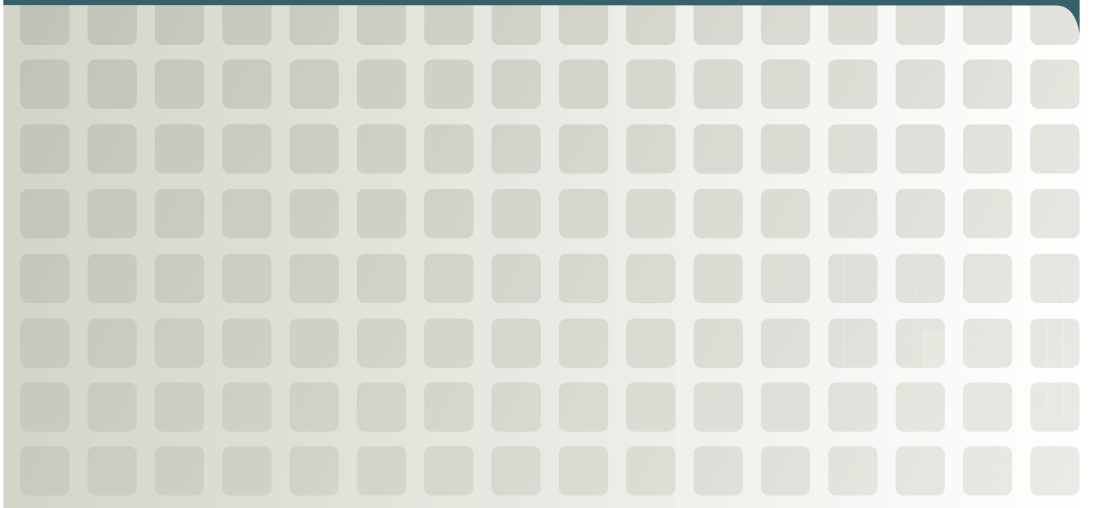
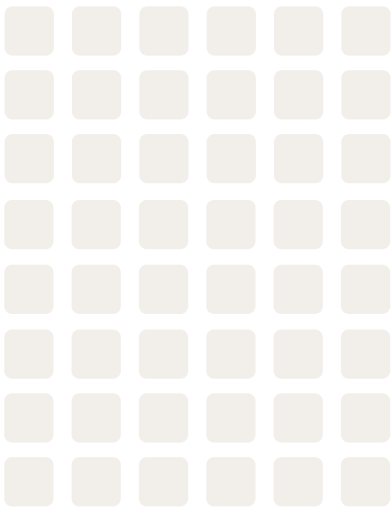




**MAINSTREAMING DATABASE CLUSTERING:
RDMA AND INFINIBAND**



MAINSTREAMING DATABASE CLUSTERING: RDMA AND INFINIBAND

Faced with shrinking budgets and increasing demands, administrators are looking to enhance database performance while reducing costs. One solution is database clustering which, historically, has only been deployed at the high end of the market. This whitepaper describes existing clustering architectures and how emerging technologies such as RDMA and InfiniBand promise to deliver the benefits of clustering to the mainstream market.

Clustering Today

Clustering is not a new concept. The ability to distribute a single, logical, relational database across multiple servers has existed for years and is deployed for two reasons: (1) to add high availability; and (2) to increase scalability.

High Availability

Clustering is primarily deployed today to add high availability. Many operating systems and applications natively lack this ability. To compensate, middleware products can add cluster fail-over capabilities. These packages allow two or more server nodes to be clustered and present the cluster to the outside world as a single system. This is accomplished by sharing common resources, such as disk storage, IP addresses, and application instances. Each node in the cluster can also have active resources simultaneously. For instance, two independent databases could be clustered in such a way that each database is active on its own node. Normally, each node is actively running its own independent database. In the event that one of the nodes fails, the other node will immediately take over active ownership of all of the resources from the other node. Some common middleware packages that do this include Microsoft Cluster Server, PolyServe Matrix HA, and Veritas Cluster Server.

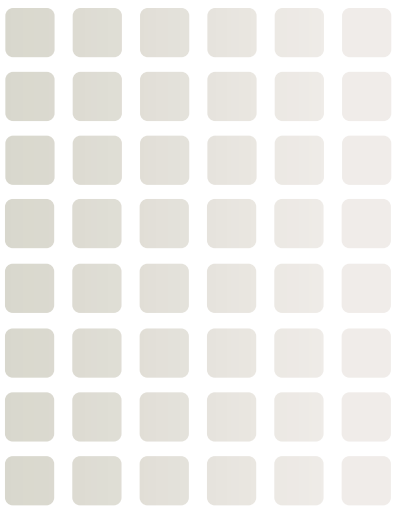
Scalability

Clustering is also deployed to add scalability, or the ability to scale out. Clusters are designed to support additional nodes, with two primary database architectures: Shared Storage and Shared Nothing.

The Shared Storage architecture stores the entire database on disk storage which can be directly accessed by each node. This model has typically been the best for scaling the database to handle larger numbers of simultaneous clients, especially in an Online Transaction Processing (OLTP) environment. Oracle's RAC is the only real example shipping today of an enterprise database engine that uses this architecture.

The Shared Nothing architecture usually involves partitioning the database into private slices which are then maintained by each node. This model has traditionally been the best for scaling query performance in very large Decision Support Systems (DSS). The Shared Nothing architecture has been implemented by several database software vendors in the past including IBM DB2, Informix (now owned by IBM), Tandem Non-stop SQL, and NCR Teradata DB.

The ability to combine high availability with scalability depends on the architecture. The Shared Storage model natively provides both capabilities. Since all of the disk storage associated with the physical databases is actively shared by all nodes, individual failures do not affect the overall integrity of the database. In contrast, Shared Nothing databases do not natively support fail-over, but when coupled with Cluster Failover middleware packages they can support fail-over by setting up a hot standby node for every active node.



Latency and Bandwidth: Keys to Scalability

Although databases are designed for scalability, they are still significantly limited by how effectively cluster nodes can communicate. For this, clusters rely on a specialized network between cluster nodes, called an interconnect, and two key characteristics—latency and bandwidth.

Latency directly affects cluster scalability. Clustered databases require synchronization between nodes, and the slower the synchronization, the less they can scale. With Oracle giRAC, scalability of a cluster directly relates to the time it takes to access physical storage. And for database instances to access physical storage, they must have exclusive access (lock) to data. Thus, the longer it takes to acquire locks from other nodes, the lower the overall performance of the database. In OLTP environments this means that the number of transactions that can be performed is directly proportional to the latency of the cluster interconnect.

In contrast to Oracle's distributed lock manager, IBM's DB2 Parallel Edition delegates control of database slices to individual nodes, but also requires low-latency communication. For example, when tables span several nodes, requests may be issued across the cluster in parallel, resulting in individual requests being funneled back to the requested node for aggregation. In this case, latency also impacts system scalability which depends on the time necessary for all nodes to receive the initial request, perform the table scan and get the results back to the requesting node.

Bandwidth also affects scalability due to the large amount of data transferred between database nodes. Oracle giRAC introduces more traffic between nodes via CacheFusion which sends cached database blocks directly from one instance to another within a cluster. This eliminates the need to go directly to physical storage, decreasing latency and increasing scalability.

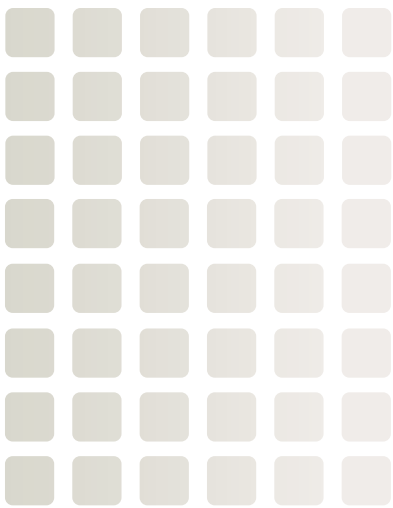
The ideal database cluster interconnect would have very low latency (on the order of microseconds) and very high bandwidth (on the order of multiple gigabits per second) to accommodate both small remote lock message (<512 bytes) and larger cache to cache transfers (>4K).

Traditional Cluster Interconnects

Unfortunately, this kind of interconnect is not available today at attractive prices. Because standards-based interconnects with the necessary latency and bandwidth do not exist, vendors have developed proprietary and expensive interconnects on proprietary platforms (Figure 1). Although these systems have run—large scale, highly-available database applications for years, they are also very expensive. A more attractive alternative would be the commodity economies of scale offered by Intel Architecture Servers. Unfortunately, Intel servers have not traditionally been used for enterprise-class database applications because they haven't scaled to desirable levels with high availability.

Similarly, Ethernet does not scale as a commodity-based server interconnect. An inherently unreliable network, Ethernet runs the TCP protocol to ensure reliable communication. TCP adds a large amount of overhead to the network, including non-useful payloads and processing overhead. This overhead severely limits how bandwidth can be used and burdens hosts with packaging and processing network traffic, stealing valuable CPU cycles. Moreover, network overhead adds a tremendous amount of latency to individual messages. As a result, clusters that use an Ethernet interconnect cannot scale to a level that justifies the cost of its components. Because of this, products such as Oracle gi RAC and IBM DB2 Parallel Edition have not seen wide deployment on Intel-based servers.

The combination of high cost, disappointing scalability, and administrative complexity has prevented clustering from being adopted in the mainstream database environment.



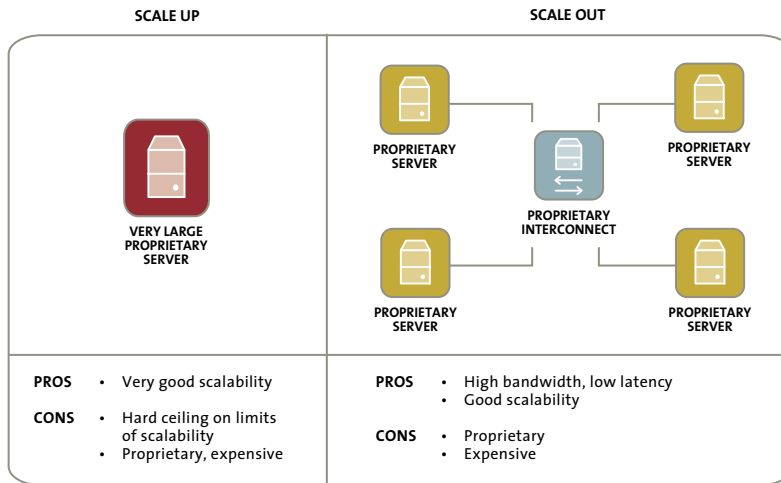


Figure 1: Proprietary Scalability Options

The Emergence of standards-based RDMA Networks

Fortunately, new networking technologies have emerged that can help make database clustering a viable solution. Based on industry standards, these interconnect technologies help build highly-scalable and highly-available clusters of commodity-based Intel servers running Windows and Linux.

InfiniBand is a new technology that offers the ability to dramatically free up CPU cycles by slashing overhead. Speeds range from 10 Gbps to 30 Gbps over a single connection, and it carries not only standard network traffic but also the I/O connectivity for servers. InfiniBand is poised to become the standard data center network for interconnecting servers and I/O resources. This standard has been developed and backed by all major computer server vendors, operating system vendors, and database vendors, as well as a long list of emerging companies.

About RDMA and InfiniBand

InfiniBand is based primarily on a concept known as Remote DMA. RDMA provides a way to reliably communicate between servers—as well as server to I/O—without the need for heavy protocols like TCP/IP (Figure 2). RDMA is built into the lowest levels of network interfaces that can read and write data directly into the memory subsystem of a server or I/O device without the need for a high-overhead network protocol driver to verify integrity and de-multiplex messages to applications. Instead, messages are moved directly into or out of the memory that is owned by an application. Also, communications reliability is built directly into the underlying network protocol. With RDMA, InfiniBand allows servers and applications to reliably transfer small messages across the network with extremely low latency and handle large bulk data transfers with extremely high throughput and minimal burden on the application servers.

Low latency and high bandwidth allow large mission-critical databases to be deployed on commodity platforms—for a fraction of the current cost. Smaller enterprises also finally gain access to enterprise-class database technology that was previously out of reach due to cost.

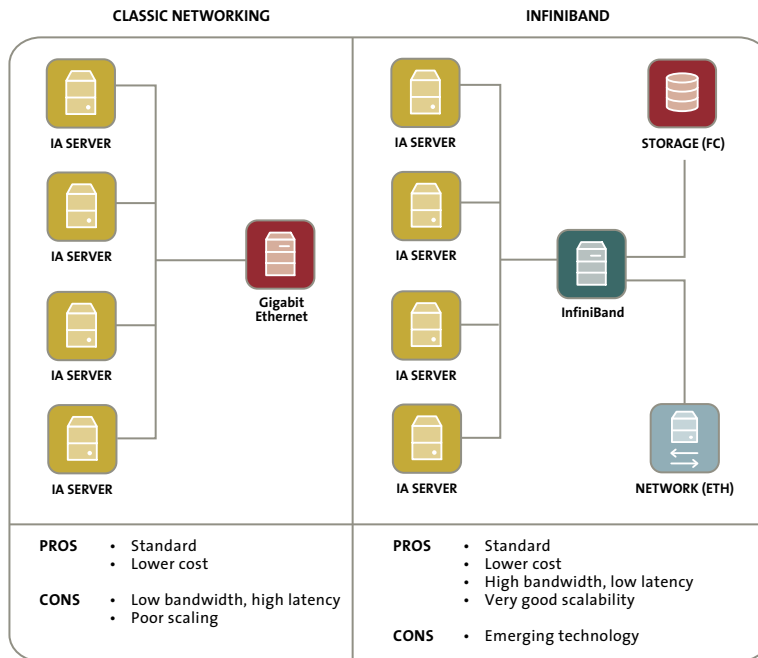


Figure 2: Commodity-Based Scalability Options

InfiniBand for I/O Disaggregation

InfiniBand also offers the ability to disaggregate I/O from servers that use it. By centrally deploying and managing I/O resources for SAN storage and traditional networking, application servers can be deployed statelessly. Instead of servers with local disk drives and redundant Fibre Channel and Ethernet connections, servers will simply have high-speed InfiniBand connections to a switched fabric, which also contains shared, redundant interfaces to Fibre Channel and Ethernet.

Enterprise Databases, Commodity Servers and InfiniBand

InfiniBand has significant support and solves a significant need. Both Oracle and DB2 have stated that they are supporting InfiniBand, and most of the Intel Server vendors have announced their intention to ship servers and I/O solutions which provide direct support for InfiniBand. In the near future, a new generation of servers, databases, and database applications will be available for much lower capital outlay and much lower cost of administration. Next-generation applications will emerge together with stateless servers and shared I/O, finally making it possible to dynamically provision servers, I/O, and storage to applications from a centrally managed location. As InfiniBand drives the adoption of clustered blades and bricks, commodity components will further drive down costs, making scalable, highly-available computing available to the mainstream data center.

© Copyright 2003, Topspin Communications, Inc. All rights reserved. Topspin is a registered trademark and the Topspin logo, TopspinOS, and Topspin Switched Computing System are trademarks of Topspin Communications, Inc. Other company, product, or service names are the property of their respective owners.