WHITE PAPER

WC
www.wintercorp.com

SPONSORED
RESEARCH
PROGRAM

WINTER CORPORATION

SCALING THE ENTERPRISE DATA WAREHOUSE

Teradata's Integrated Solution

SCALING THE ENTERPRISE DATA WAREHOUSE

Teradata's Integrated Solution

Rick Burns
Doug Drake
Richard Winter

November 2004

WINTER CORPORATION
411 WAVERLEY OAKS ROAD, SUITE 327
WALTHAM, MA 02452
617-695-1800

Executive Summary

The corporate appetite for information is mushrooming, and businesses are demanding integrated information solutions. As the data warehouse expands to encompass more of an enterprise's information assets, the requirements on the data warehouse become more demanding and the ability of database products to satisfy these requirements becomes more problematic. These forces are not new, but their effects continually push database products across new frontiers of performance and scalability.

The increasing integration of information in the enterprise data warehouse (EDW) drives its growth along several dimensions. The most obvious area of growth is in the size of the database. Today's largest databases contain more than 30 terabytes (TB) of user data; a volume that is expected to more than double in the next two years. The population of database users is also growing as database subject areas expand and as companies open their warehouses to customers and suppliers. Today, user populations in the thousands are common. Finally, competitive pressures on the business demand analytics of increasing sophistication, requiring query logic of increasing complexity. Expanding user population and query complexity drive up the database workload. These forces exert a multiplicative effect on database scalability requirements.

Along with this growth in data volume and workload, the role that the enterprise data warehouse plays within an organization has become more visible and more critical to business success. As a result, reliability and availability demands are also increasing rapidly. Data warehouses require extensive safeguards to keep them running virtually flawlessly and to protect the organization's investment in the data itself.

As data warehouses grow in size and reliability constraints, the challenge and costs of managing them must be contained. The increasingly competitive business environment forces lower unit support costs. As a result, data warehouse systems must be easier and cheaper to manage than ever.

In this WINTER CORP White Paper we examine how well Teradata meets these technical requirements by exploring and evaluating Teradata's architecture and the experiences of its leading customers.

While some database products address both transaction processing and complex decision support needs, and others target price-performance leadership for the value database market, Teradata has long focused on the most complex and demanding data warehouse problems. Teradata's massively parallel processing (MPP) architecture combines physical scalability of hardware with innovative, parallel software algorithms to overcome inherent barriers to scalability. Besides its elegant architecture, Teradata has been solving the toughest performance and scalability problems at the world's largest data warehouses for 20 years, an invaluable experience for working through the scalability barriers, large and small, obvious and subtle, that all database products face. Teradata has also overcome the intrinsic difficulties of MPP systems to deliver a world-class high availability solution out-of-the-box, and to provide an integrated management interface that masks much of the MPP system's complexity.

We conclude that, despite an increasingly competitive landscape, Teradata possesses distinct advantages for the highly complex, large-scale enterprise data warehouse. The key reasons for Teradata's distinctive advantages are:

- Teradata's low-cost, fast messaging infrastructure enables low-level interactions among components of distributed algorithms that support higher degrees of parallelization, contributing to higher scalability;
- Teradata's virtual processor architecture enables fine-grained parallelism that is resilient to parallel-efficiency sapping effects such as dynamic data skew;

- Teradata's hash-based file system promotes the balanced distribution of data so key to linear scalability;
- Twenty years of customer experience in overcoming barriers at the advancing frontier of scalability have "hardened" Teradata's parallelism and scalability to a level unmatched in the industry;
- Teradata's priority scheduler provides sophisticated workload management, allowing balanced execution of mixed workloads;
- Teradata's automated system management capabilities effectively turn managing a large MPP system with thousands of components into managing a single integrated system;
- Unlike any other DBMS marketed today, Teradata provides a built-in, fully automated availability infrastructure—at no extra cost, and with no assembly required.

Complex, large-scale data warehouses have unique requirements, and no product satisfies them all; companies should evaluate database products in light of their specific needs, preferably via quantitative measurement. Nonetheless, Teradata, with its fundamentally scalable architecture, tempered by 20 years of successful experience at the leading edge of large database practice, provides distinct advantages and merits serious consideration by organizations facing the world's toughest data warehouse challenges, now and in the future.

Table of Contents

1 Business Requirements of an Enterprise Data Warehouse

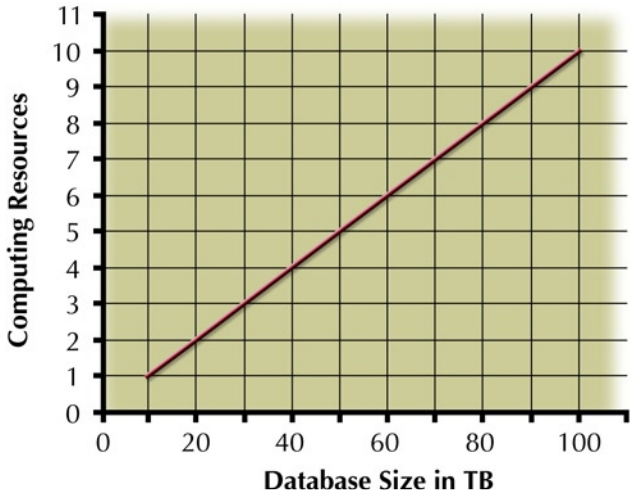
The corporate appetite for information is expanding like never before, and businesses are demanding integrated information solutions. They want to make fact-based decisions using the latest data. They want to move from debating what the facts are to analyzing what the facts mean. They also recognize that the best solutions to many business problems require insight from multiple disciplines and business areas. These forces are driving information technology (IT) organizations to integrate operational information gathered from across the corporation into a central information pool, the enterprise data warehouse (EDW). As the data warehouse expands to encompass more of an enterprise's information assets, the requirements on the data warehouse become more demanding and the ability of the hardware and software components of the warehouse to satisfy these requirements becomes more problematic.

These forces are not new, but their effects continually push database products across new frontiers of performance and scalability. As these data warehouses get larger, they also require increased levels of system reliability, availability, and manageability. Let's review the effects of these forces on the requirements for EDW systems.

1.1 SCALABILITY AND PERFORMANCE

The increased integration of information inherent in the EDW drives the growth of the data warehouse along several dimensions. The first and most obvious area of growth occurs in the size of the database. According to the 2003 Winter TopTen survey, the largest data warehouses are approaching 30 terabytes (TB) of user data and are expected to more than double in size over the next few years. In general, processing queries on 50 TB of user data will consume more computer resources than the same queries on 5 TB or 500 GB. Nevertheless, analysts still want their answers in roughly the same elapsed time, and database loads and extracts cannot exceed business requirements for timeliness of data. Such "size-up" requires the ability to complete a constant workload against larger data in constant time.

Figure 1: One-to-One Linear Size-Up



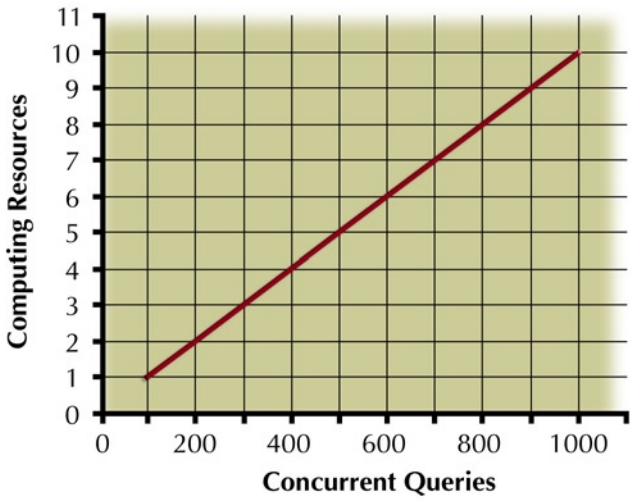
As data from more areas of the business are incorporated into the EDW, the population of database users grows. In many organizations, thousands of internal users throughout the company access the EDW. Increasingly companies are opening their databases to vendors, customers, suppliers, and other external business partners, further fueling growth of the user base. This growth of the user population drives up the volume of concurrent activity demanded from the system. Here too, however, users require the same response time despite the increased throughput demands placed on the system. Such "scale-up" requires an ability to satisfy a growing workload against a constant data volume in a fixed time.

The nature of the workload comprises a third dimension of EDW growth. More subject areas and the exploratory nature of data analysis increase the likelihood of new combinations of data. The breadth of novel data combinations generates queries that are more complex. In addition, competitive pressures on the business produce analytics of increasing sophistication, yielding query logic of growing complexity. Finally, the growing user base wants to pursue many different tasks through a growing cacophony of applications, from business operations to data mining. The total workload mix is therefore increasingly

diverse. More complex queries and a more diverse query mix both add to the workload on the system. And this workload must deliver results within ever-tightening time constraints. This constitutes another form of "scale-up" that requires an ability to satisfy an increased workload in constant time.

Of course, these separate dimensions of scalability often appear in combination—more data, more users, more subject areas, more complex and varied workload, all in the same database. These factors multiply the effects of one another; doubling both the database size and the number of concurrent queries can easily mean a quadrupling of the system throughput required. To take a simple example, twice the number of concurrent table scans against tables that occupy twice the number of pages requires reading four times the number of data pages. EDW systems require a database infrastructure that can scale across all these dimensions of growth.

Figure 2: One-to-One Linear Size-Up



The "gold standard" for scalability is linear scalability. Linear scalability is the ability to service, in a fixed time, a workload increase of a constant factor by increasing the system resources—CPU, memory, and disk—by a constant factor. Ideally, the ratio of increased workload to increased system resources is one-to-one (1:1); that is, doubling the workload requires no more than twice the resources to complete in the same elapsed time as before the workload increase. Linear scalability is increasingly effective as it approaches this 1:1 ideal. On the other hand, 1:3 scalability—doubling the workload requires a threefold increase in system resources to maintain constant response time—is still linear, but no one would applaud such limited scalability.

It is not enough to be 1:1 linearly scalable, however. Inefficient systems can scale well—in fact, inefficient algorithms may well be easier to make scalable—but they are not cost-effective. In our experience, scalable database systems can differ significantly in performance, both in hardware resources consumed and in system cost. Making database software perform well has several components. First, fundamental database operations must be both efficient and scalable. Second, and equally important, is ensuring that the query optimizer makes smart decisions when it automatically selects what operations to perform to satisfy a nonprocedural SQL request.

Meeting the business requirements driving the EDW, as ever more data becomes both accessible and critical to profitability growth, imposes serious demands on database technology across the full breadth of its functionality. Not only do database products need to demonstrate broad-based efficiency and scalability today, but since the forces fueling database growth are accelerating, the underlying database architecture also needs to support continuous, rapid performance and scalability improvements well into the future.

1.2 AVAILABILITY AND RELIABILITY

Along with a growth in data size and workload, the role that the EDW plays within an organization has become more visible and more critical to the success of the business. Through the 1990s, business intelligence systems proved their ability to help control costs and increase sales. If it was once acceptable for a data warehouse to be down for a day, that is the exception today. Many enterprise data warehouses are now considered mission-critical.

In addition, analytic results from business intelligence applications are frequently being fed back to operational systems ("closing the loop") to influence customer treatment, user behavior, inventory management, logistics, and other operational decisions. Increasingly this feedback mechanism needs to be dynamic and up-to-the-minute. These business operations are also increasingly global and need the latest analytic results round the clock.

While EDW systems are increasing in size and complexity—in short, they have more and more "moving parts"—reliability and availability demands are also increasing rapidly. Keeping the data warehouse up and running is vital. Data and data integrity must survive inevitable disk failures. Reviving a system within minutes of a processor failure, without incurring major system degradation, is increasingly important. Organizations not only require a data warehouse that can grow with them as their data needs explode, but also require that their data warehouses have extensive safeguards to keep them running virtually flawlessly and to protect the value of their investment in the data itself.

1.3 MANAGEABILITY

As these systems grow in size and complexity and acquire ever more stringent reliability constraints, the challenge and costs of managing them—health checks, preventative maintenance, system upgrades, performance monitoring—cannot be allowed to grow at anything close to the same rate. Ideally, in fact, the system should become easier to manage as it grows. The increasingly competitive business environment not only drives the effort to extract more value from a larger data volume, but also forces lower unit support costs. This climate will not permit the achievement of linear scalability in performance to imply a linear increase in administrative costs. As they grow, data warehouse systems must be easier and cheaper to manage than ever.

1.4 GOALS OF THE PAPER

In this WINTER CORP White Paper, we examine how well Teradata meets these technical requirements. We explore the scalability and performance of the Teradata platform and its database architecture and evaluate Teradata's approach to reliability, availability, and manageability. We also consider the benefits and advantages of Teradata's approach, and describe representative examples of Teradata warehouses that demonstrate these advantages.

Teradata has long claimed leadership in solving the most demanding data warehouse problems. Over the past few years, other vendors have been challenging Teradata's leadership position. We believe this review of Teradata architecture and implementations reveals continuing distinct advantages when employing Teradata for very complex, large-scale enterprise data warehouses.

2 Scalability and Performance

Teradata customers have amassed an impressive record of system growth over the past several years. At the telecommunications giant SBC, for example, the quantity of user data in its enterprise data warehouse grew by 250 percent to 25 TB between the 2001 and 2003 Winter TopTen surveys. Today SBC's data warehouse supports over 100 applications and over 12,000 tables containing data from every area of the business. The largest table is 6.4 TB and contains 35.6 billion rows. Five other tables are over one TB in size. The system services over 300,000 queries on a typical day. Since the 2001 survey, the system has grown from 178 nodes[1] to 296 nodes, with over 200 TB of spinning disk on more than 10,000 disk drives.

Another large Teradata customer, Federal Express, has grown from two TB of user data to 10 TB since 2000, and the user population has grown from 200 to over 2,100. Today the database contains over 8,000 tables and services thousands of queries every day. To handle this explosive workload, Federal Express expanded the Teradata system from 12 nodes and 10 TB of disk space to 44 nodes and 38 TB of total disk space.

At Sears, the volume of user data has grown from 1.2 TB accessed by 600 users, as reported in the 1997 Winter TopTen survey to more than 6 TB today. Today the system has over 1,000 tables containing customer, retail sales and inventory data fed by 23 operational systems across Sears' many lines of business, and supports application access and ad hoc queries from more than 5000 users. Since that initial report in 1997, the Teradata system has grown from 20 nodes to 44 nodes and 70 TB of disk today.

These are but a few examples of rapid data warehouse growth across different industries and multiple dimensions. Many other examples in telecommunications, transportation, retail, and other industries could be cited. How does Teradata successfully manage continuous customer demand for higher data volumes from increasingly diverse areas of the business, and accessed by larger user populations via more varied and complex queries? A review of the scalability of Teradata's architecture sheds light on this question.

2.1 PARALLELISM – THE KEY TO SCALABILITY

The key to achieving database scalability in all its many dimensions is effective exploitation of parallelism. From its start in scientific computing, parallelism has been easiest to exploit, and has shown the greatest benefit, handling problems that are "embarrassingly parallel." Typically, these are cases where the same operations are applied to each unit of data, whether you are executing a complex algorithm on each element of an array, or performing a data type transformation on each record in a file. In these cases, little or no communication between the processing units is required to complete the desired operation successfully. As a result, achieving efficient parallelism is relatively straightforward—parallelize the input data, process each stream independently in parallel, return the result to the next processing step. As long as independent parallel processes can run freely, without dependence on or contention with one another, linear scalability is attainable.

2.1.1 Parallelism via Grids

The latest incarnation of this parallel processing paradigm is grid computing, where processors in a server farm perform small, discrete computations under the supervision of a director process. For example, SETI@home harnesses the excess capacity of thousands of individual computers on the Internet to analyze small packets of extraterrestrial radio signals for signs of intelligent communications. Closer

[1] In MPP systems like Teradata, a node is a collection of CPUs memory and disk, connected to the system interconnect. It is the smallest physical unit of scalability in the system.

to home, the Google search engine runs on thousands of PCs deployed at geographically dispersed data centers. A search request goes to the nearest Google site and is distributed to multiple index processors that each check a slice of the global keyword index for hits. Document IDs for each keyword are accumulated and intersected with hits on other keywords in the search. Individual documents are then retrieved from the document store in parallel by document ID and the results are assembled in HTML pages for display. Here too, the fact that little or no interprocess communication is required allows a straightforward "divide and conquer" approach exploiting parallelism to achieve linear or near-linear scalability.[2]

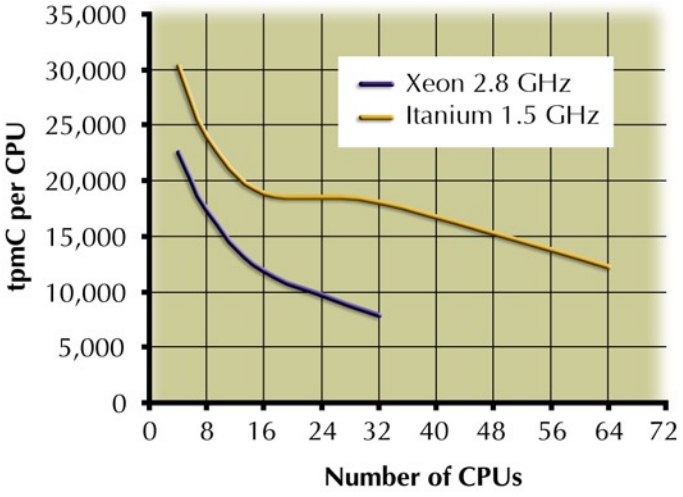
Applying the grid model to database processing can show impressive results for OLTP and simple retrieval applications; a large volume of small, repetitive operations characterizes both. In other words, grid processing excels at problems amenable to a simple scale-out solution. The processing of analytic queries typical in a data warehouse however, frequently requires significant communication among processing steps. For example, joins (especially non-co-located joins[3]), sorts, and grouping aggregations all require movement of potentially large amounts of data among units of parallel processing to produce a correct result. Achieving linear scalability for complex, data-driven query processing is a far more difficult engineering problem requiring a more thorough and sophisticated approach to parallelism. It remains to be seen whether a grid approach can be harnessed to master this class of problem.

2.1.2 Parallel Architecture: Shared Resource versus Shared-Nothing

Two architectural approaches to parallelism have emerged to address more complex parallelization problems. One is the shared resource approach; the other is called the shared-nothing approach. Both have been applied to solving the problem of efficiently parallelizing complex queries. In the shared resource approach, all parallel processing units can access all the data, both in memory and on disk. While this cleanly handles the data sharing needs of complex queries, it is prone to both decreasing resource accessibility and increasing resource contention as the degree of parallelism increases.

Large symmetric multiprocessing (SMP) computers containing large distributed memories enable very large shared database buffer pools, but they are subject to latency delays associated with accessing memory that is not located on the local processing board. Over the past decade, remote memory latencies have been reduced dramatically, but may still take two to three times as long, on average, as accesses to local memory. On the other hand, CPU speeds have increased at an even faster rate over this period, so the latency difference, while smaller in absolute terms, can actually have a larger relative impact. Similarly, as the number of processes sharing a resource,

Figure 3: SMP TCP-C Throughput per CUPU

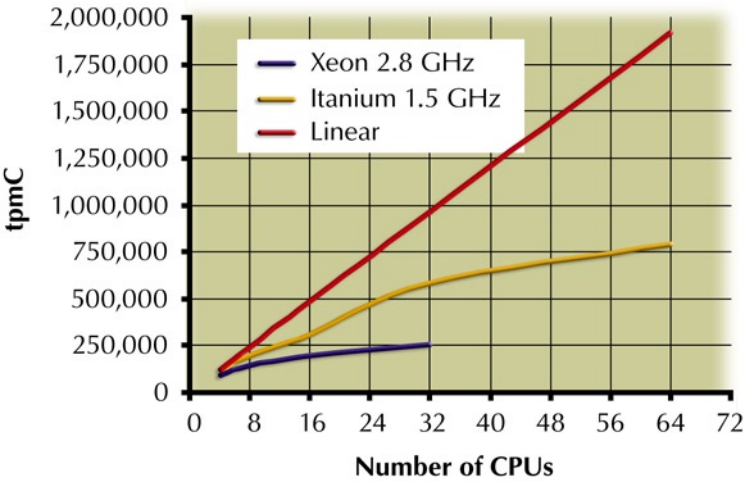


[2] Luiz André Barroso, Jeffrey Dean, and Urs Hölzle, "Web Search for a Planet: The Google Cluster Architecture", IEEE Micro, March-April, 2003, p22-28

[3] In a co-located join, corresponding rows of tables being joined exist on the same node.

such as a buffer pool or data page, increases, the opportunities increase for lock contention on resource requests for busy resources. As a result, it takes longer to perform operations that use heavily demanded shared resources. Memory delays, lock contention, and other resource sharing problems introduce inefficiencies that decrease overall system scalability. *Figure 3* vividly shows this effect. It graphs recent TPC-C transaction throughput results by the number of CPUs in the system configuration across two processor types, the Intel Xeon 2.8 GHz processor and the Intel Itanium 1.5 GHz processor. In both cases, transaction throughput per CPU declines rapidly as the number of processors in the configuration increases. *Figure 4* shows the impact of the decline in throughput per CPU on overall system scalability; for both the Xeon and Itanium SMP families scalability drops to 40 percent or less of 1:1 linear scalability. [4]

Figure 4: SMP Scalability - TCP-C



In addition, once the limits of the largest SMP system are exceeded, shared resource database systems encounter a different class of resource sharing problems, such as cache coherency, defined as the ability to maintain data integrity across multiple buffer pools running on different systems and under the control of separate instances of the database manager. The extra effort required to maintain cache coherency adds another layer of inefficiency and further inhibits system scalability.

The second approach to parallelism, called shared-nothing, eliminates resource sharing inefficiencies by avoiding resource sharing itself as much as possible. Each node in a shared-nothing multiprocessing system, also called a massively parallel processing (MPP) system, is a self-contained SMP server that uses only its own processors, memory, and disk pool and is responsible for all activity on the predetermined slice of the data that resides in its private disk pool. To handle the data sharing required for complex query processing, MPP systems rely on a dedicated communications link to facilitate high-speed data movement among nodes.

It is generally accepted that excellent scalability is attainable in MPP systems. To achieve this potential scalability however, two problems must be overcome. The first problem is that data sharing among nodes implies large-scale data movement. Joining customer data and monthly transaction data for example, may require redistribution of millions of records across the system interconnect. While network performance has improved steadily, it is still no match for memory speeds. In addition, as nodes are added the bandwidth of the interconnect must grow at least as fast to avoid becoming a scalability bottleneck.

The second problem is that MPP systems are much harder to build, use and maintain. Common operations must be dispatched on multiple nodes, their processing coordinated, their results collected, and their errors handled. Simple queries generate multistep query plans where each step runs on some or all processing nodes. Even simple operations entail setting up a dynamic parallel data flow that feeds appropriate inputs to each instance of the parallel operator and forwards each output to the correct instance of one or more downstream operators, often on different nodes. Further, the parallelism of

[4] Results derived from published TPC-C results as of Feb 15, 2004.

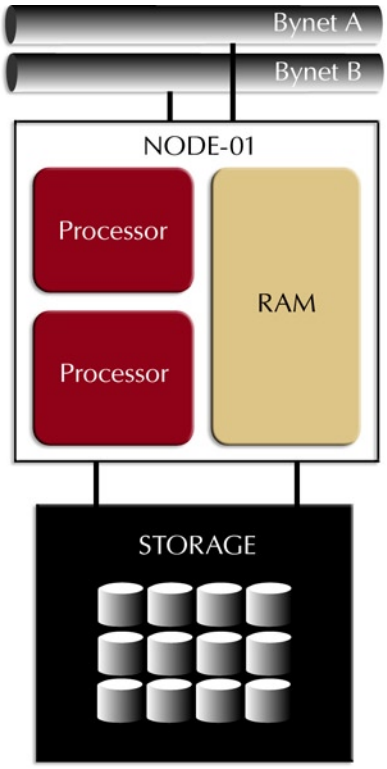
the system and the complex mechanism that controls its parallelism needs to be hidden from the user behind a non-procedural SQL interface.

The MPP system must also be maintainable; all the "moving parts" of a large complex system—and there can easily be thousands of components, processor boards, disks, power supplies, cables, etc.—should ideally act, and be managed, as a single system. Moreover, mechanisms must be provided to gracefully handle failures of individual components without compromising the availability of the entire system. These important factors impose additional demands and greater complexity on the MPP programming infrastructure. Managing both the complexity and the performance demands of MPP systems is critical to achieving the potential scalability of this environment and requires a sophisticated hardware and software infrastructure. Developing a high-quality infrastructure poses a substantial engineering challenge to building efficient and scalable MPP systems.

2.2 TERADATA'S MPP ARCHITECTURE

From its inception, Teradata adopted MPP architecture for its parallel database because only MPP offered the potential of 1:1 linear scalability. The Teradata engineering team has spent the past 20 years extending and enhancing the scalability and performance of the Teradata system, while at the same time addressing the usability and manageability difficulties inherent in MPP systems. In this section, we describe the key features underlying the scalability and performance of the Teradata system.

Figure 5: Teradata Node



Teradata is an integrated solution consisting of a distributed hardware platform and database software. The current generation hardware platform uses Intel 32-bit processors configured in dual processor SMP nodes running either a NCR branded SVR4 Unix called MP-RAS, or a Microsoft Windows 2000 Server. Each node contains its own private memory, up to 4 GB, and a private storage pool. Teradata systems scale from one to 512 nodes.[5] The nodes communicate via a scalable high-speed communication interconnect called the BYNET. The largest production Teradata system we know of runs at SBC and currently contains 296 nodes.

2.2.1 Balanced Hardware Configuration

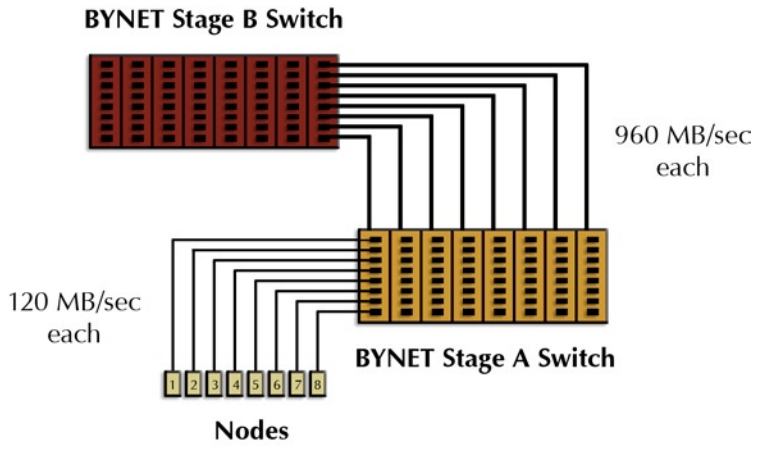
In a Teradata system, individual nodes are configured with a goal of balancing the performance of each component for the anticipated workload. The idea is to provide enough disk and interconnect bandwidth to support optimal utilization of the two processors on the node. Typically each node controls an array of mirrored disks that yield 500-600 GB of space for user data per node. Nodes are connected to the storage network via two independent adapter cards that each supports four separate fibre channel links, each link capable of 200MB/sec. For both redundancy and performance, each node has connections that total 1.6GB/sec of theoretical throughput. Nodes are connected to one another over the BYNET network, through a 4 port adapter card. Each port is capable of 120MB/sec, and each port connects to an independent, physically separate BYNET fabric for increased availability and performance.[6] The aggregate bandwidth of the interconnect scales linearly

[5] The maximum number of nodes will be doubled to 1024 in the next major release of Teradata Warehouse, Version 8.0.

regardless of the number of nodes in the configuration. The system thus scales in a balanced fashion as nodes are added to the configuration.

Historically, parallel systems have encountered serious difficulty in maintaining physical bandwidth scalability on the interconnect as the number of nodes in the system have increased. Typically, beyond some size threshold, systems can only sustain a fraction of the bandwidth between any two nodes that they are able to support below the threshold. At least one commentator has claimed that Teradata does not maintain linear bandwidth scalability beyond 64 nodes. As *Figure 5* shows, however, the BYNET architecture sustains the scalability of the physical network across Teradata's entire size range. Up to 64 nodes, a BYNET Stage A switch maintains 120 MB/sec. bandwidth per port. Beyond 64 nodes, multiple Stage A switches are linked via a higher-speed Stage B Switch to sustain this bandwidth in larger systems. The physical capacity of the BYNET has been demonstrated at Teradata sites with over 100 nodes, where concurrent bandwidth to each node in the system in excess of 70 MB per second has been reported. Achieving these throughput rates would not be possible if the BYNET were not truly scalable.

Figure 6: BYNET Linear Bandwidth Growth



The BYNET provides guaranteed message delivery and supports several message types—point-to-point messaging for communication to a specific node, such as to look up a row by primary key, broadcast messaging to communicate with all nodes, and dynamic group broadcast messages to communicate with a subset of nodes. It provides a lightweight, low-cost messaging protocol that enhances the performance of parallel algorithms by supporting rapid data distribution and fine-grained control messages. This is a powerful capability that, as we will see later, helps Teradata achieve higher levels of parallelism for many frequently used database operations.

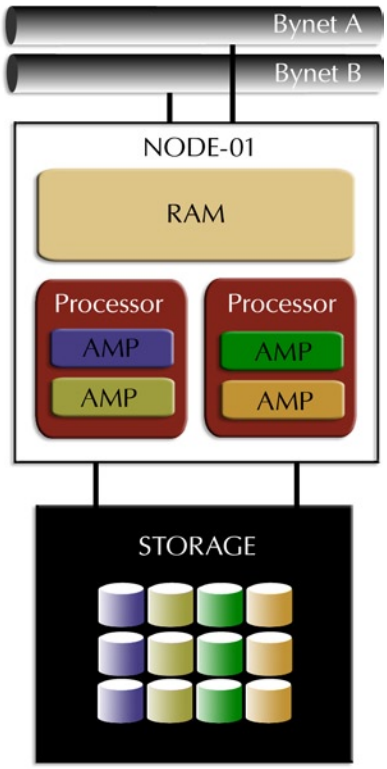
2.2.2 Database Scalability

The Teradata parallel database engine sits on top of this balanced, distributed hardware platform. Its goal is to achieve scalability by optimizing the parallelism of the system. It accomplishes this by reducing or eliminating those factors that restrict the degree of parallelism. One major factor limiting parallelism is data skew. In an MPP system, data is distributed across all the units of parallel processing

[6] All BYNET V2.x systems are preconfigured to utilize only two ports (connecting each node to two fabrics) for 240MB/sec of bandwidth per node in a dual active network configuration. Each node is "pre-plumbed" with the ability to connect to four independent fabrics should an application/system ever need that level of performance. Multiple BYNET cards per node are supported.

in the system. If the data distribution is unbalanced, or skewed, one or more units of parallelism will have more data to process, and, everything else being equal, will take longer to complete. The units of parallelism that complete earlier will most likely have to wait for the more heavily loaded parallel tasks to complete, which will detrimentally affect overall performance.

Figure 7: Teradata VPROC Architecture



The second major factor limiting parallelism is that many database processes have serial, or non-parallel, phases. The serial part of the algorithm determines the upper bound of the parallelism of the process. Not only can the process not be completed in less time than it takes to complete its serial portion, but even more importantly, the larger the parallelism of the system, the more the serial portion of the total process dominates its possible scalability.[7] Consider, for example, an operation that takes 20 seconds to complete, 10 seconds of which can be performed in parallel and 10 seconds that is inherently serial. Expanding the system from one to 10 CPUs reduces the time to complete the parallel phase tenfold to one second, but the serial part still takes 10 seconds. The net effect of increasing the scale of the system by a factor of 10 is that the processing time is reduced from 20 seconds to 11 seconds, or less than a factor of two. Now, however, the serial phase takes 91 percent of the total time. Increase the scale of the system by another order of magnitude, to 100 processors, and the task will still take more than 10 seconds and the serial part will be 99.99 percent of the total time. Clearly, minimizing serial processing is critical to achieving 1:1 linear scalability.

Let's examine how Teradata's architecture addresses the problems of data skew and serial processing elimination.

Central to the Teradata architecture is the virtual processor or VPROC. Teradata contains two types of VPROCs, one that manages the SQL interface to the user, called a Parsing Engine (PE) and one that owns and manages data, called an Access Module Processor (AMP). An AMP is an abstract machine that manages its own processor, memory, and storage resources. Each node is divided into a fixed number of identically sized AMPs. Each AMP exclusively uses a slice of the node's processor and memory resources, as well as its own private subset pool of the node's disks. As a result, each AMP owns and operates on an equally sized slice of the database. A typical configuration today allocates 10 AMPs per node. In effect, AMPs allow Teradata to extend the MPP effect to a small fraction of a node to achieve fine-grained parallelism. Finer-grained parallelism allows division of the total work among more units of parallel activity, thus reducing the magnitude and related performance penalty of data skew.

Data from every table in the database is distributed across all AMPs. Teradata has developed a hash-based file system that evenly distributes all rows across all the AMPs. Rows are hashed on a primary index defined at table create time. Teradata's hash-based file organization has proven very efficient at eliminating data skew. Measurements of data distribution on large tables—the ones that truly affect system performance—show a difference in row counts between the biggest and smallest partitions generally in the range of a fraction of one percent.

[7] This observation, known as Amdahl's Law, was first made by Gene Amdahl in 1967. It can be defined: "If F is the fraction of a calculation that is sequential, and (1-F) is the fraction that can be parallelized, then the maximum speedup that can be achieved by using P processors is 1/(F+(1-F)/P)."

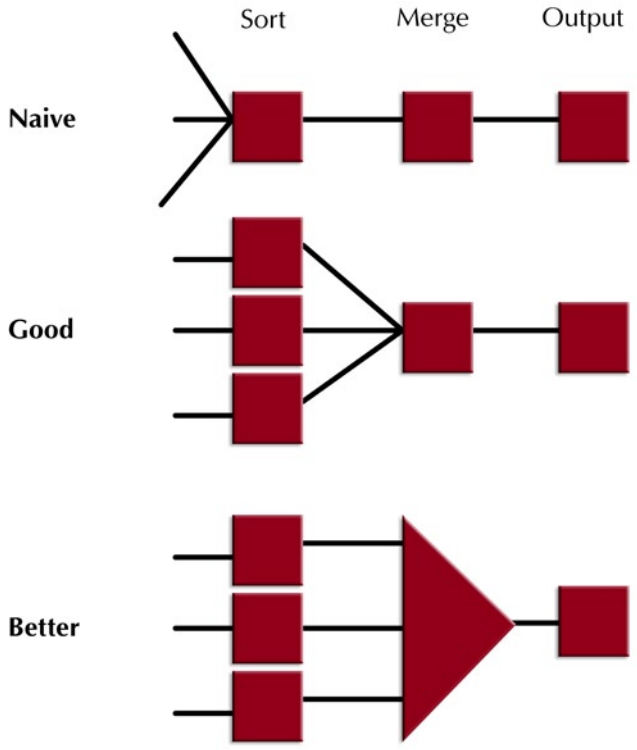
Some have complained of the inflexibility of forcing all data into one file organization. Why not sorted or entry-ordering of rows, depending on application access patterns, they argue. If, the goal is to maximize parallelism to achieve linear system scalability and very high system throughput on the largest size systems however, a hash-based file organization, by virtually eliminating data skew while enabling quick data retrieval, is an excellent choice. In addition, unlike an application-specific database, an enterprise data warehouse needs to anticipate multiple uses of the data. A hash-based organization is also an excellent choice for such general-purpose usage.

A more serious drawback to universal hash-based organization is its lack of support for data partitioning, for example, grouping data records by date. Observers have noted that query access patterns in large data warehouses favor certain data subsets. As an example, it is common to access recent data more frequently than historical data. To address this problem, Teradata introduced the partitioned primary index (PPI) feature in its most recent major release (V2 R5.0). PPI supports partitioning of data within each AMP by a partitioning key designated at table create time. PPI provides the benefits of partitioning while preserving the balanced data distribution provided by hash-based data organization.

Teradata's hash-based file system and fine-grained parallelism effectively ameliorate the negative effects of data skew on stored data. Query processing, by subsetting data unevenly, will often introduce another, dynamic form of data skew. Dynamic data skew, by its nature, cannot be anticipated. It causes wide performance variability in many parallel systems. Teradata's fine-grained parallelism, by dividing the workload into smaller parts across more units of parallelism affords the best-known approach to balancing the workload variability caused by dynamic data skew.

Because Teradata originated as an MPP system, it was forced from its inception to focus on the development of parallel database algorithms and the underlying parallel infrastructure to support them. As a result, Teradata comes as close to the "all parallel all the time" ideal as is practical. We can see this by examining how Teradata has virtually eliminated serial processing in operations that are notoriously difficult to fully parallelize, like sorting and aggregating data.

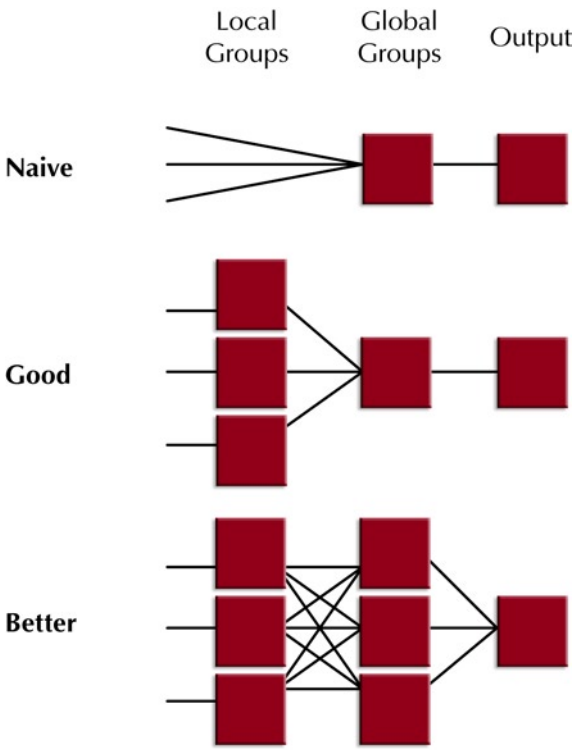
Figure 8: "Order By" Strategies



A naive but common algorithm to process an Order By query would select data from table partitions in parallel and feed the resulting partition row sets to a central serial process for sorting (*Figure 8*, Naïve strategy). In this case, the entire sort is a serial process and, for large result sets, typically dominates processing time for the query. Some leading database products still rely on this approach to sort data. A more sophisticated innovation is to perform local sorts in parallel on the partition row sets as data is selected from each table partition, and forward the ordered result set of each local sort to a central serial merge process to generate the globally ordered set (*Figure 8*, Good strategy). This approach limits the serial

portion of the algorithm to the final merge phase, a major improvement over the naïve approach. A more subtle benefit of this approach is the added performance efficiency that is derived from performing many small sorts rather than one large one; because sorting is an NlogN algorithm, reducing N improves overall efficiency. Also, this is likely to be as good an algorithm to ship data blocks among system nodes as is possible for a general-purpose communication protocol such as TCP/IP.

Figure 9: "Group By" Strategies



Because Teradata uses a low-cost messaging interface via the BYNET, it has been able to improve the local sort–global merge algorithm by parallelizing even the final merge phase. In Teradata, local sorts are performed in each AMP. AMP-level ordered sets are forwarded to BYNET logic, which first performs a parallel node-level merge on each node. Node-level ordered sets are then merged in the BYNET via a parallel binary-tree merge algorithm. This allows the BYNET to presents the globally-ordered set on demand to the requesting Parsing Engine (*Figure 8*, Better strategy). This binary-tree merge algorithm requires frequent control messages among nodes that would not be possible without an extremely low-cost messaging interface. This is an example of how Teradata uses fine-grained parallelism and low-cost messaging to gain a distinctive advantage in maximizing possibilities for parallelism.

Comparing algorithms for Group By query processing, or aggregation, presents another interesting case for Teradata's superior parallelization (*Figure 9*). The naive approach orders the row set on the group attribute in serial fashion and then performs the requested aggregation operation. An approach that improves parallelism performs local aggregations on table partitions and merges results serially. The technique for simple aggregates such as Sum is obvious, for others, such as Average, it's a bit more subtle—aggregate the component parts of the Average, the sum and count of the column values, and divide the sum by the count in the merge step to calculate the global average. In any case, the merge phase is serial. To the best of our knowledge, Teradata is the only database product to eliminate the serial merge phase of Group By processing. In a patented algorithm, Teradata merges local aggregates in parallel by hashing on the Group By value and forwarding the local aggregate for each value to the node it hashed to. This distributes global aggregation processing across the nodes of the system, where it is performed in parallel before results are collected for output. Teradata handles parallel calculation of SQL3's complex statistical functions via similar techniques.

Getting data into and out of the parallel database presents a further challenge to minimizing serial processing. In those cases where external applications can present data to Teradata in parallel streams or accept query results from Teradata via parallel streams, Teradata import/export utilities can natively accept parallel stream input or deliver parallel stream output. Much of the time, however, external applications present input data as a single stream or file and can only accept output as a single stream or file. Teradata handles single stream input by a fast, fan-out process that quickly distributes input with a minimum of processing to all AMPs where data transformation, primary index hashing, and

other preload processing can occur in parallel before input records are shipped to their destination AMP for loading into a table. Likewise, on the output side, merging to a single stream is deferred to the last possible step.

As these examples indicate, Teradata has aggressively pursued the parallelization of all algorithms. This commitment to maintaining the highest possible levels of parallelism is critical to Teradata's scalability.

2.2.3 Workload Growth - Another Scalability Dimension

Database scalability is not just about the volume of data. Growing user populations, integrating data from more areas of the business, and the increasing sophistication of business analysis, all yield more complex queries and more demanding workloads. To be truly scalable, the database system must handle these additional scalability dimensions at least as well as it handles scaling on data volume.

Teradata has a time-tested, cost-based query optimizer that readily handles the enormously complex queries devised by analysts for the most demanding data warehouses. Fifty-six-way table joins, 40 billion row tables, 12 page SQL scripts—all are reported as executing successfully and efficiently by Teradata customers. Teradata's query optimizer can quickly prune candidate query plans to rapidly determine an efficient plan for even the most convoluted query. The optimizer is fully cost-based, even for the query rewrite function, thus avoiding pathological cases where the rewrite decreases query efficiency.

In addition, Teradata's query optimizer is fully parallel-aware. For example, it takes data redistribution into account in its calculation of the costs of different execution strategies. A query optimizer that is not fundamentally parallel might miss the communications costs of data redistribution and select a suboptimal query plan. The Teradata optimizer can also anticipate the dynamic data skew that can result from different execution strategies and include in its optimization calculations the effects of skewed results on downstream stages of a query plan. Generally, this leads to cost preference for balanced plans that yields dramatic benefits for both response time of the individual query and for overall system throughput.

Teradata has many index options, including join indices and aggregate join indices. Since all relations, such as tables, indices, spool files, and join indices, are treated uniformly, the query optimizer is free to select the most cost-effective resource to satisfy a query. For example, Teradata may employ a covering index to resolve a query rather than retrieving data directly from the table itself.

Teradata's Priority Scheduler Facility dynamically supports mixed workloads by allocating system resources by job class. The priority scheduler can be configured to vary priorities of different job classes by time, so that, for example, online queries can be favored during business hours and long-running batch jobs can be favored during off hours. At times when only low priority tasks are active, resources will be fully utilized by them. Once higher-priority tasks are dispatched, resources are immediately diverted to service them, based on priority. Together, these features add up to a formidable toolkit for managing complexity and workload dimensions of scalability.

2.2.4 Performance

It is important to remember that scalability and performance are not the same; many inefficient algorithms are easy to parallelize for scalability. Being able to effectively exploit a quad processor system is no substitute for the ability to accomplish the same task in equivalent time on a uniprocessor.

Along with its commitment to scalability, Teradata has long focused on processing efficiency. At the lowest level, Teradata has developed an operating system interface called Parallel Database Extensions (PDE), that replaces low-level operating system memory management, process management, lock management, and I/O services with functions optimized for Teradata database use. It also provides additional services such as messaging, priority scheduling and the parallel infrastructure that enables

Teradata's distributed processing. PDE provides an optimized interface unmatched by general operating system services. In the past few years, Teradata has re-architected PDE as an open interface to enable Teradata on platforms other than MP-RAS. Today, in addition to MP-RAS, Teradata also runs on Windows 2000 Server and Windows XP Server. Linux and Windows 2003 versions are said to be in the works. [8]

Teradata's hash-based file system also provides many performance advantages. Since rows are stored in hash order by primary index value, disk performance for row retrieval by primary index is optimized. Not only is the correct data block and offset cheap to calculate, the calculation can be made without the overhead of reading blocks from a separate index. In effect, the primary index is built into the row structure. Join processing on tables with the same primary index is highly optimized. Not only is the join co-located, but the rows of each table are already in the correct order and can be simply joined directly as each table is read from disk. The hash-based file system also means that join processing performance is, at minimum, always linear.

Teradata's multivalued compression (MVC) feature, which supports compression of the 255 most frequent values of a column, can result in dramatic storage savings and impressive I/O reductions during data retrieval. At a large European telecommunications firm, for example, use of the MVC feature in a multi-terabyte database reduced disk usage by more than 30 percent. After conversion to MVC, load performance, measured in rows loaded per second, increased six-fold, while query execution time improved by two and one-half times. In addition, since the MVC algorithm uses a dictionary compression technique, the CPU cost of compressing and decompressing column values is minimized.

Another innovative Teradata performance feature is synchronized table scanning. This allows new scan requests to join an in-progress table scan at any point in the scan. The scan continues cycling until it returns to the start block of the latest participating request. Finally, cylinder read for table scans optimizes database I/O in one of the heaviest use cases. So while other database products approach Teradata's scalability, the wealth of performance optimizations, starting with PDE, gives Teradata a distinct edge.

2.3 SCALABILITY AND SYSTEM UPGRADES

Even for a database that delivers superior scalability, the scalability story would weaken significantly if the system could not be upgraded rapidly and non-disruptively to add additional capacity and ultimately replace obsolete processing and storage hardware once the capacity of an existing installation is exceeded. Adding system capacity is traditionally difficult in a shared-nothing system because it requires data redistribution; moving tens of terabytes of data and reallocating it across a different number of processors is truly daunting, time-consuming, labor-intensive and error-prone if it has to be done manually or semi-automatically. Even for shared disk architectures, adding nodes and disks to a partitioned database will likely require data redistribution.

Teradata's *reconfig* utility completely automates the data redistribution phase of the system upgrade process without requiring any manual retuning or redesign. Using the full power of the parallel processors and BYNET interconnect, it moves data targeted for redistribution in parallel, updates indices as needed and re-computes optimizer statistics. When *reconfig* finishes, the system is completely ready to resume its full workload. Teradata also supports multiple hardware generations in a single database instance (called Co-existence), making it easy to add nodes as workload increases while deferring the need for "floor sweep" hardware technology refreshes. Co-existence eases the upgrade

[8] John Catozzi and Sorana Rabinovici, "Operating System Extensions for the Teradata Parallel VLDB", Proceedings of the 27th VLDB Conference, Roma Italy, 2001

path while protecting existing hardware investments. In the latest release, Teradata has added the ability to migrate directly to new technology via *reconfig*. An option on *reconfig* targets redistribution off older nodes and onto newer ones.

Reconfig does require that the system be offline during the process, but Teradata has successfully streamlined *reconfig* processing in recent releases. At SBC, for example, a recent upgrade from 276 to 296 nodes was completed within a 12-hour window. Additional nodes and BYNET connections were attached to the running system beforehand. On a preset schedule, the Teradata system was quiesced and shut down. *Reconfig* was run. Teradata was then restarted and resumed regular weekend processing. No doubt the upgrade required significant planning—20 nodes and terabytes of storage were being added. The point is that such a major system upgrade was possible with minimal disruption to regular processing—no application rework or physical design changes were required. The automation of system reconfiguration is yet another area where Teradata's committed engineering effort has offset an inherent difficulty of shared-nothing architectures.

2.4 ISSUES WITH TERADATA'S INTEGRATED ARCHITECTURE

The scalability and performance of Teradata's architecture derives in part from Teradata's conception of the data warehouse platform as an integrated, dedicated hardware and software system. Observers have raised two concerns about Teradata's integrated solution strategy. The first concern is that Teradata requires that nodes performing database activities be dedicated to database work. Despite the general-purpose nature of Teradata's hardware and system software, running application processes on database nodes—parallel ETL services, for example—is strongly discouraged. Teradata recommends that all application tasks be placed on non-database nodes or on other servers connected to the Teradata system. This rule allows Teradata to better manage system throughput, especially for demanding workloads. While some commentators have considered this restriction to be a limitation of the Teradata architecture, dedicating servers to specific applications has become commonplace in today's IT world. In the era of standardized hardware components and utility computing, Teradata's approach may be viewed as more prescient than proprietary.

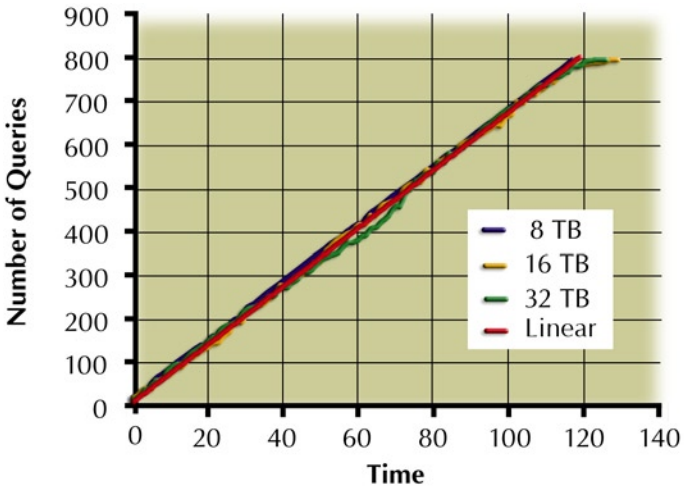
The second concern is with the cost of such a fully integrated solution. Justified or not, Teradata has a reputation in the marketplace for delivering excellent value at a relatively high price. Some argue that Teradata's pre-integration, by saving both installation cost and time, justifies a system acquisition premium. Teradata itself argues that today no system acquisition premium exists, that Teradata's prices are equivalent to competitive systems sized for comparable workloads. In addition, Teradata claims it offers lower operating costs, particularly due to reduced staff requirements, yielding favorable comparative total cost of ownership comparisons. Certainly, the market for very large data warehousing has grown more competitive, and the technology and market strategies of data warehouse products are reacting to these changes. In this light, yesterday's maxims about product capabilities and costs need to be re-evaluated by users building or expanding their data warehouse solutions.

2.5 LARGE-SCALE TEST RESULTS

Theoretical discussions of database architecture may indicate scalability potential, but do not prove scalability. Unfortunately, most database benchmarks are performed on a relatively small scale—hundreds of gigabytes of data, tens of users, handfuls of nodes. Teradata users regularly present such results at their annual user conference, called Teradata Partners. Teradata representatives can point to other benchmarks that have been run internally. All of these tests show excellent scalability for Teradata within the tested range. Since the highest scale point of these tests usually stops far below the scale reached by the largest enterprise data warehouses—tens of terabytes of user data, thousands of tables, hundreds of concurrent users, and large numbers of processors—they generally do not demonstrate that the scalability of Teradata extends this high.

In 2003, Winter Corporation had an opportunity to supervise a benchmark that tested the scalability of Teradata for very large databases. This test was run on a 40-node Model 5380 system with 1600 73-GB disk drives at the Teradata Benchmark Center in San Diego California. It involved a multi-user, mixed-query workload run against a database at three scale points: 8 TB, 16 TB, and 32 TB. At the largest scale, the database contained over one trillion rows. As *Figure 10* shows, the Teradata system demonstrated virtually linear scalability across all three tested database sizes. While a more comprehensive benchmark would be required to test scalability across multiple dimensions, this test does indicate that Teradata scalability extends to today's largest data warehouses.

Figure 10: Teradata Query Rates by Data Volume



2.6 TERADATA'S SCALABILITY ADVANTAGE

Teradata is built upon an architecture engineered for scalability from the beginning. It is based on a shared-nothing, or MPP architecture to support linear scalability from the smallest scale single node SMP systems to 512-node, multi-petabyte data warehouses. Its hash-based file system, fine-grained parallelism, low-cost messaging infrastructure and parallel database extensions are the cornerstones of Teradata's implementation of MPP architecture that provide a unique capability to exploit the scalability potential of that architecture. That capability is supported by benchmark results.

Beyond having a strong, elegant architecture and demonstrating excellent benchmark results, Teradata systems have been continuously tested on some of the largest real-world data warehouse applications known to exist. From its earliest releases, Teradata has been used at hundreds of the biggest organizations across multiple industries for the most demanding enterprise data warehouse problems. Its performance and scalability have been repeatedly tempered in the furnace of that experience as the frontier of database scalability has advanced year after year. That type of experience is invaluable for working through the scalability barriers, large and small, obvious and subtle, that all database products face. It is an experience that Teradata has gained over 20 years that is unmatched by any other database product. The table below, which displays the largest installed commercial Teradata system by year, provides evidence of the depth and scale of this experience.

Teradata's scalable architecture and wealth of experience in building the world's largest, most heavily used data warehouses provides a distinct scalability advantage for solving the biggest data warehousing problems.

Table 1: Largest Commercial Teradata Systems Shipped by Year

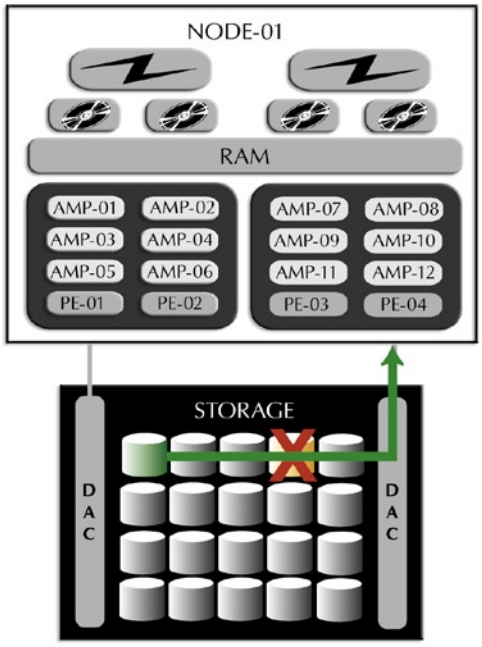
	1997	1998	1999	2000	2002	2003
CPU Type	Pentium	Pentium Pro	P II Xeon (500 MHz)	P III Xeon (700 MHz)	P III Xeon (900 MHz)	P 4 Xeon (3 GHz)
CPU Count	496	384	512	700+	900	900+
Memory Size	62 GB	192 GB	256 GB	700+ GB	900 GB	976 GB
Aggregate Continuous Memory BW	9.6 GB/sec	19.2 GB/sec	66.5 GB/sec	90+ GB/sec	125+ GB/sec	250+ GB/sec
Continuous Disk I/O BW	5 GB/sec	14 GB/sec	32 GB/sec	44+ GB/sec	55+ GB/sec	60 GB/sec
Raw Disk Capacity	24 TB	34 TB	72 TB 4096 disks	120+ TB 7000+ disks	160 TB 9000+ disks	175 TB 9300+ disks

3 Availability and Reliability

3.1 NO SINGLE POINT OF FAILURE

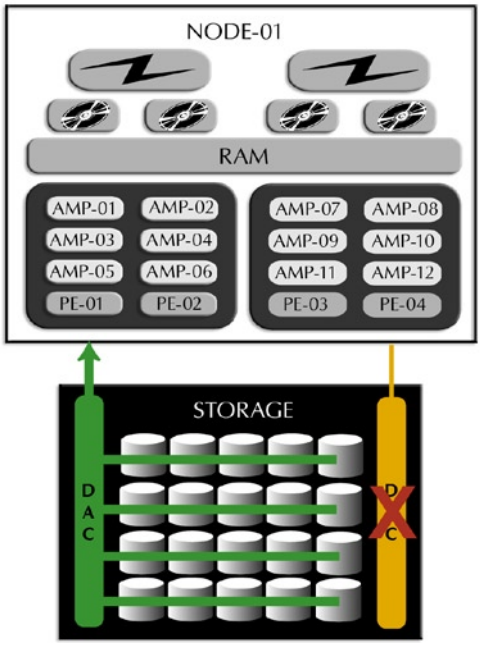
Successful data warehouses experience rapid growth in data volume and workload complexity. Many data warehouse are also becoming more integrated with the operational side of the business. For both of these reasons users are demanding higher levels of reliability and availability of their warehouse systems, but the challenges of increasing the size and complexity are often at odds with improving reliability and availability. Larger systems contain more parts that can fail. Load windows become tighter as data volumes increase, complex queries start to creep into the load window, and the system is subject to more stress as utilization increases.

Figure 11: Disk Failure



Teradata provides a comprehensive, integrated approach to satisfying the higher availability and reliability needs of today's enterprise data warehouses. All availability and reliability features are built into the system and are virtually automatic. Teradata systems are fully redundant with no single point of failure across the full spectrum of components. In Teradata, the management of the reliability and availability features does not require any significant effort from the support staff. By default, Teradata is configured to recover from the failure of any individual component by having the other components in the system automatically pick up the additional work of the failed component. Let's look at the major components of the system from the bottom up and discuss how each contributes to a highly available environment.

Figure 12: Disk Controller Failure



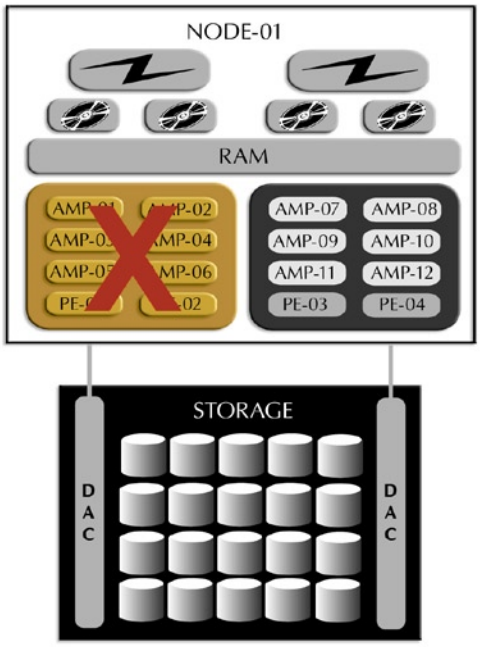
Because of their sheer volume and their lower unit reliability, the most common failures in a large data warehouse system occur in disk drives—a very large data warehouse may contain more than 10,000 disk drives. Everyone deals with this problem the same way, via tried-and-true RAID technology and hot-swappable drives. If one drive fails in the rank, as shown in the figure on the left, the other drive(s) in the array will be able to continue servicing I/O requests. The failing drive can be replaced while the system is operational by swapping in a new drive and having the system automatically rebuild the data on the new drive.

The disk array controller (DAC), which coordinates all access to disks, is another common source of failure. Here too, redundancy is used to mitigate failures. Dual active disk array controllers allow both controllers to access the disk

during normal operations. If one DAC fails, the other DAC in the disk array module will be used to service all module 1/O. As with all failure scenarios, the surviving components take over the workload for the failed components.

Both the DAC and disk drive failures are completely transparent to the end user. Queries continue to be serviced without interruption. The only noticeable user impact may be a degradation in query response times. Recovery from both disk and DAC failures is common across most platforms today. These features are just the starting point for Teradata, however. Teradata supports many more availability features as standard equipment, features that typically require significant extra cost and manual support on other platforms. These unique availability features are described below.

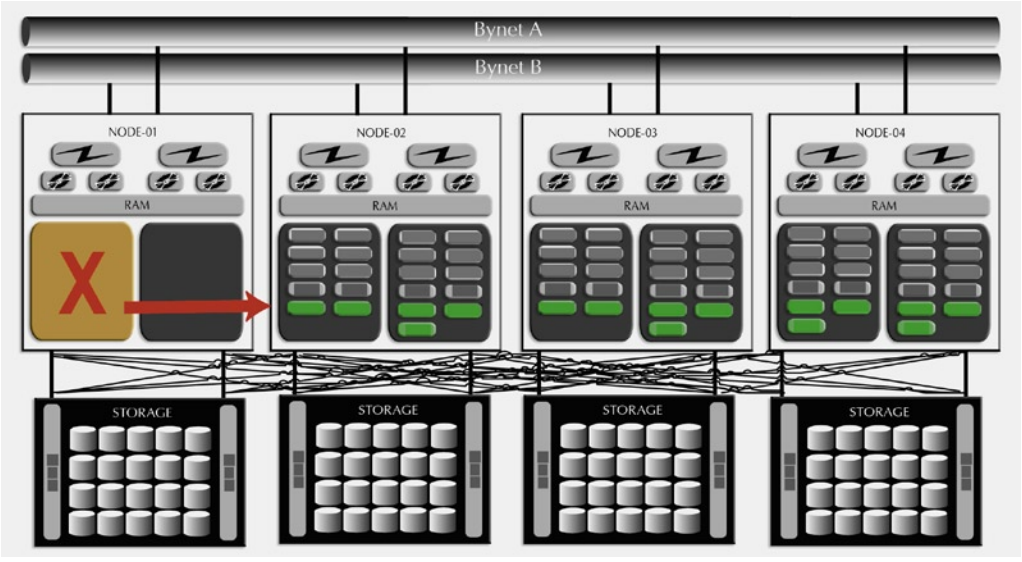
Figure 13: Processor Failure



Teradata provides automatic recovery from processor and node failures. This class of hardware failure would be fatal or require significant manual intervention in many architectures, and can be particularly troublesome for MPP systems. Key to the Teradata approach is the concept of cliques. A clique is a group of nodes, normally four, that share connectivity to the same disk array modules as shown in the figure below. Cliques are the standard unit of deployment of Teradata systems.

When a processor failure occurs, the node will automatically detect the failure and suspend execution of all queries running on the AMPs on that processor. In the same fashion, all queries managed by PEs on the failing processor are suspended. All of the AMPs from the node containing the failed processor are automatically migrated to the other nodes in the clique. This typically takes only a few minutes. Once the AMPs have migrated, the system will automatically resubmit those queries that

Figure 14: AMP Migration During Node Failover



were in-flight at the time of the processor outage. The system will continue to operate with access to all data because the AMPs from the failing processor and node are now running on other processors within the clique that, by design, have full connectivity to the disks of the failed node or processor. After the failed processor is replaced, the console operator specifies that the processor is back online and the AMPs will be migrated back to their original configuration, a process that also normally takes only a few minutes. Teradata's automatic failover feature is powerful enough to continue operations even after a second or third failure within the same clique, albeit with degraded performance due to depletion of hardware resources.

In its latest release, Teradata has expanded the clique idea by providing support for Large Cliques and Hot-Standby Nodes. Large Cliques of up to eight nodes now provide the option to lessen the impact of a node failure by spreading the migrating AMPs over seven surviving nodes rather than three. A Hot Standby Node is also an option for those situations that can tolerate little or no performance degradation following a node failure. Unlike other databases systems where failover is an optional feature, Teradata systems are shipped with cliques and failover preconfigured.

Another major hardware component, the BYNET, Teradata's high-speed interconnect, is comprised of two or more separate high-speed networks. Like other Teradata components, both are always active and servicing requests. Should one fail, the other network manages all requests. Teradata is the only MPP system that offers a fully redundant interconnect and, on Teradata, it is a standard feature of the system. All of the other secondary types of components such as power supplies, fans and adapters are normally configured for redundancy. Failures to these types of components are normally transparent to the end user.

Another distinctive feature of Teradata, fallback, can be used for tables that require a higher level of redundancy than is commonly provided by RAID. Fallback provides this added redundancy by automatically creating a copy of each row in another part of the system. Fallback guarantees that the data for the fallback row will always be stored physically separate from the primary row. This approach allows the fallback copy of a row to be accessed should a very severe problem cause all the nodes within a clique to be inoperable or multiple disk drives within the same rank to fail. Fallback is implemented by a simple Data Definition command issued against the desired table. Beyond that, no additional work must be performed by the Teradata DBA to manage the fallback copy.

Despite its high value, fallback may be cost-prohibitive to implement it on all tables across the board—this would double the space requirement and also double I/O and processor load times. Within most data warehouses, the actual requirement for availability will vary by subject area or table. For example, a retailer's replenishment team can live with up to two days of outage to data pertaining to store attributes but must have access to yesterday's item movement every business day with no more than four hours of downtime per quarter. Another team might require higher availability for more recent data while lowering the requirement for older data. Fallback, by providing table level control, can provide higher availability to only those resources that require it. This controls costs because higher availability features do not have to be implemented on all or large segments of the database (e.g., tablespace or file system) when only certain subsets of the data might require it.

While it is not possible to develop completely bug-free software, it is possible to reduce the severity and disruption of bugs when they are encountered. Teradata can gracefully recover from many classes of software failures without causing a system outage. Software failures are normally detected, logged, and the offending request automatically aborted and resubmitted. The Teradata system also contains separate diagnostic hardware and software to detect, collect, fix, and report problems. These diagnostic components continually monitor the BYNET, processors, nodes, adapters, disk array modules and other vital components. Enterprise data warehouses are frequently comprised of hundreds or thousands of

individual components. The tight integration of the hardware and software enables errors and failures to be automatically detected and remedied with little or no human intervention. Recovery is fully automatic—no additional software must be acquired, installed, configured, or managed.

Major failures caused by either multiple concurrent failures or a disaster such as a fire, while improbable, are also possible. In addition to the built-in features described above, Teradata offers three further levels of disaster recovery protection: BAR (backup and recovery) and load tools that can be assembled to construct a private solution for disaster recovery; Teradata Recovery Centers for offsite coverage providing a range of service levels; and a new solution called Teradata Dual Active, involving multiple active systems sharing the query workload and providing failover for each other in the event of a disaster.

3.2 HIGH AVAILABILITY

As we have discussed above, an outage of virtually any component will result in little or no noticeable impact to users. Teradata provides a comprehensive availability solution—not just redundant components, but more importantly, the tight integration of the hardware and software to enable reliable self management. This results in a system where the impact of outages is minimized. These high availability capabilities affect the first dimension of availability—the frequency of outages.

The other dimension of availability is the duration of outages once a failure has occurred. Teradata provides rapid recovery from a major outage such as a processor, node, or even clique failure. The actual outage time observed by users once such a major failure is encountered is normally in the two- to four-minute range. This recovery enables the system to continue running after migrating all work to the operating components. A similar period is required once the component is replaced and the system is restarted to bring the replacement component online. This final restart may be deferred to a convenient system window.

A system that is scalable in all dimensions must be able to support a recovery within a reasonable period as the size of the system grows. With Teradata, the time to recover is predictable and remains constant at even as the system grows in data volume and number of nodes.

3.3 A REAL-WORLD EXAMPLE

Most failures that occur actually go unnoticed. The system automatically takes care of the problem without human intervention. For most problems, the only interaction occurs when a field engineer replaces a failed drive and notifies the system that the component has been replaced. During a recent very resource-intensive benchmark conducted by Winter Corporation, we tested many of the availability features of Teradata. This test involved powering down various components to simulate an outage. All of the availability features worked as designed. For simulated component failures, the system continued running; it was barely noticeable that a failure had actually occurred. We also tested severe cases like node failures. A node was powered off while the system was simultaneously loading data and running a set of queries. As expected, the system stopped processing while the AMPs were migrated to the surviving nodes in the clique, which took about three and a half minutes. After that point, all of the workload was automatically resubmitted and the system continued to operate in spite of the fact that one of the nodes was now offline. Up to this point, everything worked as planned.

The next step in our test plan was to restore power to the powered down node so that it could be brought back on-line. However, much like real life, where examples of Murphy's Law abound, the

engineer accidentally powered down another node that was operational and processing its share of the system workload. We watched while the system went through its normal recovery process, and in three and a half minutes the system was back and operational. However, this time two nodes in the same clique were completely powered down. This unplanned outage showed that the system could gracefully recover from multiple as well as single outages.

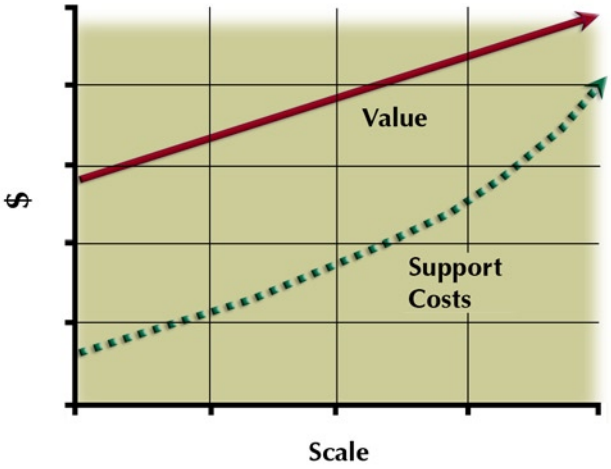
Overall, our availability tests demonstrated that Teradata delivers a world-class high availability solution out-of-the-box, one that automatically provides very high levels of reliability and rapid failure recovery with minimal human intervention.

4 Manageability

As data warehouses grow, the ease of managing the system becomes a factor that drives cost and, therefore, the return on investment (ROI). Information technology executives are demanding low, predicable increases in cost for each unit of value anticipated from the enterprise data warehouse As the system grows, or as new subject areas, applications, or historical data are brought into the warehouse it should not result in a disproportionate increase in support cost (see *Figure 5*).

Teradata's system managed approach makes system software automatically perform as many management tasks as possible. Many of the mundane tasks that the DBA or support staff must normally perform on other DBMS platforms are automatically performed by Teradata, resulting in lower overall support costs. Ease of management not only applies to the ease of ongoing warehouse support, but also the ease of setting up and making major changes in the environment. These distinctive manageability features of Teradata are discussed below.

Figure 16: Undesireable ROI Curve



4.1 EASE OF MANAGEMENT

Since the introduction of the Teradata parallel database computer (DBC 1012) in 1984, the product has focused on providing an integrated approach to manageability. As Teradata has continued to evolve, an integrated management interface had been developed to mask much of the system complexity. Teradata follows the principle of "Do not burden administrators with management of the physical details when the system can handle these details itself more efficiently and more effectively." Teradata provides sophisticated hardware and software monitors—the Administration Work Station (AWS) and Teradata Manager—that give administrators both a global system view and the ability to drill down to individual components. This provides efficient management of large, distributed systems.

Managing space in other large data warehouse environments can be a daunting task. Disk space must be properly configured with drives, slices, files, extents, and tablespaces. To ensure balance and optimal operation, these units of storage must be continually monitored and tuned. By contrast, with Teradata the only comparable space management task is monitoring the overall utilization of disk space by database. This task can be easily automated to have the system issue a warning when a certain threshold is exceeded.

A Teradata system is normally delivered to a customer with the disk and operating system preconfigured. The DBA simply creates a database and limits how much space it can use. No storage space is preallocated. Tables can then be created in databases using standard DDL. As tables are created, Teradata ensures balanced distribution of data across the AMPs. That's it. Unlike other database management systems that have multiple storage pools that have to be separately managed, partitioned, and monitored, Teradata provides a much simpler, easier-to-understand scheme. As the system gets larger with more tables, the advantage of Teradata's simplicity only grows. The little work that is required under Teradata can be performed directly within the DBMS via SQL statements. Operating system parameters are seldom if ever modified. There is little need to tune memory, sort heap size or buffer pool size, as Teradata dynamically manages memory resources.

In other database environments, as tables are updated their physical space and secondary indexes become fragmented, leading to wasted space and suboptimal I/O operations. Database products typically have a reorganization utility to reclaim this wasted space and improve I/O. Teradata automatically reclaims space as needed in small increments, as the system has processing cycles available in the background. Hence, no reorganization utility is needed. Complete database reorganization never has to be manually performed and access to all tables never has to be interrupted. While other DBMS products have improved in this area, the reorganization function must often be invoked manually, and can cause a table to be unavailable during its operation.

Workload balancing is critically important in an MP environment, but almost impossible in a VLDW environment without a high degree of automation.

One design goal with Teradata is to distribute data uniformly across all of the AMPs to achieve even workload balancing. After all, in an MPP environment, an operation can only be completed as fast as the slowest unit of work. As noted earlier, the rows in a table are hashed on its primary index to a specific AMP and disk location via a hashing algorithm. This automates data distribution, and therefore workload balancing, so the database designer or DBA seldom has to put any significant effort into the distribution of data. This automatic workload balancing is maintained even after a failure. If a particular component fails, the operating components pick up the work that would have been executing on the failed component. While the work may not be perfectly balanced across the entire system, the work will be balanced within the components that are capable of participation.

Failures rarely require human intervention. Some vendors are touting new autonomic or self-manageability features. Teradata has had these autonomic features integrated into its product for many years. All of the components in the system are continually monitored, with actions taken or early warnings delivered to avoid more serious failures. Many of these diagnostics are run without any human intervention. Diagnostics that require additional resources are initiated by a human, but most of the monitoring work is performed by the system itself. And should a failure occur, recovery is fully automatic. Of course, humans are required to physically remove and replace failed components. All of these sophisticated diagnostics and automatic recovery contribute to an environment that is easier to manage.

As discussed earlier, availability is built-in and automatic. No effort has to be devoted to installing the availability software, node groups do not have to be defined, and recovery mechanisms do not have to be configured and tested. On other platforms, the effort required to properly configure the availability software can be extensive.

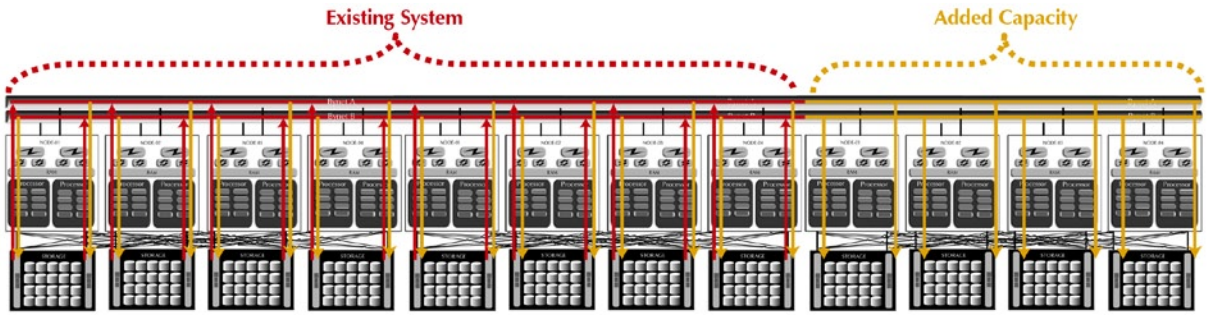
Another aspect of manageability that is sometimes overlooked is automatic parallelism. The designer does not have to carefully plan how and to what degree parallelism should be exploited. The Teradata optimizer constructs the optimal plan, parallelizing each step across the involved units of parallelism (AMPs) and between each independent step. In addition, loads don't require any special pre- or post- processing to split the load file into separate fragments to be loaded in parallel. The load utilities only need to know the location of the file, layout of the fields, any optional transformations, and the destination columns. After that, the system automatically parallelizes both the load process and the resulting table data.

This automatic parallelism is especially beneficial after a system upgrade. With some DBMSs a system upgrade often would require substantial manual effort to rebuild the partitioning scheme, redistribute data, rebuild indices, recomputed statistics, and change queries and load scripts. Since Teradata fully

controls the degree of parallelism, no such work needs to be performed either before or after an upgrade. An added benefit of the simplicity of the load process is that it often results in minimal data movement.

The system upgrade process itself normally requires a major administrative effort. However, with Teradata, once the new hardware has been installed, virtually all of the remaining work is performed by the *reconfig* utility. There is no need to delete and recreate secondary indexes or statistics. No queries or load processes have to be changed. Teradata has improved the performance of this process so that it can normally be completed within one day, regardless of the size of the system. Teradata systems can support up to four generations of technology within a single environment. This extends the useful lifespan of the hardware, maximizing capital investment without increasing the difficulty of data migration.

Figure 17: Data Redistribution via *Reconfig*

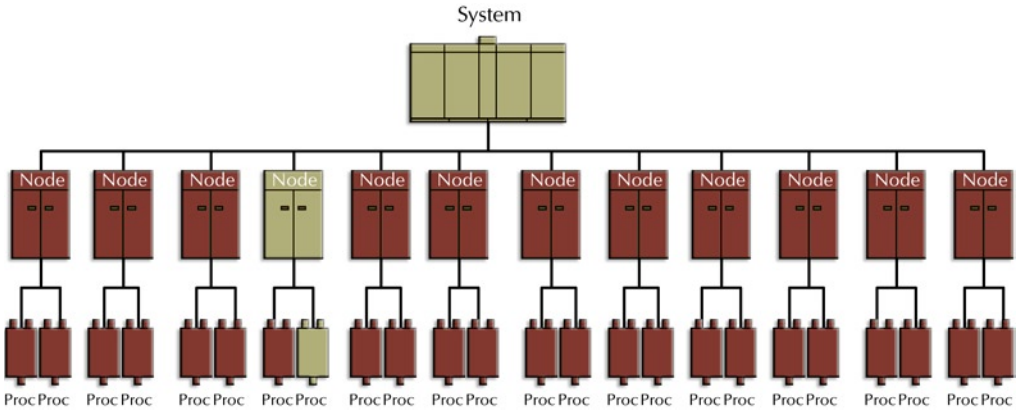


Teradata will collect statistics efficiently in parallel so that the optimizer can immediately have the latest data demographics to generate optimized plans. There is never a need to rebind any of the applications to take advantage of updated statistics or an upgraded system with a new number of AMPs. Other DBMSs may require substantial redesign work to perform optimally in the new environment.

4.2 A SINGLE UNIFIED SYSTEM

Successful data warehouses grow—into thousands of disk drives, hundreds of processors and dozens of nodes. Without a unified management approach, this growth can be a management and support nightmare. Teradata's AWS presents these many independent parts as a single unified system.

Figure 18: Unified Management of a Distributed System



Software upgrades are managed by the Parallel Upgrade Tool (PUT) as a single, integrated installation process, regardless of the number of nodes—not a separate process for each unit of parallelism or node. Contrast this to clustered systems where each node is like a separate system. Software maintenance is applied to each node of the cluster individually. This obviously requires more work, and more importantly, is more prone to error because each node must have the software applied in exactly the same way across the entire cluster.

The utilities that are used to monitor a Teradata system all operate under this principle. At the AWS, the hardware is presented in a hierarchical fashion. All issues are trickled up and reflected in the hierarchy of components. For instance, a problem with a single processor is not only shown as a processor issue, but also an issue within that node and at the system level. The operator does not have to monitor each of the individual processors or nodes. The operator can easily see the severity of issues at any level of the system—drilling down to those areas that are of interest.

The main performance-monitoring tool, Teradata Manager, allows operations staff to view performance and other critical information across the entire system. Individual components such as nodes and processors can be viewed side by side to get a wide view of workload balance. Additional detail can be obtained by successively drilling down into the areas of interest. All of the management and administration tools operate under this unified system model. Databases and all objects are created in the system-wide data dictionary and not in the individual units of parallelism. This allows all of the potentially thousands of objects to be centrally managed. Management of the system is simpler and requires fewer resources as the system grows. As a result, support requirements grow more slowly than the growth rate of the database size, number of applications, and users supported.

4.3 ROBUST WORKLOAD MANAGEMENT TOOLS

Teradata provides a suite of sophisticated workload management tools that simplify the job of the DBA. Rather than continually responding to problems like killer queries and slow response times, Teradata provides tools to manage many of these everyday issues proactively.

Consider, for example, the infamous product join or Cartesian product query that occurs when a user mistakenly omits a joining predicate or improperly constructs a predicate, resulting in a query where every row in one table is joined to every row in a second table. The system then attempts to carry out the process as instructed, potentially generating a result set of billions of rows. Not only is the result set meaningless, but the system must also expend considerable resources to produce this meaningless result. These and other types of human errors can bring a database system to its knees if not properly handled.

The Teradata Dynamic Query Manager can recognize and reject resource-intensive or killer queries prior to their execution. The DBA controls TDQM behavior by specifying rules to either terminate or defer questionable queries based on several factors. Different rules can apply to certain times of the day or week, and to the estimated processing times of the submitted query. With TDQM, the DBA no longer has to continually monitor the workload and manually abort killer queries.

Once a query begins execution, the Priority Scheduler Facility manages the quantity of system resources allocated to it. While Teradata has always had a basic priority scheme, namely to allocate more resources to more important tasks, recent product enhancements have resulted in a more comprehensive solution. These newer features enable Teradata's concept of an Active Data Warehouse (ADW). ADW is designed to support mixing short tactical or operational (quasi-transactional) queries with larger, longer, strategic queries in the same database leveraging the same copy of enterprise data. Distinct query types can be run in different priority classes. Each priority class has a relative weight that determines its priority in accessing system resources, to assure quick turnaround for queries that require it. Query milestones differentiate short- and long-running queries submitted by the same group of users by automatically

demoting the priority of a running query after it has consumed up to a specific threshold of CPU. These and other sophisticated scheduling techniques enable the right quantity of resources to be applied to the right tasks.

4.4 CUSTOMER EXPERIENCE

Some of the VLDW projects that we have directly participated in have proved to be interesting examples of Teradata's manageability features. One such project has a large Teradata data warehouse with a total support staff of two full-time DBAs and one part-time operations support person. The primary reason for the second DBA was to insure coverage, should one be unavailable. Over a 12-month period, the system doubled capacity, tripled the number of subject areas and quadrupled the number of users—all without adding new support staff. This was achievable due, in large part, to the manageability features of Teradata.

The data warehouse at Continental Airlines collects data from 41 sources, including flight schedules, seat inventory, revenue and ticketing data, customer profiles, frequent flier information, and employee payrolls and supports access from 1300 employees in 35 departments via query and reporting software. It services an expanding list of applications including revenue management, customer relationship management, fraud detection, and payroll management. A parts and maintenance application is in the pipeline. The system posts updates in near-real-time via automated data transformation processes so analysts are working with data that is only seconds old. This system, built and managed by a total staff of 15, earned Continental a recent TDWI Best Practices award. Continental estimates a cost savings and revenue increases totaling several hundred million dollars because of its Teradata-based enterprise data warehouse.

5 Conclusion

In the increasingly competitive enterprise data warehouse market, Teradata continues to possess distinctive advantages that make it the system to beat for the largest, most complex data warehouse applications. All major DBMS products have made enormous strides across multiple fronts in recent years. Nonetheless, Teradata continues to maintain technology and market leadership in performance and scalability, availability and maintainability for the most demanding enterprise data warehouse applications. The key reasons for Teradata's distinctive advantages are:

- Teradata's low-cost, fast messaging infrastructure enables small-scale interactions among components of distributed algorithms to support higher degrees of parallelization, contributing to higher scalability. The parallel merge phase of SQL order by processing is a prime example of this capability;
- Teradata's virtual processor architecture enables fine-grained parallelism that is resilient to parallel-efficiency sapping effects, such as the dynamic data skew commonly created by many business intelligence queries;
- Teradata's hash-based file system promotes the balanced distribution of data, so key to linear scalability;
- Twenty years of experience with leading data warehousing users overcoming barriers at the advancing frontier of scalability have "hardened" Teradata's parallelism and scalability to a level unmatched in the industry;
- Teradata's priority scheduler provides sophisticated workload management that allows balanced execution of mixed workloads;
- Teradata's automated system management capabilities (AWS and Teradata Manager) effectively turn management of a large MPP system with thousands of components into management of a single integrated system; and,
- Unlike any other DBMS marketed today, Teradata provides a built-in, fully automated availability infrastructure—at no extra cost and with no assembly required.

Teradata technology is not standing still. In recent releases Teradata has added a wealth of product enhancements that include:

- Partitioned Primary Index—supports partitioning of data without sacrificing balanced data distribution, resulting in parallel execution of partitioning logic;
- Multi-valued Compression—dramatically reduces database storage and I/O cost; and,
- Dynamic Group AMP Processing—optimizes queries that touch data on only a few AMPs, for example, rows satisfying highly restrictive conditions
- Large Cliques and Hot Standby Nodes—reduces or eliminates performance degradation following node failover

Every complex, large-scale data warehouse has unique requirements, and no product satisfies them all; companies should evaluate database products in light of their specific needs, preferably via quantitative measurement. Nonetheless, Teradata, with its fundamentally scalable architecture, tempered by 20 years of successful experience at the leading edge of large database practice, provides distinct advantages and merits serious consideration by organizations facing the world's toughest data warehouse challenges, now and in the future.

A leading center of expertise in very large databases,
Winter Corporation provides services in
consulting, research, architecture and engineering.

We help users and vendors understand their opportunities;
select their database and data warehouse platforms;
define and measure the value of their strategies, architectures and products;
plan, architect and design their implementations;
and manage their scalability, performance and availability issues.

Our focus is databases near, at and beyond the frontier of database scalability.

WINTER CORPORATION
411 WAVERLEY OAKS ROAD, SUITE 327
WALTHAM, MA 02452
617-695-1800

visit us at www.wintercorp.com