June 3, 2009

Massive But Agile: Best Practices For Scaling The Next-Generation Enterprise Data Warehouse

by James G. Kobielus

for Information & Knowledge Management Professionals

Includes case studies

June 3, 2009

Massive But Agile: Best Practices For Scaling The Next-Generation Enterprise Data Warehouse

by James G. Kobielus
with Boris Evelson, Noel Yuhanna, Charles Coit

EXECUTIVE SUMMARY

Information and knowledge management (I&KM) professionals continue to expand the scale, scope, and deployment roles for their enterprise data warehouse (EDW) investments. Today's most demanding EDW environments support petabytes of aggregated data, trillions of records, thousands of concurrent users and queries, complex mixed-query workloads, subsecond latencies, and continuous, high-volume data loading. Information managers are adopting EDW best practices that push the scalability and performance envelope without sacrificing the agility to optimize this critical infrastructure to ever-changing analytic workloads. Some key best practices involve deploying 64-bit multicore EDW processing nodes, scaling out through shared-nothing massively parallel processing (MPP), pushing query processing to grid-enabled intelligent storage layers, applying efficient compression in the storage layer, and deploying preconfigured high-end EDW appliances.

TABLE OF CONTENTS

NOTES & RESOURCES

Forrester interviewed the following vendor companies for this report: Aster Data Systems, Greenplum, IBM, Infobright, Kognitio, Microsoft, Netezza, Oracle, ParAccel, SAP, Sybase, and Teradata. To create case studies, Forrester interviewed the following user companies in several verticals: CVS Caremark (retailing/prescription benefits management), LGR Telecommunications (hosted telecommunications analytics processing), Merkle (hosted database marketing service provider), MySpace (Web 2.0 social networking), and NYSE Euronext (securities trading exchanges).

Related Research Documents

"The Forrester Wave™: Enterprise Data Warehousing Platforms, Q1 2009"
February 6, 2009

"Forrester's Business Intelligence Data Architecture Decision Tool"
January 12, 2009

INFORMATION TSUNAMI: PUTTING PRESSURE ON THE ENTERPRISE DATA WAREHOUSE

The information tsunami continues unabated, even in a down economy. Even though organizations are feeling a severe crunch on the bottom line, their transactional applications continue to flood information workers with steady streams of intelligence that drive increasingly do-or-die business decisions. Enterprises everywhere continually struggle to keep their enterprise data warehousing (EDW) environments from being swamped by the inexorable pressure from growing data and usage volumes for business intelligence (BI), predictive analytics, data and content mining, and other key applications that hinge on information maintained in the EDW.

Information and knowledge management (I&KM) professionals feel the full brunt of the trend toward ever more massive, all-encompassing, flexible EDWs. They use every approach in their arsenals to scale, accelerate, and optimize the EDW, but do so under more stringent budget and resource constraints. At the same time, the enterprise's EDW platform — the data management hub of many BI applications — often lags two, three, or more years behind the times, complicating efforts to tune it for newer workloads. Given the significant sunk cost of the EDW and the range of analytic applications that depend on it, organizations often wait until they have maxed out its capacity before getting serious about scaling it further.

For I&KM professionals, the EDW scalability pain is growing more acute. Through all the turmoil in the corporate and competitive environment, they must continue to scale their EDW, accelerate queries and other operations, and expand the platform's agility to support mixed-query workloads, data volumes, and concurrent usage requirements.

From a performance-optimization perspective, the key pressure points are several:

- **EDWs consolidate ever more massive data sets.** Enterprises are growing their EDWs' production databases by one or more orders of magnitude, from the high gigabytes into the tens, hundreds, even thousands of terabytes (in the latter case, petabytes). Just 10 years ago, few EDWs grew as large as a terabyte. Today, in 2009, Forrester sees anecdotal evidence that approximately more than two-thirds of EDWs deployed in enterprises are the 1 to 10 terabyte range. Forrester estimates that by 2015, a majority of EDWs in large enterprises will be 100 TB or larger, with petabyte-scale EDWs becoming well-entrenched in such sectors as telecommunications, finance, and Web commerce. Driving this trend is the evolution toward EDWs that consolidate many formerly separate data marts and aggregate unstructured content for customer and market intelligence. More and more, EDWs also process, store, and aggregate trillions of transactional data such as event logs, clickstreams, and numerous device-generated digital signals in the healthcare, oil and gas, manufacturing and numerous other industries. EDWs need to persist — in other words, aggregate, store, and maintain — these large time-series data sets for in-database mining and support quick detail-level query for compliance reporting.[1]

- **Hub-and-spoke EDWs proliferate data domains, marts, and cubes.** I&KM pros must ensure that their production EDW data is being fed into an expanding range of dependent data marts, from a few to dozens, even hundreds, of separate physical databases and/or subject areas built on star, snowflake, and other dimensional data models.[2] Information managers need these downstream data marts and cubes to support fast queries against very large aggregates of structured data, and they need to do this in a flexible and agile manner, with the ability to respond quickly to ever-changing business requirements.

- **User and query concurrency expands against the EDW.** Enterprises must support steady growth in concurrent access to the EDW and mart data, from a handful of named users and queries into the hundreds or thousands of concurrent sessions. The number of concurrent users, sessions, and queries against the EDW will grow as more enterprises adopt business intelligence self-service and other pervasive BI technologies. Indeed, as the soft economy spurs enterprises to move toward self-service user dashboard and report development, thereby controlling IT costs, the trend toward pervasive BI will accelerate and concurrency against the EDW will skyrocket.[3]

- **Real-time BI comes to the fore.** Users are demanding continual reductions in end-to-end data latency — in other words, delay — through the EDW and out to marts and BI applications, from overnight batch loading to truly real-time, guaranteed, end-to-end latencies. Many BI professionals remark that users have come to expect that the data in their BI reports, dashboards, and scorecards — hence, in their EDW — be fresh and accurate up to the second. Clearly, this need for continual feeds, plus real-time interactive slice-and-dice of complex data sets, is putting fresh pressure on information managers and more stringent workloads and service-level agreements on the EDW.[4]

- **Complex content types flow in greater volumes into the EDW.** Many organizations are expanding the range of content types — from structured relational through semi- and unstructured data — that can be loaded, persisted, and processed in the EDW. Indeed, customer data integration (CDI), which is one of the principal applications of the EDW, is practically screaming for information managers to find a way of consolidating and mining the text-based customer and market intelligence bubbling up continually from blogs, social networks, and other Web 2.0 sources. Integrating unstructured data — content — into EDW will allow I&KM pros to run truly differentiated analysis, with richer data sets and more precise customer segmentation, potentially resulting in more-effective, more-focused, and better-targeted marketing campaigns with higher cross-sell, upsell ratios.

- **EDW becomes a key platform for predictive model execution.** Organizations are moving a wider range of compute- and data-intensive analytical workloads — such as data mining, predictive analysis, data scrubbing, and statistical analysis — to execute natively on the EDW, taking full advantage of the platform's sophisticated scale up and scale out parallelism. Many time-series analytical data sets are far too large to move in batch throughout enterprise networks. Consequently, data mining specialists have come to recognize that it's best to leave these massive data sets where they normally "live" — in the EDW — and to move the algorithms to that platform, rather than attempt bandwidth- and storage-clogging downloads to analytical workbenches.

BEST PRACTICES: EDW SCALING, ACCELERATION, AND PERFORMANCE OPTIMIZATION

To understand how some of the most successful information managers address the scalability challenge, Forrester spoke with database administrators (DBAs) and other practitioners regarding their best practices. Our interviews included CVS Caremark, LGR Telecommunications, Merkle, MySpace, NYSE Euronext, and others to provide a cross-industry perspective on the topic. Our expert interviews provide perspectives on what works well for I&KM pros to address a dizzying range of scalability, performance, and optimization concerns.

To effectively scale your data warehouse, you must first adopt a platform that can grow linearly with your organization's BI, ETL, and other workloads.

One key consideration is whether you should adopt a general-purpose EDW platform — such as those from Teradata, Oracle, IBM, Microsoft, Netezza, and HP — that you can scale for a broad range of capacity and performance requirements, including various database sizes, user and query concurrency levels, data-loading speeds and volumes, and load and query latencies. Alternately, you might wish to adopt a more special-purpose EDW platform scenario that is optimized for a particular application or go with a specialized EDW that is optimized for particular deployment scenarios, queries, and/or latencies. One category of special-purpose EDW platform includes Sybase, Vertica Systems, and ParAccel, which incorporate columnar databases and are best suited to deployment as scalable data marts in support of online analytical processing (OLAP) query acceleration against large table aggregates. Another category includes application-specific EDWs like SAP Business Information Warehouse (BW).

Another key scaling consideration is whether you have the optimal end-to-end data management architecture, which should balance resource utilization and workloads across the EDW, data marts, BI platforms, staging nodes, operational data sources, and other key platforms. In many circumstances, the best scalability practice may be to take loads off the EDW and migrate them to other platforms — such as moving ETL and data cleansing jobs to persistent staging nodes — and thereby improve performance levels for all applications. DW administrators should understand the scalability and performance pros and cons of different data management architectures, including centralized EDW, hub-and-spoke EDW, and data federation.[5]

Information managers may apply the following four best practices with either type of EDW platform, general-purpose or specialized, to varying degrees. Although these best practices may seem obvious, successful DW professionals take a serious attitude and rigorous approach to them:

- **Best practice No 1.** Scale your EDW through parallelism.

- **Best practice No. 2.** Accelerate the performance of your EDW with appliances.

- **Best practice No. 3.** Optimize your EDW's distributed storage layer.

- **Best practice No. 4**. Retune and rebalance your EDW's workloads regularly.

EDW SCALING BEST PRACTICE NO. 1: SCALE YOUR EDW THROUGH PARALLELISM

Information managers should scale their EDW by implementing parallel processing throughout its architecture. Note that at all levels of parallelization, you will also need to make sure your SQL queries and application code have been parallelized to make the most of this massive, virtualized EDW grid or cloud (see Figure 1).

While academia and technology vendors have multiple unsettled disputes as to what scale-up or scale-out approaches work best, here are capsule discussions of some of these best practices:

- **Scale-up EDW server nodes through shared-memory symmetric multiprocessing (SMP).** Query processing, ETL jobs, and other EDW workloads have many fine-grained processes that can be accelerated though the server platform's native shared-memory SMP features. Most EDW deployments are single-node installations that can be scaled up through more intensive application of SMP on machines with ever-speedier CPUs, more RAM, more I/O bandwidth, and bigger, faster disk subsystems. All of the case studies included in this report leverage SMP features native to their EDW appliance platforms, or to the underlying hardware operating environment, for scaling and optimization at the "node" (i.e., EDW server) level. Although this may seem like an obvious best practice, DBAs should be careful not to repurpose an ancient pre-SMP server in their data centers as an EDW, operational data store (ODS), or data mart — unless their users can tolerate query response times in minutes or hours.

- **Scale out through clustering and shared-nothing massively parallel processing (MPP).** Multinode server scale-out enables DBAs to maintain service levels as their EDW workloads grow and data sets expand into the hundreds or thousands of terabytes. Due to the contention inherent in shared-memory SMP, scale-out, at least to a second clustered server, often becomes necessary when the aggregate EDW data set grows much beyond 1 terabyte. When the number of EDW server nodes is in the typical range of two to four, DBAs should leverage the clustering features of their server operating system. When the number of nodes grows to the dozens, hundreds, or thousands, shared-nothing MPP is the best approach. A shared-nothing architecture eliminates the dependencies of nodes on common storage, memory, and processing resources, thereby maximizing linear scalability. All of the case studies included in this report use server clustering and shared-nothing MPP for scale-out, with each of the nodes incorporating SMP for internal parallelization.

- **Partition your shared-nothing MPP EDW grid into distinct query, hub, and staging tiers.** An EDW platform supports a pipeline of processes that extract data from sources, transform and load it into production tables and cubes, and facilitate fast queries. As you scale out your EDW through shared-nothing MPP, you should partition your single-system image multinode grid into distinct architectural tiers that are optimized to the principal pipeline processes. A back-end EDW tier should support staging, ETL, and data quality (DQ) processes. Another tier should support the principal EDW hub functions: managing data schemas, hierarchies, dimensions, and metadata, and maintaining the master production tables. A third tier should support online analytical processing, cubing, caching, and query processing. By maintaining this architecture, you will be able to scale and optimize each of these tiers independently to optimize performance on their diverse workloads.[6]

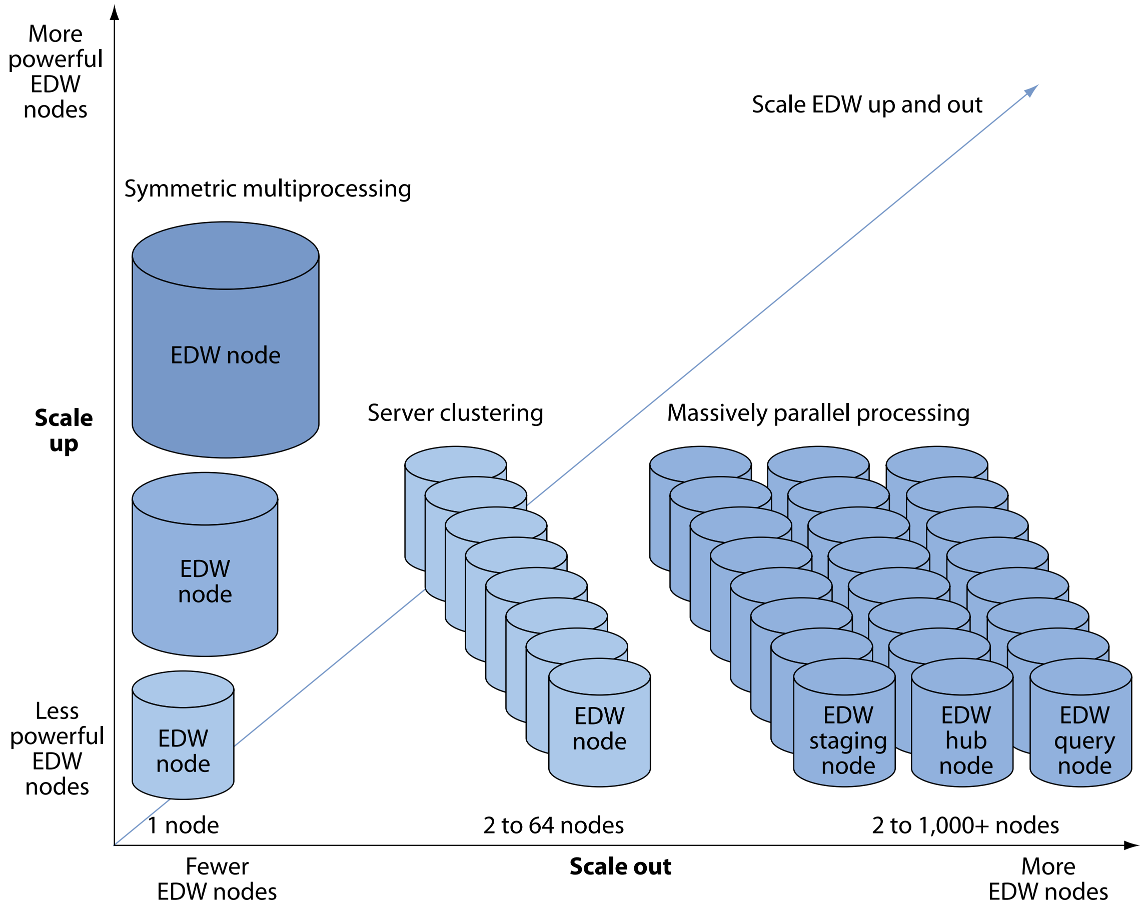
There are some best practices around a two-axis "scale-up" (SMP) versus "scale-out" (server clustering, MPP) graph (see Figure 2).

Figure 1 Approaches To EDW Scalability

Approach	Description	When to use	Pros and cons
Symmetric multiprocessing (SMP)	• Scale-up approach • Parallelizes multithreaded task execution • Native to server operating environments	• Deployment with single-node EDW server with up to 10 TBs of data • Multinode EDW deployments requiring internally parallelized nodes	• Pros: relatively inexpensive; SMP natively available on many server OS platforms; enables efficient resource utilization on existing EDW node • Cons: not scalable into 100s of TBs on single node; shared-memory and shared-disk architecture limits intranode scalability
Server clustering	• Scale-out approach • Symmetric or asymmetric partitioning and load balancing across nodes • Single system image	Deployments with two to 64 database server compute/storage nodes and up to low-100 TBs of data	• Pros: server clustering natively available on many server OS platforms; scalable across dozens of servers into 100s of TBs • Cons: moderately expensive; rarely scales beyond six to eight clustered nodes in real-world deployments, or much beyond the low 100s of TBs
Massively parallel processing (MPP)	• Scale-out approach • Symmetric or asymmetric task and data partitioning and load balancing across nodes • Single system image	Deployments with 64+ database server compute/storage nodes and up to petabytes of data	• Pros: scalable into 1,000s of TBs and 1,000+ nodes in a single cluster; shared-nothing architecture enables linear scale-out into the petabytes • Cons: very expensive; complex; difficult to parallelize and optimize workloads across grid

46489 Source: Forrester Research, Inc.

Figure 2 Scale Your EDW Through Parallelism



46489

Source: Forrester Research, Inc.

Scale Your EDW Through Parallelism: Pitfalls To Avoid

Information managers should make sure that they avoid such EDW scalability constraints as single-core CPUs, nonparallelized application code, and shared-disk architectures. Failure to address these constraints will keep EDWs from scaling to support larger data sets, faster data loads, more diverse mixed queries, and greater user and transaction concurrency:

- **Don't process DW loads on single-core CPU platforms.** Many enterprises may still be running their EDWs on older single-core CPU platforms or may attempt to repurpose older, slower hardware for EDW functions such as staging nodes, multidomain hubs, or data marts. Considering the large and growing load on all EDW nodes, every one of those nodes should be running on the latest, fastest server platforms available. All of the case studies interviewed for this report are leveraging the performance benefits of 64-bit multicore CPUs such as Intel Itanium in their EDW server nodes.

- **Don't forget to parallelize your application code.** Often, parallelizing older SQL code can be very difficult, costly, and time-consuming. Indeed, the ability to write new SQL code to take advantage of fine-grained SMP is not within every programmer's skill set. At the very least, you should use a parallelizing compiler. However, even many auto-parallelizing compilers provide spotty support for SQL and other domain-specific languages. A better practice than parallelizing the code manually is to leverage the EDW vendor's auto-parallelization features or transactional DBMS vendor's query-optimization tool to ensure that each query step is fully parallelized and that there are no single-threaded operations (scans, joins, index accesses, aggregations, sorts, inserts, updates, or deletes) in your SQL query access plan. In its case study, NYSE Euronext stresses the importance of continuing to rewrite queries to optimize them for each EDW vendor's specific database and parallelization architecture.

- **Don't rely on a shared-disk architecture.** One of the principal performance issues in EDWs is I/O bandwidth, which is often limited to the constraints of shared-disk architectures such as storage area networks (SANs). If you implement shared-nothing MPP only in, say, the EDW's query-processing and production data-hub tiers, but retain a shared-disk I/O architecture, you will be perpetuating an I/O bottleneck that will slow retrieval from disk, hence introducing latency into query responses. Consequently, you should introduce shared-nothing MPP in a "single-tier" architecture across all nodes, providing each node with access to its own direct-access storage, thereby effectively expanding I/O bandwidth by an order of magnitude. It's also important to ensure that all nodes in a single-tier shared-nothing MPP grid have access to a high-bandwidth interconnect backbone, such as 20 Gbps switched Infiniband network.

EDW SCALING BEST PRACTICE NO. 2: ACCELERATE YOUR EDW WITH APPLIANCES

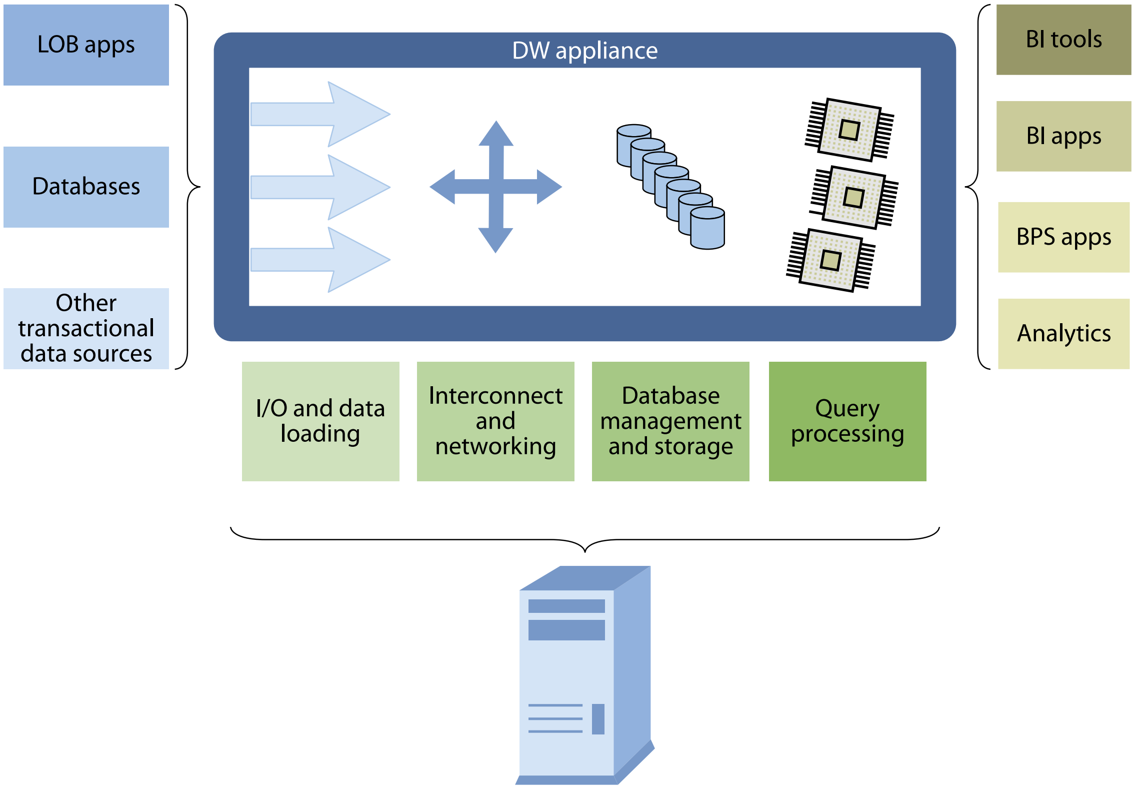
Information managers should scale their EDW by implementing high-performance hardware/software appliances at every node, thereby preventing slow nodes from dragging down the performance of a distributed EDW. Appliances should be deployed to address key performance pain points — such as for front-end OLAP acceleration in the data mart — and then, to maintain a balanced, high-performance deployment, throughout your distributed EDW cluster or grid:

- **Adopt EDW appliances to speed performance.** EDW appliances are preconfigured, modular devices that support quick deployment for core functions. They prepackage and pre-optimize the processing, storage, and software components for fast queries, data loads, and other operations. Appliances can help you reduce time-to-value and cut life-cycle costs at all EDW compute and storage nodes. When evaluating commercial EDW appliances, you should use the same criteria as with any equivalent DW software solution, including price-performance, functionality, flexibility, scalability, manageability, integration, and extensibility. Most of the case studies in this report involved EDW appliances, and all of those reported significant improvements in query and/or load performance vis-à-vis traditional non-appliance-based EDW deployments.

- **Deploy EDW appliances initially at nodes requiring immediate performance boosts.** Consider deploying EDW appliances in specific tactical roles — such as standalone data marts for department- or function-specific data marts. Rethink your enterprise data management architectures with an eye to offloading as much CPU- and storage-intensive functionality to appliances as possible, as commercial appliance offerings become available. Cost-effective scaling of your DW, OLAP, and BI environments depends on accelerating complex queries and bulk data loading through pre-optimized solutions that were designed for these functions. OLAP acceleration has been a key application where appliances have paid off, speeding complex queries against large table aggregates maintained in dimensional data structures (see Figure 3). Proving out DW appliances in specific, quick-payback deployments will help you gain support for more broad-based enterprise deployments going forward.

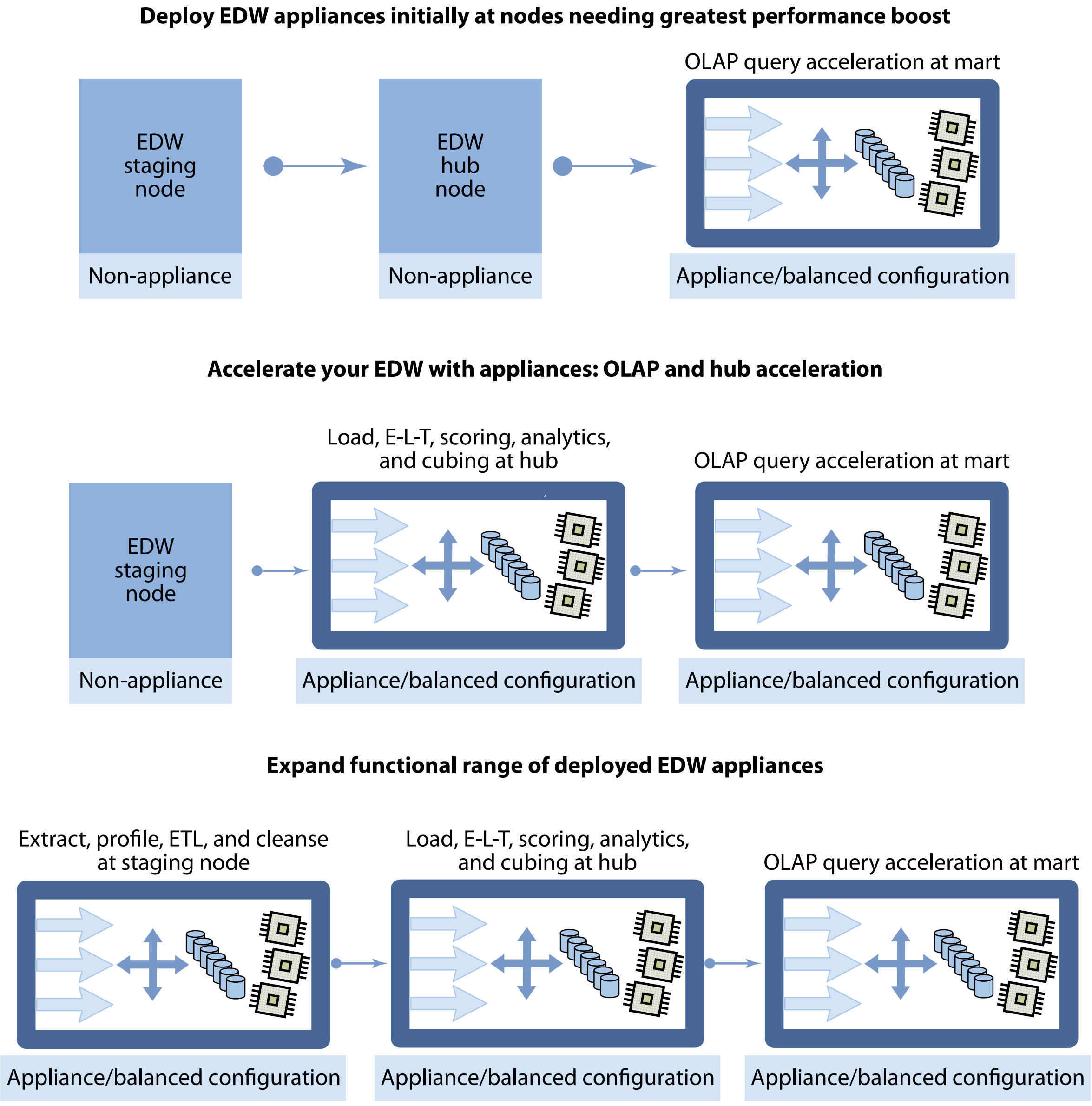
- **Expand the functional range of your deployed EDW appliances.** Everything in your EDW can benefit from a hardware-accelerated performance boost. Enterprises should deploy hardware-optimized EDW appliances in a shared-nothing MPP grid, in which specific appliance nodes are specialized to such functions as processing OLAP queries, maintaining production data tables, processing complex transforms, scoring records for data-mining and predictive modeling, cleansing and de-duplicating data, and managing continuous and batch loads from sources. An example of this type of deployment could be an acceleration of EDW hub functions in addition to OLAP queries through appliances deployed at each of those tiers, as well as appliance-based acceleration of functions from end-to-end data integration, hub tier, and OLAP/mart tier — throughout a distributed EDW (see Figure 4). You may be able to acquire these features as add-on hardware/software modules from your EDW appliance vendor. Alternately, to the extent that your EDW appliance vendor supports an extensibility framework such as MapReduce, you may be able to add these features to your appliance deployment through custom programming.[7]

Figure 3 Accelerate Your EDW With Appliances: OLAP Acceleration



46489 Source: Forrester Research, Inc.

Figure 4 Accelerate Your EDW With Appliances


Deploy EDW appliances initially at nodes needing greatest performance boost
OLAP query acceleration at mart
EDW staging node
EDW hub node
Non-appliance
Non-appliance
Appliance/balanced configuration
Accelerate your EDW with appliances: OLAP and hub acceleration
Load, E-L-T, scoring, analytics, and cubing at hub
OLAP query acceleration at mart
EDW staging node
Non-appliance
Appliance/balanced configuration
Appliance/balanced configuration
Expand functional range of deployed EDW appliances
Extract, profile, ETL, and cleanse at staging node
Load, E-L-T, scoring, analytics, and cubing at hub
OLAP query acceleration at mart
Appliance/balanced configuration
Appliance/balanced configuration
Appliance/balanced configuration

46489 Source: Forrester Research, Inc.

Accelerate Your EDW With Appliances: Pitfalls To Avoid

I&KM professionals should steer clear of the temptation to regard appliances as a panacea for EDW scalability and performance problems. Rather, you should benchmark commercial EDW appliances against your workloads, configure the chosen EDW appliance with a balanced configuration, and rewrite your queries and ETL to optimize them for that specific deployed appliance.

- **Don't fail to benchmark commercial appliances against your specific workloads.** Comparing diverse EDW appliance solutions is difficult if you do not consider the types of query execution that you need them to execute. If you don't benchmark EDW appliances yourself, you won't have the information needed to determine which offering is optimized to your needs. For example, brute-strength table scans are the forte of DW appliances that incorporate relational databases, but these table scans are suboptimal if your BI application is doing mostly single-row lookups. In a similar vein, DW appliances that incorporate columnar databases support a high degree of compression that is tuned to each column/attribute's specific data type, but columnar databases lose their efficiency advantage on queries that involve SELECT operations across many columns or even a "SELECT *" in a worst case. Hands-on testing is needed to determine which approach and vendors best support requirements for processing complex OLAP workloads, concurrent DW usage, fast bulk loading, workload optimization, and analytic-database scalability.[8]

- **Don't deploy an appliance that lacks a balanced configuration.** An EDW appliance, indeed any EDW deployment, needs to have a balanced configuration of CPU, memory, I/O, and storage to avoid introducing performance-sapping bottlenecks. One of the most common EDW bottlenecks is fast-querying processing CPUs waiting for data to arrive from slow disks over narrow I/O bandwidth. This is one of the chief symptoms of an unbalanced EDW architecture, and appliances are not immune to this problem. An appliance must be configured to balance storage, I/O, memory, and processors so that all queries and other jobs receive adequate resources to meet their service levels, and also to ensure efficient capacity utilization across all EDW subsystems. For I&KM pros, it is best to follow the vendor's recommended balanced configuration for an EDW appliance that is suited to your requirements rather than to try balancing your appliance's configuration yourself through trial and error.[9]

- **Don't neglect to tune queries and ETL scripts to work with your chosen appliance.** EDW appliances are not a magic bullet for superior performance. At the very least, you need to retune your queries and load scripts to each appliance you deploy in your environment. One query plan is not necessarily optimal across all DBMS or DW platforms, nor is every ETL or data cleansing script. Determine the extent to which commercial EDW appliances integrate with your existing BI, DI, DQ, and DBMS environments — or will require extensive modifications to existing enterprise software in order to operate at full efficiency. For example, ask if you will need to make modifications to your SQL queries, analytic applications, and ETL scripts to integrate completely with a DW appliance. Most EDW appliance vendors will provide you with as-needed implementation support to tune your queries and ETL jobs, but you should obtain this commitment before committing to their solution. More generic appliances may require less tuning: CVS Caremark reports that it did not need to make significant modifications to its MicroStrategy BI queries or Informatica ETL scripts to run optimally in the firm's Teradata EDW grid.

EDW SCALING BEST PRACTICE NO. 3: OPTIMIZE YOUR EDW'S DISTRIBUTED STORAGE LAYER

I&KM professionals: Optimize your EDW's distributed storage layer to ensure maximum scalability and performance for specific workloads. You should use compression and other techniques to minimize data's footprint, while deploying the optimal database technology at each node and tuning your EDW's joins, partitions, and indexes.

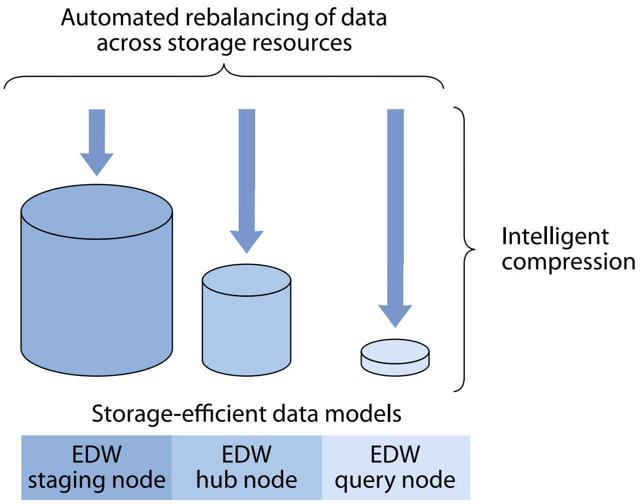
- **Intelligently compress and manage your EDW data to realize storage efficiencies.** DBAs should apply intelligent compression to their EDW data sets, including tables and indices, to reduce their footprint and make optimal use of storage resources. Intelligent compression encompasses diverse metadata-driven techniques for generating compact data footprints for storage, transmission, and/or processing. Also, some physical data models are more inherently compact than others (e.g., tokenized and columnar storage are more efficient than row-based storage), just as some logical data models are more storage-efficient (e.g., third-normal-form relational is typically more compact than large denormalized tables stored in a dimensional star schema).

  One downside to intelligent compression is that it can degrade EDW performance by making CPUs shoulder the workloads associated with on-the-fly compression and decompression. DBAs can achieve balance — keeping data compressed in storage while supporting on-the-fly decompression — by performing the decompression in specialized hardware close to disk, thereby taking that load off the front-end query processors. DBAs can realize compact EDW storage by applying these various approaches (see Figure 5).

- **Deploy nontraditional databases to optimize to particular EDW and BI roles.** No one data storage, persistence, or structuring approach is optimal for all deployment roles and workloads. For example, no matter how well-designed the dimensional data model is within an OLAP environment, users eventually outgrow these constraints and demand more flexible decision support. By requiring that relational data be denormalized and prejoined into star schemas and other fixed, subject-specific structures, traditional multidimensional OLAP denies users the flexibility to drill down, up, and across data sets in ways that were not designed into the underlying cubes. Other storage approaches — such as columnar, in-memory, and inverted indexing — may be more appropriate for such applications, but not generic enough to address other broader deployment roles.[10] DBAs can optimize EDWs through deployment of role-fitted databases at various tiers (see Figure 6).

- **Optimize schemas, joins, partitions, and indexes to your EDW's queries and data structures.** No fixed EDW data-modeling approach — physical and logical — can do justice to the ever-shifting mix of queries, loads, and other operations being performed against your growing, evolving pool of data and storage resources. The more dynamic the demands on your EDW, the more often you will need to revisit your data schemas, join strategies, and partitioning and indexing approaches to maintain acceptable performance for all users and applications. Make sure you adopt tools that let you focus on logical data models — such as star or snowflake schemas — while the tool automatically reconfigures the underlying EDW physical data model for optimal query and data load performance. Tools can accomplish this by constantly monitoring actual query usage statistics. You should also make full use of the EDW vendor's tools that allow you to assess the performance impacts of various joining, partitioning, and indexing options.

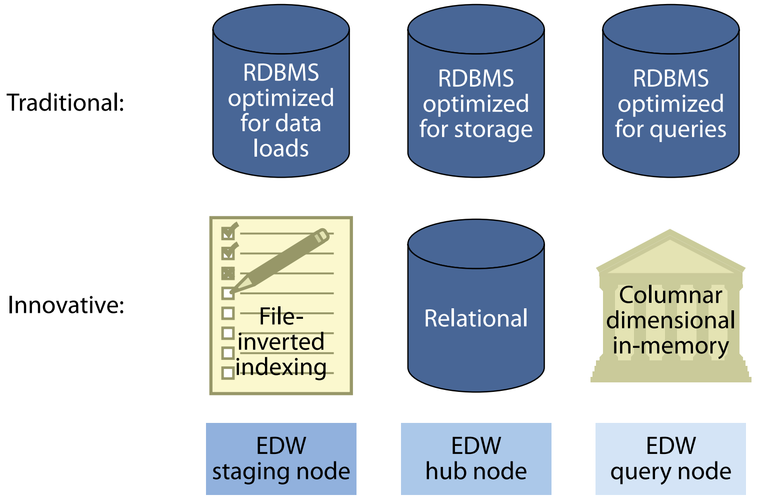
Figure 5 Intelligently Compress And Manage Your EDW Data To Realize Storage Efficiencies


Automated rebalancing of data across storage resources
Intelligent compression
Storage-efficient data models
EDW staging node
EDW hub node
EDW query node

46489

Source: Forrester Research, Inc.

Figure 6 Optimize Your EDW's Distributed Storage Layer: Deployment Role-Fitted Databases


Traditional:
RDBMS optimized for data loads
RDBMS optimized for storage
RDBMS optimized for queries
Innovative:
File-inverted indexing
Relational
Columnar dimensional in-memory
EDW staging node
EDW hub node
EDW query node

Consider deploying columnar, in-memory, dimensional, and other nontraditional databases where optimized to particular EDW and BI deployment roles.

46489 Source: Forrester Research, Inc.

Optimize Your EDW's Distributed Storage Layer: Pitfalls To Avoid

I&KM professionals should continue to tune their distributed storage layer, recognizing that changing requirements and applications may make yesterday's approach suboptimal. To ensure that your EDW retains the agility to support evolving workloads, you should regularly reassess your compression, database, joining, partitioning, and indexing schemes. Don't treat any particular storage approach as the one true religion that fits all requirements forever and ever.

- **Don't apply a compression scheme ill-suited to your EDW's data sets.** DBAs sometimes fail to apply any compression to their stored records or only use the default compression schemas without exploring more storage-efficient intelligent-compression options available on their EDW platform. Consequently, the data may consume excessive storage and I/O bandwidth, thereby slowing query response and throughput on the EDW. Considering the expense of these resources, and the frequent availability of underutilized EDW CPU power, it makes more sense to compress the data for storage and transmission and then decompress on the fly at the front-end query. Compression efficiencies of 60% to 70% are typical. DBAs should leverage metadata, dictionaries, and statistics to tailor compression/decompression functions to data types, ranges, distributions, partitioning, and other characteristics. They should also explore using tokenized physical storage for maximum de-duplication of repetitive patterns, sequences, and regions.

- **Don't treat narrowly focused database architectures as the silver bullet.** As noted above, DBAs must realize that no one database architecture can be the universal, preferred solution for all scenarios. Though some in the EDW industry regard particular architectures — such as columnar and inverted-indexing — as superior, the market has not migrated en masse to these approaches, as they are not generic enough and may require different data modeling and administration techniques. Indeed, the fact that most EDW platform vendors — including such leaders as Teradata, Oracle, IBM, and Microsoft — continue to base their solutions on traditional relational/row-wise databases shows that this approach still has considerable legs. Columnar's advantages in support of complex queries against large denormalized table aggregates can be addressed in relational environments through such techniques as vertical partitioning, materialized views, results caches, projections, and cost-based query optimization. Likewise, the storage efficiencies of inverted indexing can be approximated on relational databases through intelligent compression.

- **Don't forget to regularly reassess your data structures to suit changing workloads.** When queries slow, DBAs often look first at capacity issues with servers, storage, and interconnects on the EDW, though in fact the problem may be that underlying data structures have grown suboptimal for a changing query workload. For example, users may be doing more complex queries that require a large number of table joins, which not only introduces latency but consumes excessive CPU and storage resources. Also, many queries may now primarily be accessing data on one overworked storage device, due to the DBA having neglected to repartition data sets in keeping with changing access patterns. Likewise, you may not be indexing all the principal fields that users are now looking up, a deficiency that will add considerable latency to queries against your EDW. Query statistics and DBMS resource monitoring tools can help you pinpoint these data structure issues and quickly rectify them.

EDW SCALING BEST PRACTICE NO. 4: RETUNE AND REBALANCE WORKLOADS

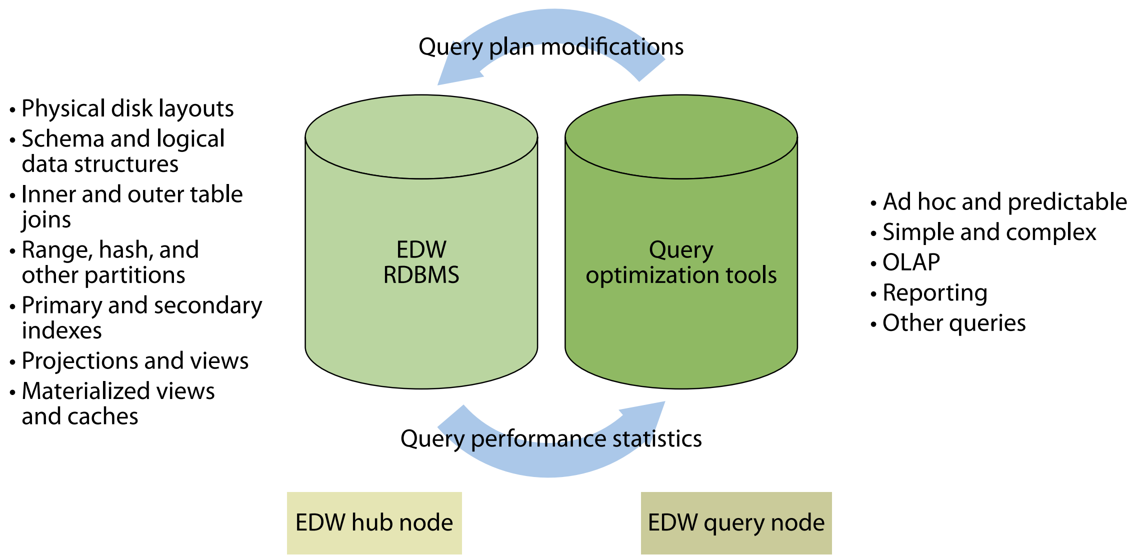
I&KM professionals should implement an agile EDW that can be quickly, flexibly, and automatically optimized to ever-changing workloads. You should retune your queries regularly, use workload management tools to dynamically allocate EDW workloads and resources, and optimize the distribution of key functions across distributed EDW mart, hub, staging, and other nodes.

- **Optimize your queries as often as necessary.** Predictably, fast queries are the key criterion for EDW optimization, and lack thereof is often the chief source of user dissatisfaction. I&KM professionals should use all the query monitoring, planning, and optimization tools at their disposal and retune their queries regularly. You can speed queries through such approaches as cost-based optimization, query preprocessing, results caching, materialized views, predicate pushdown, and rewrite-based auto-tuning (see Figure 7). DBAs should create baseline performance numbers and review them against increased workloads or increased database size. When you see degradation of 5% or more, that's a good time to start exploring further query optimization. NYSE Euronext regularly retunes queries to run efficiently against its Greenplum and Netezza EDW appliances, due to stringent query performance requirements from users in its trading exchanges.

- **Use EDW workload management tools to dynamically optimize workloads.** Even if you retune your queries regularly, you still run up against the fact that your EDW may be processing many types of queries side by side, including simple and complex, ad hoc and production, single-table and multitable, batch and real-time.[11] At the same time, the EDW may be handling concurrent access by tens or hundreds of users. The EDW may also be processing ETL, continuous CDC, bulk data cleansing, and compute-intensive data-scoring jobs. On top of all that, your DBAs may be performing regularly administrative tasks, such as backup and recovery of data sets. The only way to ensure that you meet service levels on all of these disparate jobs is to use policy-based workload management tools that dynamically allocate EDW CPU — memory, storage, and bandwidth among them — thereby warding off resource contention (see Figure 8).

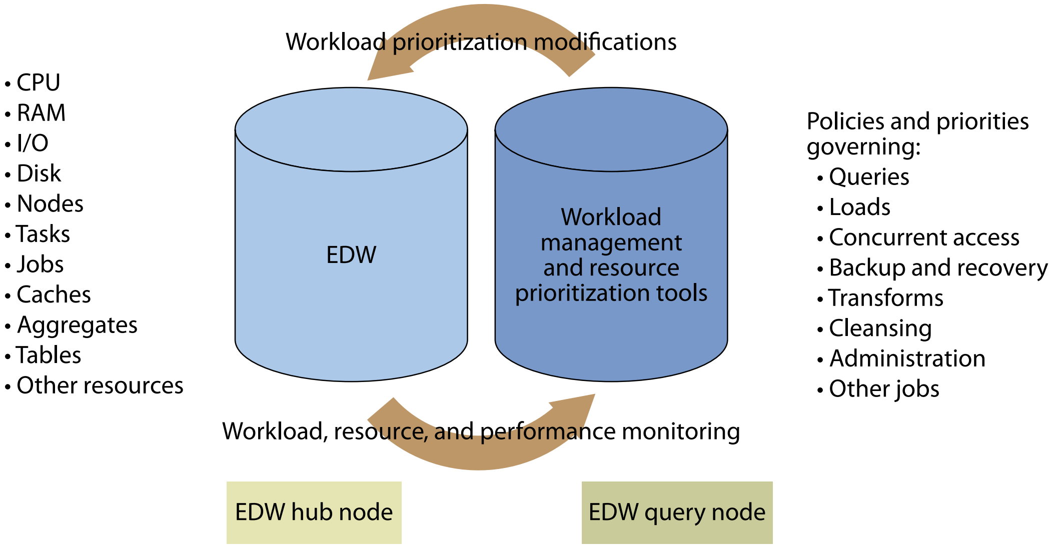
- **Balance resource utilization across distributed EDW ecosystem.** An EDW is not an island, and its workloads are integrally connected to processes performed by other systems, including data sources, staging nodes, cubing engines, and data marts. DBAs should balance the workloads processed across the entire EDW ecosystem. For starters, you should schedule batch ETL jobs for windows that minimize resource burdens on source systems and staging nodes. If you move toward more real-time BI through the EDW, be sure to provision the necessary bandwidth, processing, and caching infrastructure from end to end to guarantee low latencies. If you move toward a hub-and-spoke EDW with dependent OLAP data marts, you should make sure that cube refreshes are scheduled to minimize performance impacts on the EDW hub. DBAs can balance end to end EDW resource utilization throughout a distributed EDW deployment (see Figure 9).

Figure 7 Optimize Your Queries As Often As Necessary



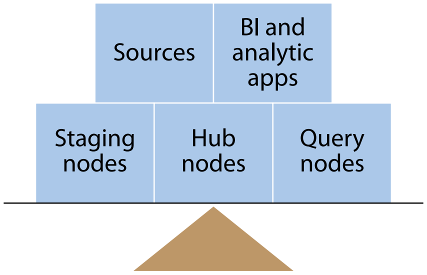
46489 Source: Forrester Research, Inc.

Figure 8 Use EDW Workload Management Tools To Dynamically Optimize Workloads



46489 Source: Forrester Research, Inc.

Figure 9 Balance Resource Utilization Across All Nodes In A Distributed EDW Ecosystem



46489 Source: Forrester Research, Inc.

Retune And Rebalance Your EDW's Workloads Regularly: Pitfalls To Avoid

I&KM professionals should avoid flying blind where EDW workload optimization is concerned. You should base ongoing query optimization on a steady stream of performance statistics, while also using EDW performance consoles and policy tools to maintain service levels across mixed workloads. In a distributed EDW, you should also avoid overloading the front-end compute nodes with all query-processing functions. Balancing EDW query-processing workloads requires that you move the most resource-intensive function — predicate processing — into an intelligent storage layer.

- **Don't neglect to collect or leverage query-performance statistics.** Query statistics are a DBA's best friend, because they can help you do root-cause analysis related to performance issues.

Statistics, both historical and live, can also help identify suboptimal query plans, spot resource bottlenecks, understand performance trends, plan capacity upgrades and reconfigurations, and preempt quality-of-service issues before they manifest themselves. But statistics cannot deliver any of these benefits if DBAs fail to collect them and leverage them for ongoing query optimization. If you imagine that your queries will magically auto-optimize themselves, and that no purpose is served by doing statistical analysis of their past performance, your increasingly frustrated users will soon enough call your attention to intolerable wait times.

- **Don't fail to enforce service-level policies across mixed queries, loads, and latencies.** Workload management and resource/query governor tools are the traffic cops of the EDW platform. They help isolate, prioritize, allocate, and balance loads so that the entire EDW can hum smoothly. They are typically used to control CPU, memory, and disk I/O usage per query, user, or workload. If you find that your ETL jobs are consuming so much CPU capacity that critical user queries are suffering, you have failed. You should be using your EDW platform's workload management tools to define policies that meet all service levels pertinent to all operations, ensuring the requisite latencies on all queries, loads, and other operations.[8]

- **Don't forget to push query processing to an intelligent data-storage layer.** Query processing is a time-sensitive EDW function that must often be distributed over two or more distributed nodes. To the extent that you perform all query processing functions in the EDW's front-end host/compute tier, you risk introducing a bottleneck that will bog down your queries. One approach for easing this constraint is to perform query preprocessing at the intelligent storage layer continuously, as data is fetched from direct-access storage (while continuing to do results aggregation and delivery in the compute tier). To accelerate this query preprocessing functions — which is often called "query predicate pushdown" — it should be handled by dedicated hardware that is tightly integrated with the EDW appliance's direct-access storage devices. For example, query predicate pushdown is a core feature of Netezza's architecture, and also of the recently released HP Oracle Database Machine with Exadata Storage. Both of these intelligent-storage-enabled EDW platforms are represented in this report's case studies.

FORRESTER'S EDW SCALING NEXT PRACTICES

While our research uncovered a number of EDW scaling best practices, here are some next practices that I&KM professionals should focus on once they've mastered the basics:

- **Leverage shared-nothing MPP for in-database analytics.** I&KM professionals are adopting an emerging best practice known as "in-database analytics." Under this practice, data mining, predictive analysis, and other compute-intensive analytic functions are migrating to the EDW platform to leverage its full parallel-processing, partitioning, scalability, and optimization functionality. DBAs, data modelers, and advance analytics specialists should explore the various vendor-proprietary and open industry frameworks and APIs from such vendors as Teradata, Oracle, Netezza, Greenplum, and Aster Data. Most noteworthy in this regard is the Google-developed MapReduce framework and SQL extensions, which have been adopted by several EDW platform vendors. In-database DW analytics can either replace or supplement traditional analytics execution approaches.

- **Federate EDWs through SOA, EII, and semantic abstraction layer.** Data federation — sometimes known as enterprise information integration (EII), semantic data integration, or data virtualization — may allow some organizations to do without an EDW if they wish. Data federation is an umbrella term for a wide range of operational BI topologies that provide decentralized, on-demand alternatives to the centralized, batch-oriented architectures characteristic of traditional EDW environments. Within an on-demand data federation environment, data delivery usually relies on an EII middleware layer, data persistence involving distributed caches, and data transformation relying on a semantic abstraction layer and registry. Essentially, data federation allows organizations to scale out into a "virtual EDW" that aggregates many scattered EDWs, operational data stores (ODSs), independent data marts, line-of-business databases, and other repositories. In addition, the data federation approach allows you to effectively balance query loads across these diverse data domains throughout your decentralized data management fabric.

- **Selectively deploy cloud-based EDW services for particular functions or workloads.** Cloud-based virtualization is beginning to seep into analytic infrastructures. To support flexible mixed-workload analytics, the EDW, over the coming five to 10 years, will evolve into a virtualized, cloud-based, and supremely scalable distributed platform. The virtualized EDW will allow data to be transparently persisted in diverse physical and logical formats to an abstract, seamless grid of interconnected memory and disk resources and to be delivered with subsecond delay to consuming applications. EDW application service levels will be ensured through an end-to-end, policy-driven, latency-agile, distributed-caching and dynamic query-optimization memory grid, within an information-as-a-service (IaaS) environment. Analytic applications will migrate to the DW platform and leverage its full parallel-processing, partitioning, scalability, and optimization functionality. At the same time, DBAs will need to make sure that cloud-based DW offerings meet their organizations' most stringent security, performance, availability, and other service-level requirements. Given the immaturity of cloud-based DW services, public cloud providers will find themselves severely challenged to prove they are indeed enterprise-grade.

CASE STUDIES

CVS Caremark

CVS Caremark— a large US prescription benefits management (PBM) and retailing firm — focuses on EDW scale-out. It maintains a multinode Teradata-based EDW that has grown over time to support a growing analytical data set and query workload for its PBM business. From a single Teradata 5500 appliance node early in 2008, the deployment grew to three 5500 nodes by year-end

2008 and is expected to scale out to 14 5500 nodes in mid-2009; all nodes run on 64-bit Linux. Another factor behind the steady growth of CVS' PBM EDW is ongoing consolidation of formerly siloed data marts and an operational data store (ODS) into a Teradata grid, which reduces ongoing costs associated with hardware, software, and operations. A third factor behind CVS' PBM EDW's growth is the continual development of new PBM applications that both serve transactional data to and query data that is delivered through the Teradata grid.

CVS expects that the PBM Teradata grid's storage capacity, 20 TB, will be totally maxed out by mid-2009 in support of transaction growth, mart consolidation, and new applications, and that it will need to add more storage/compute nodes to keep pace with that demand. Over the next three years, the firm predicts that data size and transaction volumes on its EDW will quadruple. For its PBM business, CVS has standardized on Teradata's EDW platform for several performance-related reasons. First, Teradata's EDW grid offers linear scalability that the firm cannot obtain on existing EDW and data mart platforms that lack a shared-nothing MPP architecture. Also, the firm did not need to modify any queries from its existing BI environment, MicroStrategy, to get those queries to run as efficiently as possible against Teradata's database. Furthermore, Informatica PowerCenter, CVS' principal ETL/extract-load-transform (E-L-T) tool, runs much faster on Teradata than on the existing third-party DW database. Also, Teradata did not require as much storage for indexes as does the rival DBMS on which it has been running many of its siloed marts (caveat: that rival DBMS is a version three generations old).

LGR Telecommunications

Global service provider LGR Telecommunications pursues both a scale-up and scale-out approach. It has implemented Oracle Database 10g to support a hosted EDW subscription offering for the telecommunications industry, focusing on high-volume call detail record (CDR) analysis. LGR runs the service from two US-based data centers, with active-active failover between the main and mirror sites, both of which have enough capacity to process the full workload. Each datacenter has a two-node, active-active Oracle Database Real Application Cluster (RAC) deployment per site. Each node is a bladed HP Superdome server that incorporates 128 Intel Itanium CPUs, 5 GB RAM, and two HP StorageWorks XP 24000 direct-access storage subsystems. The nodes pull CDR details from local storage area networks (SANs). Each node currently persists and processes around 310 TB of CDR data.

The company has many customers, the largest of which processes 3 billion new CDRs per day on its hosted service, and the transaction volumes and associated storage and processing requirements continue to grow. For another customer, it hosts a CDR database consisting of 310 TB of user data on 1.2 petabytes of raw storage resources. On the aggregate, the service manages concurrent access and queries to the hosted EDW from thousands of end users. Data loads from CDR sources — carrier network switches — come in near real time, with as much as 40,000 new CDRs per second being loaded on the Oracle 10g nodes. Though it has not put an appliance-based EDW into production, LGR is beta testing the recently released HP Oracle Database Machine with Exadata

Storage, which accelerates query-predicate processing in an intelligent, shared-nothing MPP storage layer. It has deployed the Exadata storage layer behind its existing HP Superdome servers running Oracle 10g. It reports that Exadata has reduced response times on some queries by half.

MySpace

MySpace — a Web 2.0 social-networking service provider owned by News Corp.'s Fox Interactive Media group — is scaling out its clickstream analytics EDW to petabyte scale and has designed a tiered MPP grid to support fine-grained ongoing optimization of loading, query, and other key functions. The service provider has implemented Aster Data Systems' nCluster MPP EDW software solution, which is not packaged as an appliance, over a distributed MPP grid of commodity Dell hardware servers with local direct-access storage. The service provider has implemented the Aster solution in a tiered cluster architecture of "queen," "worker," and "loader" nodes to support precision parallel scaling of specific functions in keeping with dynamic changes in the data, queries, and applications being performed on the EDW grid. The MPP grid runs across the firm's several data centers throughout the US.

Currently, the service provider's Aster MPP EDW grid includes 100 server nodes managing an aggregate 250 TB of clickstream data. The EDW grid loads 7 billion new clickstream event data records — amounting to 2 TB of raw data — per day, in hourly batches, from its thousands of Web and application servers, as well as from a data-collection server farm. These clickstream events are from the continuous activities of the 250 million users worldwide of its social-networking services. In addition to clickstream analysis, the firm's analysts access the distributed data on the Aster grid in support of ad hoc query, data mining, marketing, and customer experience optimization applications. Going forward, the service provider plans to rely more heavily on the Aster MPP EDW grid to execute a lot of the compute and data-intensive tasks it currently runs on other platforms. These tasks include ETL, data cleansing, predictive analytics, and data mining. It plans to continue fine-tuning its functional tiered clustering of Aster nodes, creating dedicated clusters for the above-listed EDW applications, as well as for various subject domains, for very large data sets, and for real-time-update transactions.

NYSE Euronext

NYSE Euronext, a global trading exchange with operations in North America and Europe, is scaling and accelerating its EDW through the deployment of appliances for massively parallel scale out. It uses both EDW appliances for reporting, surveillance, and other analytics on detail-level transaction data. A key driver behind NYSE Euronext's adoption of EDW appliances has been 60% to 100% yearly growth in transaction volumes across its equities, options, and futures exchanges on both sides of the Atlantic. The exchange's EDW platforms must support near-real-time loading and ad hoc queries with response times in the milliseconds and microseconds. In addition, each deployed EDW platform must be able to support ad hoc queries of 100 users daily, where queries are often very complex and can access as much as 40 to 50 TB of detail-level transaction data per query.

These requirements have spurred the company to adopt both Greenplum and Netezza for different analytic applications. Though those vendors' appliances have different parallel-processing architectures, they both offer the query-processing horsepower and scalability to address the most demanding decision support requirements. In the US, the exchange has implemented a 20-node shared-nothing MPP EDW using Greenplum's appliance, which runs on Sun X4500 servers and external disk arrays. The grid stores 100 TB of transaction data. The firm has also implemented several Greenplum nodes in a smaller grid for development and quality assurance. It plans to expand the production Greenplum grid to 40 nodes by the end of 2009. It anticipates that over the next several years, it will scale the Greenplum grid to a petabyte and beyond. To control storage costs, the exchange currently maintains no more than the latest 30 days of transaction data on the Greenplum appliances before archiving, and compresses the data on the appliance. It loads more than a terabyte of new data on the Greenplum grid every day, in near real time.

Merkle

Merkle, a US-based database-marketing company, maintains several EDWs, operational data stores (ODS), and data marts on different DBMS platforms, with a growing adoption of EDW appliances for near-real-time query acceleration and data loading. EDW platforms support the firm's hosted subscription services for marketing campaign management and customer data integration. The firm has deployed EDWs from Microsoft, Netezza, Oracle, and ParAccel. Customer preferences have spurred the firm to deploy several EDW/DBMS platforms side-by-side in its data centers.

Most of the firm's analytical data is in Microsoft SQL Server 2005. It maintains a large EDW on a cluster of SQL Server 2005 nodes, maintaining more than 100 data marts and an aggregate 250 TB across different physical servers. It is in the process of migrating the cluster to SQL Server 2008 to take advantage of that DBMS version's better data compression, which promises a 3-to-1 improvement over the compression on SQL Server 2005. The firm will expand its adoption SQL Server and phase out Oracle Database for data marts. However, it plans to take a wait-and-see attitude on Microsoft's promised "Project Madison" shared-nothing MPP appliance for SQL Server, which will not be available for beta-testing till late 2009, at the earliest.

SUPPLEMENTAL MATERIAL

Companies Interviewed For This Document

Aster Data Systems

CVS Caremark

Dansk Supermarket

Disney

Greenplum

IBM

Infobright

Kognitio

LGR Telecommunications

Loyalty Management Group

Merkle

Microsoft

MySpace, a division of News Corp's Fox Interactive Media

Nationwide Mutual Insurance Company

Netezza

NYSE Euronext

Oracle

ParAccel

Premier Bankcard

SAP

Sybase

Teradata

Turkcell

ENDNOTES

[1] The information revolution is producing mountains of digital data that are becoming more and more challenging to process and analyze. Not only are businesses generating more data every day, but our approach to data analysis (structured databases, indexes, and distributed data architectures) generates even more data. Furthermore, the volume of unstructured content is also increasing, adding to the proliferation. To help organizations get business value out of all this data, vendors — both platform vendors and business intelligence (BI) specialists — are offering multiple technology and architecture solutions for very large database (VLDB) BI. Understanding that BI for multiterabyte data sets may require different architectures and technologies than smaller data sets is key for successful VLDB BI implementations. Information and knowledge management professionals should implement the most appropriate VLDB BI option and plan for continued explosive data growth. See the July 23, 2007, "Data, Data Everywhere!" report.

[2] Financial bailouts, downward consumer spending, and roller-coaster stock markets are starting to put pressure on information and knowledge management professionals. I&KM pros must do three things during this period: squeeze more value from existing investments like enterprise data warehouses (EDWs), control their current costs, and seek out incremental project opportunities, such as real-time business intelligence (BI), that deliver fast, visible results. See the October 29, 2008, "Topic Overview: Must-Read I&KM Research For An Economic Downturn" report.

[3] Scalability, performance, and optimization are the paramount criteria in today's EDW market. In Forrester's 54-criteria evaluation of enterprise data warehousing (EDW) platform vendors, we found that Teradata,

Oracle, IBM, and Microsoft lead the pack because each offers mature, high-performance, flexible, secure, and robust solutions. Teradata is tops in scalability, with its petabyte-scale massively parallel platform, though Oracle has shown great progress recently in EDW appliance scale-out through an intelligent storage layer. IBM has the broadest range of EDW appliance packaging options, while Microsoft has one of the most impressive midmarket-focused EDW solution portfolios. Netezza, Sybase, and SAP are Strong Performers in the EDW platforms market, with a primary focus on niche markets. Among them, Netezza demonstrates the most rapid evolution into a robust EDW platform, though the vendor's data warehousing (DW) appliance primarily addresses tactical deployment for front-end online analytical processing (OLAP) acceleration. Sybase recently launched a family of cost-effective EDW appliances, with its core columnar database suited primarily for very large data marts. SAP's NetWeaver BI platform offers a solid EDW hub capability for existing SAP users, but it aims to evolve into a more flexible platform for real-time business intelligence (BI). See the February 6, 2009, "The Forrester Wave™: Enterprise Data Warehousing Platforms, Q1 2009" report.

[4] Interactive, self-service BI is a growing enterprise practice for decision support. For years, traditional BI technologies have provided tools for reporting, analysis, and visualization of information. While these technologies remain the core staples of enterprise-grade BI solutions, Forrester recognizes an emergence of newer technologies and approaches to analyzing data. One such approach is the "BI workspace," where power users, especially power analysts, can explore data without their IT departments imposing any limitations or constraints, such as fixed data models, security, and production environment schedules. Information and knowledge management (I&KM) professionals should consider adding workspace capabilities to the list of BI functions that are necessary for leading-edge BI environments. See the June 23, 2008, "BI Workspaces: BI Without Borders" report.

[5] I&KM professionals need help evaluating their options for logical data architectures to support business intelligence, such as physical BI repositories versus federated data access to operational data stores and enterprise data warehouses. Forrester's Business Intelligence Data Architecture Decision Tool provides concrete guidance to evaluate the pros and cons of various approaches. See the January 12, 2009, "Forrester's Business Intelligence Data Architecture Decision Tool."

[6] Single system image is a core feature of both massively parallel processing (MPP) and server clustering environments. Single system image refers to the fact that a distributed system appears as a single logical resource to users, applications, and administrators. A single system image provides a virtualization layer that abstracts the underlying components of the distributed infrastructure, which may include multiple nodes with different underlying data models, development environments, operating systems, and hardware platforms.

[7] In mid-2008, EDW vendors Greenplum and Aster Data announced that they had implemented the Google-developed parallel computation API called MapReduce in their respective products. For its part, Google has been using MapReduce in its massive search environment to efficiently query petabytes of data — unstructured, semistructured, and structured — through MPP-optimized SQL extensions. One of the key innovations with MapReduce is that it provides a framework for parallelizing any in-database analytical algorithm — not just SQL queries.

[8] Appliances are taking up permanent residence in the heart of the enterprise data center, the data warehouse. DW appliances — in all their bewildering proliferation — are moving into the mainstream. The reason? They are preconfigured, modular devices that support quick deployment for DW killer applications — most notably, accelerating OLAP queries against large, multidimensional data sets. DW appliances prepackage and pre-optimize the processing, storage, and software components for fast OLAP and fast data loading. See the April 4, 2008, "Appliance Power: Crunching Data Warehousing Workloads Faster And Cheaper Than Ever" report.

[9] Online analytical processing is a mainstay of traditional EDW and BI environments. OLAP was invented almost 20 years ago to address a burning business need — fast analytics against large tabular data sets — for which traditional relational database management systems are not optimized. OLAP cubes, overlaying a logical dimensional structure over a denormalized data-storage layer — provide fast slicing, dicing, and pivoting of complex data along multiple dimensions. To support speedy analysis against aggregated tabular data sets, OLAP cubes involve prejoining records into "fact" and "dimension" tables, often with key derived values precalculated. See the November 7, 2008, "OLAP: In Fashion Or Old-Fashioned?" report.

[10] The emerging next-generation EDW will support extreme flexibility, enabling data to be transparently persisted in diverse physical and logical formats, each optimized for a particular deployment role, in support of mixed-query workloads. One such EDW data-persistence approach, "value-based storage," enables the EDW's analytic database — without manual redesign, tuning, or optimization — to continually deliver fast response to complex, dynamic, ad hoc queries. See the August 8, 2008, "Vendor Snapshot: Illuminate Solutions Breaks Through Traditional BI Barriers" report.

[11] Most BI environments continue to rely on EDWs as an aggregation point for historical data loaded in batch from operational repositories. However, I&KM professionals are increasingly rethinking their EDW architectures to optimize their infrastructures for real-time applications. Often they add real-time support, leveraging short batch windows, trickle-feed loading, changed data capture, and other ETL-acceleration approaches. Alternatively, I&KM pros may rely on architectural approaches like operational data stores, event stream processing, data federation, and information fabric, approaches that either supplement the EDW or bypass it altogether. See the August 11, 2008, "Really Urgent Analytics: The Sweet Spot For Real-Time Data Warehousing" report.

FORRESTER®

Making Leaders Successful Every Day

Headquarters

Forrester Research, Inc.
400 Technology Square
Cambridge, MA 02139 USA
Tel: +1 617.613.6000
Fax: +1 617.613.5000
Email: forrester@forrester.com
Nasdaq symbol: FORR
www.forrester.com

Research and Sales Offices

Australia
Brazil
Canada
Denmark
France
Germany
Hong Kong
India
Israel
Japan
Korea
The Netherlands
Switzerland
United Kingdom
United States

For a complete list of worldwide locations, visit www.forrester.com/about.

For information on hard-copy or electronic reprints, please contact Client Support at +1 866.367.7378, +1 617.613.5730, or clientsupport@forrester.com. We offer quantity discounts and special pricing for academic and nonprofit institutions.

FORRESTER®