TDWI RESEARCH

DECEMBER 2011

TDWI MONOGRAPH SERIES

Seven Keys to High-Performance Data Management for Advanced Analytics

By David Stodder

SPONSORED BY

sas

tdwi.org

tdwi

Table of Contents

About the Author

David Stodder is director of TDWI Research for business intelligence. He focuses on providing research-based insight and best practices for organizations implementing BI, analytics, performance management, data discovery, data visualization, and related technologies and methods. He is the author of TDWI Best Practices Reports and Checklist Reports on mobile BI, data discovery, and information management and has chaired recent TDWI conferences focused on BI agility and big data analytics. Stodder has provided thought leadership on BI, analytics, information management, and IT management for over two decades. Previously, he headed up his own independent firm and served as vice president and research director with Ventana Research. He was the founding chief editor of *Intelligent Enterprise* and served as editorial director for nine years. You can reach him at dstodder@tdwi.org.

About Our Sponsor

SAS is the leader in business analytics software and services, and the largest independent vendor in the business intelligence market. SAS big data and big analytic technologies are embedded in a framework that supports the entire decision-making process. This integrated framework combines the strengths of SAS solutions and technologies with SAS' commitment to innovation, continuous technical support, professional services, training, and partnerships. SAS helps customers at more than 50,000 sites improve performance and deliver value by making better decisions faster. Since 1976 SAS has been giving customers around the world THE POWER TO KNOW®.

Executive Summary

In the modern economy, innovation and excellence are strongly tied to the ability of organizations to draw uncommon insights from data. Advanced analytics—which include data mining, predictive analytics, machine learning, and more—are central to enabling organizations to translate "data smarts" into better customer service, more efficient processes, and products and services that are responsive to trends in demand. Advanced analytics are complementary to standard business intelligence (BI) because the insights help users answer the "why" questions behind the numbers they see in BI reports. With the combination, users can make better decisions and optimize future actions toward desired outcomes.

To get the most out of advanced analytics, organizations need to leverage high-performance computing in their data management practices and technology deployments. High-performance computing can enable organizations to overcome performance bottlenecks and other challenges that inhibit the power and growth of advanced analytics and their consumption by BI users. High-performance data management options that are important to advanced analytics are in-database processing, in-memory processing, grid computing, and stream processing.

High-performing analytics depend on choosing the most appropriate technology option to fit the workload. This selection is especially critical as organizations seek to deploy analytics against "big data" sources that are high in volume and variety. A key goal is to harness the power of database engines to speed both analysis and consumption of analytics. Scoring predictive models, which involves applying statements or codes to records in the database pertaining to the subject area of the model, is one process in particular that can benefit from in-database and in-memory options. For real-time computing, many organizations want to score models dynamically as records arrive; complex event processing (CEP) and stream processing are options to consider for this level of dynamic scoring.

This monograph focuses on how organizations can align data management practices and technologies with requirements for advanced analytics. Focusing on seven key steps, it explores how organizations can use high-performance computing options to support advanced analytics through in-database processing, in-memory analytics, event processing, and alternative approaches to data transformation and integration.

Meeting Rising Demand for Advanced Analytics

To prosper in competitive markets where margins are tight, customer loyalty is fragile, and "the way we've always done it" is no longer the safe route, organizations need to be smarter and faster. They must take full advantage of data flowing—sometimes in torrents—into their organizations from both traditional applications and new sources such as online customer behavior, clickstreams, event streams, and social media. To anticipate the future, they must apply predictive models to these large and fast-moving data flows to discover patterns and associations that will help them identify the best customers, tailor marketing programs to the appropriate segments, manage risk, reduce exposure to fraud, and allocate resources based on accurate forecasts of future conditions.

Advanced analytics thrive when coupled with high-performance computing technologies that are rapidly becoming the mainstream. In-database processing, in-memory analytics, and more take advantage of dramatic advances in hardware, middleware, and distributed networks for commodity scalable computing. Among these advances are multi-core processing, massively parallel processing (MPP), and very large memory (VLM). Organizations can use high-performance computing to improve the scale and speed of advanced analytics, including upgrading the ability to deliver analytics through current BI and data warehousing systems. Organizations can also apply predictive analytics models in applications that have automated processes that tolerate little or no latency between data capture, analysis, and resulting actions.

However, to support advanced analytics, organizations need to think differently about data management. First, advanced analytics often demand larger, detailed data volumes than the smaller pools of aggregated data frequently used for BI and online analytical processing (OLAP). Organizations need to consider how to make this data available, particularly when analytic models need to be deployed against real-time or at least very timely data. Second, traditional approaches to data integration; movement; and extract, transform, and load (ETL) processes may not be optimal for advanced analytics, requiring organizations to evaluate different approaches. Finally, to support analytic models running against streams of live transactions and real-time events, organizations need to consider CEP technology to complement analytic applications and databases.

Advanced Analytics and BI: Complementary Technologies

What are "advanced analytics" and how do they relate to BI? There are a number of practices and technologies under the umbrella of advanced analytics, including data mining, predictive analytics, natural language processing, and artificial intelligence disciplines such as machine learning, decision trees, and neural networks. What unifies them are first that they involve statistical, quantitative, or mathematical analysis of data; secondly, most center on developing, testing, training, scoring, and monitoring models. Advanced analytics often involve iterative explorations—using many variables —to discover why something is happening, what will happen next, and how to optimize actions so that desired results will occur.

Leading-edge organizations are using advanced analytics to understand customer behavior and discover which variables are most influential in determining customer loyalty, particularly for those customers who are deemed most profitable and valuable. Another important use case is credit scoring, which uses models to predict whether new applicants for credit will be able to repay loans on time or if they'll default. Credit scoring is obviously important to financial services

institutions, but it is becoming an important mode of analysis for many kinds of service providers to understand the risk of taking on certain customers.

Advanced analytics are also critical for fraud detection, quality control, and an expanding range of purposes. Organizations use advanced analytics tools to speed up complex inquiries so they can determine in real time where they should focus, who they should focus on, and what decisions and actions should be taken to achieve a desired outcome.

BI and OLAP typically specialize in querying, reporting, and analyzing historical data to understand and compare results to date or for specific time periods in the past. Organizations can use BI and OLAP calculations to project a view of what the numbers say is likely to occur in the future. However, advanced analytics can provide an even deeper understanding of why and a scientifically based, predictive view of the future. Advanced analytics provide users with the ability to explore many variables to refine insight. To provide this deeper level of understanding, advanced analytics often need to explore raw, detailed data rather than smaller samples and aggregations, which are customarily used for BI and OLAP.

Today, many users interact with BI systems through dashboard interfaces that integrate data access and visualizations such as charts and graphs with alerts, checklists, and other devices for tracking important changes. Dashboards have improved upon traditional BI reports, which provided only static and limited views of historical performance. Modern BI systems can refresh data in dashboards more frequently; users can track metrics to keep tabs on business performance and be alerted to spikes, dips, or other deviations from expected norms in something closer to real time.

What BI systems lack is both the deeper, more exploratory perspective that advanced analytics can provide, and the insights driven by predictive and other analytic models. By interacting with dashboard portals, BI users can consume advanced analytics through visualizations, and use data discovery capabilities to gain a “why” understanding of what the BI performance metrics are showing. Organizations can go further and make advanced analytics operations themselves the drivers, and implement BI dashboards and metrics to provide views into the results of the analytic operations. Examples include analytics that provide insight into customer satisfaction, success in fraud prevention, and so on.

Advanced analytics should not live in a world separate from BI systems. Where BI systems implementations provide the most value through dashboards, reports, and tracking of performance metrics, advanced analytics can offer highly complementary technology. BI dashboards can consume advanced analytics to give users a deeper and wider perspective, enabling users to explore why trends or events are occurring and to gain a predictive understanding of what will or could happen in the future. With credit scoring, for example, a BI dashboard report could show credit scores by customer and other data, but the heart of the application would be models that allow discovery of why the scores are what they are, through a variety of views such as average score by ZIP code. In this way, analytic scoring routines can complement BI reporting.

Models for advanced analytics typically undergo three major phases: training, deployment, and monitoring. This monograph focuses primarily on the deployment phase, where analytic models are running against data or event sources and need the power that high-performance computing options can provide. High performance in the model deployment phase is critical to the consumption of advanced analytics by users of BI systems. The ability to successfully score the models—which involves transforming predictive models into SQL statements or program codes and then applying them to all database records—is critical to model deployment. Organizations

can improve the speed and breadth of model scoring using high-performance computing. In the following sections, we will highlight seven key steps that are important for data managers to consider as they set their technology strategy to support advanced analytics.

Step One: Exploit in-database processing to speed analysis.

Traditionally, analytics have been supported through the creation of separate file systems that obtain their data from databases via replication procedures. This approach may still be appropriate for multidimensional OLAP on a smaller scale, or for analysis that is less time sensitive and does not involve movement of detailed data that might expose sensitive information as it travels across networks.

However, as organizations increase their reliance on analytics to support real-time decisions in frontline operations, they need an alternative to massive increases in data movement and replication to serve analytic model deployment. Organizations need to evaluate how they can exploit SQL database engines and data warehousing appliances—particularly those running on MPP platforms—to provide the level of performance and management oversight they need. (See Figure 1.)

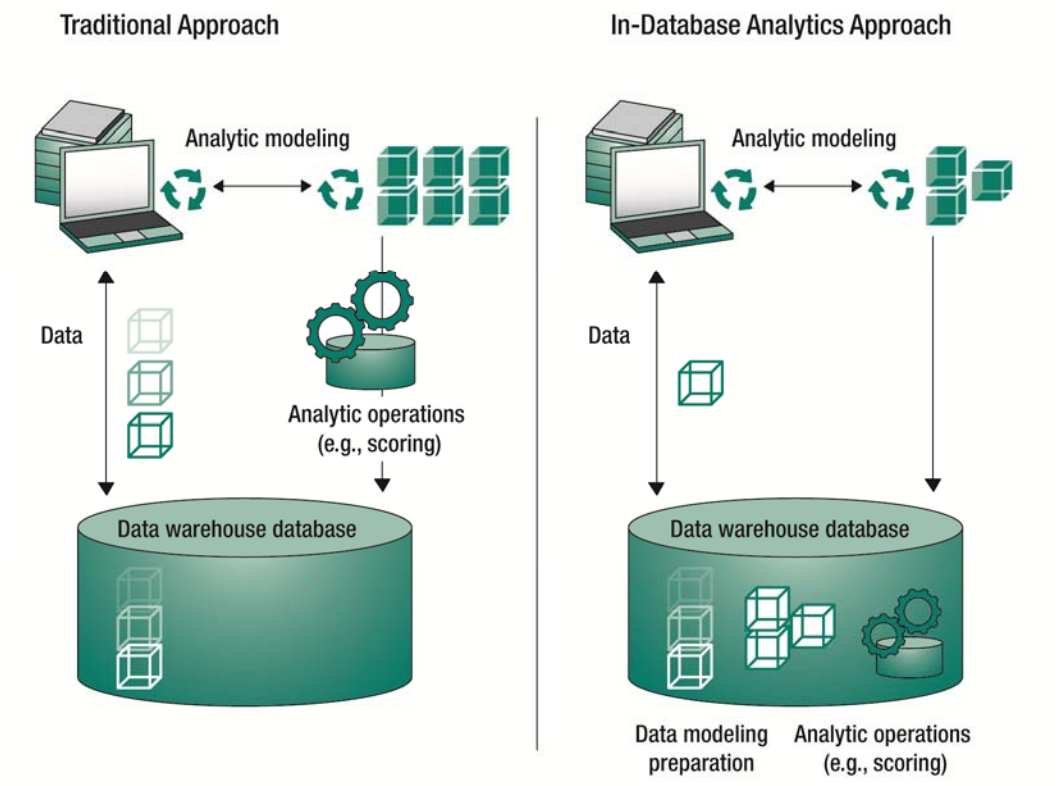

Figure 1. *Traditional architecture compared with in-database analytics approach.*

Many leading relational database systems support in-database processing for analytics, which enable organizations to integrate analytical processing with SQL functions to solve performance and management problems. Models generated with data mining tools can be translated into database-specific functions for scoring and other procedures, including data preparation, exploration, and further modeling steps, which are then run using MPP database engines.

Two concerns often arise with in-database processing for analytics:

- It will require frequent, lengthy, and difficult new rounds of coding, testing, and validation
- The challenges of conversion will restrict flexibility in the types of models that can be deployed

In-database processing need not involve manually converting analytic model scoring logic into SQL (or C). Instead, organizations can use tools to automate model scoring and conversion. This automation allows organizations to benefit from the speed and scale of database systems without incurring new overhead. It also makes it easier to implement a greater variety of model types to take advantage of database processing. As the number of models grows and analytics operations demand more frequent scoring across larger data volumes, automation becomes critical.

Performing analytical calculations inside the database allows organizations to skip slow and costly data replication and loading procedures. With less data duplication, organizations can govern access to sensitive information more effectively; they can also save costs by reducing the personnel and computing systems required to support duplicate data sets on data marts or files systems outside the database.

Increasing the scalability of analytics. Two key steps in data preparation for analytics—sorting and summarization—can benefit from in-database processing. Using SQL syntax, the database system can perform the sorting that is needed and deliver tables and views to users. Letting the database engine perform as much summarization as possible inside the database can enable analytics to take advantage of parallel processing for better scalability as data volumes rise. Organizations can use in-database processing to significantly reduce the time that would normally be spent bringing entire summarized data sets from external stores over the network to specialized data files dedicated to analytics.

Running analytical processes inside massively parallel SQL database platforms can increase performance and scalability as long as organizations choose the methods that are best suited to the type of analytics they are performing. Most model building and scoring processes, for example, still need flattened, detailed data sets, not the normalized or star schema forms typically used in data warehouse designs. Tools can help organizations assign reusable SQL or C routines that best fit the performance requirements of each analytic model.

With the in-database approach, analysts can model and test on these special data sets using memory-based “sandboxes” or virtual repositories that are intended for short-term analysis needs. The flattened, detailed data sets they need can live as sandboxes on the same database platform as the data warehouse and use the platform’s storage and manipulation functions. Along with avoiding the costs associated with separate systems, the integration of analytic sandboxes and data warehouses can make it easier for BI systems to consume the analytics along with the data the systems access normally from data warehouses.

Perhaps most important for performance, in-database analytics enables users to access *all* the data, rather than subsets or aggregations, and be able to run analytics on platforms that can support multiple passes through large data sets. This is critical for the iterative, questions-leading-to-more-questions processes that are common with advanced data analysis. In addition, to eliminate redundant effort, organizations can publish models to the data warehouse, where analysts can call on them as shareable and reusable functions for other analytic processes and BI systems.

Step Two: Increase the flexibility and power of analytics with ELT.

One of the most important uses of in-database processing is for extract, load, and transform (ELT) processes. In recent years, ELT—an alternative to the more familiar ETL—has attracted growing interest and implementation. ETL processes are commonly used in data warehousing to gather data from multiple tables or data sources and prepare it for users of BI and OLAP tools. However, they can become numerous, redundant, costly, and complicated over time, which has prompted organizations to consider alternatives to satisfy BI needs. Now, with advanced analytics growing, organizations have additional reasons to go with ELT.

In particular, the ELT process pushes the transformation steps into the database engine, where they can be performed though SQL statements and procedures. Data is extracted and loaded in raw format and then transformed and moved into tables for access by users. Whereas ETL usually relies on a separate layer of integration systems, ELT exploits the power of database engines. If these engines are running on multi-core and parallel processing hardware platforms, ELT can scale to high levels of performance. Most steps associated with ETL for data integration and preparation, including sorting, merging, summarization, and profiling, can benefit from the scale and power of having a database engine run them as ELT processes. (See Figure 2 on the next page.)

ETL processes consume a lot of computing time and power, which requires organizations to run them during off-peak hours. Thus, it can make sense to use ELT to shift some of the transformation workload away from ETL servers and middleware to the data warehouse's database platforms, where there is likely spare capacity during off-peak hours. ELT tends to be demand driven, responding to users' or analytic operations' requests for transformed data, rather than running according to a prearranged, and periodic schedule. Thus, ETL and ELT can be complementary, giving organizations tools to handle real-time, on-demand requests along with the regularly scheduled reporting transformations.

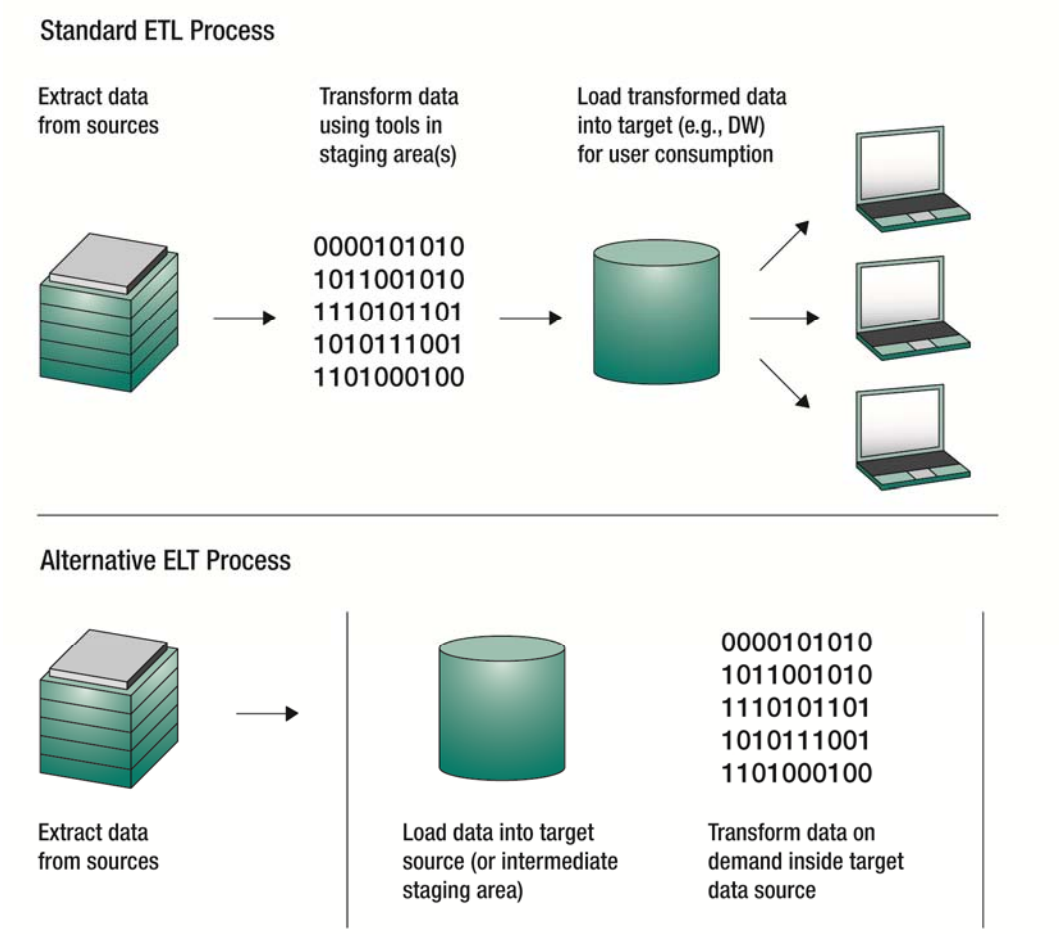

Figure 2. *Standard ETL compared with the ELT alternative.*

Enabling multipass, iterative discovery. One of the key reasons to use ELT for analytics is the number of passes through the data that are typically needed for advanced analytic models. To satisfy preparation requirements for each variable, it is often necessary to make an additional pass through the data for each variable. Data that is fed into models for testing and scoring could have a structure that is different from what organizations are used to for standard ETL processes. ELT can exploit the power of parallel database engines to run each pass through the data in parallel. With defined ELT processes, organizations can develop repeatable and potentially automated routines instead of relying entirely on one-off, manual coding for each new data structure.

ELT is not suitable for every type of project, however. ETL may still be a better choice when requirements call for integrating and transforming data from a large number of high-volume source systems. However, ELT can improve the speed of transformations for analytics that demand considerable joining of and access to historical data, because of the use of powerful

parallel database engines to run the routines and the efficiency gained from not having to replicate and load the data elsewhere. When the objective is to quickly extract and load raw data and do very little, if any, transformation, ELT again can be the better choice because of speed gained by reducing data movement.

The ELT approach aligns with trends toward moving functions closer to the data, instead of moving data outside of the database to perform transformations and other processes. The reasoning is that in general, the more you can avoid moving the data, the better you can control and improve analytic operations being performed on the data. Reduction in data movement can also improve network performance and allow organizations to execute data governance and regulatory policies in one place.

Whether organizations are using ETL, ELT, or a hybrid of the two, it is vital to the efficiency of high-performance analytics to enable flexibility and reusability. Organizations need to pair their algorithms with the right extract, load, and transform processes—no matter what the order is—and potentially on the fly to meet urgent, real-time requirements. In addition, to reduce time and cost, organizations need to reuse transformations for new analytical processes and applications. Unfortunately, most ETL systems and processes lack the software automation and tooling needed to provide greater flexibility and reusability. Organizations should evaluate automated software alternatives that can help them manage ETL and ELT processes and make it easier for developers to reuse them for new applications.

Step Three: Implement data federation to reduce data movement and broaden access.

ELT is one method of alleviating cost, complexity, and performance problems associated with data movement in support of analytic model deployment. Another method is data federation. Data federation is particularly useful when access to real-time operational data is required. It also allows users to gain comprehensive views of data that is physically located in diverse sources across a business network. The data is joined together virtually by data federation middleware. Federation leaves data in place rather than moving it to a separate store. Although the data stores in federated systems are all relational in many cases today, the ultimate goal is to provide virtual views of heterogeneous types of data, which could include nonrelational data such as legacy file systems, spreadsheets, XML streams, content stores, search results, and more. (See Figure 3 on the next page.)

Data federation has proved a useful option for BI systems and can similarly give analytic applications a quicker path to operational, real-time data. Federation is an alternative to batch ETL operations that provide only periodic movement and migration of data to an enterprise data warehouse (EDW). Although initially regarded as a rival to EDW, in practice federation complements EDWs, ELT, and classic ETL processes by providing a “loosely coupled” approach to data integration in circumstances where it would be too costly, slow, or constraining to try to centralize the data—especially for near-real-time access. As data is being read from disparate sources, federated systems can perform join operations that deliver comprehensive data results without users having to know the metadata or access schema details of each source. Analytic applications that need a steady, frequently updated flow of data from across business functions, regions, and from outside organizations are good matches for data federation.

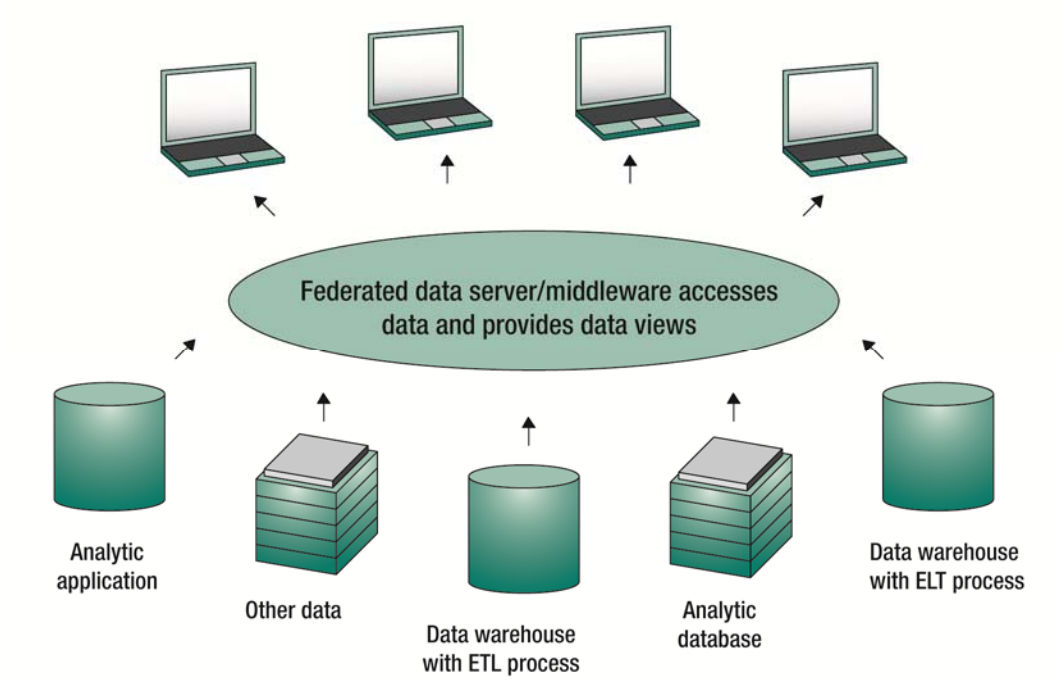

Figure 3. *Data federation enables access to data without movement.*

Federated systems must be able to optimize query plans to deliver high performance; they must ensure that only the data that is needed to satisfy a request is returned across the network. One approach to optimization is to push segments of federated query processing to the database engine, rather than depending entirely on external middleware. Some federated systems are also able to cache the results of frequent queries in memory; in this way, data is more easily available for analytics and can be updated at a set level of frequency. When organizations evaluate data federation solutions as part of their data management strategy to support analytics, they should examine these and other features for enabling high performance.

Step Four: Manage in-memory processing to achieve high performance.

Perhaps more than any other technology development, the availability of much larger random access memory (RAM) is opening up new possibilities for high-performance analytics. In recent years, the cost of computer memory has continued to fall while the amount of addressable memory has increased. Growing adoption of 64-bit operating systems by software developers has made it easier for users of BI, analytics, data marts, and warehousing systems to exploit very large memory (VLM)—up to a terabyte, with more promised in the future. In addition, advances in compression techniques are enabling organizations to pack more data into memory and make effective use of specialized approaches such as columnar databases. Organizations can use VLM to bring analytic compute functions closer to the data.

Traditionally, administrators running BI and analytic queries against a data warehouse have relied on the database management system (DBMS) to read information from tables stored on disk. To get around the disk I/O performance bottleneck, organizations have used caches as a place to temporarily hold tables or small subsets of data to service queries. However, these caches are generally temporary and limited in size; also, if they are located on disk, they do not fully solve the bottleneck problem. Using expanded main memory opens up a much larger territory for running analytic functions, such as model scoring locally against data rather than through access to remote disk-based sources. With data in memory, organizations can deploy analytic models against potentially real-time data that is updated through continuous, incremental loading and transformation. (See Figure 4.)

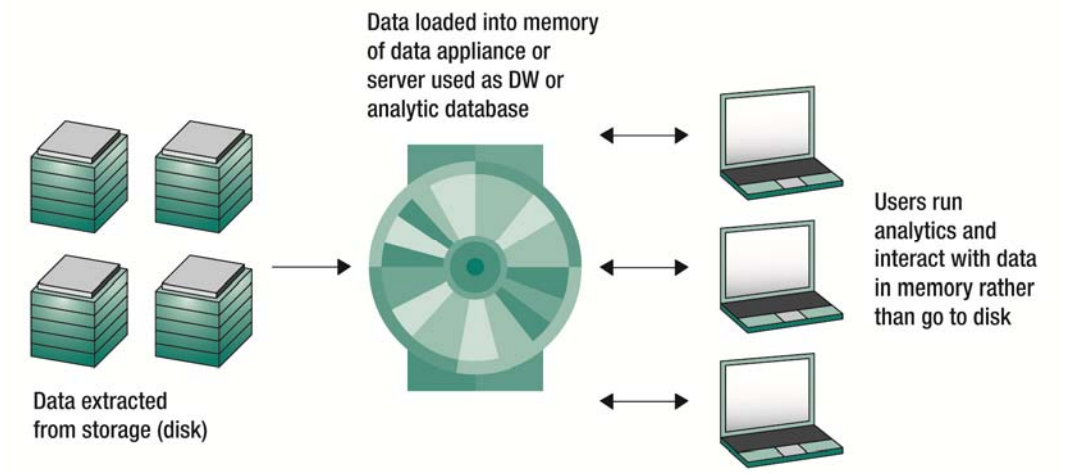

Figure 4. *Data provided in memory to support advanced analytics.*

Analytic applications can work with VLM to improve both the performance and scale of analysis. Real-time, "speed of thought" analysis becomes much more possible with in-memory systems, even with large numbers of variables. The preprocessing steps that usually are required for reducing the number of records moving through the I/O bottleneck—steps that have effectively slowed analysis and limited the scope of discovery—are not as necessary when all the data is loaded into memory.

Analytics using in-memory databases can be fully contained in main memory and take advantage of the low latency potential of accessing data in RAM. In-memory systems give IT administrators a break because they do not have to tune queries to go through the I/O bottleneck. However, growth in the use of large memory for analytics hardly means that data management can relax. The following are three priorities for data management: managing data models and data location; optimizing in-memory analytics by getting the most out of compression; and managing shared memory and pipelining for high performance.

Manage data models and data location. VLM may seem ample, but organizations should look for ways to optimize its use and not let the space become chaotic. Lack of order, poor understanding of a data or analytic model's size and complexity, and incomplete knowledge of

data location can make critical operations on the data—such as incremental updates for real-time analytics—slow and difficult. In particular, administrators should focus on the following:

- How the system manages where the data is located in rows, columns, and tables
- How it may use pointers and vectors to reference data stored in different memory locations
- How (and how often) the data is refreshed and loaded
- How updates are synchronized with data in storage

Better management and knowledge of these factors are critical to the integration of in-memory analytics with applications elsewhere in the organization. Organizations should avoid turning in-memory analytics into yet more silos that are difficult to integrate.

Optimize in-memory analytics by getting the most out of compression. Compression techniques can be critical to the success of in-memory analytics because they enable systems to keep more data in memory. If the underlying database system is column oriented, compression is even more essential.

Most columnar database systems use a variety of compression techniques to suit different requirements. Advanced columnar approaches can use similarities between adjacent data elements to guide compression, which can make it easier to later decompress and access the data. Administrators should examine how their systems are compressing data and evaluate whether techniques could be refined to improve the ratio of data in memory to the total size of the database held in disk storage. Administrators should also review how data is updated, and how the use of compression to keep more data in memory might impact standard buffer-caching procedures for discarding least-used data to make room for new data for analysis.

Manage shared memory and pipelining for high performance. In-memory analytics performance depends on shared memory space management and pipelining. Let's look first at managing the memory space for analytics. Some analytic applications and databases are flexible in how they work with memory, enabling users to implement data models, analytic models, and data volumes that are actually larger than what will fit in a single addressable memory space. These systems can work with the operating system and the computer platform's underlying virtual memory management to determine how to assign processes to address spaces in virtual memory, with the goal of providing the best in-memory analytics performance for large models and data volumes.

However, administrators need to examine how to optimize analytic applications to avoid the potential downsides of virtual memory "swapping," which happens when the system tries to allocate more physical RAM space than is available. One easy solution, of course, is to simply add technology to increase the memory available to the analytic application. This addition will decrease processing time and allow organizations to avoid the penalties of memory swapping.

Along with space management, administrators should evaluate how the in-memory system uses shared memory and pipes (that is, "pipelining") to support high-performance analytics. Both are important to how analytic application processes read and exchange data and execute operations. When properly managed, these shared memory methods help deliver the benefits of in-memory analytics by ensuring that data is easily accessible to multiple processes running in the system, more so than with traditional means of reading and writing data between client and server processes.

Pipelining—also known as pipeline processing—concerns the rapid and carefully sequenced (if not overlapping) execution of multiple instructions contained in analytic queries. Pipelining allows systems to start operations for analysis as data arrives in memory, without having to wait for other operations in the sequence to finish completely. The downside of pipelining is that there is still data movement; data must move from one segment to the next, which can create complexity and make it difficult to track the data's location at any given moment. Administrators should therefore evaluate additional or alternative procedures such as data vector processing on multiple data sets, which can reduce data movement, increase awareness of data location, and allow for operations on arrays, not just single data elements. Pipelining will be discussed in more detail later in this monograph.

Step Five: Achieve dynamic scalability by integrating grid computing with in-memory and in-database technology.

Organizations can fit in-database and in-memory processing options together with grid computing to create a complete and more dynamic high-performance analytics system. This system can effectively address the range of performance challenges that are inherent in deploying analytic models for scoring and in enabling the consumption of advanced analytics by users of BI systems and other applications.

Grid computing allows organizations to break analytic operations into subparts and distribute the work across hardware resources in a network. In most grid computing architectures, the system will process operations on the data in parallel on a network or cluster of servers, and then pull the results together for the user's view. Grid computing resources are generally loosely coupled, so you can expand the system easily by adding nodes and handling increases in workloads dynamically. An organization might, for example, want to gain a near real-time view of a large volume of customer data for marketing analysis. Rather than wait out a long provisioning and deployment cycle for single large server, the organization could use a grid computing architecture of commodity blade servers as nodes to expand computing power incrementally to support the analytic operations.

The dynamic scale-out approach of grid computing is a good fit for organizations needing to support advanced analytics to meet immediate, time-sensitive business objectives. These objectives may come up suddenly and be short in duration, but it doesn't mean that they don't need to touch large volumes of data. Analytic operations such as credit scoring require multiple terabytes of source data to be processed, scored, prepared, and summarized—all potentially within the tight time frame of a real-time customer interaction. Using grid computing, organizations are able to address such requirements as they unfold, rather than being forced to purchase the maximum capacity an organization can afford on single servers in anticipation of peak workloads.

In sum, grid computing makes high-performance computing for advanced analytics complete by giving organizations scale-out capabilities to go with the "scale-up" capabilities enabled by in-database and in-memory processing. Organizations can then determine the best technology fit for different kinds of analytic operations and choose how to scale for faster results as analytic functions tap larger data volumes and different kinds of information.

As an example of how scale-out and scale-up can come together, consider the choice of deploying a scoring model in memory. If underlying grid computing is available, the organization can scale

out economically by creating a logically shared address space that is physically implemented on the grid in a networked cluster of VLM, multi-core blade servers. Distributing work across such a grid can increase the power and speed of analysis without incurring a linear increase in costs. Organizations can use the dynamic growth potential of grid computing to support an increase in the frequency of scoring to daily, hourly, or even near-real-time, on-demand intervals.

Understanding the role of message-passing interfaces (MPIs). For advanced analytics, an MPI is one of the most important components of a parallel grid computing architecture. A standards-based MPI communications protocol enables developers and analysts to program for parallel computing platforms, including those configured in a grid computing architecture.

Figure 5 shows (in a simplified fashion) how grid computing completes the high-performance system and where MPI fits into the architecture. An organization's grid could include "CPU-heavy," VLM nodes for operations to be undertaken in memory; another grid computing architecture could have nodes optimized for in-database processing to manage real-time data for analytics that need that level of currency. A grid control server would provide management functions and communicate with central file servers and tools, as well as the metadata server and tools. Implementation of the MPI protocol would allow developers to deploy advanced analytics applications that can take advantage of the parallel computing platforms in the grid architecture to share data and work required by analytic operations.

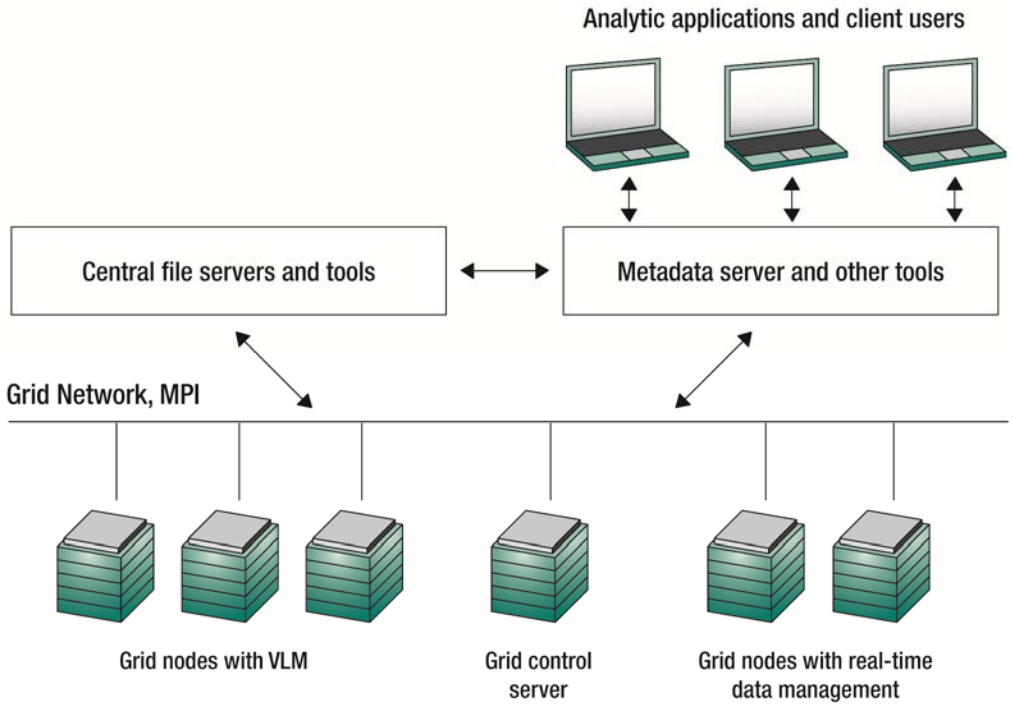

Figure 5. *View of grid computing with role of MPI protocol shown.*

While MPI is critical for communication among loosely coupled processors in a parallel computing environment, the protocol can also be used in conjunction with memory pipes to efficiently move data in shared memory environments. These could include situations where some of the work would be distributed to VLM systems in a grid computing architecture. MPI can therefore be used to support the running of analytic operations in memory on data that could not be easily broken apart to work in a disk-heavy MPP system. This data could include large images, spatial data, and data for modeling logistic regression. MPI implementation gives organizations the flexibility to perform analytic operations on data using the most appropriate technology.

Step Six: Employ workload management to align technology with analytic requirements.

As daily business decisions grow more dependent on the consumption of analytics, the data management infrastructure that supports analytic models and applications must be highly available and reliable. Operational executives and users—much less customers—will not be as tolerant of performance slowdowns and outages as "data scientists" who may be accustomed to grabbing off-hour machine cycles to test models and algorithms wherever and whenever they can find them. In addition to ensuring fault tolerance and availability, IT needs to be able to balance workloads, direct jobs to the appropriate resources, and maintain components in the grid without having to bring the entire system down.

Grid management tools are important for meeting these requirements and for responding to defined service-level agreements so that analysts and IT can set and meet business user expectations. Tools can help administrators balance workloads and identify priority jobs so that they can be moved to the front of a queue and given the appropriate computing resources. Tools can help administrators to align technology capabilities with the requirements of specific analytic workloads. If these workloads can be parsed and broken into smaller tasks to run in parallel, the management tools can help administrators oversee their execution in the grid architecture.

Understanding and managing parallel processing is vital for administrators to get good workload performance from grid computing as well as the entire high-performance computing system of multi-core processors and MPP systems. The basic goal of parallelism is to use networks and middleware to harness multiple CPUs to work on analytics in parallel, so you get results faster. For high-performance analytics, two important aspects of parallel processing are pipelining and partitioning.

- **Pipeline parallelism** describes a multistep process, usually involving at least three (but potentially more) processors. Each processor will perform a step—in data warehousing, for example, the steps are typically transformation, matching, and loading—as the data moves through the "pipeline" from source systems to a target data warehouse. Analytic operations would be similarly divided into steps. Pipeline parallelism allows processors to begin executing on new step instructions even though all the steps in a pipeline may not yet be complete. Data that is waiting in between steps for the next one to complete can be put in memory buffers rather than written to disk, which would slow down performance and require storage resources.

- **Partition parallelism** focuses on data rather than process steps. Partitioning splits data records into subsets and gives the subsets to multiple servers; the servers then work

simultaneously in parallel to perform each operation. Partitions could be subsets based on data from a particular region, customer segment, and so on. Advanced analytics can take advantage of a range of partitioning methods, with the choice depending on which is the best way to split the data and execute the particular type of analytic workload on each processor. The methods include range, hash, round robin, and random partitioning. Some analytical applications will additionally speed up processing by accessing or reading blocks of data from each partition, rather than waiting for all the results to come back together.

Figure 6 shows how data extracted from source systems could be divided into partitions and/or run through a pipeline process for analytic operations or transformation steps, ending with the data available in a target data warehouse for BI and analytics.

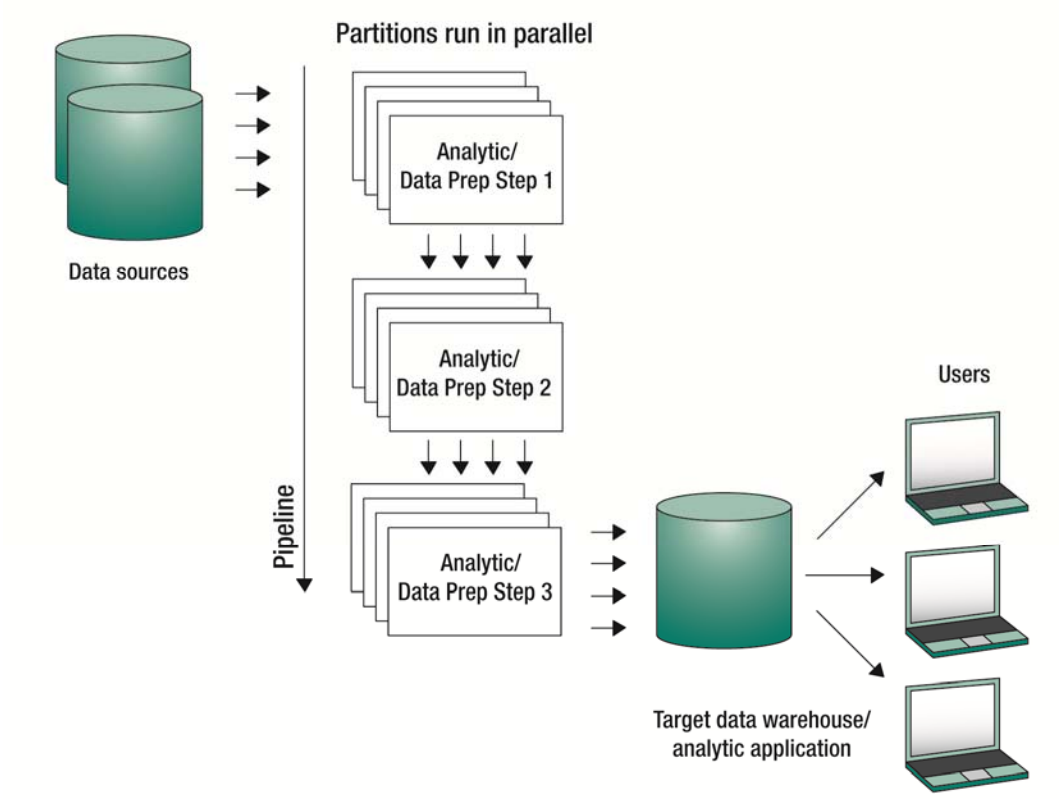

Figure 6. *Pipeline and partition parallel processes to distribute work for faster results.*

Organizations need to give themselves flexibility to match pipelining and partitioning options with analytical needs and workloads. At the same time, however, it is better if the choice of method occurs at a separate layer from the underlying job so that an ETL or ELT routine, for example, does not have to be completely recoded when a different method is chosen.

For most parallel processing environments, pipeline and partition parallelism are both necessary and can be used in combination. Partition parallelism is often the best choice when there are large data volumes and it would be faster to divide the work in each step of a process among multiple

servers. Workload management and other tools can help administrators decide which approaches and methods are best for particular analytic operations; software tools could be used to automate some of these decisions. However, the key point is to have systems that are capable of implementing multiple approaches and methods so that organizations have flexibility to meet all analytic application needs.

Step Seven: Leverage high-performance computing for real-time analytics and complex event processing.

High-performance computing technologies come together to support a key objective in many firms: gaining the ability to sense, analyze, and respond to new information in real time, or something very close to it. Organizations want to move the point of decision to events in real time and thereby eliminate delays in decisions to extend or deny credit to customers, prevent fraud and abuse, trade securities or detect improper trades, correct flaws in manufacturing processes, and more. If it takes too long to churn through data and execute an informed decision, conditions will have changed and the opportunity to respond will have passed.

Advanced analytics can play a critical role in enabling organizations to reduce latency and act in real time. Analytics such as clustering algorithms allow organizations to rapidly make sense of large volumes of big data flowing into the organization in real time. An example is data contained in e-mails; an algorithm could separate clusters of those with high value from those with low or uncertain value that require human analysis. Predictive models can be deployed to score events or transactions in real time to filter out the noise and concentrate attention on those items that require decisive action. Manufacturing organizations can use neural networks or other analytic models to monitor automated processes and immediately raise awareness to either humans or automated decision management systems when there are mistakes, imbalances, or problems in the supply chain—before it's too late and the ripple effects escalate costs across other parts of the business network.

Taking advantage of complex event processing. As this monograph has discussed, in-database and in-memory analytics can use high-performance computing technology to eliminate delays in deploying analytic models so that BI and other applications can consume them more quickly. However, rather than wait for data about events and transactions to enter databases and go through preparation and transformation steps, organizations in many industries are moving one step further and are implementing complex event processing (CEP). CEP engines can detect events as well as event patterns and relationships, enabling organizations to apply business logic, rules, and analytic models to automate and manage decisions in response.

Combining analytics with CEP allows organizations to perform model scoring or other operations on data, literally as it is being created. Figure 7 on the next page shows how data and events captured by various source systems can be brought to the attention of CEP, real-time analytic engines, and model scoring operations via a message bus. Data could additionally be stored in operational data stores. Users could consume insights generated by CEP and analytics in downstream activities, such as providing users with alerts and notifications, creating real-time trend analysis dashboards and impact assessments, or executing automated actions and processes. Business rules managed by rules engines or applications could guide actions according to application objectives, metrics, governance policies, and more.

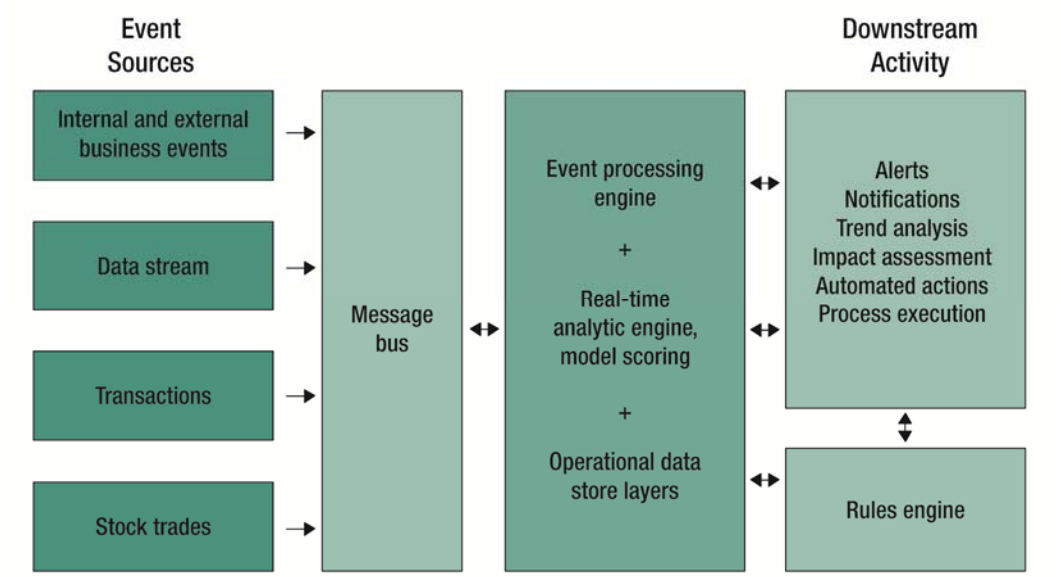

Figure 7. *CEP and analytic operations for tapping real-time data and events for analytics.*

Organizations can apply real-time analytics, such as scoring models, on these streams rather than wait for the data to arrive in database systems. Advanced analytics and CEP together enable organizations to automate decisions in complex situations where there are many variables and it would take too long for humans to sort through possible causes for event patterns and determine how to respond. Dealing with "complex" events that are actually combinations of many events flowing in a constant stream is a huge challenge, especially as the volume rises. Big data generated by sensors and online customer behavior are good examples of the kinds of event streams and data flows that offer opportunities to apply analytic models to guide automated decisions, so that organizations can react in real time, as events are happening.

Because decisions ultimately impact both human and automated processes, the real-time analytics need to be integrated with business rules, logic, and operational management practices. Tools are maturing to make it easier to embed analytics in operational processes and applications so that systems can fire rules in response to events in real time. Embedded, real-time analytics can increase the value of kiosks, such as ATMs, as well as human-centered activity in contact centers, for example, where organizations need to respond to related events across multiple channels. Fraud detection, which often involves continuous monitoring and the need for immediate response to events, is also a key implementation area for real-time analytics integrated with rules and logic.

Recommendations

High-performance computing technologies and methods differ from traditional approaches that focus on simply adding bigger servers and storage. Advanced analytics can benefit from alternatives that encourage smarter use of resources, greater efficiency in data management, and higher flexibility in how organizations can scale up to meet business priorities. In closing, here are six recommendations to help you set your strategy:

Apply advanced analytics to enable more intelligent decisions. Executives, managers, and operational employees need more than just BI reports to make the right decisions. They need insights that help them understand where they should focus attention, why events are occurring, and what the desired outcome should be. Advanced analytics can help them gain a deeper understanding and improve the success of their decisions.

Enable BI systems and other applications to consume analytics. Organizations lose many of the potential advantages of advanced analytics if they cannot discover insights in BI dashboards, performance management systems, and other applications that decision makers use every day. Deploy analytic models to enrich BI, data warehousing, and other systems so that users can consume them in their decision processes.

Reduce latency between data capture, analysis, and business execution. In many business scenarios—including customer interaction, fraud detection, and process optimization—reducing latency can result in huge advantages. High-performance computing technology can support real-time analytics, embedded analytics, and other options for speeding discovery and making insights actionable. Consider implementing CEP, so that predictive models and other analytics can operate on events as they are happening.

Evaluate in-database and in-memory analytics and grid computing to increase scale and performance. High-performance computing gives organizations new options that could offer better performance for model scoring and other advanced analytic processes than what is available from traditional data architectures. Because many analytic models need to be deployed against large volumes of detailed data, it makes sense to use the full power of database engines and grid architectures. Evaluate the potential of performing operations on data in memory to avoid the penalties of landing data to disk.

Decrease data movement to save steps in preparing data for analysis. Some of the most time-consuming phases of data analysis and the deployment of analytic models involve extracting, moving, and replicating the data to separate stores, where it is then prepared for BI and analysis. Organizations should evaluate not only in-database processing but also ELT and data federation to exploit the power of database engines and reduce the data movement that is involved in traditional approaches to integrating and transforming data.

Understand workloads so you can match technology options with performance requirements for analytics. As the consumption of analytics spreads throughout organizations, it is imperative that IT managers prioritize and balance workloads to ensure that resources are properly allocated. Growing business reliance on analytics also means that systems supporting analytics have to be reliable and available. Organizations should consider implementing tools that can help them manage analytic workloads, analyze how resources are being consumed, and gather usage metrics to support the acquisition of new technology where appropriate.