehouse, no updates are permitted at all — except when queries are halted once a day for a "mass download" of fresh data.

In decision support situations, therefore, scalability is less about parallelism and more about getting large chunks of data from disk to main memory (so it can be processed) as quickly as possible. Note that because "read locks" do not bar other queries from accessing the same data item, queries can operate in parallel with no delays. Also, grouping data on disk to reflect the most likely pattern of queries (e.g., all orders in France go together) is vital to optimizing performance and scalability, because data grouped on disk can be loaded into main memory as swiftly as possible. Thus, some relational data is "de-normalized" by creating extra data-item copies, so that performance will be improved where several different query patterns are frequent.

Decision support also differs from OLTP in its limits to scalability. OLTP systems run out of gas when too many end users submit too many updates at once, or when too many updates take too long to back up each day. Decision support systems reach their upper limit when too much data is stored on disk, and cannot be loaded into main memory fast enough. OLTP systems often reach their upper limits in the 100 GB (gigabyte) to 1 terabyte range; some decision support systems are nearing 100 terabytes, or even a petabyte.

The typical decision support architecture is a data warehouse, or smaller data marts than feed a central data warehouse. The data warehouse uses ETL (extract, transform, load) software to load (and "cleanse") data from both OLTP databases and application databases (e.g., ERP), typically once a day. Trained "data miners" perform queries on the database using specialized analytics tools, in order to gain insights about sales patterns, and the data warehouse may also be used to generate global "enterprise reports."

OLAP (OnLine Analytical Processing) takes decision support one step further. It generates a special set of indexes ("cubes") to allow more detailed data analysis for specific purposes, such as budgeting, forecasting, and accounting. The end user is typically not a data miner, but rather a company CFO or sales executive. OLAP is typically applied to a data warehouse, but may also be applied to "mixed" or OLTP databases.

Most databases optimized for decision support provide one or more of the following decision-support scalability technologies:

1. A "cost-based query optimizer". This software "pre-parallelizes" queries based on their "cost", or the likely processing time of each subquery, thus improving query performance and scalability.
2. Bit-mapped indexing. Certain types of data can be represented as "yes-no" bits for certain types of query. By keeping indexes representing this data in main memory and performing operations on the indexes rather than the data, decision support databases can achieve up to 3 orders of magnitude improvement in performance for these queries.
3. Replication. Replication software automates and speeds the ETL process, improving the timeliness of data-warehouse data.

Cost-based Query Optimizers

"Intelligent" or "cost-based" optimizers use statistics and directives from administrators to determine the likely processor and disk-access times of subqueries of a query sent to the database, and then determine the best way to split the query (and to run subqueries in parallel with other queries) in order to minimize these "costs." The result is highly parallelized, rapid-load-from-disk querying.

Note that because the cost-based query optimizer itself can slow performance, the optimizer code is itself parallelized. Also, the database will "dynamically partition" main memory for further speedup of multiple queries.

The best cost-based query optimizers are highly sophisticated engines that incorporate a decade of real-world experience. The IBM, IBM/Informix, Microsoft, Oracle, and Sybase relational databases have all accrued the necessary experience to provide exceptional cost-based query optimizer performance and scalability.

Bit-Mapped Indexing

Bit-mapped indexing is based on the idea that many data items' values (e.g., numeric and simple text values) are duplicated or take only two or a few values. Bit-mapped indexing creates one-bit yes/no references to these values, and stores these references in an index kept typically in main memory. Then, when a query asks whether a data item has a particular value ("how many of my sales are in the U.S.?"), the database scans the index for values, rather than accessing each data item on disk. Because main-memory access is typically at least 1000 times as fast as disk access, bit-mapped indexing can achieve up to three orders of magnitude improvement in simple-query performance. Moreover, because complex queries can sometimes be carried out by Boolean operations between indexes, a surprisingly wide array of queries is susceptible to this kind of speedup.

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

The performance gains from bit-mapped indexing can be phenomenal. For example, a query that requires examination of an entire 100-terabyte database could theoretically demand 30 hours simply to load all of the data into main memory. Processing a bit-mapped index that corresponded to those 100 terabytes would take less than a second!

However, bit-mapped indexing is not a tool for all queries. For example, if there are too many values for a data item, indexing takes more space than the data item itself does. Also, the queries and data items to which bit-mapped indexing applies tend to change over time, so administrators must be prepared for ongoing index creation and deletion, even when the database is running.

A key factor for IT buyers assessing a vendor's bit-mapped indexing is the degree of its automation. For example, Sybase claims that Sybase IQ product automates a great proportion of the index creation and deletion tasks, while the system is running, with minimal performance overhead. The Microsoft SQL Server bit-mapped indexing implementation, by contrast, trades high administrator flexibility for added administrator effort and cost.

Replication

Replication is one-way copying of a data item between data stores. As noted above, replication is used most frequently to deliver fresh data to data warehouses. In general, replication is also used to pass data between distributed databases. Thus, to understand the value of replication, we need to understand that replication is applied when a data store reaches its "limits to growth." In this case, some users choose to partition the data store (separate it into several pieces, with no data item in common between the pieces). However, partitioning rarely works for long in the real world — partitions get unevenly large or small, slowing performance or requiring constant "rebalancing" performance overhead and administrative costs.

The alternative that typically is chosen, sooner or later, is a distributed database, in which there is a significant amount of data copied between data-store pieces. These copies are kept "synchronized" — i.e., every transaction must see the same value for the data item at all times, no matter which copy is changed first.

Where there are a significant number of updates in the transaction stream (OLTP and "mixed" streams), the technology of choice is *two-phase commit*. However, two-phase commit has significant performance overhead (it must exchange "handshakes" between the two data stores twice while "write" locking both data items). As a result, two-phase commit is really feasible only for a low amount of data copies.

Replication, by contrast, cannot handle situations in which two copies must be kept in synchronization at all times, but is fast where updates from one data store must be flowed to another. Thus, replication is ideal for data warehouses and operational data stores, which accept one-way data flows from OLTP and application data stores, and can take their time about recording data-item changes.

IT buyers seeking to find the best replication solution should focus especially on the performance overhead of the solution, especially on the source database. The act of copying a data-item update each time it occurs, and of sending that update on to the destination data store, can cause significant slowdown in vital OLTP systems. Log-based replication is typically best, because it uses the same mechanism that databases employ to back up transactions for recovery purposes. On the destination system, such as a data warehouse, various techniques, such as periodic "mass downloads" at times of low querying demand or offline, reduce the performance overhead.

"Mixed" Scalability Technologies

Infostructure Associates research shows that there are now three general types of "mixed" transaction streams:

1. The application stream. Packaged or in-house applications typically involve updates, reads, and queries of relational-type "structured" data in relatively even proportions. For example, ERP systems with "embedded" databases log orders, allow verification of particular orders, and support analysis of buying patterns.
2. The operational data store stream. This involves a high proportion of both updates and queries of primarily structured (but also semi-structured [text] and unstructured [graphics]) data, as "fresh" business-critical data is continually arriving, to be used for "real-time" decision-making based on data analysis.
3. The Web stream. This typically involves a high proportion of reads of large unstructured data objects (graphics, video), and to a lesser extent semi-structured text, but with some amount of querying (search) and updates (e-Commerce).

"Mixed" scalability is now a highly important characteristic of a database, because all three types of "mixed" transaction stream have seen growth far beyond the OLTP and decision-support streams, and in fact unprecedented growth. IT buyer that wishes to choose a "corporate standard" database should emphasize above all finding one that delivers "mixed" scalability.

IT buyers should look for the following "mixed" scalability technologies in today's databases:

- "Transaction-balancing" technologies that are tuned for "mixed" transaction streams (e.g., by using different caches for updates and queries) and adapt quickly to changes in the mix of updates, reads, and queries.
- "Rich-media" support, i.e., specific performance and scalability features for semi-structured and unstructured data (e.g., "object" and "object-relational" database technology, complex-data-type support, and content managers).
- Stored procedures and business rules. These "mini-applications" can run inside the database engine for greater performance.
- Two-phase commit. As noted above, two-phase commit allows a database to avoid "limits to growth" — up to a point.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

- The ability to write an application to run "inside the database." Typically, this involves "Java inside", which allows data-accessing Java applications and EJBs (Enterprise Java Beans) to improve their performance.

Business Rules

Business rules are "mixed-transaction" code that reflects the rules and practices of the enterprise, embedded in an enterprise database that can capture the resulting complex relations between data items. For example, a business rule may sound an alert when a large customer's payment is overdue by 30 days, or (in order to require that suppliers meet certain quality and financial criteria) automatically compute each supplier's rating for these criteria, alert the enterprise when suppliers fail to meet the criteria, and carry out security checks on end users accessing supplier data.

Business rules are often highly useful — they provide a cost-effective way to clarify and enforce company policies, they can improve data quality and information security, they can simplify programmer's jobs, and they can improve program quality. They are also high-performance — they are precompiled, they are usually more highly optimized, and they run in the same process as the database.

Triggers, alerts, and *stored procedures* are key technologies for creating business rules. The stored procedure is a piece of code stored in the database and that runs in the same process as the database engine. An alert is a specialized stored procedure that delivers a message to an end-user, programmer, administrator, or application. A trigger contains a set of criteria and a stored procedure that is invoked when these criteria are met. A business rule can be expressed as a trigger, an alert, or just a plain vanilla stored procedure.

However, business rules have a downside. Today's enterprise business rules have typically accumulated over two decades, often with little documentation. As a result, IT is increasingly noting a "legacy business rules" problem, as complex masses of existing business rules make changes in business rules or superseding code difficult to implement.

Enterprise database vendors provide some business rules and other useful stored procedures. IT buyers should note whether the vendor provides particularly useful business rules for a particular vertical industry.

Two-Phase Commit

As noted above, where there are multiple data stores (often created in order to scale higher than a single data store) and one of the data stores is dedicated to decision support, replication is the way to pass data between copies. However, for OLTP and "mixed" distributed databases, the way to go is two-phase commit, or 2PC.

Where a data item has more than one copy in different databases, 2PC aims to make every change to that data item appear as if it happened simultaneously in all copies. The result will be that copy A in information base A appears to every transaction preceding and following the change as identical to copy B in information base B. 2PC does this by "locking out" all other

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

transactions on all data item copies (phase 1), changing all copies, verifying that all copies have been changed (phase 2), and then unlocking all the copies.

The outstanding example of a "mixed" transaction stream relatively well suited to 2PC is the Web site. Web-site implementations typically involve many reads, a few updates, and a few queries. To scale the Web site, the user can create a new copy of the data store on a new server, then update all copies of a data item simultaneously via 2PC, blocking only global queries, and those only for a short time.

Databases vary in their ability to support 2PCs. IT buyers should look for 2PC support that makes the programmer's (or administrator's) job of creating and synchronizing data-item copies easy.

Running the Application Inside the Database

Stored procedures already run "inside the database"; but running an application "inside the database" can also have a surprisingly large effect on "mixed-transaction-stream" performance. The reason is that today's database may often be running on the same machine as other processes — such as applications invoking the database — and has therefore learned to exercise control over the machine's scheduler, so that the database gets first priority and other processes second priority. Today's major enterprise databases now allow users to create applications that run "inside the database" by running in the same process as the database engine. Thus, these applications also get first priority.

Typical Web implementations, in which applications are on second-tier servers and databases on third-tier servers, therefore do not run the applications "inside the database." However, many enterprise applications, packaged or in-house-developed, run on the same server as the database, and therefore placing the application inside the database can yield a major performance and scalability boost. Likewise, applications invoking an operational data store typically benefit from being placed inside the database. Where the vendor advertises "Java inside", that means that new applications written in J2EE can be written to run inside the database.

IT buyers should look particularly at what programming languages the database vendor supports "inside the database". For example, many support "Java inside", but not all support "C^{++} inside."

Infostructure Associates Robustness and Manageability Criteria

The general move in today's administrative tools towards "self-healing" and "proactive" tools has not left database administration unscathed. A decade ago, administrators focused on avoiding unexpected downtime; today's administrative technologies try also to minimized expected downtime, recover rapidly from system failures, and make the administrator's job as easy as possible — to the point where, in some real-world cases, people with no database-administration training can keep a database running almost indefinitely, by following the directions for backup on weekends.

However, as databases have improved in these areas, the requirements for database management have "raised the bar" even faster. One key factor causing today's stringent database-administration requirements is the enormous rate of growth in information storage. Overall, storage sizes required by the average large enterprise have nearly doubled every year over the last decade, and show no signs of slowing down. These growth rates, in turn, place enormous stresses on the high-end database's administrative tools — simply trying to keep the "backup window" for a large application from growing is an ongoing concern not only for a storage manager but also for the database administrator.

The other key factor in increasing database requirements is the rise in prominence in database-administration costs in figuring five-year total cost of ownership. A decade ago, with solutions whose license costs could run to the millions of dollars, administrative costs at the high end were relatively negligible. Today, PC and blade servers are driving down hardware costs, and network and storage costs are also decreasing sharply — but the fully loaded salary of a database administrator is going up. Therefore, to minimize TCO, the large enterprise must make do with more databases per administrator — and only "manageable" and "proactive" database-administration tools can achieve this.

To achieve the new "quality of service", IT buyers should look especially to the following technologies:

- Online, parallel tools, e.g., load, backup/recovery, monitoring, and index management.
- "Zero-administration" or "near-lights-out" tools, such as automated reorganization.
- Cross-database administrative tools.
- "Grid" or "clustered" tools with rapid failover.

Online, Parallel Database Administration Tools

Typical database administration tasks today include backup/restore, roll-back/roll-forward, reorganization, and mass load/copy. Of these, the most important is usually backup/restore. This is because restoring a database after a backup is the final option that ensures that most of the transactions applied to a database are performed, even after a failure; and because backup of a large database typically causes either the most performance overhead or the most scheduled downtime of any database administrative task.

Backup involves copying a data store's contents at a point in time to permanent storage (typically tape; but disk that is slower and lower-cost than usual is now sometimes used). Restore involves copying a point-in-time backup from permanent storage back to the data store. The more frequent the updates to the data store (as in OLTP), the more frequent the need for backup, because backup only allows restoration to a point in time, and if a failure occurs well after the backup, all transactions between the two points in time are lost (although roll-back/roll-forward technology can often recover to a point much closer in time to the system failure).

Long experience shows that as the data store grows, the burden of backup/restore on the administrator and the effects on IT grow as well. For example:

- *Administrative costs increase*. As the amount to be backed up increases, "backup windows" grow to encompass nights and weekends, with administrator monitoring required in case the backup runs into trouble — resulting in a major increase in administrator costs.
- *System availability decreases.* Typically, administrators carrying out night- and weekend-long backups bring down the database to do so. If the database offers online backup, then administrators do not have to bring the database down, but the backup software causes major performance overhead, as it competes for both processor time and use of data items.
- *Recovery after a system crash is slower.* The restore process, being the backup process in reverse, likewise slows as the amount of data-store contents backed up at a point in time increases. For example, restoring a terabyte database with 100 GB/hour restore can require half a day.
- *System performance decreases over time.* If backup takes too long, users often feel that there is not enough time for data-store reorganization — and without reorganization, database performance inevitably becomes slower and slower over time.
- *Effectively, the data store cannot grow above a certain limit.* Because most users will not accept a backup time greater than a night or weekend, the speed of backup can determine the point at which the database creates more trouble that it is worth.

As noted above, *parallel* and *online* technologies provide partial answers to these problems. Parallel technology allows the administrator to perform the I/O for each data item to be backed up/restored at the same time as other backup I/O, resulting in 10x or 100x backup speedup in many cases. Online technology performs backup while the database is running, improving system availability dramatically at the cost of lower performance during backup. Parallel technology speeds offline or online backup; online technology can avoid the backup window altogether. Most of today's database suppliers offer parallel, online backup/restore.

In recent years, subtly, database backup/restore has become more and more similar to the backup/restore performed by storage managers (which typically involves end-user spreadsheets, files, and text). For example, most enterprise databases now advertise the ability to store any enterprise data in their data stores. As a result, Oracle has announced storage management software integrated with Oracle Database 10g that carries out many of the functions of a SAN (storage area network), and can replace a SAN.

The task of the IT buyer in assessing database administrative tools for their "scalability" is relatively straightforward: look for the widest possible array of parallel, online utilities; look for the highest performance in backup/restore, if possible by solutions that integrate database and SAN backup; and search for solutions that offer the greatest flexibility in carrying out "partial restores", which avoid the need for a time-costly full restore.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

"Zero Administration" Tools and Reorganization

We have referred to the need for reorganization before now; the real reason that reorganization is needed is "data fragmentation." As data items are added and deleted over time, gaps between items that should be "clumped" for optimum performance grows greater and greater; thus disk seek times get longer and longer; and thus database performance gets slower and slower. In effect, what happens is "storage sclerosis" much like the buildup of fatty deposits that prefigures a heart attack. The cure is a database reorganization utility, which backs up the data store, then restores it optimized on disk for today's typical transactions and without fragmentation.

IT buyers should look for a parallel, online reorganization utility that "self-activates" or starts itself on a fixed schedule, without administrator intervention. Because other key administrative tools in today's database can be scheduled likewise, having a parallel, online, automated reorganization utility is a good sign that the database can be operated with minimal administrator intervention ("near-lights-out" administration). It is interesting to note that users of today's "niche" or "Low-IT" databases report that these provide better "near-lights-out" support and are more robust than today's enterprise databases.

Cross-Database Administrative Tools

The rationale for cross-database tools begins with the fact that for many organizations, it simply is not practical to insert all information in one database. End users constantly generate their own text files and spreadsheets; many packaged applications use their own databases, and it is too costly to substitute the "corporate-standard" database; local offices need autonomy to thrive, and use that autonomy to violate IT standards. In fact, some organizations may not know that their "art department" is using the FileMaker database on Macintosh computers — and is quite productive doing so.

Every so often, enterprise database suppliers attempt to dominate the market by trying to persuade their users to move all information to multiple copies of their databases. This would indeed cut users' administrative costs in some cases; but, in the real world, this "global data migration" has proven almost impossible to achieve.

The "second best" solution is to create administrative tools that span multiple vendors' databases — so that, for example, users can back up all data in a local office in a coordinated way. There are two main sources for this type of software: storage software firms, which provide backup and restore in SANs and NASs (network-attached storage) on a "block" and "file" basis; and utility suppliers such as Compuware and Computer Associates, which provide a common interface to carry out major administrative tasks across enterprise databases (IBM, Microsoft, and Oracle) as well as legacy mainframe databases (IMS, ADABAS, and DATACOM DB).

IT buyers for whom cross-database administrative tools offer significant database-administrator cost savings should look particularly for tools that coordinate backup (and restore) across multiple vendors' data stores, with a high degree of flexibility about scheduling backup. This will allow the administrator to adapt "globally" to changes in storage growth trends rather than focusing on each database at a time.

"Grid" or "Clustered" Database Administration Tools

These tools manage administration across databases from the same vendor. Typically, they manage a "cluster" of databases running a single application that are already linked by a clustering (or application server) product, or a "grid" of tens and hundreds of databases (many of which run on PC or blade servers).

Oracle offers examples of both "clustered" and "grid" database administration. Oracle provides cross-Oracle-database administration as part of its application server product suite, and as part of Oracle Database 10g's grid capabilities.

IT buyers should look for clustered or grid administrative tools only when the database already supports this architecture, not when a new architecture must be introduced to support the administrative tool — the administrative cost savings may not counterbalance the added cost and complexity of architecture implementation. The tools should allow the administrator to "load balance" between nodes in the cluster or grid dynamically.

Infostructure Associates Flexibility Criteria

Today's database is often required to be far more flexible than in years past. Software infrastructure standards proliferate, especially those associated with the Internet; the database must support a wide array of new data types, transaction streams, and end users; and in order to deliver performance while interoperating with application servers and Web servers from different vendors, today's database must integrate effectively with popular Web architectures (such as the service-oriented architecture) and development tools.

Thus, IT buyers should look especially for the following flexibility features in a database:

3. *Effective support for key standards*. That is, there are too many standards for effective support of them all; but the database should at least support key transaction standards such as SQL [and perhaps XQuery]), plus key Web-service standards such as XML.

4. *Easy addition of, and development using, new complex data types*. Key complex data types introduced by the Internet include text files, graphics, video, audio, and "compound" data — so-called "rich media" data.

5. *Relatively tight integration with infrastructure software*. Today's database should not only integrate with application servers and with business-component libraries, but also support creation of and access by portals and Web services.

Transactional and Web-Service Standards

Structured Query Language (SQL) is common to all major databases. It is an English-like command language for coding transactions on relational data. The relevant standard is ANSI SQL.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

XQuery is an attempt to extend SQL to objects and rich-media data, formatted as XML messages. So far, XQuery has not gained wide acceptance in the database world, although it is now close to a check-list item in EII products. Infostructure Associates anticipates that within the next few years, XQuery will gain far wider acceptance by major database vendors.

The ODBC (Open Database Connectivity) standard allows developers to access data from over thirty relational and legacy databases using a common SQL format; its variant, JDBC, does the same thing specifically for Java programmers. Every major popular database supports ODBC and/or JDBC.

While object and XML databases have had relatively little success at the high end of the market, the XML standard is now supported in one way or another in all major databases. Moreover, XML databases have had some success recently as repositories, message brokers, and mid-tier ODSs.

A key Web-service feature that some enterprise databases now support is the ability to present the database's stored procedures, or the database itself, as a Web service. While it is unclear as yet what advantages accrue from this — after all, SQL itself already provides a simple interface — certainly the ability to invoke administrative commands may have significant future benefits.

IT buyers should particularly focus on XML databases as flexible alternatives to enterprise databases in some circumstances. Because XML databases do not have to convert (XML to relational) and "deconvert" (relational to XML) data for which the data store is one stop on its journey, XML databases have significant performance advantages in areas such as RFID (which, as noted below, can be thought of as "metadata in motion"). For any database, IT buyers should emphasize the basic standards outlined above rather than standards such as ebXML, because these are not as well established and may yet prove a waste of time.

Complex Data Type Support

Over time, databases have accrued a broad and varied set of "end users" — internal programs, Web site end users, workgroups within the enterprise, partners/suppliers, internal workgroups, data miners, intranets via a LAN (local area network) or remotely, and IT administrators. The aim of today's database is therefore to be "many-to-many" — to draw from a wide variety of data types, and present it to end users in a wide variety of ways (see Table 2).

One key technology in doing so is XML. Either by storing data as XML, or wrapping it in XML when it is delivered to the end user, the database can provide any type of complex data and allow the "end user" to determine the format, in a standardized way that makes this determination fairly straightforward. Note also that XML support makes the task of exchanging data with suppliers and customers (supply-chain management) easier.

Table 2: "End Users" and Their Data Needs

"End User"	Data Typically Accessed	Data Format Typically Required
Internal programs	Relational	Relational
Web-site End-Users	Text, graphics, relational	HTML
Partners/Suppliers	Relational, text	XML
Internal Workgroups or Intranets	Spreadsheet, text	HTML, graphical
Data Miners	Relational	Graphical
Administrators	All types	No requirements

Source: Infostructure Associates, June 2005

Most database vendors now provide some form of XML support. IBM, Oracle, and Microsoft also allow storage of and transactions on rich-media data, although not necessarily in XML format. In one way or another, these vendors also allow "mixed transactions" that operate on both relational and rich-media data. In XML databases, of course, these capabilities are built in. IT buyers should emphasize the degree to which operations on "mixed" and non-relational data are transparent to the end user, seeming like traditional relational operations.

Infrastructure Software Support

The key infrastructure software with which today's databases should be tightly integrated includes:

- *Major application servers and TP (transaction processing) monitors.* In a 3-tier architecture, both application servers and TP monitors (typically used only for mainframe databases) provide "pre-multiplexing" and "pre-load-balancing" of transactions arriving at the database. Thus, the more effectively these work with a database's own load-balancing capabilities, the better the performance and scalability. Oracle and IBM are notable for integrating their own database with their own application servers.
- *Portal infrastructure.* Because enterprise portals typically seek to combine data from multiple back-end databases, the database should deliver data to the portal infrastructure wrapped in XML, as most databases can do nowadays.
- *Web services infrastructure —particularly XML, SOAP, UDDI, and WSDL (Web Services Definition Language).* Often a database is used as a Web services repository. Thus, it should be able to support UDDI communications; store the Web-service information in WSDL; deliver data in XML; and send the data via SOAP.

Application Servers and TP Monitors

IT buyers should seek a database that supports the application server that the user wishes to employ — preferably an application server that provides transaction load-balancing and multiplexing, and, if possible, coordinated load-balancing between application server and database.

Enterprise Portal Infrastructure
IT buyers should look especially for a database that delivers rich-media data to the portal in an accepted format, such as XML. Here, XML databases are especially effective as second-tier caches for enterprise databases that can deliver data to the portal in the requisite format.

Web Services Infrastructure
While, as noted above, a database can serve as the repository of multiple data types, it typically does not provide the global access to all data sources via a common format that an EII tool does. Therefore, to simplify the programmer's job and improve productivity dramatically, IT buyers should look to pair a Web services development tool with an EII tool rather than just adding another database.

Infostructure Associates Programmer-Productivity Support Criteria

In 2005, increasing numbers of CIOs and CEOs are finally beginning to realize that *the choice of development solution and platform matters to the bottom line*.

Why now? The short answer is "***speed to value***." Today, as never before, IT and ISVs can cut the time to create a high-value application from 2 years (or never!) to 12 months, simply by choosing a different development solution. At the same time, the value of that additional 1-1 ½ years to the organization has skyrocketed, because computer applications are now a major source of competitive advantage, and competitors can copy that advantage in shorter and shorter periods of time. In other words, when product turnover is 3 years or less, being alone in the market for half that time is worth far more than in the old days, when IT mattered less to competitive advantage and long product lifecycles meant that competitive advantage lasted for less of the product's life span.

However, the short answer does injustice to the full sweep of the new trend. This is about speed to *value*, not just speed to market. Speed to value involves two other key criteria for the organization:

1. Cost
2. Risk

That is, an application may be valuable not merely because it gives competitive advantage, but also because it cuts costs, or mitigates corporate risk.

Obviously, an application can be aimed directly at cutting costs or implementing business continuity. However, it is also becoming increasingly clear that the same tools and infrastructure components that speed development also cut application cost of ownership (which is a significant corporate cost), and reduce the risks that the application will fail (which is a major factor in business continuity). And of these infrastructure components, the database is by far the most important in delivering speed to market, reduced cost of ownership, and reduced corporate risk. This is confirmed not only by Infostructure Associates TCO/ROI re-

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

search, but also by the negative impacts of reduced attention to data-access coding since the advent of the Web.

The unfortunate fact is that many of the Web-based business-critical applications built from scratch have been written in the low-level Java programming language, which has a complex interface with databases (sometimes referred to as the "object-relational mismatch"), or have been retrofitted to access data because the application was not scalable enough, with little reusability. The result has been overall *decreased* programmer productivity.

The object-relational mismatch has been perpetuated by Java standards organizations' decision to foster EJBs (Enterprise Java Beans). In many cases, these make data-access slower than alternative solutions, applications less scalable, and Web architectures more complex.

The database can play a major role in overcoming these problems (along with the development toolset). Specifically, the database should include:

- Relatively high-performance EJB libraries and access to "EJB-like" stored procedures.
- Support for and/or partnership with higher-level, more productive development toolsets.
- Provision of a "framework" that includes high-level components or interfaces to the database, as well as application server, security, administration, and e-Business APIs that are as high-level as possible.

Because the programmer-productivity tools that most databases offer are presently uneven at best, IT buyers should look for real-world demonstration of speed to value, as well as relatively high-level components tailored for a specific vertical industry.

Infostructure Associates "Database Plus" Criteria

The IT buyer should also take note of additional relatively new database technologies that target a particular area of transaction processing. These include:

- *Main-memory technology*. This operates the database entirely in main memory. Main-memory databases such as TimesTen are especially appropriate for medium-scale applications where performance is vital (such as keeping track of and analyzing the stock market "in real time"), or as a cache to speed up an enterprise database. For example, an RFID software infrastructure supplier uses a main-memory database to process local-warehouse RFID data in real time. IT buyers should emphasize the ability of the main-memory database to coordinate with major popular enterprise databases.
- *Small-footprint mobile/wireless technology*. This is described more fully in the analysis of Sybase's iAnywhere Solutions. This technology is exceptionally useful in extending informational power to mobile and wireless devices and creating the "mobile workplace" for on-the-go or remote workers.
- *Content management technology*. These are databases tuned specifically for rich-media data, including a very different set of transactions than the typical relational joins and selects. While content managers have not been very successful in the market, they have achieved

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

enough of a presence that EII tools find it very useful to provide SQL-type access to them. Over time, it appears that enterprise databases will incorporate many of their functions.

- *Non-relational and "post-relational" technology*. The analyses of Intersystems and Ipedo (an XML database) describe alternative approaches to data management. As noted, these have achieved success in the market by providing scalability for data that relational databases have traditionally not focused on, such as complex data types and semi-structured data. They can be especially useful, paired with an EII tool, in creating a VOS (virtual operational store), as described in the next chapter. IT buyers should look especially for performance in transactions on both relational and non-relational data.

- *OLAP technology*. This is described in the next Chapter. OLAP extends the benefits of decision support beyond data-mining experts to key organization executives, and gives deeper insights into the data.

Whatever the "database plus" technology may be, it still must typically operate in an environment with relational databases. Therefore, it is important for the IT buyer to verify that the "database plus" technology interoperates well with major popular relational enterprise database such as IBM's, Microsoft's, and Oracle's.

.

Chapter Three:
Key New Database Tasks — What Should You Watch Out For?

Infostructure Associates finds that four new IT information strategies can have an especially powerful positive impact on an organization over the next 2-3 years:

1. Handling the upcoming flood of RFID data.
2. Implementing "*strategic information management*."
3. Creating a VOS, or *virtual operational store*, to support a "virtual real-time enterprise."
4. Implementing OLAP.

RFID Data — "Metadata In Motion"

RFID (radio frequency identification) implementation arrives now. One vendor estimates that 90 of the top 100 WalMart suppliers have provided some sort of RFID tags on at least some of their products by the end of January, 2005.

RFID implementations yield data — but data of a different kind. Typical implementations use an RFID printer to slap a tag on a product or component during production or on entry into a warehouse, then monitor its progress through a warehouse by periodic "pinging" using stationary RFID readers. Thus, the "core data" in RFID implementations — unique identification of a product — is always accompanied by semantics, or metadata, or data about data, describing where that product is in a business process, or its physical location. The core data typically never changes; the metadata changes very frequently. To put it another way, RFID is "metadata in motion."

Three other characteristics also make RFID data management different. First, RFID "reading" is imperfect, and likely to stay so. Thus, when stacked on a "pallet," a widget is readable from some angles and not from others. A sudden inability to detect the widget may mean that it has been lost, or that it cannot be read. Sophisticated filtering software can reduce the amount of these "false negatives," but data management at the local level must monitor, alert, and handle "false negatives" in order to aid the manager of the warehouse, while "cleaning" the data as far as possible so that it can be used by enterprise data miners.

Second, RFID data has the potential to be "the flood to end all floods." That is, each widget as it moves through the production and distribution process is constantly generating a new "ping." Even with sophisticated filtering software, there may be thousands or tens of thousands of updates of RFID data (i.e., changes to where tens of thousands of products are in a business process or where they are physically) per minute at "peak load" — an exceptional OLTP situation. If we add the requirement that we track where each product has been, say, every half hour (a "process trail"), then we also generate an enormous amount of static (unchanging) data.

Third, more than any previous type of data, RFID data is metadata in motion *across organizations*. The intent is that each WalMart or Department of Defense supplier not only tag its products but also pass the RFID data for a product from supplier to customer, all the way down the

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

supply chain. Thus, WalMart can query its supplier for the status of an item, or Dell can track a shipment from customer order to delivery — and beyond. Every RFID data store is potentially accessible from outside the organization; every RFID data management system potentially queries across organizational boundaries.

Because RFID has arrived, and because RFID data management is different from anything that has come before, databases need to be retuned or re-architected to handle the new needs — as has happened with data warehousing, content management, and Web data in the recent past. Specifically, enterprise information strategists need to answer two questions:

1. What is the appropriate database architecture for RFID data?
2. What type of database is best for handling RFID-type transaction streams and data operations?

Note that in the long term, there is a third question about RFID data: how can my organization leverage RFID data most effectively, by itself or in combination with existing data?

RFID Architecture: The Three-Level Solution

The appropriate database architecture for an RFID implementation reflects the appropriate overall architecture for that implementation. Real-world implementations are tending toward a "three-level" overall architecture, as follows (see also Figure 1):

1. At the lowest level, a "*buffer*" receives the data from RFID readers and printers, filters it for duplicates, "cleans" it if possible to eliminate "false negatives", and adds semantics to the data for use at the next level up. As of yet, no database is attached to this level — but it could be.
2. The middle level is the *local* level. This typically involves one physical location, such as a warehouse, and the aim here is to allow the manager at the location (such as a warehouse manager) some ability to monitor, fix problems, and do analysis. Often, also, the local level allows the user to define workflow within the physical location. A local database supports these tasks. The database may also add semantics to the core data, both for local workflow and to support global analysis at the top level.
3. The top level is the *organization* or *enterprise* level. At this level, the user may monitor across locations (and across organizations), define workflow across the entire business process, and perform business analysis on the combined data from multiple local databases, either separate from or combined with other enterprise data. Thus, the RFID data store may be separate from other enterprise databases' data stores, or may be flowed to a data warehouse or OLAP database. Note that at the organization level, a user may "reach across" not only to access another organization's enterprise RFID database, but also to probe its local-level databases.

As a result, the database architecture is often "double three-tier", in which three client/-application/database-server tiers operate at both the local and enterprise level.

The Right Database for the RFID Job

Infostructure Associates therefore believes that RFID implementers should consider the following when choosing databases for the local and enterprise levels:

- The local-level database should be highly scalable, because of the potential for a flood of data, and close to "real-time" in its performance, because of the need for occasional quick reaction to RFID problems.
- The local-level database should support network and systems management capabilities, allowing alerts, console-type monitoring, and coordination with enterprise systems management solutions.
- The local-level database should support workflow semantics, allowing the user to track where in the process a widget is.
- The local-level database should provide some ability for the warehouse manager to analyze RFID data, e.g., in order to optimize the local part of the business process.
- The enterprise-level database, separate from or combined with other data, should provide semantics to allow a combined analysis of RFID and non-RFID data (e.g., POS data and customer records).
- The enterprise-level database should support powerful analysis tools (e.g., OLAP).
- The enterprise-level database (or an EII tool) should allow access to other RFID data stores, at least at the enterprise level and if possible at the local level, in other organizations.
- The enterprise-level database should be highly scalable, to handle a flood of RFID data. It should do especially well at update- or insert-type transactions. It may make sense to move "historical" data to a separate "static" database, to avoid problems with the "query from hell."
- The enterprise-level database should have some capability of *pushing* data down to the local level, as when the local manager needs to be alerted that products are scheduled to arrive. The ability to *pull* data (e.g., trigger a local-level upload of the latest data about a particular product) may also be useful.

It should be understood that using an existing database, or a "standard" database vendor, without verifying whether that database can be tuned for the unique characteristics of RFID, is a Bad Idea. Today's databases have been optimized for different transaction patterns than RFID affords. Perhaps the technology that comes closest to having the necessary capabilities is in-memory databases or "post-relational" databases, at the local level, and mid-tier operational data stores, at the enterprise level.

Leveraging RFID Data

Clearly, the major benefits of an RFID database accrue from use of an organization-wide RFID database at the enterprise level — all that the local warehouse manager can improve is turnover for one warehouse.

In the long term, RFID infrastructure — including an RFID database — can deliver positive bottom-line impact. RFID infrastructure allows greater control over the supply chain, and therefore greater optimization for bottom-line expense-cutting. RFID infrastructure ensures that RFID delivers a large amount of new actionable information to the corporate decision-maker, potentially both at the retail level (how does product placement on the shelves relate to buying behavior?) and at the production and distribution levels (are we forcing product on a vendor further down the chain?). In fact, RFID data, when enhanced by RFID-infrastructure "semantics", can provide a source of customer satisfaction, allowing buyers to monitor shipment more closely across the supply chain.

To achieve this, RFID users should emphasize the analytic capabilities of their RFID database, either via BI tools or OLAP. Moreover, RFID implementers need to create a "global metadata repository" that captures the relationships between the RFID product-process data and existing customer and product data. An EII tool is effective at doing this; or if higher performance is needed, a new data mart may be created. It is less likely that inserting RFID data in the organization's existing data warehouse will do the trick, as RFID data is only semi-structured, and the likely large size of the RFID data store may overburden an already-stretched data warehouse.

The Benefits of Strategic Information Management

Infostructure Associates continues to argue that we are seeing a fundamental shift in effective strategies for achieving competitive advantage via IT. Previously, in-house-developed or packaged applications would do the trick. Today, the lag time before a competitor duplicates the application is becoming shorter and shorter — less than two years, in many cases.

The answer, and the new strategy that competitors cannot match, is to use proprietary information about your customers and suppliers, gaining insights and offering value-add that your competitors cannot duplicate. The competitive advantage gained by putting together previously unexamined relationships between bits of proprietary data is also long-term, because no counter-moves by a competitor can duplicate the knowledge of customers and suppliers gained by proprietary relationships with them.

The key barrier to achieving competitive advantage by leveraging proprietary information is that this information is scattered across a wide range of data sources. Therefore a BI (business intelligence) tool or in-house-developed querying tool cannot realistically support identification of key competitive-advantage insights. Instead, IT needs to implement a broad strategy of (a) placing information where it is rapidly accessible, and (b) ensuring that the relationships between the data are better identified — *strategic information management*.

Key tasks in strategic information management are:

- *Identify* the enterprise's key information resources.
- *Manage* them (i.e., make sure that at any given time they are in the right data repositories).
- Support users' efforts to *leverage* them (i.e., provide powerful cross-data-source tools and metadata repositories identifying data relationships).
- *Add* to them (identify new data sources that crop up and place them in the overall information framework).

Note that strategic information management has benefits well beyond competitive advantage — although competitive advantage by itself is sufficient to justify an effort at strategic information management. Enterprise-wide ability to cross data sources also:

- Enables better enterprise portals, data mining, and Web searches;
- Gives a clearer view of the enterprise's information assets;
- Speeds development (because programmers only need to code one data-access routine rather than one for each project or for each data source); and
- Cuts administrative costs (because database administrators manage across multi-vendor data sources).

As an information integration tool, EII has an important role to play in strategic information management. Many tools can be used to support strategic information management, including ETL tools, EAI tools, data warehouses, EII tools, and ODSs. Of these, the two key tools are EII and an ODS, because only these aim to support enterprise-wide cross-data-source metadata. And, as noted, these work more effectively in combination. Therefore, a combined EII-ODS solution can act as the keystone of a strategic information management framework.

Strategic information management, in turn, can allow better implementation of key enterprise strategies. Table 3 suggests specific ways in which a combined EII-ODS solution can be useful in some typical business strategies.

Table 3: Ways to Use EII and ODS in Strategic Information Management

Aim	Ways to Use EII and ODS
Combining business processes ("composite applications")	Replicate data involved in each business process (e.g., a customer or job record) to the XML database. Write the composite application to XQuery or SQL and the ODS, replicating the changes back to the business processes. This speeds development and ensures that data is updated in a consistent fashion (e.g., that if the customer cancels an order, then both sales and manufacturing see the order as cancelled immediately). Persist data views aggregated across applications in the ODS. This improves performance for cross-application queries and updates.
Implement Web services "back-end" consumer interfaces in existing applications to their databases	Write a Web services consumer application to invoke EII.
Implement "real-time" or "on-demand" reaction to new customer or supply-chain data	Replicate high-update customer or supply chain data to the ODS. Use the ODS to offload processing from enterprise databases. Use XML's extensibility to implement changes to a wider variety of data.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Improve development "time to value" for applications leveraging proprietary information (e.g., enterprise portals)	Leverage the ODS and EII to front-end proprietary information (e.g., in the data warehouse). Developers will thus write one module instead of one for each data source.
Audit key enterprise information for government requirements	Create an enterprise-wide metadata repository using EII that imports metadata semi-automatically from key repositories such as the data warehouse and rich-media sites. Note that the EII and ODS typically allow incorporation of email data — a likely demand from regulators and attorneys. For legal discovery and auditing, write an XQuery or SQL query to the EII interface.
Cut IT costs	Cut development costs as noted above.
Improve multi-tier application scalability	Aggregate structured and unstructured information from back-end systems of record in the ODS, and synchronize the information with these systems. By allowing the user to offload query processing from the back-end systems, a mid-tier ODS can deliver results to front-end users faster.

Source: Infostructure Associates, June 2005

The Usefulness of XML Standards for Strategic Information Management

XML as a "lingua franca" for disparate data types has long ago proven its worth, as a common format and as a widely-adopted technology for message handling and Web messaging. XML has been implemented in most EII tools and mid-tier ODSs, and XQuery is steadily gaining ground in those markets. In the risk mitigation and regulatory areas, the XBRL (Extensible Business Reporting Language) extension of XML is proving useful for preserving key documents' original formats and supporting Sarbanes-Oxley regulators' detailed queries.

XQuery allows SQL-like querying and updating against disparate data types. XQuery also permits data to be transformed, EII-style, into a common XML format during querying, as well as result formats to be generated suitable for applications and end users. Therefore, XQuery is especially appropriate for EII-type queries against rich-media and file as well as relational data, or to access data in XML format in an ODS.

An Assessment of Strategic Information Management Tools

Strategic information management has both a short-term component and a long-term component. The short-term job of strategic information management is to deliver available information consistently, no matter what the variations in end-user demand are — "on-demand" information. The long-term jobs of strategic information management are to identify, manage, support the leveraging of, and add to the information resources of the enterprise. Tools to support the short-term job include storage management and SAN (storage area network)/NAS (network-attached storage) management (especially virtualization), database and cross-database administration, system and systems file administration, and Web-site administration. Tools to support the long-term jobs include metadata repositories and cross-database administration — tools to provide full support for strategic information management have yet to surface, and tools to allow automated searches for sources of new proprietary or extra-enterprise information that can be used for competitive advantage are not yet visible in the market.

Strategic information management tools allow users to combine data — either by physically moving or copying it, or by combining it temporarily as part of a transaction — across data-source frontiers.

Strategic management tools include:

- Data migration and mass copy/download tools.
- Replication and ETL (extract, transform, load) tools.
- Data warehouses and data marts.
- Enterprise and mid-tier ODSs.
- EII tools.

Table 4 shows one way to categorize strategic information management tools.

Table 4: Taxonomy of Strategic Information Management Tools

Tool	Tasks	Uses
Data migration and mass down-load	• Usually *moves* large amounts of data at once, permanently or at scheduled intervals, typically always when the target data store is not online • Typically one source, one target	• Terminate a data store by moving its data and operations to another data store (e.g., standardize on one database) • Apply database-type transactions to a new data type (e.g., spreadsheets) by copying the data into a database • Backup and recovery
Replication and ETL tools	• *Copies* data one piece at a time, one way, either continuously as data changes or in packets (typically offline) • Can transform data to the different format of a data warehouse and "cleans" the data for higher quality • Typically many sources, one target	• Update a data warehouse or data mart with new or changed data from business-critical OLTP relational databases, often hourly, daily, or weekly • Reflect local-office data to a central site (e.g., backup or combining the data for reporting) • Synchronize laptop and local or central data stores • Copy data between packaged applications,

		e.g., to integrate business processes (EAI)
Data warehouse or data marts	• Receives *copies* of large amounts of data, often offline (data marts may copy to data warehouse) • Typically many sources, one target • Allows querying of combined data but not update • Often very large (terabytes)	• Support querying, data mining, and business intelligence • Provide metadata repository of key business information
Enterprise ODS	• *Copies* (one way), typically changes in data (immediately) — often *from* the store/cache to multiple targets, always online • Usually many sources, one target • Often uses replication or ETL tool • Allows querying and update • Often large, but not as large as data warehouse	• Combine customer and supplier information to detect relationships and use for competitive advantage • Combine business-critical application data for business process integration • Improve time-to-value and speed-to-react by seeing more up-to-date data than data warehouse can provide • Improve visibility of current (less than a day old) information
Mid-tier ODS	• *Copies* (one way) typically changes in data (immediately) — often *from* the store (used as a cache) to many targets, always online • Usually many sources, one target • Often uses replication or ETL tool • Allows querying and update • Smaller than enterprise ODS	• Combine customer and supplier information to detect relationships and use for competitive advantage • Access file and rich-media data • Combine business-critical application data for business process integration • Improve time-to-value and speed-to-react by seeing more up-to-date data than data warehouse can provide • Improve visibility of current (less than a day old) data • Improve performance/scalability of key applications, especially “composite” ones
EII	• *Preserves* the data sources as they are	• Supports queries/reads/updates across a broad range of data sources (including file

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

	• Often uses real-time ETL tool or "adapter • Allows querying and (in some cases) update • Transforms data to combine it for each transaction	and rich-media data) without creating copies to synchronize • Creates a more comprehensive metadata repository than any other strategic information management tool • Allows development that includes cross-data-source transactions • Potentially, supports cross-data-source administration

Source: Infostructure Associates, June 2005

Table 5 notes ways in which particular strategic information management tools can support specific strategic information management tasks.

Table 5: Uses of Tools in Strategic Information Management

Task	Value-Add of Tool
Identify and Add	• ETL tools and data warehouses can combine metadata from relational production databases • EII tools can semi-automatically "inhale" relational metadata and add file, rich-media, spreadsheet, and extra-enterprise Web metadata.
Manage	• Data migration and ETL tools can copy relational data stores to one central data warehouse or operational data store, which supports allow full administrative operations on the combined data • EII databases potentially allow users to carry out administrative operations across all relational, file, spreadsheet, and rich-media data sources
Leverage	• Data warehouse and EII APIs (application programming interfaces) enable faster development of transactional code that accesses a broader range of data. • Enterprise portals give a wider range of users access to a broader scope of data. • Operational data stores and EII can support updates as well as queries and reads.

Source: Infostructure Associates, June 2005

If IT implements strategic information management across the enterprise, each of these tasks can be carried out enterprise-wide. Thus, IT can create an "almost enterprise-wide" metadata repository via EII and/or a data warehouse; an "almost-enterprise-wide" data-source administration tool via EII; and an "almost enterprise-wide" "information-leveraging tool" by combining EII and an ODS or data warehouse, plus BI or OLAP tools to sift the resulting enterprise-wide "virtual data store".

Data Migration and Mass Copy/Download Tools

Data migration and mass copy/download tools show up in today's ETL tools as well as various proprietary backup tools (i.e., download to backup disks). For example, Evolutionary Technologies Inc. (ETI) offers data migration as part of its "data integration" solution. Also, tools from major relational database suppliers allow mass copying from one relational database to another, for particular database vendors (e.g., Sybase to IBM DB2). Most of the available tools handle the case of copying relational data from one data store to another.

General-purpose data migration and ETL tools not only pass data between multiple vendors' databases but also convert between a wide array of data types. Therefore, they are especially suited to "permanent" data movement, e.g., database consolidation.

Replication and ETL Tools

If two copies of the same data in different data stores must be kept "in synchronization" (meaning that their values should appear the same no matter what the sequence of transactions on each), the typical technology to use is two-phase commit. However, two-phase commit locks up the database engine for each data store for significant amounts of time — and thus, for any reasonable stream of updates on either data store, there is a major performance penalty for keeping multiple copies of data. In other words, in production OLTP systems, few data copies are possible.

In data warehouses, however, no updates are permitted; hence, the data warehouse can accept replication, i.e., a steady stream of one-way data updates. The only performance barrier is the time to "write lock" a piece of data as the incoming data replaces it. Even this can slow querying in a terabyte data-warehouse data store, so replication can be either instantaneous or "bursty", e.g., done as a mass download while the data warehouse is offline. Replication tools are readily available, some from major vendors such as IBM.

ETL tools, like data migration tools, translate differing source data formats into a common target format, with the typical steps of the process including Extracting, Transforming (with data cleansing), and Loading. Like replication tools, ETL tools can copy data "in real time" or at intervals, and in small or large amounts. A tool can combine replication and ETL technology to ensure that as data is changed in a source data store, that change is sent on to a target data store.

ETL tools are used both with data warehouses and EAI. ETL tools are sometimes referred to as "adapters". Many EII vendors also offer adapters. ETL vendors include Compuware, ETI, IBM, Sybase, and webMethods.

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

Data Warehouses and Data Marts

Data warehouses aggregate data primarily, and often exclusively, for decision support. A data warehouse's data is derived from outside OLTP and "mixed" databases. The databases' data is fed to the decision-support information base via either mass "downloads" involving careful "cleansing" of the data (typically via an ETL tool) or by one-way synchronization in which changes in specific data items are "replicated" in the data-warehouse information base. Specialized querying or analytics tools allow expert "data miners" to access the data and derive insights about patterns of sales or similar competitive-advantage information. Data marts are smaller data warehouses, often for specific functions within an enterprise. Data marts often pass data on to a large central data warehouse.

Experience has shown that the data warehouse has major limitations that prevent it from serving as the ultimate "information aggregation" of all data in the enterprise. First, it can only grow to a certain point before performance becomes too poor; hundreds of terabytes may sound like a large number, but the typical major organization stores far more data than that. Second, in order to service the "query from hell," the data warehouse must focus on queries to the exclusion of all other operations, including the updates that accompany key operations like customer-order entry. Third, holders of line-of-business data or other data are often reluctant to register it in the data warehouse's metadata repository, because registration brings no immediate business benefit. Fourth, refreshing a large data warehouse daily is a massive operation that prevents warehouse usage for hours, so that keeping the warehouse more up to date than one day or one week behind is, practically speaking, impossible.

That said, the data warehouse is often the best base on which to begin implementing strategic information management. It is not only the largest aggregation of data in the enterprise; it is also the largest aggregation of metadata.

All major enterprise database suppliers — IBM, Microsoft, Oracle, and Sybase — offer extensive data warehousing support, as does NCR-Teradata.

Operational Data Stores and EII Tools

These are described below, in the section on VOSs.

Note that EII has an enormous potential as the cornerstone of any strategic information management effort. It is the best tool for gathering metadata that spans the global enterprise's data sources, it can administer across these data stores, and it can aid data miners in identifying and leveraging relationships across an unmatched range of data.

Note also that a combination of EII and mid-tier ODS is often better for strategic information management purposes than either alone. EII tools can cause performance overhead, because the data does not "persist" between transactions; but the mid-tier ODS cannot hold as much data for querying as the EII can span.

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

Strategic Information Management Conclusions

Currently available tools can be combined today to form a powerful strategic information management solution; but few IT shops have begun to do so. While strategic information management is not urgent in the sense that enterprises will collapse without it, strategic information management does potentially offer major benefits in competitive advantage and major cost savings in database administration. Therefore, IT should investigate this new approach to data management sooner rather than later.

A key part of successful implementation of strategic information management is implementation of EII on an enterprise-wide basis. Without it, the comprehensive metadata repository on which everything else depends will almost never be built.

Note that implementation of strategic information management is exceptionally low-risk. In most large enterprises, most of the tools (ETL, data warehousing, and operational databases) already exists; the key add-on is EII, which user experience shows to be easy to integrate, and the key "repurposing" is using an operational database as an ODS.

The Virtual Operational Store (VOS) and the Real-Time Enterprise

The "real-time enterprise" (RTE) — the promise of an organization that can react to changes in its environment immediately — is in danger of becoming an over-hyped, under-performing failure — like artificial intelligence before it.

While definitions of "real-time enterprise" abound, they have in common these characteristics:

- A real-time enterprise supports *on-demand* access to key *information*.
- Effective use of this "on-demand information" means *competitive advantage* for the companies like Wal-Mart and Dell that seem to achieve it.
- To become an RTE, companies must make mission-critical business processes (and decision support/business intelligence capabilities) that access this information *faster*.

In other words, the prescription for becoming an RTE is seductively simple:

1. Identify the information and business processes that are mission- or business-critical and for whom improvements in speed will give you a competitive advantage.
2. Speed them up.

Just like that.

Infostructure Associates asserts that this "RTE strategy" is far more likely to result in long-term failure than the "virtual" approach described both in the Infostructure Associates white paper entitled "Wanted: the Virtual Real-Time Enterprise; Needed: EII + ODS = VOS", and in this report:

1. No database or data-access mechanism can deliver instantaneous access to the terabytes and even petabytes of mission-critical and business-critical information in the average large enterprise, as

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

the failure of the data warehouse to do so proves — and the situation is getting worse, not better, as storage size at Fortune 2000 enterprises doubles every two years or less while major-database speed improves by 40% at most every two years.

2. The proliferation of in-house and extra-enterprise data sources that can be used for competitive advantage means that, like Alice in the Looking Glass, the faster the IT executive runs to include new operational data sources in the RTE, the more new data sources there are to probe, so that the enterprise is falling farther and farther behind the ideal of the RTE. And this is true even before RFID looses a flood of new time-critical data on Wal-Mart and its competitors, not to mention increasing use of data from extra-enterprise sources such as information from suppliers or customers, or even Web services that provide real-time information such as commodity prices, interest rates, or even weather.

To solve this problem, IT (and ISVs aiding them) should go back to the original technical meaning of the word "virtual" — *a relatively small amount of fast-access operational data that allows the system to deliver, on average, 90% of the speed of an "ideal" on-demand cross-enterprise database*. By creating a "virtual operational store" (VOS), IT avoids the major problems with the RTE:

1. By accessing a much smaller VOS database, IT ensures almost-real-time response to data-mining queries, customer-facing updates, and business-process-speeding complex transactions.
2. By semi-automating addition of new data sources and data types to the VOS, IT can keep pace with the rapidly growing demands of rapidly growing data sources.

Where is the technology to support the VOS and the virtual RTE? Certainly not in today's major popular databases and data-warehouse solutions alone. These are often topping out and proliferating, with no clear solution (despite the promise of grid computing) in sight. Moreover, these solutions are relational (optimized for structured data) and do not adapt easily to handling the increasing percentage of unstructured and semi-structured operational data, nor to storing XML's complex data types.

Infostructure Associates finds that the technologies that offer real promise of solving upcoming RTE problems are EII tools and ODSs — that is, an EII solution and an ODS specifically designed to handle a wide range of operational data, to scale, and to add new data sources and types semi-automatically. Separately, these two technologies have already proven themselves to be highly useful to the savvy enterprise. Together, they can deliver all the benefits of the separate products plus VOS capabilities that will enable long-term enterprise success in achieving the virtual real-time enterprise.

Below, we describe the key considerations that IT should take into account in creating the virtual RTE via a VOS.

The Nature of a Virtual Operational Store

A virtual operational store harks back to the idea of virtual memory. Virtual memory provides a "cache" of fast main memory to front a much larger (by 1-3 orders of magnitude) disk storage area. When the processor in a computer accesses the main memory and finds that the

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

data it needs is not there, it places some of the data in main memory on disk ("swaps it out"), accesses the disk to get the data it needs, loads that data into main memory ("swaps it in"), and then proceeds. By using sophisticated "rules of thumb", the computer can ensure that, most of the time, the data it needs is in memory — and therefore, on average, the computer will perform at 90% or more of its potential performance if all of the disk storage had been main memory instead. The old joke about virtual memory is to cup your empty hands and say "I have in my hands ten terabytes of virtual memory" — because, effectively, the system acts as if all of storage was main memory.

In the same way, a virtual operational store aims to ensure that, most of the time, key data that is likely to be accessed "on demand" is available in the data store, and can be retrieved at speeds comparable to that of a major popular database — or faster, because an ODS is optimized for just such a situation. To achieve this, the VOS uses "rules of thumb" for "swapping in" and "swapping out" similar to the virtual-memory approach:

- *Age (or "Currency")*. The older the data, the less likely to be accessed for on demand purposes — often, week-old data is unlikely to be accessed again except in rare circumstances. The typical enterprise without a VOS determines this ad-hoc, intermittently, and occasionally; the VOS determines this semi-automatically, flexibly, and continually.
- *Locality of Reference*. The RTE typically uses data in some data sources far more than others. The typical enterprise without a VOS sets up a data warehouse, which uses these data sources exclusively, comprehensively, and permanently. The VOS keeps track of usage patterns and adjusts accordingly, thus using all data sources as necessary, using only the data needed, and adapting dynamically as the patterns change.
- *Least Recently Used*. Not all data is used once and then forgotten; some (such as financial data) may be used frequently for at least a year or two. The enterprise without a VOS typically does not archive data based on least recent use; the VOS uses Least Recently Used to avoid constant swapping of data based on its Age.
- *Data fixing*. Sometimes, data must be rapidly available forever. For example, data to enable systems to get up and running quickly after a disaster, although rarely used and often quite old, needs to be rapidly available. The enterprise without a VOS typically solves this problem separately for each different data source; the VOS can "data fix" this information to remain in the data store until further notice, thus centralizing this type of key data and allowing readjustment over time.

In order to carry out its tasks, a VOS must therefore include:

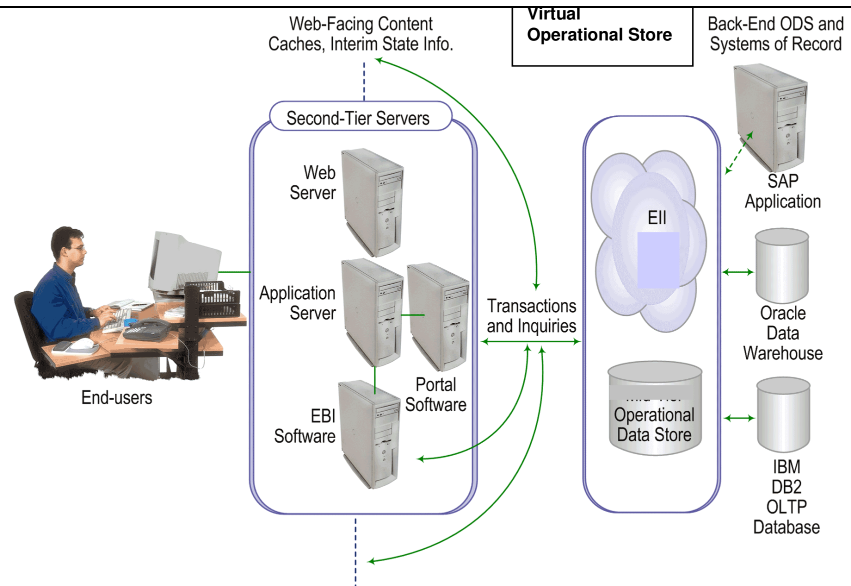
- *All of the key features of an ODS*. These would include the ability to perform queries and updates rapidly, to scale, to import data from a wide variety of mission-critical and business-critical databases rapidly (with data transformation and cleansing), and to support access from key business applications such as data mining and reporting. They might also include the ability to synchronize with mission- and business-critical databases via two-phase commit, to export data to these databases, and to synchronize metadata with these databases.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

- *A global metadata repository that updates and expands semi-automatically*. The repository would support a flexible data model that adapts dynamically to changes in the enterprise's data and is not constrained, for example, by relational schema definitions. This repository would be partially imported from, and refreshed automatically from, other databases' data dictionaries, and would contain information on the data's age, other locations, recent use, and whether it should be "data fixed". The repository should allow data to be indexed in a way that allows business users to get the information they need. Ideally, a VOS will provide fully granular access to key information stored in business documents as well as relational databases. XML is a key enabling technology for reaching this goal, because every piece of data can be labeled for easy access (and indexing). Several EII suppliers offer most of these capabilities.
- *Cross-database and composite-application development (and, if possible, administration) tools*. Some ODSs and most EII suppliers now provide these development tools. These days, Web service development support is especially useful.
- *Cross-data-source querying and updates*. In some cases, it makes the most sense for the VOS not to load the data into the "cache" data store before carrying out a query. In these cases, most EII suppliers allow querying across a wide variety of data sources containing structured, semi-structured, and unstructured data types. Many EII suppliers also allow access to information outside the enterprise, such as data from trading partners, supply chain information, and data derived from Web services such as stock ticker data or even eBay prices.

Figure 1 shows how a VOS might fit into an overall enterprise architecture.

Figure 1: Example of a VOS in an Enterprise Architecture



Source: Infostructure Associates, June 2005

The Benefits of a VOS for the Real-Time Enterprise

The first benefit of a VOS is that it makes the real-time enterprise far more feasible. Therefore, a VOS should typically enable all the key benefits of an RTE:

1. *Rapid response* to changes in the environment for competitive advantage and a better profit margin — e.g., tweaking prices locally and hourly, identifying new opportunities based on changing market conditions, adjusting the supply chain rapidly to new orders, and detecting and reacting quickly to new regulations, competitive challenges, and disasters. In particular, the enterprise should see faster and more flexible reporting and analysis, complementing traditional BI.
2. *Speed-up of business processes* to cut costs and increase customer satisfaction, because information from each stage of a process is supplied to the next more quickly (i.e., "information latency" decreases).
3. Increased ability to *leverage proprietary information for competitive advantage*. Today's IT strategist, seeking to increase competitive advantage, can focus primarily on the application resources or the data resources of the enterprise. Until recently, it made most sense to focus on applications. However, these are increasingly easy for competitors to duplicate — and proprietary information is not. Proprietary information is typically locked in widely differing and often aged data sources; a VOS makes that information more widely available.

Second, a VOS delivers significant "side-effect" benefits. For example:

- Today's storage efforts focus particularly on Information Lifecycle Management (ILM), attempting to move data from device type to device type (e.g., nearline disk to nearline tape) based on changing requirements for that data. Storage systems necessarily treat this "data aging" in a very crude and broad-brush manner, because they do not have fine-grained and up-to-date information about the likely uses of each piece of data in the near future. A VOS provides a centralized "information base" that ILM storage systems can plunder for insights into how useful various data sources and files will be to the organization in the near future, and therefore when they should be "aged."
- A VOS can fit seamlessly into an enterprise's existing IT infrastructure, and leave existing databases intact and unaffected.

Third, if a VOS includes EII and ODS capabilities, it can deliver all the benefits that these technologies have already provided in the real world (see Table 6).

Table 6: Ways to Use a VOS in Key IT Tasks

Aim	Ways to Use a VOS
Integrate business processes	Replicate data from the data sources involved in the business processes to the ODS, allowing all related transactions in the business processes to be carried out at once within the ODS in a consistent fashion — and then the resulting data is resynchronized with the original data sources. Persist data views aggregated across applications within the ODS for improved

	query and update performance across the applications Program using the EII interface rather than the component applications, in order to carry out querying across business processes. The resulting code should use the EII tool's ability to combine data from different sources and allow analysis of data relationships across data sources, not just view data from different source side-by-side on the same screen.
Implement Web services interfaces to existing information	Write Web services consumer code for each key enterprise application to invoke the VOS rather than each data source
Improve RTE capabilities by speeding time to react to current customer or supply chain data	Replicate high-frequency-of-update customer or supply-chain data in the ODS Use the VOS to offload processing from back-end enterprise data stores and systems of record, speeding notification of new data and changes caused by reacting to new data. Leverage the VOS's extensibility to implement changes to a broader array of key data.
Reduce time to develop new applications leveraging proprietary information, such as portals	Use an EII tool to front-end the data warehouse plus other proprietary information, allowing one set of code for all information access Use data persistence in the ODS to reduce the need for mapping and reconstruction programs for unstructured content
Audit all information in the enterprise for government requirements or Sarbanes-Oxley	Create a metadata repository in the VOS by importing or generating metadata from data warehouses/marts, ETL tools, and file systems and rich-media sources. Write audit queries to the resulting cross-data-source EII interface, or create a reporting application with modular report templates and reusable components.
Cut IT costs	Cut development costs via the VOS's ability to allow developers to "write once" to supply data access to many applications
Attain better scalability of existing multi-tier applications	Implement a VOS to aggregate information (both structured and unstructured) from back-end systems of record and store it in the operational store. Synchronize the aggregated data views with back-end systems via the VOS. Thus, the user can offload query processing from back-end systems and satisfy the data requirements of multi-tier front-end applications with higher performance.
Merge companies	Using EII, create integrated reports about customers, suppliers, and the overall enterprise's financial position, e.g., over multiple data warehouses/marts.

Infostructure Associates, June 2005

Key Criteria for Acquiring an EII and ODS to Create a VOS

EII tools and ODS tools differ markedly in their ability to support developers and end-users. Infostructure Associates recommends that IT buyers use different criteria according to the "main constituency" for a VOS. Key criteria for developers are as follows:

- *Performance/scalability* of the applications built over the VOS. For example, support for clustering and load balancing allows developers to create applications that scale out and up. Other useful features include indexing of locally-stored data and ability to optimize cross-database queries at the EII and local-data-store level.

- *Programmer productivity and simplified maintenance/upgrade*. For example, by "componentizing" key operations at the business level, a good EII tool or ODS allows the developer to swiftly write applications such as enterprise portals.
- *Flexibility* to support a wide variety of platforms and architectures — especially Linux/Unix and Windows — as well as to support various data models and easily integrate with back-end data sources (with both query and update support).

Key criteria for end-users are as follows:

- *Comprehensive support* in both EII and ODS for connection to and synchronization with a wide variety of back-end databases and other data sources.
- *Easy "plugability"* into front-end and enterprise portal implementations, as well as popular Web and application servers.
- Powerful EII *cross-database query features.*

IT buyers should also consider the following overall criteria:

- The availability of "lights-out" or "near-lights-out" administrative tools;
- The EII tool or ODS supplier's ability to serve as a "one-stop storefront" for system design and integration.
- The ability of the solution to be a good foundation for a Web services architecture — over the next two years, Web services implementations will become much more common.

State of the Art in ODSs

An operational data store combines key data from multiple data sources, or systems of record, in one database that — unlike a data warehouse — allows update of the data. Today, there are two kinds of ODS — an enterprise ODS, which focuses on collecting enterprise-level mission-critical data, and a mid-tier ODS, a database delivering high performance on queries, updates, and data access that serves as a "persistent long-term cache" for several relational databases from one or several suppliers, as well as for unstructured data sources.

The enterprise ODS offers the ability to aggregate the data of a data warehouse plus "near–real-time" information "currency" that gives users the ability to react almost instantaneously to changes in business conditions — such as shifts in buying patterns. However, adding updates to the capabilities of a data warehouse means a sharp decrease in the amount of data that can be stored in the operational data store and the amount and complexity of the data mining that can be applied to it. Using any ODS, users can react more quickly to insights, but can gain fewer of these insights.

Moreover, enterprise ODSs typically aim to aggregate data from structured sources, such as enterprise applications that run on top of a relational database. As such, they are especially suited for environments where the structure of data, or "metadata," rarely changes.

All enterprise databases can be used as enterprise ODSs when combined with "near-real-time" ETL tools. Most major enterprise database suppliers now offer support for implemen-

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

tation of enterprise ODSs, and the idea of an enterprise ODS, pioneered by suppliers such as Unisys, is generally accepted.

A mid-tier ODS provides all of the "core" functionality of a scalable and secure enterprise data engine, as well as a flexible and extensible repository. This allows users to improve the performance of each underlying data source and to combine key data from databases, repositories, and shared file systems, "reflecting" changes to this key data back to the back-end data sources. A mid-tier ODS can provide all of the benefits of an enterprise ODS (if sometimes on a smaller scale), as well as improved performance/scalability for business-critical databases and applications.

Mid-tier ODSs are more likely than enterprise ones to leverage XML for more flexible aggregation and simplified data access and transformation. A mid-tier ODS can use XML support to achieve structural data independence. Thus, users can simplify the process of picking the right subset of the data to be aggregated, because more data sources can be accessed dynamically. As the amount of unstructured content created in today's typical enterprise increases rapidly, VOS implementers can use a mid-tier ODS to make this content available to enterprise applications for decision support, demand forecasting, and revenue maximization.

Mid-tier ODSs have functionality similar to EII tools, making their integration more effective. Like EII or EAI tools, mid-tier ODSs can store metadata about relationships across databases or other sources of operational data, and can import data from a wide variety of data sources.

Because its support for a wide variety of data types and coordination with operational databases is "designed in" rather than added later, the mid-tier ODS is frequently better suited for a VOS than an enterprise ODS is. Strong examples of a VOS-friendly mid-tier ODS are Raining Data's TigerLogic™ XML Data Management Server (XDMS) software and the ODS in Ipedo's XML Intelligence Platform (Ipedo XIP).

State of the Art in EII Tools Supporting a VOS

The potential for EII as an enabler of the RTE cannot be overestimated. EII can serve as the focal point for gathering an unprecedented amount of metadata about the enterprise's proprietary information. Potential EII tools can administer across as wide a swathe of the enterprise's data sources as desired. EII allows leveraging data relationships across data sources without the effort of moving or copying the data.

Users should note that because EII tools typically do not persist data between transactions, that data must be extracted, transformed, and integrated anew for each operation, causing performance overhead compared to an ODS — although the ODS does not permit access to as large a data store as an EII tool. Thus, a combination of EII and ODS — as in a VOS — often delivers a better tradeoff between data-store size and performance than either one separately.

The EII market is still relatively new, and some key EII suppliers are small but growing rapidly, such as Attunity, Ipedo, and MetaMatrix. Large firms that have entered the market include BEA (Liquid Data for WebLogic), and IBM, whose WebSphere Information Integrator has been highly successful in the market. Newer suppliers that add distributed-EII-server scalability as well as other features include Composite Software and Avaki.

State of the Art in Integrated EII and ODS

Already, suppliers are beginning to consider the merits of integrating EII and an ODS. Ipedo's XML Intelligence Platform is a notable example of a solution that combines EII and mid-tier ODS capabilities, and Intersystems' Cache and Ensemble, as well as IBM's DB2 UDB and WebSphere Information Integrator, although not specifically targeted to the EII, ODS, and VOS markets, provide similar capabilities. Moreover, Raining Data's TigerLogic XDMS allows programmers to create EII-like cross-data-source transactions, and thus supports ODS-plus-EII functionality. Some EII suppliers offer an EII plus ODS combination by partnering with an ODS supplier. As of yet, no supplier specifically targets the VOS market or provides full "virtual" capabilities such as data fixing.

One concern often expressed by IT about EII, ODSs, and their combination is that it will add a significant new administrative cost — yet another database administrator for yet another database. On the contrary, Infostructure Associates finds that in many real-world cases, today's EII tools and mid-tier ODSs can decrease administrative costs, because both provide a cross-database repository of the metadata that the administrator can manage. Moreover, both EII tools and mid-tier ODSs are "administrator-light"; they automate administration to such a degree that they are suitable for "near-lights-out" use in smaller enterprises.

Given the nascent state of the VOS market, Infostructure Associates recommends that enterprises consider *both* picking a supplier with an EII-ODS combination *and* taking a "roll your own" approach that chooses an EII tool and ODS tool separately and integrates them in-house. However, EII-ODS suppliers do deserve special consideration, because, in some cases, they have tackled some of the challenges of creating a VOS already.

Conclusions: EII + ODS = VOS

It is said that the largest danger for a business startup — and the most likely outcome — is neither immediate success nor immediate failure, but rather a long-drawn-out process in which the enterprise always sees local successes (within an overall pattern of slight losses) that lead it to throw good money after bad. In the same way, an enterprise's effort to become an RTE runs a large risk of continually pointing to specific successes in reducing information latency, with excellent bottom-line effects, while the overall trend is of more and more information arriving, resulting in less and less timely access, leading to greater information lags and greater competitive disadvantage.

The answer is a global, flexible approach to creating an RTE — the virtual operational store (VOS). A VOS can potentially deliver nearly the performance and scalability of an ideal (but never-to-be-realized) enterprise-wide database, while adapting semi-automatically to new data and new data sources. Thus, a VOS can deliver all the benefits of the RTE, plus side benefits such as meshing with Information Lifecycle Management and all the benefits of the EII tool and ODS that should form its core components.

By choosing a good EII tool and ODS (or their combination), such as are already available in today's market, the savvy CIO can get nine-tenths of the way to implementing a VOS — and can begin to reap some of the benefits of a full VOS, even before the customization that will be necessary to implement the full VOS. Over the next year, Infostructure Associates anticipates that support for key "rules of thumb" will make the job of IT in creating a VOS much easier. Therefore, now is the right time for IT buyers to set forth a strategic plan for identifying key mission-critical data to be put in the VOS, and to choose a supplier for the initial stages. If, moreover, the suppliers that the IT buyer chooses support standards for Web services, the IT buyer may rest assured that suppliers may be switched if they don't pan out, and multiple suppliers can integrate effectively.

The real-time enterprise means real-time information, and real-time information requires virtual technology. The sooner the enterprise recognizes that fact and acts accordingly, by creating a VOS, the better for that enterprise.

The Benefits of OLAP

OLAP (Online Analytical Processing) means digging deep into data to make every type of vital business decision — hour-by-hour adjustment of stock delivery to stores, yearly budgeting, sales management.

Highly effective OLAP is about digging deeper into the data, more frequently and faster, at any time, cost-effectively, with higher-quality decisions.

The key findings of Infostructure Associates from recent OLAP interviews include:

- *Businesses need OLAP*. Highly competitive markets demand that the successful organization collect ever more data, perform ever deeper analysis, and deliver decisions ever more rapidly. Most medium- to large-scale organizations now have large, rapidly growing data stores that they use with reporting, querying, and business intelligence tools to support key decisions. Current decision-support solutions do not scale well as the data stores grow, do not support "deeper" analysis, and are not built to support forecasting and "what-if" analysis. OLAP does all of these things.
- *A good OLAP solution can provide bottom-line value-add for the organization that uses it.* The value-add derives especially from deeper insights into customer data relationships that allow investments that maximize ROI (return on investment). A less easily quantified, but potentially equally important, value-add is increased ability to identify and fine-tune investments based on "what-if" analysis. For example, a sales executive can use OLAP to segment customers and prospects by geography, buying habits, and income (the "dimensions" of the data), identify the likeliest groups of target customers, and forecast the outcome of ad campaigns targeted at each of these groups. OLAP is typically much faster than traditional querying tools because OLAP queries are mostly pre-calculated, resulting in efficient, rapid analysis of huge quantities of data.
- *IT buyers should use the following criteria to choose an OLAP solution: performance/scalability, flexibility, cost-effectiveness, value, and risk mitigation (availability, security, and data quality).* Performance and scalability are especially important in OLAP, because OLAP-based investment

decisions are often time-critical, because OLAP scalability allows users to apply analysis to larger terabyte-sized databases than typical data warehouses, and because OLAP performance/scalability allows "deeper" analysis than traditional BI tools.

- *IT buyers should note Oracle OLAP Option's innovative architecture.* Traditional OLAP solutions separate the OLAP database from relational databases, creating administrative complexity/costs, decreased performance and scalability in many cases, and more risk (lower availability and less security). Oracle OLAP Option's architecture puts OLAP in the same process as the Oracle relational database, with the same security and administrative tools, the same data store, and the same querying commands. Because Oracle OLAP is part of the database, this eliminates the need to reconcile data between the two systems, and is often more performant. Likewise, clustering solutions such as Real Application Clusters (RAC) have long proven their ability to deliver additional availability via failover.

The Need for OLAP

Increasingly, medium- and large-scale businesses performing business analysis are finding that data warehousing and BI tools are not enough. They do not scale to the terabyte sizes that new data sources such as RFID and the inevitable growth of corporate information require; they do not allow deeper analysis than two or three "dimensions"; and they are not well adapted to forecasting.

Suppose that you are CEO of a value-added reseller (VAR) of computer software, hardware, and services, and your business has undergone a significant contraction in the month of March, for no apparent reason. Using a querying tool, you look at your revenues for March by region (a fairly complicated query); the answer takes a long time to come, and all it shows is that the North American region is under-performing. Clearly, determining why will take quite some time, and require a querying expert.

By contrast, a good OLAP tool enables rapid, in-depth analysis and forecasting for both immediate tactical and longer-term strategic decision making. Within the general class of analytic tools, OLAP differentiates itself by ability to analyze data across dozens or even hundreds of dimensions.

If you are the CEO of that VAR, your OLAP tool will let you easily specify a 6-dimension OLAP query to look at revenue per salesperson, for March for the hardware division for Canada, compared to last year. The answer is immediate: one of your most productive salespeople had a disastrous month. Now that you know where at least one of the problems is, you can test the probable effect of transferring another salesperson to that territory, or aiming a marketing campaign there.

That brings up another point: the ones who often need OLAP the most are not computer nerds or statistical experts, but high-level business types. Typical users of OLAP solutions include CFOs, strategic planners, and sales and marketing executives. These use OLAP solutions not only for traditional analysis but also for budgeting, financial reporting and consolidation, monitoring of performance against budget, planning, business forecasting, support for the budgeting/planning process, sales forecasting, clickstream analysis, profitability analysis, quality analysis, and analysis of data for more general decision-making.

Business Requirements for an OLAP Solution

Today's OLAP solutions differ markedly in their ability to support business needs. Buyers should emphasize the following criteria in assessing and comparing OLAP solutions:

- *Performance/scalability*. It is vitally important that an OLAP solution give you answers rapidly, no matter what the amount of data or the sophistication of the analysis required, for "real-time" or fast-reaction decisions. Analysis is stifled or even stopped if users have to wait even a few minutes for a response to a question. Many of today's OLAP users need analysis on terabytes and even tens of terabytes of data. You also don't want your analysis limited to a certain number of dimensions because of the OLAP tool's limitations.
- *Flexibility*. You should be able to implement and upgrade your OLAP solution, and extend it to new operations and types of data, rapidly. You should be able to use your favorite user interface, whether you like Microsoft Excel spreadsheets or a reporting tool's charts and graphs.
- *Cost of Ownership*. The solution should minimize your total cost of ownership (TCO), and especially the administrative costs of maintaining the data (e.g., backup/restore and security). Infostructure Associates research shows that administrative costs are a large and increasing proportion of TCO for most major applications.
- *Ease of use*. The OLAP solution should give you clear, simple ways to ask queries, vary dimensions, and "drill down" to more detailed analysis. When you need a sophisticated statistical analysis, the OLAP tool should provide it. You should see results in a clear and visually appealing way.
- *Risk mitigation – including availability, security, and data quality*. You should look for a solution that minimizes the risks that OLAP solutions face. Most users need OLAP solutions to deliver answers at any time, when an emergency occurs, so they need OLAP solutions that are *available* as much of the time as possible. You need a solution that keeps tight *security* over your proprietary data. Because it is very easy to enter bad data into the system that can cause you to analyze a problem incorrectly, you need a system that improves *data quality*.

Problems With Traditional OLAP Solutions

While today's long-popular OLAP solutions do better than data warehouses with BI (business intelligence) tools, they have have serious limitations (see Figure 2). Many of their problems stem from their architecture — by making the OLAP database separate from any other database, they:

- Complicate the enterprise's IT infrastructure, increasing administrative costs and security problems dramatically.
- Make the data that you are analyzing more "out of date".
- Slow down your analysis to a crawl when they can't find the right data in the OLAP database and have to "drill through" to a relational database.

- Force users to constantly reconcile and synchronize between OLAP and source data.
- May also have annoying limits on your analysis — for example, some of them limit the number of "dimensions" that you can analyze.

Oracle's Innovative OLAP Architecture

Oracle OLAP Option is an integral part of the Oracle 9i/10g database (see Figure 3). Thus, Oracle OLAP Option runs in the same process as the Oracle database, and uses the same data — there is no need to move data from the database's data store to the OLAP data store. There is no "drill-through" needed — the OLAP engine can run against OLAP data, relational data, or a combination of the two, and so can the relational engine. OLAP and relational database share the same process, the same storage, the same metadata repository, the same administrative tools, the same security model, and the same (OLAP API and SQL) interfaces.

The Oracle architecture, in which OLAP is part of the database, also lets you leverage the advanced features that are part of the Oracle database, such as clustering for availability, automated and sophisticated administrative tools, and a strong security scheme. Security is consistent across Oracle relational and OLAP data, and serviced by a single administrative interface. Oracle OLAP Option users can also leverage RAC (Real Application Clusters), which enables cost-effective load balancing, failover, and scaling. With RAC, businesses can tie together inexpensive Intel-based servers for failover and load balancing; moreover, if users need more power, servers can be easily added to the cluster. BI tools and Microsoft Excel can operate against both relational and multidimensional data in the same database. To improve data quality, the OLAP database can use the "data cleansing" features of the Oracle database. Thus, the Oracle architecture can offer significant advantages in availability, security, data quality, ease of use, and cost of ownership as well.

Performance/Scalability — The Most Important OLAP Criterion

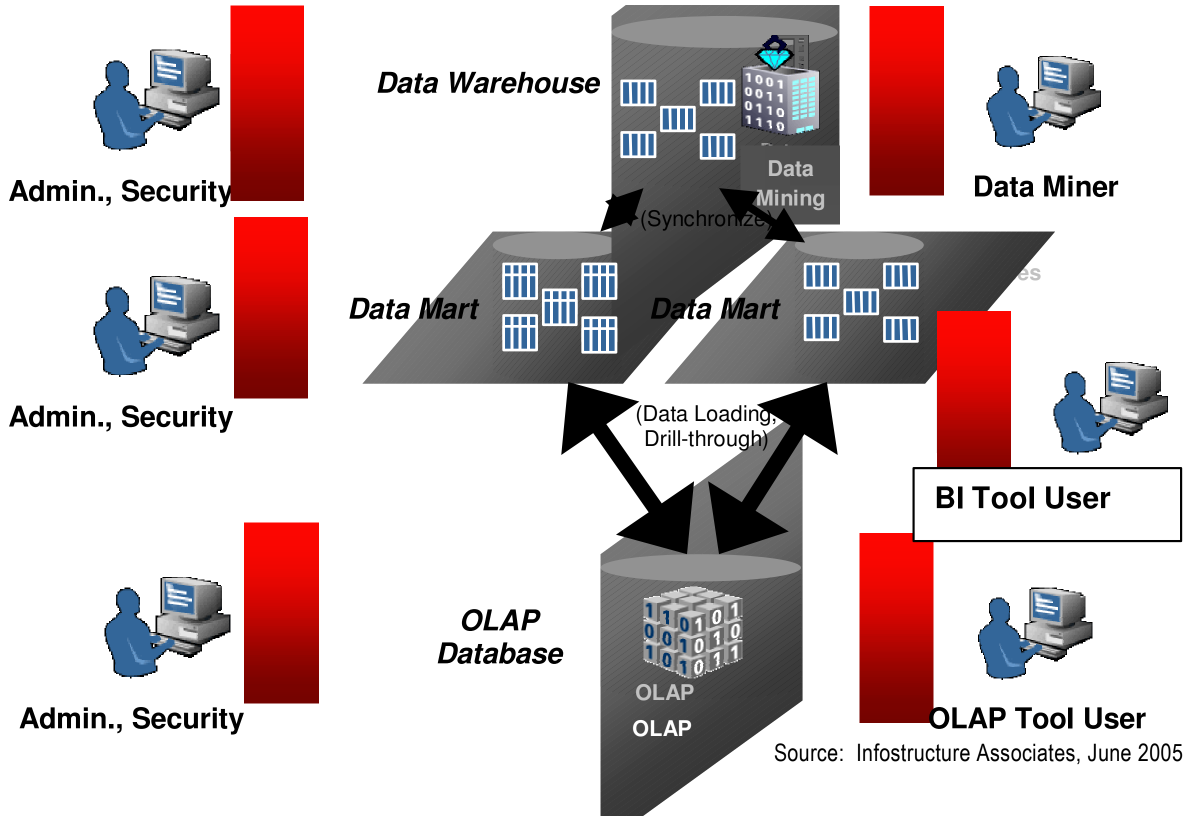
A rule of thumb in OLAP is that most queries should be carried out in five seconds or less. Key factors in achieving OLAP performance and scalability are:

- A database engine and data storage focused on and tuned for multidimensional data.
- A database engine and storage effective in optimizing queries, and especially OLAP ones.
- Efficient data storage. This is particularly important in avoiding "data explosions", in which generated information related to the data can take ten or one hundred times the amount of space that the data itself uses.

Related technologies that boost scalability include caching, parallelism, and partitioning.

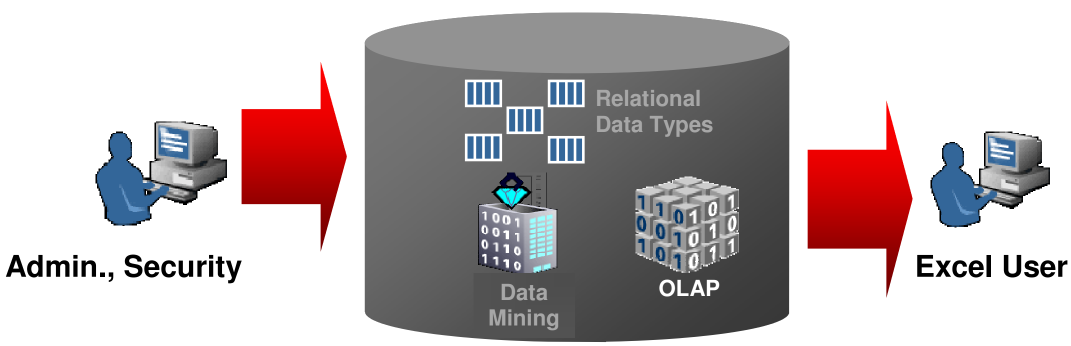
Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Figure 2: Traditional Architecture



Source: Infostructure Associates, June 2005

Figure 3: Oracle Architecture



Source: Infostructure Associates, June 2005

Flexibility Criterion

Flexibility considerations include:

- The ability to implement and upgrade the OLAP solution rapidly.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

- Access to a wide variety of data types and sources, and the ability to extend that access to new data types rapidly.
- Easy integration with major popular BI and analysis tools and platforms.

Cost of Ownership Criterion
Infostructure Associates has conducted extensive research in embedded-database (that is, a database supporting an application) TCO over the last six years. Typical significant costs included in TCO are: license costs (including license costs for software infrastructure such as application servers), deployment/upgrade costs, training costs, and administrative costs. Infostructure Associates has found that:

- The cost of managing and maintaining the database is the largest long-term cost of an embedded-database solution such as an OLAP solution.
- Administrative costs continue to increase as a proportion of overall TCO.
- Vendor technology continues to offset rising people costs.

Thus, over the long term, the most cost-effective OLAP solution minimizes administrative costs.

Ease of Use Criterion
Ease of use includes:

- The power that the OLAP solution gives to the end user — that is, the range of functions that the end user can apply for analysis.
- The user-friendliness of the OLAP solution — that is, how intuitive the user interface is, how easy it is to carry out desired analysis tasks, and how visually concise and clear the display of results is.
- How easy it is for the developer to create new applications on top of the OLAP solution, and how easy it is for the administrator to carry out the usual maintenance tasks.

Most of these sub-criteria are highly subjective, and user experience may vary.

Risk Mitigation Criterion
OLAP end users are often key decision-makers making time-critical or fast-reaction decisions, potentially at any time on any day of the week. Therefore, it is highly important that the OLAP solution be available as much of the time as possible, and if possible 7x24. Moreover, these decisions often involve proprietary information; hence security is important too. Business-critical decisions must be based on correct data, so data quality is a third important way of reducing the risks of OLAP.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Chapter Four: The Database Market

The database market continues to be massive ($11 billion in revenues in 2004) and growing at a moderate pace (15%). Growth should continue at a steady pace over the next two years, as users continue to refresh their IT infrastructure.

Functions

Infostructure Associates, as noted in Chapter Two, sees most applications of databases centering around three types of transaction stream:

1. *OLTP* — common in customer-related software, e.g., airline reservation systems, order processing systems, and stock market data stores.

2. *Decision support/data warehousing* —used by data miners to gain insights for organizational decision-making, e.g., BI and enterprise reporting.

3. *"Mixed"* — may include "embedded" databases supporting an application, Web-site support, and ODSs.

Table 7 shows trends in the percentage of database uses for each of the three types of transaction stream. Over the next five years, "mixed" transaction streams should come to dominate the market.

Table 7: Trends in Database Function (Share of Overall Database Revenues)

Function	1999	2004	2009
OLTP	50%	25%	20%
Decision Support	20%	30%	25%
"Mixed"	30%	45%	55%

Source: Infostructure Associates, June 2005

Vertical Submarkets

No one vertical industry dominates the database market; and this should continue in the foreseeable future. Table 8 shows the major vertical database markets, in approximate order of revenues. All industries appear healthy and able to grow database needs in the near future.

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Table 8: Key Database Vertical Markets

Vertical Market	Companies	Typical Database Use
Financial	Investment, credit card, bank, insurance	Run the business; banks to handle ATM data; and investment to analyze stock information ("quants")
Telecommunications	Telephone companies and PTTs, wireless suppliers	Run the business; phone companies to deliver robust storage of voice and data switch information
Retail	Department store chains, restaurants	Point-of-sale data (OLTP) and to run the business
Manufacturing	Car makers, e.g., Ford; equipment makers, e.g., Caterpillar; semiconductor suppliers	ERP applications ("mixed"); car makers use databases for design (CAD/CAM)
Health Care	Hospitals and hospital chains	Run the business, and increasingly to store and protect key rich-media data, such as charts
Oil/Gas/Aerospace	Exxon Mobil, Boeing	Oil/gas companies store and analyze geological data; aerospace companies store designs (CAD/CAM) and handle production (ERP)
Travel/Entertainment	Hotels, luxury cruise lines; Television, movie, and music companies	Hotel chains store "movies on demand"; increasingly, news organizations use content management to handle rich-media data; movie companies store video
Transportation	Truck, train, and shipping companies	Store key scheduling data
Government	NSA, Defense	Store rich-media satellite data
Education	Universities	Store student and administration data securely

Source: Infostructure Associates, June 2005

Platforms

Database revenues on particular platforms reflects the degree to which these platforms are used in larger-scale or enterprise computing. Table 9 shows a steady increase in Windows usage in the last few years; the increasing popularity of Linux has created a market for open-source databases, which should not impact revenues significantly. Over the next few years, both Linux and Windows should continue to erode mainframe market share.

Telephone: 508-265-1558
24 Muzzey St. *Fax: 781-862-6236*
Lexington, MA 02421 *www.valleyviewventures.com*

Table 9: Trends in Database Platforms (Share of Overall Database Revenues)

Operating System/-Platform	1999	2004	2007
Unix/Linux	55%	55%	60%
NT	20%	25%	35%
Mainframe/proprietary	25%	20%	15%

Source: Infostructure Associates, June 2005

Suppliers

Table 10 shows Infostructure Associates' estimates of particular vendors' market share. IBM, Oracle, and Microsoft should continue to dominate, with little relative change in market share, over the next two years.

Table 10: Trends in Database Platforms (Share of Overall Database Revenues)

Vendor	Market Share
IBM	31%
Oracle	29%
Microsoft	16%
Sybase	3%
Progress	3%
Computer Associates	2%
Intersystems	2%
Other	14%

Source: Infostructure Associates, June 2005

Customer Size

Table 11 shows database market revenue breakdowns by size of customer. The categories are defined as follows:

1. *High* — enterprise with more than $75 million in revenues.
2. *Medium* — $25 million to 75 million in revenues (also includes revenues from use of a database strictly at the workgroup or departmental level within a larger organization).

3. *Low* — an SMB with less than $25 million in revenues.

We anticipate some increase in percentage of revenues from the Low market over the next two years, as IBM's push into that market is showing some signs of success.

Table 11: Database User Size and Market Share (of Overall Database Revenues)

Size of Database User	High	Medium	Low
Market Share	50%	30%	20%

Source: Infostructure Associates, June 2005

Telephone: 508-265-1558
24 Muzzey St.
Fax: 781-862-6236
Lexington, MA 02421
www.valleyviewventures.com

Infostructure Associates
24 Muzzey St.
Lexington, MA 02421
USA
Telephone: 508 265 1558
Fax: 781 862 6236
www.valleyviewventures.com

June 2005

Infostructure Associates is an affiliate of Valley View Ventures that aims to provide thought leadership and sound advice to both vendors and users of information technology.

This document is the result of research performed by Infostructure Associates, which was not sponsored or underwritten. Infostructure Associates believes its findings are objective and represent the best analysis available at the time of publication.

Email the analyst:
wayne.kernochan@valleyviewventures.com