Trading Floor Architecture

Executive Overview

Increased competition, higher market data volume, and new regulatory demands are some of the driving forces behind industry changes. Firms are trying to maintain their competitive edge by constantly changing their trading strategies and increasing the speed of trading.

A viable architecture has to include the latest technologies from both network and application domains. It has to be modular to provide a manageable path to evolve each component with minimal disruption to the overall system. Therefore the architecture proposed by this paper is based on a services framework. We examine services such as ultra-low latency messaging, latency monitoring, multicast, computing, storage, data and application virtualization, trading resiliency, trading mobility, and thin client.

The solution to the complex requirements of the next-generation trading platform must be built with a holistic mindset, crossing the boundaries of traditional silos like business and technology or applications and networking.

This document's main goal is to provide guidelines for building an ultra-low latency trading platform while optimizing the raw throughput and message rate for both market data and FIX trading orders.

To achieve this, we are proposing the following latency reduction technologies:

- High speed inter-connect—InfiniBand or 10 Gbps connectivity for the trading cluster
- High-speed messaging bus
- Application acceleration via RDMA without application re-code
- Real-time latency monitoring and re-direction of trading traffic to the path with minimum latency

Industry Trends and Challenges

Next-generation trading architectures have to respond to increased demands for speed, volume, and efficiency. For example, the volume of options market data is expected to double after the introduction of options penny trading in 2007. There are also regulatory demands for best execution, which require handling price updates at rates that approach 1M msg/sec. for exchanges. They also require visibility into the freshness of the data and proof that the client got the best possible execution.

In the short term, speed of trading and innovation are key differentiators. An increasing number of trades are handled by algorithmic trading applications placed as close as possible to the trade execution venue. A challenge with these "black-box" trading engines is that they compound the volume increase by issuing orders only to cancel them and re-submit them. The cause of this behavior is lack of visibility into which venue offers best execution. The human trader is now a "financial engineer," a "quant" (quantitative analyst) with programming skills, who can adjust trading models on the fly. Firms develop new financial instruments like weather derivatives or cross-asset class trades and they need to deploy the new applications quickly and in a scalable fashion.

In the long term, competitive differentiation should come from analysis, not just knowledge. The star traders of tomorrow assume risk, achieve true client insight, and consistently beat the market (source IBM: http://www-935.ibm.com/services/us/imc/pdf/ge510-6270-trader.pdf).

Business resilience has been one main concern of trading firms since September 11, 2001. Solutions in this area range from redundant data centers situated in different geographies and connected to multiple trading venues to virtual trader solutions offering power traders most of the functionality of a trading floor in a remote location.

The financial services industry is one of the most demanding in terms of IT requirements. The industry is experiencing an architectural shift towards Services-Oriented Architecture (SOA), Web services, and virtualization of IT resources. SOA takes advantage of the increase in network speed to enable dynamic binding and virtualization of software components. This allows the creation of new applications without losing the investment in existing systems and infrastructure. The concept has the potential to revolutionize the way integration is done, enabling significant reductions in the complexity and cost of such integration (http://www.gigaspaces.com/download/MerrilLynchGigaSpacesWP.pdf).

Another trend is the consolidation of servers into data center server farms, while trader desks have only KVM extensions and ultra-thin clients (e.g., SunRay and HP blade solutions). High-speed Metro Area Networks enable market data to be multicast between different locations, enabling the virtualization of the trading floor.

High-Level Architecture

Figure 1 depicts the high-level architecture of a trading environment. The ticker plant and the algorithmic trading engines are located in the high performance trading cluster in the firm's data center or at the exchange. The human traders are located in the end-user applications area.

Functionally there are two application components in the enterprise trading environment, publishers and subscribers. The messaging bus provides the communication path between publishers and subscribers.

There are two types of traffic specific to a trading environment:

- Market Data—Carries pricing information for financial instruments, news, and other value-added information such as analytics. It is unidirectional and very latency sensitive, typically delivered over UDP multicast. It is measured in updates/sec. and in Mbps. Market data flows from one or multiple external feeds, coming from market data providers like stock exchanges, data aggregators, and ECNs. Each provider has their own market data format. The data is received by feed handlers, specialized applications which normalize and clean the data and then send it to data consumers, such as pricing engines, algorithmic trading applications, or human traders. Sell-side firms also send the market data to their clients, buy-side firms such as mutual funds, hedge funds, and other asset managers. Some buy-side firms may opt to receive direct feeds from exchanges, reducing latency.

Figure 1   *Trading Architecture for a Buy Side/Sell Side Firm*

AON-Instrumented High Availability Multicast Network
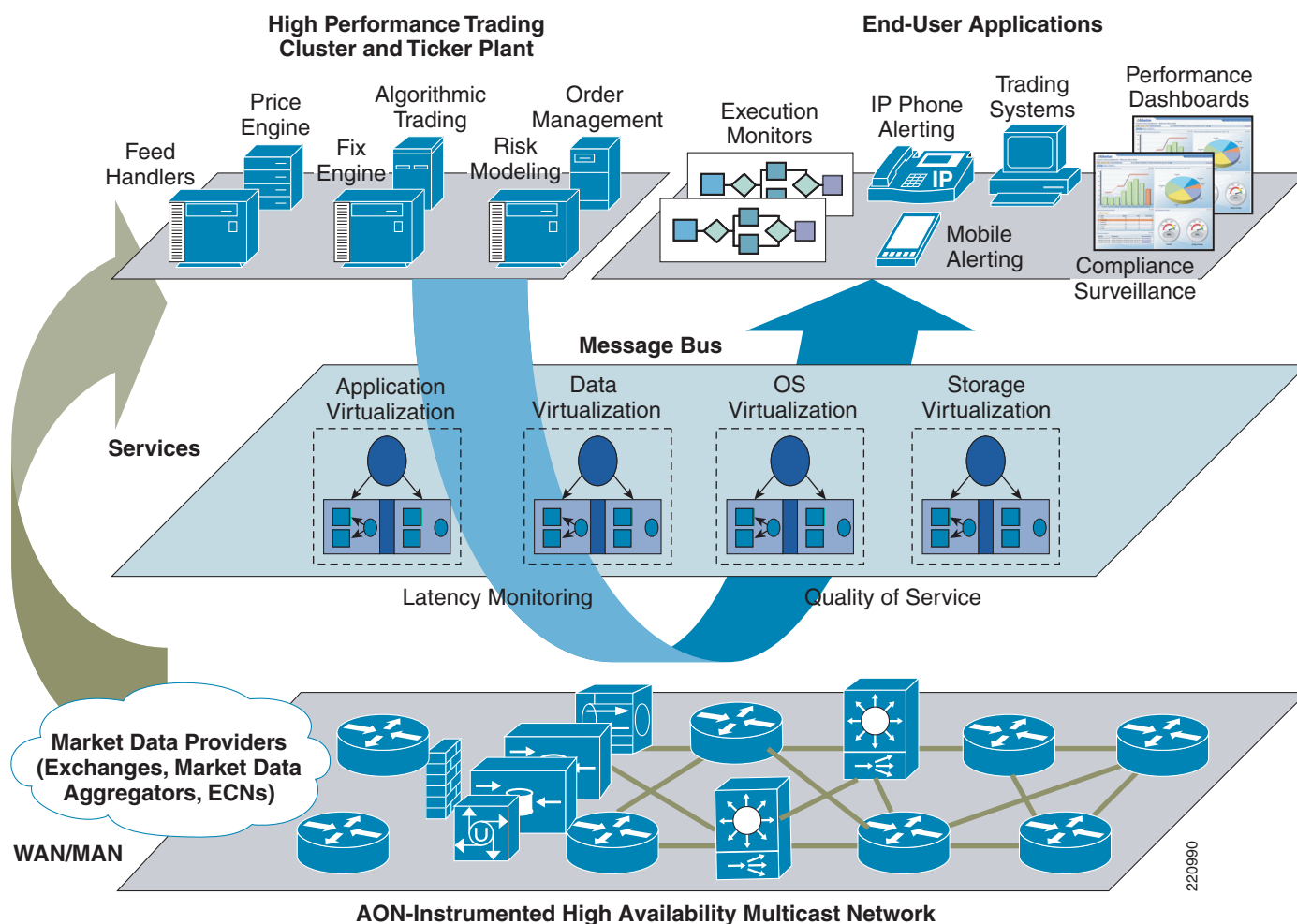
There is no industry standard for market data formats. Each exchange has their proprietary format. Financial content providers such as Reuters and Bloomberg aggregate different sources of market data, normalize it, and add news or analytics. Examples of consolidated feeds are RDF (Reuters Data Feed), RWF (Reuters Wire Format), and Bloomberg Professional Services Data.

To deliver lower latency market data, both vendors have released real-time market data feeds which are less processed and have less analytics:

– RDF-D (Reuters Data Feed-Direct) (http://about.reuters.com/productinfo/datafeeddirect/)

– Bloomberg B-Pipe—With B-Pipe, Bloomberg de-couples their market data feed from their distribution platform because a Bloomberg terminal is not required for get B-Pipe. Wombat and Reuters Feed Handlers have announced support for B-Pipe.

A firm may decide to receive feeds directly from an exchange to reduce latency. The gains in transmission speed can be between 150 milliseconds to 500 milliseconds. These feeds are more complex and more expensive and the firm has to build and maintain their own ticker plant (http://www.financetech.com/featured/showArticle.jhtml?articleID=60404306).

• Trading Orders—This type of traffic carries the actual trades. It is bi-directional and very latency sensitive. It is measured in messages/sec. and Mbps. The orders originate from a buy side or sell side firm and are sent to trading venues like an Exchange or ECN for execution. The most common

format for order transport is FIX (Financial Information eXchange—http://www.fixprotocol.org/). The applications which handle FIX messages are called FIX engines and they interface with order management systems (OMS).

An optimization to FIX is called FAST (Fix Adapted for Streaming), which uses a compression schema to reduce message length and, in effect, reduce latency. FAST is targeted more to the delivery of market data and has the potential to become a standard. FAST can also be used as a compression schema for proprietary market data formats.

To reduce latency, firms may opt to establish Direct Market Access (DMA).

DMA is the automated process of routing a securities order directly to an execution venue, therefore avoiding the intervention by a third-party (http://www.towergroup.com/research/content/glossary.jsp?page=1&glossaryId=383). DMA requires a direct connection to the execution venue.

The messaging bus is middleware software from vendors such as Tibco, 29West, Reuters RMDS, or an open source platform such as AMQP. The messaging bus uses a reliable mechanism to deliver messages. The transport can be done over TCP/IP (TibcoEMS, 29West, RMDS, and AMQP) or UDP/multicast (TibcoRV, 29West, and RMDS). One important concept in message distribution is the "topic stream," which is a subset of market data defined by criteria such as ticker symbol, industry, or a certain basket of financial instruments. Subscribers join topic groups mapped to one or multiple sub-topics in order to receive only the relevant information. In the past, all traders received all market data. At the current volumes of traffic, this would be sub-optimal.

The network plays a critical role in the trading environment. Market data is carried to the trading floor where the human traders are located via a Campus or Metro Area high-speed network. High availability and low latency, as well as high throughput, are the most important metrics.

The high performance trading environment has most of its components in the Data Center server farm. To minimize latency, the algorithmic trading engines need to be located in the proximity of the feed handlers, FIX engines, and order management systems. An alternate deployment model has the algorithmic trading systems located at an exchange or a service provider with fast connectivity to multiple exchanges.

Deployment Models

There are two deployment models for a high performance trading platform. Firms may chose to have a mix of the two:

- Data Center of the trading firm (Figure 2)—This is the traditional model, where a full-fledged trading platform is developed and maintained by the firm with communication links to all the trading venues. Latency varies with the speed of the links and the number of hops between the firm and the venues.

Figure 2   *Traditional Deployment Model*
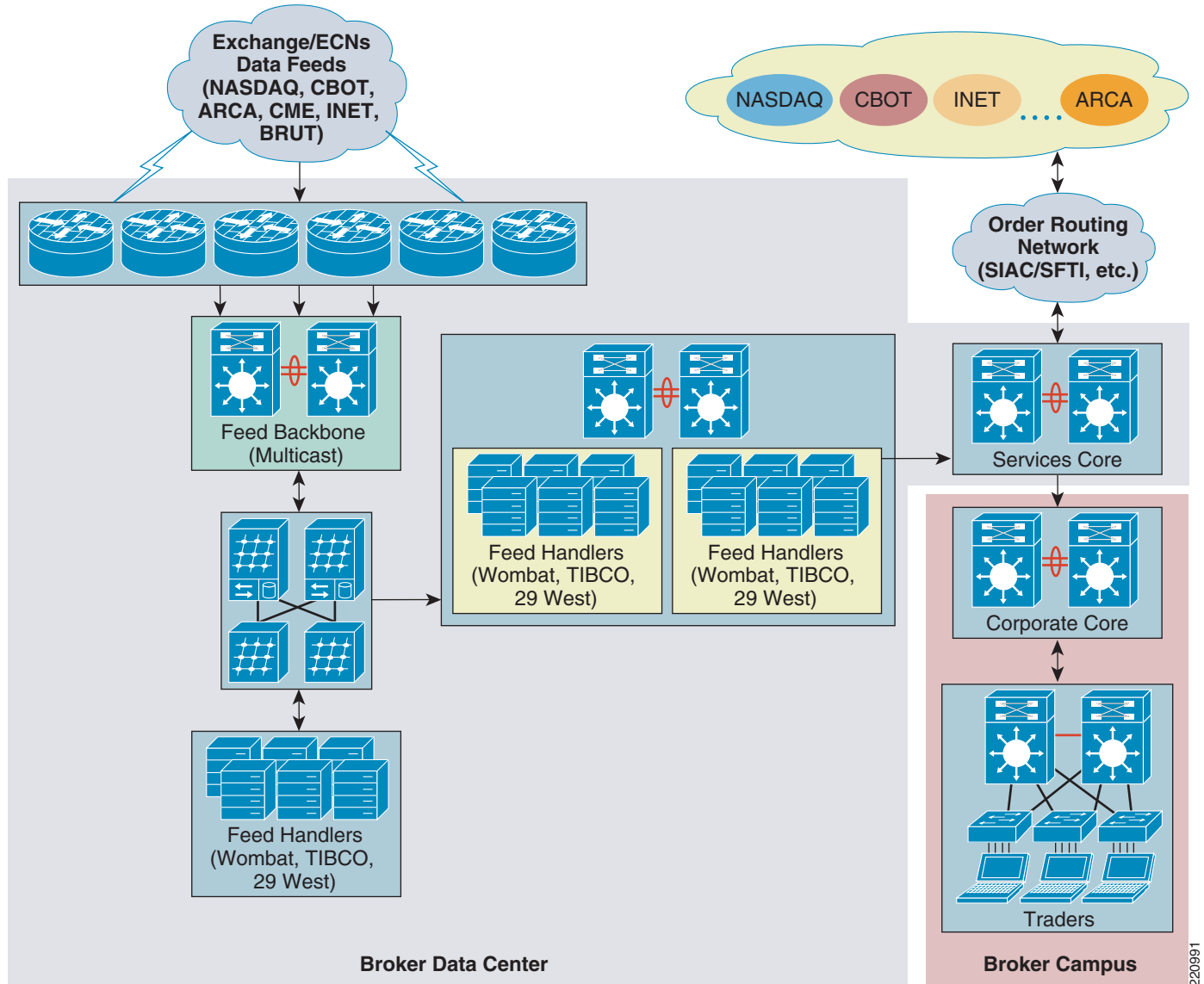
- Co-location at the trading venue (exchanges, financial service providers (FSP)) (Figure 3)

  The trading firm deploys its automated trading platform as close as possible to the execution venues to minimize latency.

Figure 3   Hosted Deployment Model

Services-Oriented Trading Architecture

We are proposing a services-oriented framework for building the next-generation trading architecture. This approach provides a conceptual framework and an implementation path based on modularization and minimization of inter-dependencies.

This framework provides firms with a methodology to:

- Evaluate their current state in terms of services
- Prioritize services based on their value to the business
- Evolve the trading platform to the desired state using a modular approach

The high performance trading architecture relies on the following services, as defined by the services architecture framework represented in Figure 4.

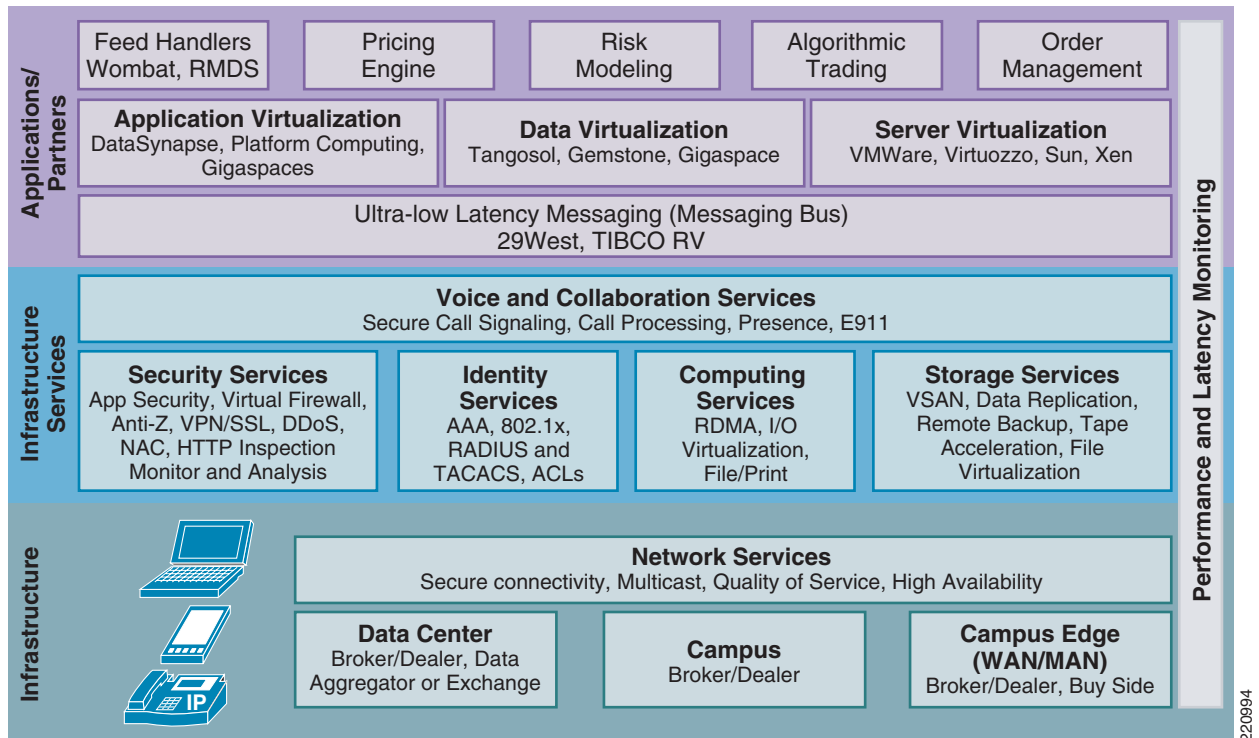*Figure 4*　　　*Service Architecture Framework for High Performance Trading*

Table 1　　　*Service Descriptions and Technologies*

Service Description	Technology
Ultra-low latency messaging	Middleware
Latency monitoring	Instrumentation—appliances, software agents, and router modules
Computing services	OS and I/O virtualization, Remote Direct Memory Access (RDMA), TCP Offload Engines (TOE)
Application virtualization	Middleware which parallelizes application processing
Data virtualization	Middleware which speeds-up data access for applications, e.g., in-memory caching
Multicast service	Hardware-assisted multicast replication through-out the network; multicast Layer 2 and Layer 3 optimizations
Storage services	Virtualization of storage hardware (VSANs), data replication, remote backup, and file virtualization
Trading resilience and mobility	Local and site load balancing and high availability campus networks
Wide Area application services	Acceleration of applications over a WAN connection for traders residing off-campus
Thin client service	De-coupling of the computing resources from the end-user facing terminals

Ultra-Low Latency Messaging Service

This service is provided by the messaging bus, which is a software system that solves the problem of connecting many-to-many applications. The system consists of:

- A set of pre-defined message schemas

- A set of common command messages

- A shared application infrastructure for sending the messages to recipients. The shared infrastructure can be based on a message broker or on a publish/subscribe model.

The key requirements for the next-generation messaging bus are (source 29West):

- Lowest possible latency (e.g., less than 100 microseconds)

- Stability under heavy load (e.g., more than 1.4 million msg/sec.)

- Control and flexibility (rate control and configurable transports)

There are efforts in the industry to standardize the messaging bus. Advanced Message Queueing Protocol (AMQP) is an example of an open standard championed by J.P. Morgan Chase and supported by a group of vendors such as Cisco, Envoy Technologies, Red Hat, TWIST Process Innovations, Iona, 29West, and iMatix. Two of the main goals are to provide a more simple path to inter-operability for applications written on different platforms and modularity so that the middleware can be easily evolved.

In very general terms, an AMQP server is analogous to an E-mail server with each exchange acting as a message transfer agent and each message queue as a mailbox. The bindings define the routing tables in each transfer agent. Publishers send messages to individual transfer agents, which then route the messages into mailboxes. Consumers take messages from mailboxes, which creates a powerful and flexible model that is simple (source: http://www.amqp.org/tikiwiki/tiki-index.php?page=OpenApproach#Why_AMQP_).

Latency Monitoring Service

The main requirements for this service are:

- Sub-millisecond granularity of measurements

- Near-real time visibility without adding latency to the trading traffic

- Ability to differentiate application processing latency from network transit latency

- Ability to handle high message rates

- Provide a programmatic interface for trading applications to receive latency data, thus enabling algorithmic trading engines to adapt to changing conditions

- Correlate network events with application events for troubleshooting purposes

Latency can be defined as the time interval between when a trade order is sent and when the same order is acknowledged and acted upon by the receiving party.

Addressing the latency issue is a complex problem, requiring a holistic approach that identifies all sources of latency and applies different technologies at different layers of the system.

Figure 5 depicts the variety of components that can introduce latency at each layer of the OSI stack. It also maps each source of latency with a possible solution and a monitoring solution. This layered approach can give firms a more structured way of attacking the latency issue, whereby each component can be thought of as a service and treated consistently across the firm.

Maintaining an accurate measure of the dynamic state of this time interval across alternative routes and destinations can be of great assistance in tactical trading decisions. The ability to identify the exact location of delays, whether in the customer's edge network, the central processing hub, or the transaction application level, significantly determines the ability of service providers to meet their trading service-level agreements (SLAs). For buy-side and sell-side forms, as well as for market-data syndicators, the quick identification and removal of bottlenecks translates directly into enhanced trade opportunities and revenue.

Figure 5   *Latency Management Architecture*

	Sources of Latency	Latency Reduction Solutions		Monitoring	
Application Layer	Application Software (OS, App) Program Trading, Ticker capture, Smart Order Routing, Analytical	MPI, SDP	Direct Market Access	Cisco Application Analysis	
	Application Hardware (CPU, Memory, Storage)	Grid computing, SAN, RDMA, In-Memory Caching			
Transaction Layer	Market Data DistributionTriarch, Tibco/RV, RMDS	Trade Orders FIX	Acceleration Appliances	FIX Adapted for Streaming (FAST)	Cisco AON	Trading Metrics Analysis Engine
			Emerging MD platforms			
Network Layer	Security (Firewall, Identity Server, Encryption)	HW Assisted Security	HW Assisted Multicast Replication	Security monitoring	
	TCP/IP Overhead	Multicast replication	TCP Optimization	QoS Policy	Cisco Multicast Manager	QoS Policy Mgr.
Interface Layer	Buffering, serialization, fragmentation	CBWFQ, LLQ	Serialization Optimization	IP SLA	Cisco Bandwidth Quality Analyzer
	Physical Layer (Ethernet, WAN)	Infiniband, Low-latency Ethernet Infiniband over WAN, Fiber Optics		RMON	

220995

Cisco Low-Latency Monitoring Tools

Traditional network monitoring tools operate with minutes or seconds granularity. Next-generation trading platforms, especially those supporting algorithmic trading, require latencies less than 5 ms and extremely low levels of packet loss. On a Gigabit LAN, a 100 ms microburst can cause 10,000 transactions to be lost or excessively delayed.

Cisco offers its customers a choice of tools to measure latency in a trading environment:

- Bandwidth Quality Manager (BQM) (OEM from Corvil)
- Cisco AON-based Financial Services Latency Monitoring Solution (FSMS)
- Cisco IP SLA

Bandwidth Quality Manager

Bandwidth Quality Manager (BQM) 4.0 is a next-generation network application performance management product that enables customers to monitor and provision their network for controlled levels of latency and loss performance. While BQM is not exclusively targeted at trading networks, its microsecond visibility combined with intelligent bandwidth provisioning features make it ideal for these demanding environments.

Cisco BQM 4.0 implements a broad set of patented and patent-pending traffic measurement and network analysis technologies that give the user unprecedented visibility and understanding of how to optimize the network for maximum application performance.

Cisco BQM is now supported on the product family of Cisco Application Deployment Engine (ADE). The Cisco ADE product family is the platform of choice for Cisco network management applications.

BQM Benefits

Cisco BQM micro-visibility is the ability to detect, measure, and analyze latency, jitter, and loss inducing traffic events down to microsecond levels of granularity with per packet resolution. This enables Cisco BQM to detect and determine the impact of traffic events on network latency, jitter, and loss. Critical for trading environments is that BQM can support latency, loss, and jitter measurements one-way for both TCP and UDP (multicast) traffic. This means it reports seamlessly for both trading traffic and market data feeds.

BQM allows the user to specify a comprehensive set of thresholds (against microburst activity, latency, loss, jitter, utilization, etc.) on all interfaces. BQM then operates a background rolling packet capture. Whenever a threshold violation or other potential performance degradation event occurs, it triggers Cisco BQM to store the packet capture to disk for later analysis. This allows the user to examine in full detail both the application traffic that was affected by performance degradation ("the victims") and the traffic that caused the performance degradation ("the culprits"). This can significantly reduce the time spent diagnosing and resolving network performance issues.

BQM is also able to provide detailed bandwidth and quality of service (QoS) policy provisioning recommendations, which the user can directly apply to achieve desired network performance.

BQM Measurements Illustrated

To understand the difference between some of the more conventional measurement techniques and the visibility provided by BQM, we can look at some comparison graphs. In the first set of graphs (Figure 6 and Figure 7), we see the difference between the latency measured by BQM's Passive Network Quality Monitor (PNQM) and the latency measured by injecting ping packets every 1 second into the traffic stream.

In Figure 6, we see the latency reported by 1-second ICMP ping packets for real network traffic (it is divided by 2 to give an estimate for the one-way delay). It shows the delay comfortably below about 5ms for almost all of the time.

Figure 6   *Latency Reported by 1-Second ICMP Ping Packets for Real Network Traffic*
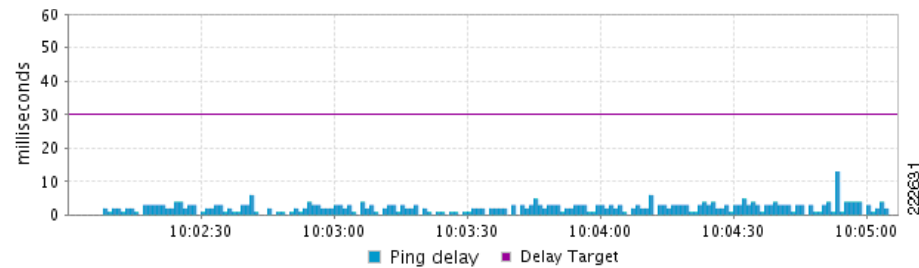
In Figure 7, we see the latency reported by PNQM for the same traffic at the same time. Here we see that by measuring the one-way latency of the actual application packets, we get a radically different picture. Here the latency is seen to be hovering around 20 ms, with occasional bursts far higher. The explanation is that because ping is sending packets only every second, it is completely missing most of the application traffic latency. In fact, ping results typically only indicate round trip propagation delay rather than realistic application latency across the network.

Figure 7   *Latency Reported by PNQM for Real Network Traffic*
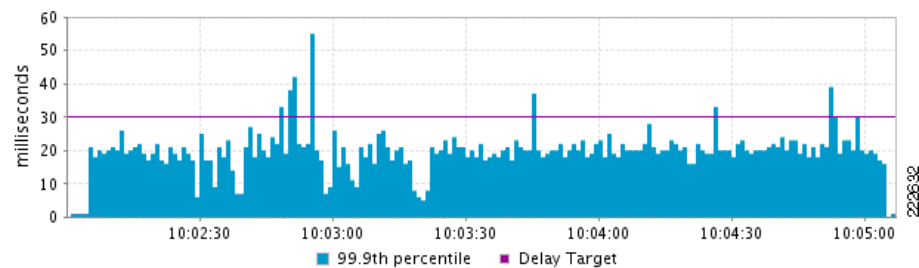

In the second example (Figure 8), we see the difference in reported link load or saturation levels between a 5-minute average view and a 5 ms microburst view (BQM can report on microbursts down to about 10-100 nanosecond accuracy). The green line shows the average utilization at 5-minute averages to be low, maybe up to 5 Mbits/s. The dark blue plot shows the 5ms microburst activity reaching between 75 Mbits/s and 100 Mbits/s, the LAN speed effectively. BQM shows this level of granularity for all applications and it also gives clear provisioning rules to enable the user to control or neutralize these microbursts.

Figure 8   *Difference in Reported Link Load Between a 5-Minute Average View and a 5 ms Microburst View*
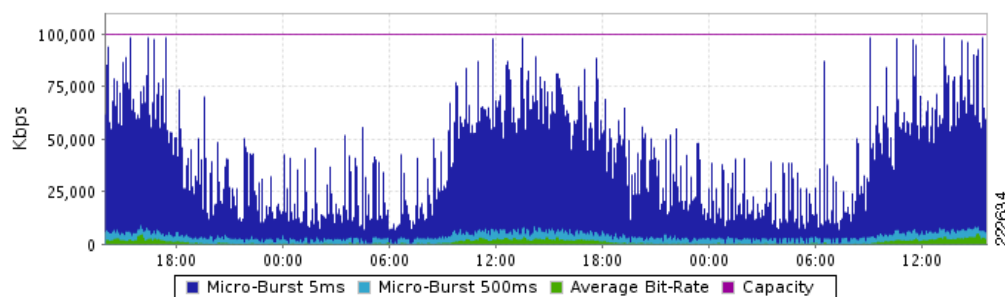
BQM Deployment in the Trading Network

Figure 9 shows a typical BQM deployment in a trading network.

Figure 9   *Typical BQM Deployment in a Trading Network*
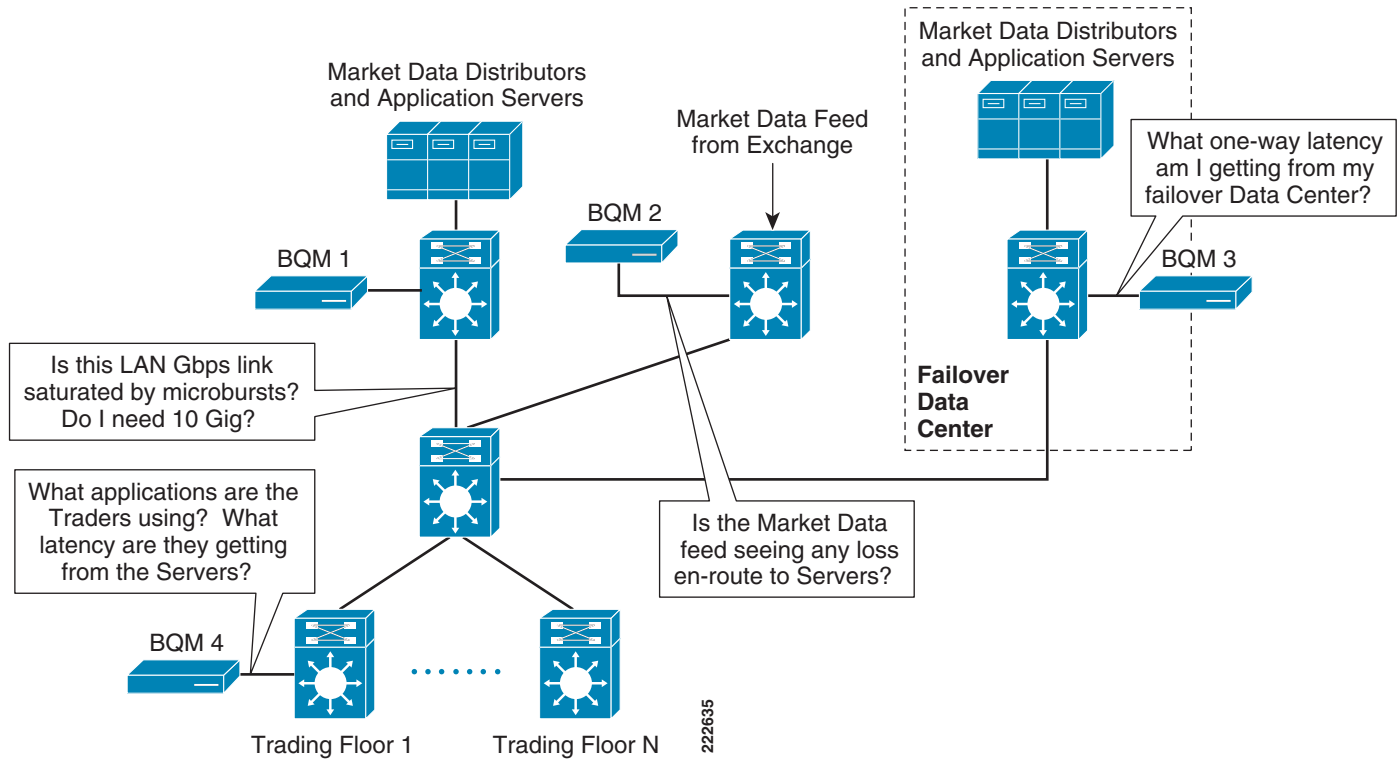
BQM can then be used to answer these types of questions:

- Are any of my Gigabit LAN core links saturated for more than X milliseconds? Is this causing loss? Which links would most benefit from an upgrade to Etherchannel or 10 Gigabit speeds?

- What application traffic is causing the saturation of my 1 Gigabit links?

- Is any of the market data experiencing end-to-end loss?

- How much additional latency does the failover data center experience? Is this link sized correctly to deal with microbursts?

- Are my traders getting low latency updates from the market data distribution layer? Are they seeing any delays greater than X milliseconds?

Being able to answer these questions simply and effectively saves time and money in running the trading network.

BQM is an essential tool for gaining visibility in market data and trading environments. It provides granular end-to-end latency measurements in complex infrastructures that experience high-volume data movement. Effectively detecting microbursts in sub-millisecond levels and receiving expert analysis on a particular event is invaluable to trading floor architects. Smart bandwidth provisioning recommendations, such as sizing and what-if analysis, provide greater agility to respond to volatile market conditions. As the explosion of algorithmic trading and increasing message rates continues, BQM, combined with its QoS tool, provides the capability of implementing QoS policies that can protect critical trading applications.

Cisco Financial Services Latency Monitoring Solution

Cisco and Trading Metrics have collaborated on latency monitoring solutions for FIX order flow and market data monitoring. Cisco AON technology is the foundation for a new class of network-embedded products and solutions that help merge intelligent networks with application infrastructure, based on either service-oriented or traditional architectures. Trading Metrics is a leading provider of analytics software for network infrastructure and application latency monitoring purposes (http://www.tradingmetrics.com/).

The Cisco AON Financial Services Latency Monitoring Solution (FSMS) correlated two kinds of events at the point of observation:

- Network events correlated directly with coincident application message handling
- Trade order flow and matching market update events

Using time stamps asserted at the point of capture in the network, real-time analysis of these correlated data streams permits precise identification of bottlenecks across the infrastructure while a trade is being executed or market data is being distributed. By monitoring and measuring latency early in the cycle, financial companies can make better decisions about which network service—and which intermediary, market, or counterparty—to select for routing trade orders. Likewise, this knowledge allows more streamlined access to updated market data (stock quotes, economic news, etc.), which is an important basis for initiating, withdrawing from, or pursuing market opportunities.

The components of the solution are:

- AON hardware in three form factors:
  - AON Network Module for Cisco 2600/2800/3700/3800 routers
  - AON Blade for the Cisco Catalyst 6500 series
  - AON 8340 Appliance
- AON software
- Trading Metrics M&A 2.0 software, which provides the monitoring and alerting application, displays latency graphs on a dashboard, and issues alerts when slowdowns occur (http://www.tradingmetrics.com/TM_brochure.pdf).

Figure 10 AON-Based FIX Latency Monitoring
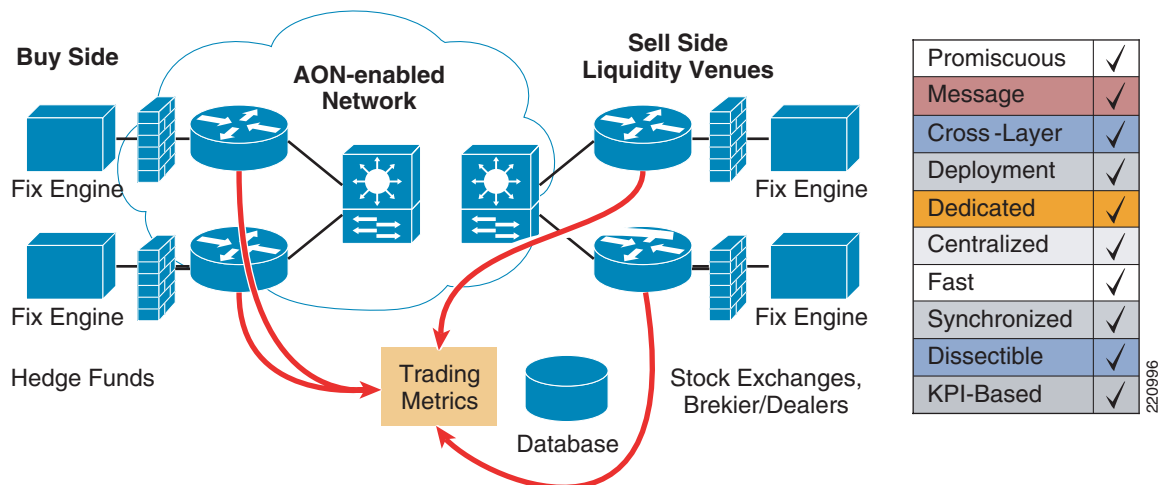
Cisco IP SLA

Cisco IP SLA is an embedded network management tool in Cisco IOS which allows routers and switches to generate synthetic traffic streams which can be measured for latency, jitter, packet loss, and other criteria (http://www.cisco.com/go/ipsla).

Two key concepts are the source of the generated traffic and the target. Both of these run an IP SLA "responder," which has the responsibility to timestamp the control traffic before it is sourced and returned by the target (for a round trip measurement). Various traffic types can be sourced within IP SLA and they are aimed at different metrics and target different services and applications. The UDP jitter operation is used to measure one-way and round-trip delay and report variations. As the traffic is time stamped on both sending and target devices using the responder capability, the round trip delay is characterized as the delta between the two timestamps.

A new feature was introduced in IOS 12.3(14)T, IP SLA Sub Millisecond Reporting, which allows for timestamps to be displayed with a resolution in microseconds, thus providing a level of granularity not previously available. This new feature has now made IP SLA relevant to campus networks where network latency is typically in the range of 300-800 microseconds and the ability to detect trends and spikes (brief trends) based on microsecond granularity counters is a requirement for customers engaged in time-sensitive electronic trading environments.

As a result, IP SLA is now being considered by significant numbers of financial organizations as they are all faced with requirements to:

- Report baseline latency to their users
- Trend baseline latency over time
- Respond quickly to traffic bursts that cause changes in the reported latency

Sub-millisecond reporting is necessary for these customers, since many campus and backbones are currently delivering under a second of latency across several switch hops. Electronic trading environments have generally worked to eliminate or minimize all areas of device and network latency to deliver rapid order fulfillment to the business. Reporting that network response times are "just under one millisecond" is no longer sufficient; the granularity of latency measurements reported across a network segment or backbone need to be closer to 300-800 micro-seconds with a degree of resolution of 100 μseconds.

IP SLA recently added support for IP multicast test streams, which can measure market data latency.

A typical network topology is shown in Figure 11 with the IP SLA shadow routers, sources, and responders.

Figure 11   *IP SLA Deployment*
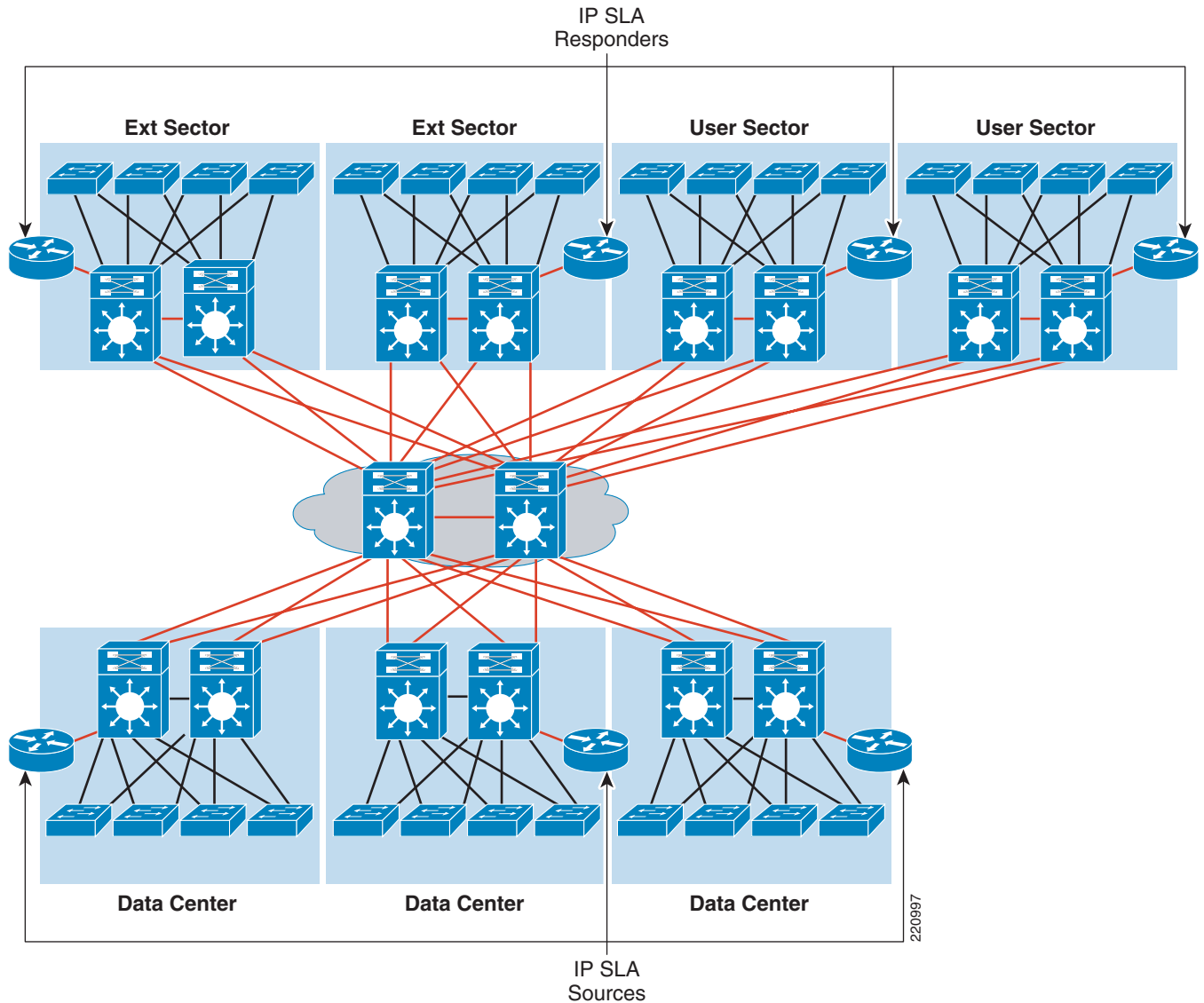
Computing Services

Computing services cover a wide range of technologies with the goal of eliminating memory and CPU bottlenecks created by the processing of network packets. Trading applications consume high volumes of market data and the servers have to dedicate resources to processing network traffic instead of application processing.

The problems:

- Transport processing—At high speeds, network packet processing can consume a significant amount of server CPU cycles and memory. An established rule of thumb states that 1Gbps of network bandwidth requires 1 GHz of processor capacity (source Intel white paper on I/O acceleration http://www.intel.com/technology/ioacceleration/306517.pdf).

- Intermediate buffer copying—In a conventional network stack implementation, data needs to be copied by the CPU between network buffers and application buffers. This overhead is worsened by the fact that memory speeds have not kept up with increases in CPU speeds. For example, processors like the Intel Xeon are approaching 4 GHz, while RAM chips hover around 400MHz (for DDR 3200 memory) (source Intel http://www.intel.com/technology/ioacceleration/306517.pdf).

- Context switching—Every time an individual packet needs to be processed, the CPU performs a context switch from application context to network traffic context. This overhead could be reduced if the switch would occur only when the whole application buffer is complete.

Figure 12   Sources of Overhead in Data Center Servers

Sources of Overhead in Server Networking	CPU Overhead
Transport Processing	40%
Intermediate Buffer Copying	20%
Application Context Switches	40%
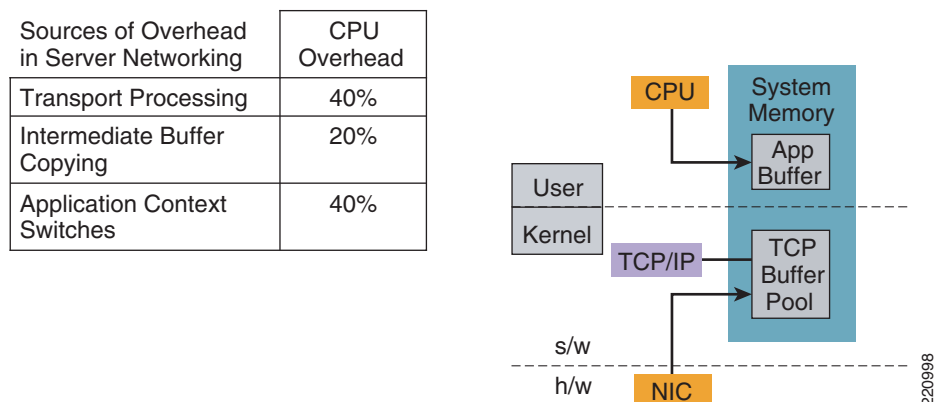
The solutions:

- TCP Offload Engine (TOE)—Offloads transport processor cycles to the NIC. Moves TCP/IP protocol stack buffer copies from system memory to NIC memory.

- Remote Direct Memory Access (RDMA)—Enables a network adapter to transfer data directly from application to application without involving the operating system. Eliminates intermediate and application buffer copies (memory bandwidth consumption).

- Kernel bypass —Direct user-level access to hardware. Dramatically reduces application context switches.

Figure 13   RDMA and Kernel Bypass
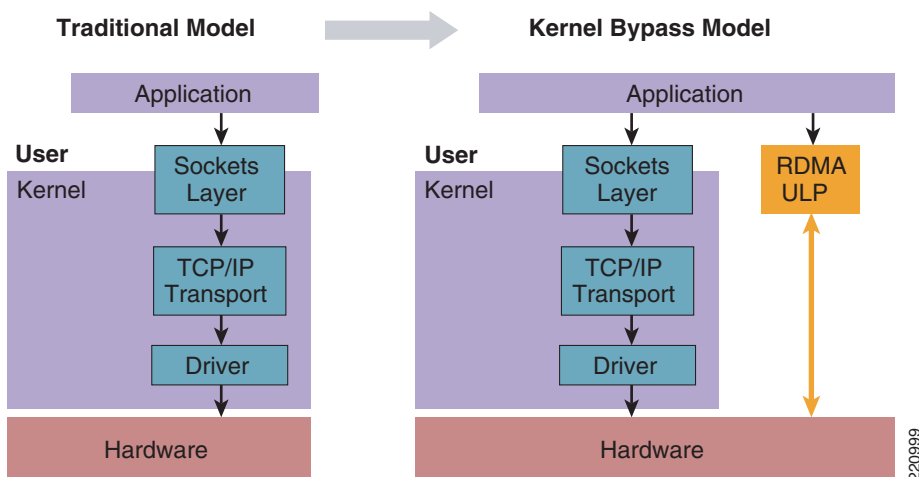
InfiniBand is a point-to-point (switched fabric) bidirectional serial communication link which implements RDMA, among other features. Cisco offers an InfiniBand switch, the Server Fabric Switch (SFS):
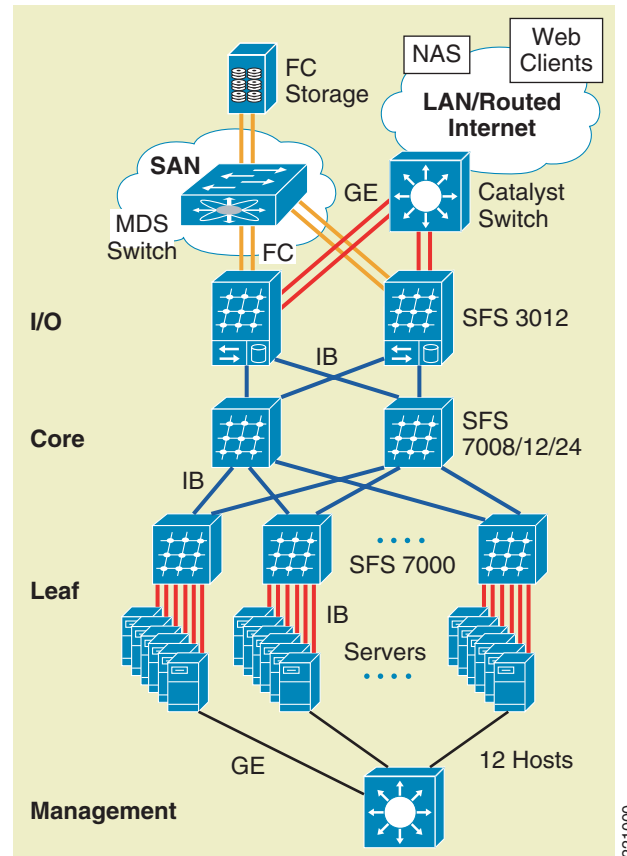
http://www.cisco.com/application/pdf/en/us/guest/netsol/ns500/c643/cdccont_0900aecd804c35cb.pdf.

Figure 14   Typical SFS Deployment

Trading applications benefit from the reduction in latency and latency variability, as proved by a test performed with the Cisco SFS and Wombat Feed Handlers by Stac Research:

http://www.stacresearch.com/index.php?option=com_content&task=view&id=36&Itemid=33

Application Virtualization Service

De-coupling the application from the underlying OS and server hardware enables them to run as network services. One application can be run in parallel on multiple servers, or multiple applications can be run on the same server, as the best resource allocation dictates. This decoupling enables better load balancing and disaster recovery for business continuance strategies. The process of re-allocating computing resources to an application is dynamic. Using an application virtualization system like Data Synapse's GridServer, applications can migrate, using pre-configured policies, to under-utilized servers in a supply-matches-demand process (http://www.networkworld.com/supp/2005/ndc1/022105virtual.html?page=2).

There are many business advantages for financial firms who adopt application virtualization:

- Faster time to market for new products and services

- Faster integration of firms following merger and acquisition activity
- Increased application availability
- Better workload distribution, which creates more "head room" for processing spikes in trading volume
- Operational efficiency and control
- Reduction in IT complexity

Currently, application virtualization is not used in the trading front-office. One use-case is risk modeling, like Monte Carlo simulations. As the technology evolves, it is conceivable that some the trading platforms will adopt it.

Data Virtualization Service

To effectively share resources across distributed enterprise applications, firms must be able to leverage data across multiple sources in real-time while ensuring data integrity. With solutions from data virtualization software vendors such as Gemstone or Tangosol (now Oracle), financial firms can access heterogeneous sources of data as a single system image that enables connectivity between business processes and unrestrained application access to distributed caching. The net result is that all users have instant access to these data resources across a distributed network (http://www.gridtoday.com/03/0210/101061.html).

This is called a data grid and is the first step in the process of creating what Gartner calls Extreme Transaction Processing (XTP) (http://www.gartner.com/DisplayDocument?ref=g_search&id=500947). Technologies such as data and applications virtualization enable financial firms to perform real-time complex analytics, event-driven applications, and dynamic resource allocation.

One example of data virtualization in action is a global order book application. An order book is the repository of active orders that is published by the exchange or other market makers. A global order book aggregates orders from around the world from markets that operate independently. The biggest challenge for the application is scalability over WAN connectivity because it has to maintain state. Today's data grids are localized in data centers connected by Metro Area Networks (MAN). This is mainly because the applications themselves have limits—they have been developed without the WAN in mind.

Figure 15   *GemStone GemFire Distributed Caching*
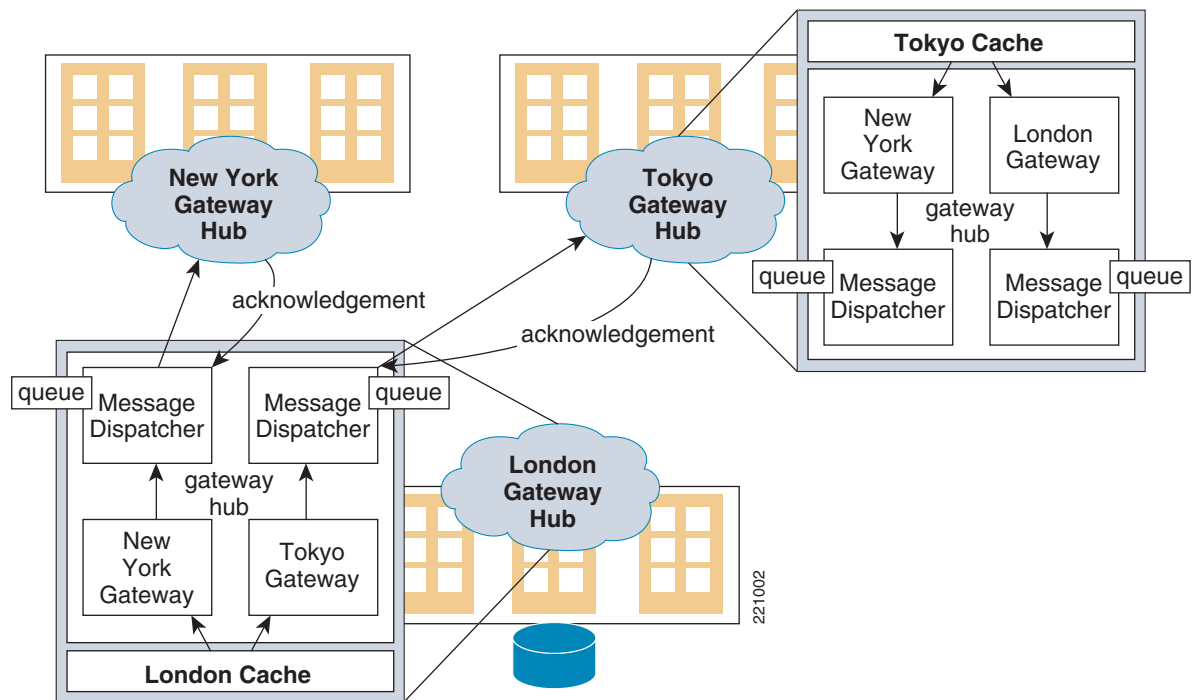
Before data virtualization, applications used database clustering for failover and scalability. This solution is limited by the performance of the underlying database. Failover is slower because the data is committed to disc. With data grids, the data which is part of the active state is cached in memory, which reduces drastically the failover time. Scaling the data grid means just adding more distributed resources, providing a more deterministic performance compared to a database cluster.

Multicast Service

Market data delivery is a perfect example of an application that needs to deliver the same data stream to hundreds and potentially thousands of end users. Market data services have been implemented with TCP or UDP broadcast as the network layer, but those implementations have limited scalability. Using TCP requires a separate socket and sliding window on the server for each recipient. UDP broadcast requires a separate copy of the stream for each destination subnet. Both of these methods exhaust the resources of the servers and the network. The server side must transmit and service each of the streams individually, which requires larger and larger server farms. On the network side, the required bandwidth for the application increases in a linear fashion. For example, to send a 1 Mbps stream to 1000recipients using TCP requires 1 Gbps of bandwidth.

IP multicast is the only way to scale market data delivery. To deliver a 1 Mbps stream to 1000 recipients, IP multicast would require 1 Mbps. The stream can be delivered by as few as two servers—one primary and one backup for redundancy.

There are two main phases of market data delivery to the end user. In the first phase, the data stream must be brought from the exchange into the brokerage's network. Typically the feeds are terminated in a data center on the customer premise. The feeds are then processed by a feed handler, which may normalize the data stream into a common format and then republish into the application messaging servers in the data center.

The second phase involves injecting the data stream into the application messaging bus which feeds the core infrastructure of the trading applications. The large brokerage houses have thousands of applications that use the market data streams for various purposes, such as live trades, long term trending, arbitrage, etc. Many of these applications listen to the feeds and then republish their own analytical and derivative information. For example, a brokerage may compare the prices of CSCO to the option prices of CSCO on another exchange and then publish ratings which a different application may monitor to determine how much they are out of synchronization.

Figure 16   Market Data Distribution Players
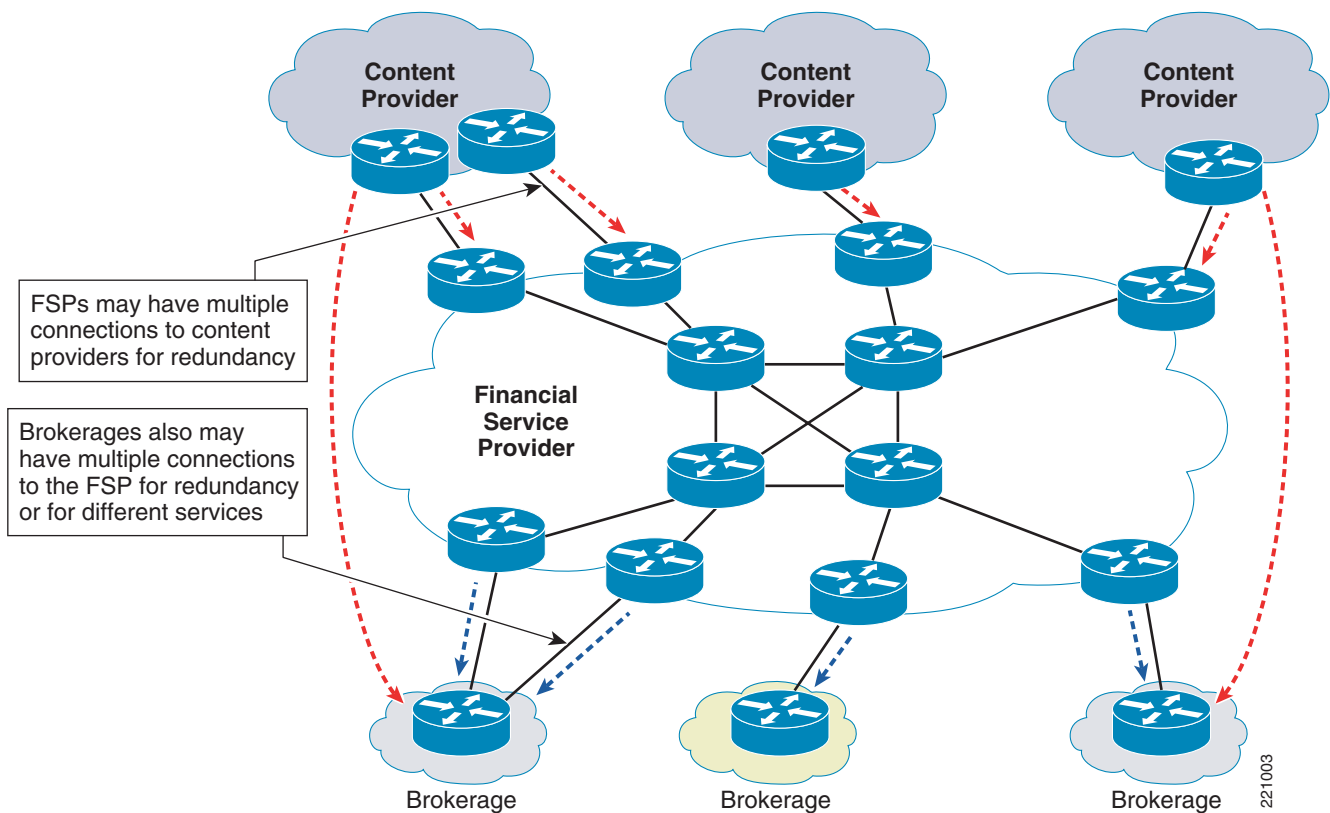
The delivery of these data streams is typically over a reliable multicast transport protocol, traditionally Tibco Rendezvous. Tibco RV operates in a publish and subscribe environment. Each financial instrument is given a subject name, such as CSCO.last. Each application server can request the individual instruments of interest by their subject name and receive just a that subset of the information. This is called subject-based forwarding or filtering. Subject-based filtering is patented by Tibco.

A distinction should be made between the first and second phases of market data delivery. The delivery of market data from the exchange to the brokerage is mostly a one-to-many application. The only exception to the unidirectional nature of market data may be retransmission requests, which are usually sent using unicast. The trading applications, however, are definitely many-to-many applications and may interact with the exchanges to place orders.

Figure 17   *Market Data Architecture*
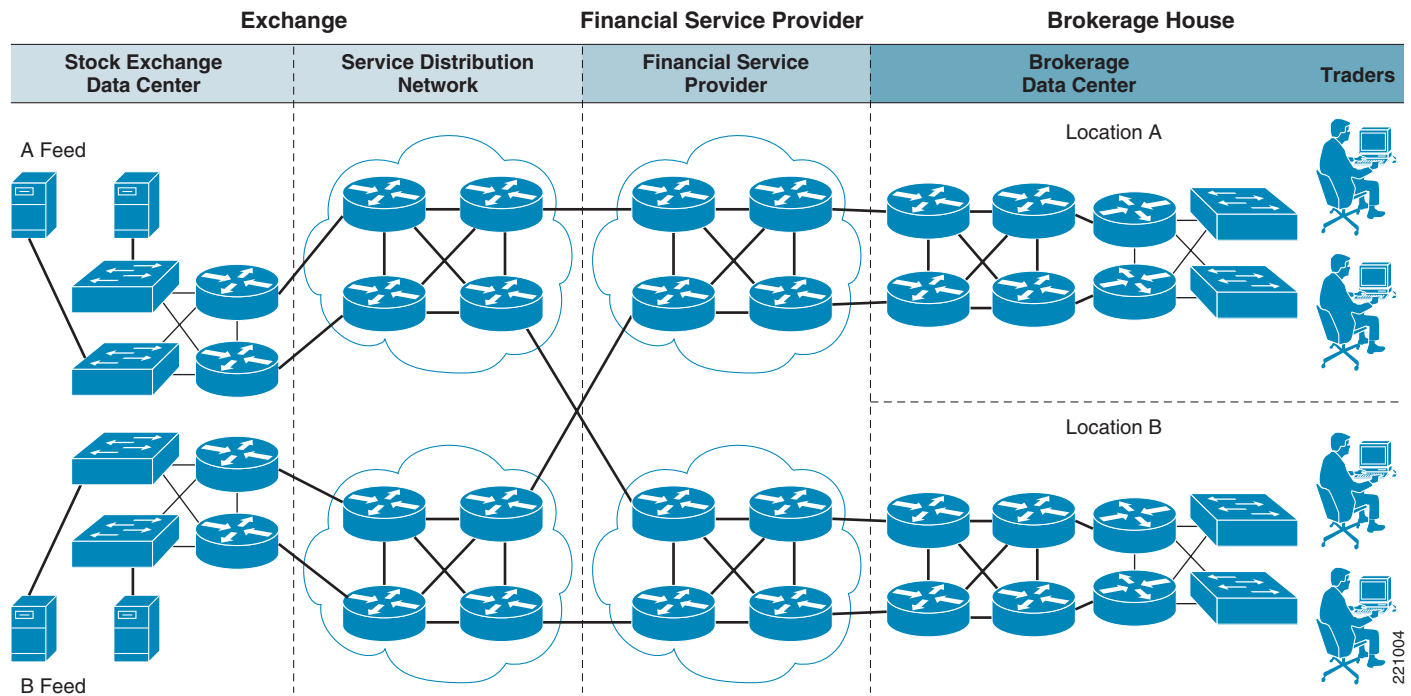
Design Issues

Number of Groups/Channels to Use

Many application developers consider using thousand of multicast groups to give them the ability to divide up products or instruments into small buckets. Normally these applications send many small messages as part of their information bus. Usually several messages are sent in each packet that are received by many users. Sending fewer messages in each packet increases the overhead necessary for each message.

In the extreme case, sending only one message in each packet quickly reaches the point of diminishing returns—there is more overhead sent than actual data. Application developers must find a reasonable compromise between the number of groups and breaking up their products into logical buckets.

Consider, for example, the Nasdaq Quotation Dissemination Service (NQDS). The instruments are broken up alphabetically:

```
NQDS (A-E)   224.3.0.18
NQDS (F-N)   224.3.0.20
NQDS (O-Z)   224.3.0.22
```

Another example is the Nasdaq Totalview service, broken up this way:

```
Data Channel           Primary Groups    Backup Groups
---------------------  --------------    -------------
NASDAQ TotalView (A)     224.0.17.32       224.0.17.35
NASDAQ TotalView (B-C)   224.0.17.48       224.0.17.49
NASDAQ TotalView (D-F)   224.0.17.50       224.0.17.51
NASDAQ TotalView (G-K)   224.0.17.52       224.0.17.53
NASDAQ TotalView (L-N)   224.0.17.54       224.0.17.55
NASDAQ TotalView (O-Q)   224.0.17.56       224.0.17.57
NASDAQ TotalView (R-S)   224.0.17.58       224.0.17.59
```

```
NASDAQ TotalView (T-Z)    224.0.17.60      224.0.17.61
```

This approach allows for straight forward network/application management, but does not necessarily allow for optimized bandwidth utilization for most users. A user of NQDS that is interested in technology stocks, and would like to subscribe to just CSCO and INTL, would have to pull down all the data for the first two groups of NQDS. Understanding the way users pull down the data and then organize it into appropriate logical groups optimizes the bandwidth for each user.

In many market data applications, optimizing the data organization would be of limited value. Typically customers bring in all data into a few machines and filter the instruments. Using more groups is just more overhead for the stack and does not help the customers conserve bandwidth. Another approach might be to keep the groups down to a minimum level and use UDP port numbers to further differentiate if necessary. The other extreme would be to use just one multicast group for the entire application and then have the end user filter the data. In some situations this may be sufficient.

Intermittent Sources

A common issue with market data applications are servers that send data to a multicast group and then go silent for more than 3.5 minutes. These intermittent sources may cause trashing of state on the network and can introduce packet loss during the window of time when soft state and then hardware shorts are being created.

PIM-Bidir or PIM-SSM

The first and best solution for intermittent sources is to use PIM-Bidir for many-to-many applications and PIM-SSM for one-to-many applications.

Both of these optimizations of the PIM protocol do not have any data-driven events in creating forwarding state. That means that as long as the receivers are subscribed to the streams, the network has the forwarding state created in the hardware switching path.

Intermittent sources are not an issue with PIM-Bidir and PIM-SSM.

Null Packets

In PIM-SM environments a common method to make sure forwarding state is created is to send a burst of null packets to the multicast group before the actual data stream. The application must efficiently ignore these null data packets to ensure it does not affect performance. The sources must only send the burst of packets if they have been silent for more than 3 minutes. A good practice is to send the burst if the source is silent for more than a minute. Many financials send out an initial burst of traffic in the morning and then all well-behaved sources do not have problems.

Periodic Keepalives or Heartbeats

An alternative approach for PIM-SM environments is for sources to send periodic heartbeat messages to the multicast groups. This is a similar approach to the null packets, but the packets can be sent on a regular timer so that the forwarding state never expires.

S,G Expiry Timer

Finally, Cisco has made a modification to the operation of the S,G expiry timer in IOS. There is now a CLI knob to allow the state for a S,G to stay alive for hours without any traffic being sent. The (S,G) expiry timer is configurable. This approach should be considered a workaround until PIM-Bidir or PIM-SSM is deployed or the application is fixed.

RTCP Feedback

A common issue with real time voice and video applications that use RTP is the use of RTCP feedback traffic. Unnecessary use of the feedback option can create excessive multicast state in the network. If the RTCP traffic is not required by the application it should be avoided.

Fast Producers and Slow Consumers

Today many servers providing market data are attached at Gigabit speeds, while the receivers are attached at different speeds, usually 100Mbps. This creates the potential for receivers to drop packets and request re-transmissions, which creates more traffic that the slowest consumers cannot handle, continuing the vicious circle.

The solution needs to be some type of access control in the application that limits the amount of data that one host can request. QoS and other network functions can mitigate the problem, but ultimately the subscriptions need to be managed in the application.

Tibco Heartbeats

TibcoRV has had the ability to use IP multicast for the heartbeat between the TICs for many years. However, there are some brokerage houses that are still using very old versions of TibcoRV that use UDP broadcast support for the resiliency. This limitation is often cited as a reason to maintain a Layer 2 infrastructure between TICs located in different data centers. These older versions of TibcoRV should be phased out in favor of the IP multicast supported versions.

Multicast Forwarding Options

PIM Sparse Mode

The standard IP multicast forwarding protocol used today for market data delivery is PIM Sparse Mode. It is supported on all Cisco routers and switches and is well understood. PIM-SM can be used in all the network components from the exchange, FSP, and brokerage.

There are, however, some long-standing issues and unnecessary complexity associated with a PIM-SM deployment that could be avoided by using PIM-Bidir and PIM-SSM. These are covered in the next sections.

The main components of the PIM-SM implementation are:

- PIM Sparse Mode v2
- Shared Tree (spt-threshold infinity)

  A design option in the brokerage or in the exchange.

- Static RP
- Anycast RP

  Details of Anycast RP can be found in:

  Anycast RP

  http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/mcst_sol/anycast.htm

The classic high availability design for Tibco in the brokerage network is documented in:

Financial Services Design for High Availability
http://www.cisco.com/en/US/tech/tk828/technologies_white_paper09186a008015a8ad.shtml

Bidirectional PIM

PIM-Bidir is an optimization of PIM Sparse Mode for many-to-many applications. It has several key advantages over a PIM-SM deployment:

- Better support for intermittent sources

  For more information, see Intermittent Sources.

- No data-triggered events

  One of the weaknesses of PIM-SM is that the network continually needs to react to active data flows. This can cause non-deterministic behavior that may be hard to troubleshoot. PIM-Bidir has the following major protocol differences over PIM-SM:

  - No source registration

    Source traffic is automatically sent to the RP and then down to the interested receivers. There is no unicast encapsulation, PIM joins from the RP to the first hop router and then registration stop messages.

  - SPT switchover

    All PIM-Bidir traffic is forwarded on a *,G forwarding entry. The router does not have to monitor the traffic flow on a *,G and then send joins when the traffic passes a threshold.

  - No need for an actual RP

    The RP does not have an actual protocol function in PIM-Bidir. The RP acts as a routing vector in which all the traffic converges. The RP can be configured as an address that is not assigned to any particular device. This is called a Phantom RP.

  - No need for MSDP

    MSDP provides source information between RPs in a PIM-SM network. PIM-Bidir does not use the active source information for any forwarding decisions and therefore MSDP is not required.

Bidirectional PIM is ideally suited for the brokerage network in the data center of the exchange. In this environment there are many sources sending to a relatively few set of groups in a many-to-many traffic pattern.

The key components of the PIM-Bidir implementation are:

- Bidirectional PIM
- Static RP
- Phantom RP

Further details about Phantom RP and basic PIM-Bidir design are documented in:

Bidirectional PIM Deployment Guide
http://www.cisco.com/warp/public/732/Tech/multicast/docs/bidirdeployment.pdf

Source Specific Multicast

PIM-SSM is an optimization of PIM Sparse Mode for one-to-many applications. In certain environments it can offer several distinct advantages over PIM-SM. Like PIM-Bidir, PIM-SSM does not rely on any data-triggered events. Furthermore, PIM-SSM does not require an RP at all—there is no such concept in PIM-SSM. The forwarding information in the network is completely controlled by the interest of the receivers.

Source Specific Multicast is ideally suited for market data delivery in the financial service provider. The FSP can receive the feeds from the exchanges and then route them to the end of their network.

Many FSPs are also implementing MPLS and Multicast VPNs in their core. PIM-SSM is the preferred method for transporting traffic in VRFs.

When PIM-SSM is deployed all the way to the end user, the receiver indicates his interest in a particular S,G with IGMPv3. Even though IGMPv3 was defined by RFC 2236 back in October, 2002, it still has not been implemented by all edge devices. This creates a challenge for deploying an end-to-end PIM-SSM service. A transitional solution has been developed by Cisco to enable an edge device that supports IGMPv2 to participate in an PIM-SSM service. This feature is called SSM Mapping and is documented in:
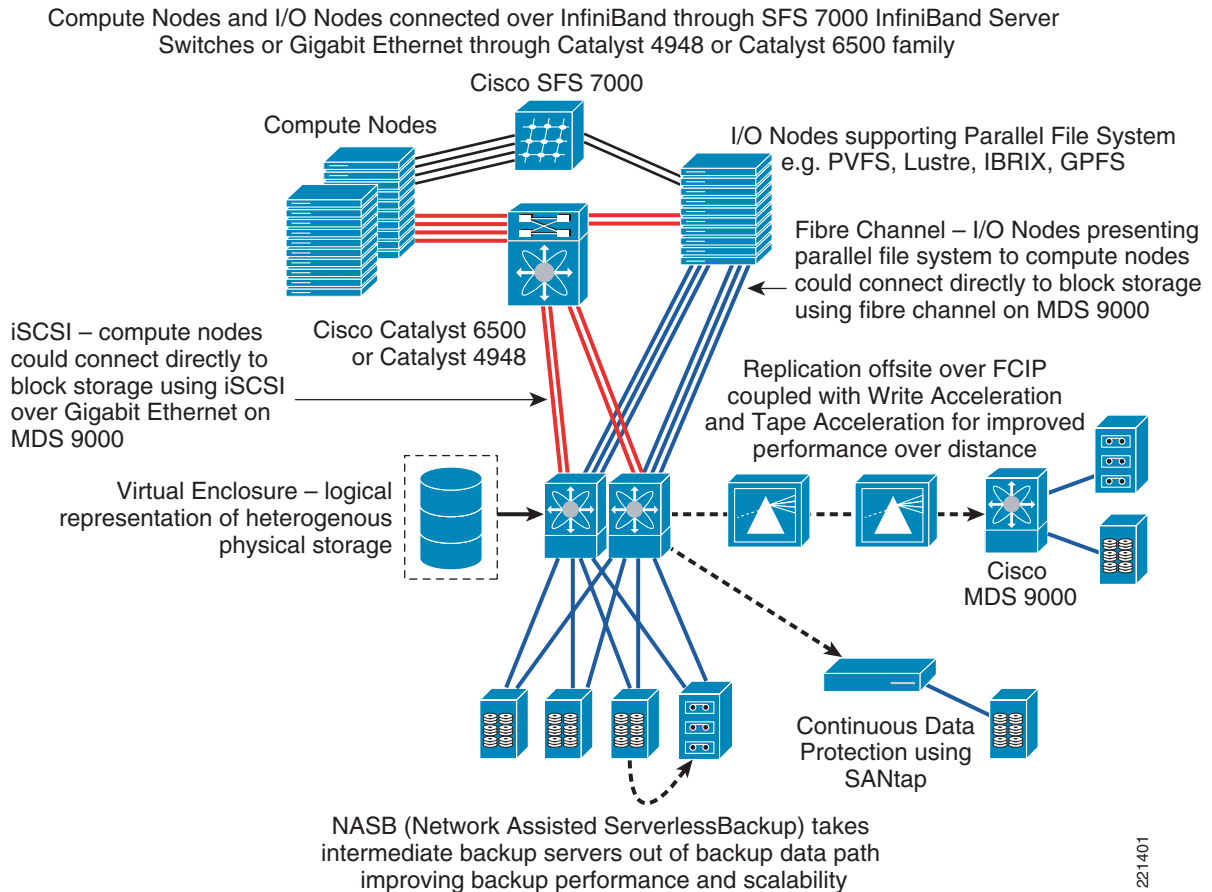
http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801a6d6f.html

Storage Services

The service provides storage capabilities into the market data and trading environments. Trading applications access backend storage to connect to different databases and other repositories consisting of portfolios, trade settlements, compliance data, management applications, Enterprise Service Bus (ESB), and other critical applications where reliability and security is critical to the success of the business. The main requirements for the service are:

- Storage virtualization

- Replication

- Backup services

Storage virtualization is an enabling technology that simplifies management of complex infrastructures, enables non-disruptive operations, and facilitates critical elements of a proactive information lifecycle management (ILM) strategy. EMC Invista running on the Cisco MDS 9000 enables heterogeneous storage pooling and dynamic storage provisioning, allowing allocation of any storage to any application. High availability is increased with seamless data migration. Appropriate class of storage is allocated to point-in-time copies (clones). Storage virtualization is also leveraged through the use of Virtual Storage Area Networks (VSANs), which enable the consolidation of multiple isolated SANs onto a single physical SAN infrastructure, while still partitioning them as completely separate logical entities. VSANs provide all the security and fabric services of traditional SANs, yet give organizations the flexibility to easily move resources from one VSAN to another. This results in increased disk and network utilization while driving down the cost of management. Integrated Inter VSAN Routing (IVR) enables sharing of common resources across VSANs.

Figure 18   *High Performance Computing Storage*

Compute Nodes and I/O Nodes connected over InfiniBand through SFS 7000 InfiniBand Server
Switches or Gigabit Ethernet through Catalyst 4948 or Catalyst 6500 family

Cisco SFS 7000

Compute Nodes

I/O Nodes supporting Parallel File System
e.g. PVFS, Lustre, IBRIX, GPFS

Fibre Channel – I/O Nodes presenting
parallel file system to compute nodes
could connect directly to block storage
using fibre channel on MDS 9000

iSCSI – compute nodes
could connect directly to
block storage using iSCSI
over Gigabit Ethernet on
MDS 9000

Cisco Catalyst 6500
or Catalyst 4948

Replication offsite over FCIP
coupled with Write Acceleration
and Tape Acceleration for improved
performance over distance

Virtual Enclosure – logical
representation of heterogenous
physical storage

Cisco
MDS 9000

Continuous Data
Protection using
SANtap

NASB (Network Assisted ServerlessBackup) takes
intermediate backup servers out of backup data path
improving backup performance and scalability

221401

Replication of data to a secondary and tertiary data center is crucial for business continuance. Replication offsite over Fiber Channel over IP (FCIP) coupled with write acceleration and tape acceleration provides improved performance over long distance. Continuous Data Replication (CDP) is another mechanism which is gaining popularity in the industry. It refers to backup of computer data by automatically saving a copy of every change made to that data, essentially capturing every version of the data that the user saves. It allows the user or administrator to restore data to any point in time. Solutions from EMC and Incipient utilize the SANTap protocol on the Storage Services Module (SSM) in the MDS platform to provide CDP functionality. The SSM uses the SANTap service to intercept and redirect a copy of a write between a given initiator and target. The appliance does not reside in the data path—it is completely passive. The CDP solutions typically leverage a history journal that tracks all changes and bookmarks that identify application-specific events. This ensures that data at any point in time is fully self-consistent and is recoverable instantly in the event of a site failure.

Backup procedure reliability and performance are extremely important when storing critical financial data to a SAN. The use of expensive media servers to move data from disk to tape devices can be cumbersome. Network-accelerated serverless backup (NASB) helps you back up increased amounts of data in shorter backup time frames by shifting the data movement from multiple backup servers to Cisco MDS 9000 Series multilayer switches. This technology decreases impact on application servers because the MDS offloads the application and backup servers. It also reduces the number of backup and media servers required, thus reducing CAPEX and OPEX. The flexibility of the backup environment increases because storage and tape drives can reside anywhere on the SAN.

Trading Resilience and Mobility

The main requirements for this service are to provide the virtual trader:

- Fully scalable and redundant campus trading environment
- Resilient server load balancing and high availability in analytic server farms
- Global site load balancing that provide the capability to continue participating in the market venues of closest proximity

A highly-available campus environment is capable of sustaining multiple failures (i.e., links, switches, modules, etc.), which provides non-disruptive access to trading systems for traders and market data feeds. Fine-tuned routing protocol timers, in conjunction with mechanisms such as NSF/SSO, provide subsecond recovery from any failure.

The high-speed interconnect between data centers can be DWDM/dark fiber, which provides business continuance in case of a site failure. Each site is 100km-200km apart, allowing synchronous data replication. Usually the distance for synchronous data replication is 100km, but with Read/Write Acceleration it can stretch to 200km. A tertiary data center can be greater than 200km away, which would replicate data in an asynchronous fashion.

Figure 19 *Trading Resilience*
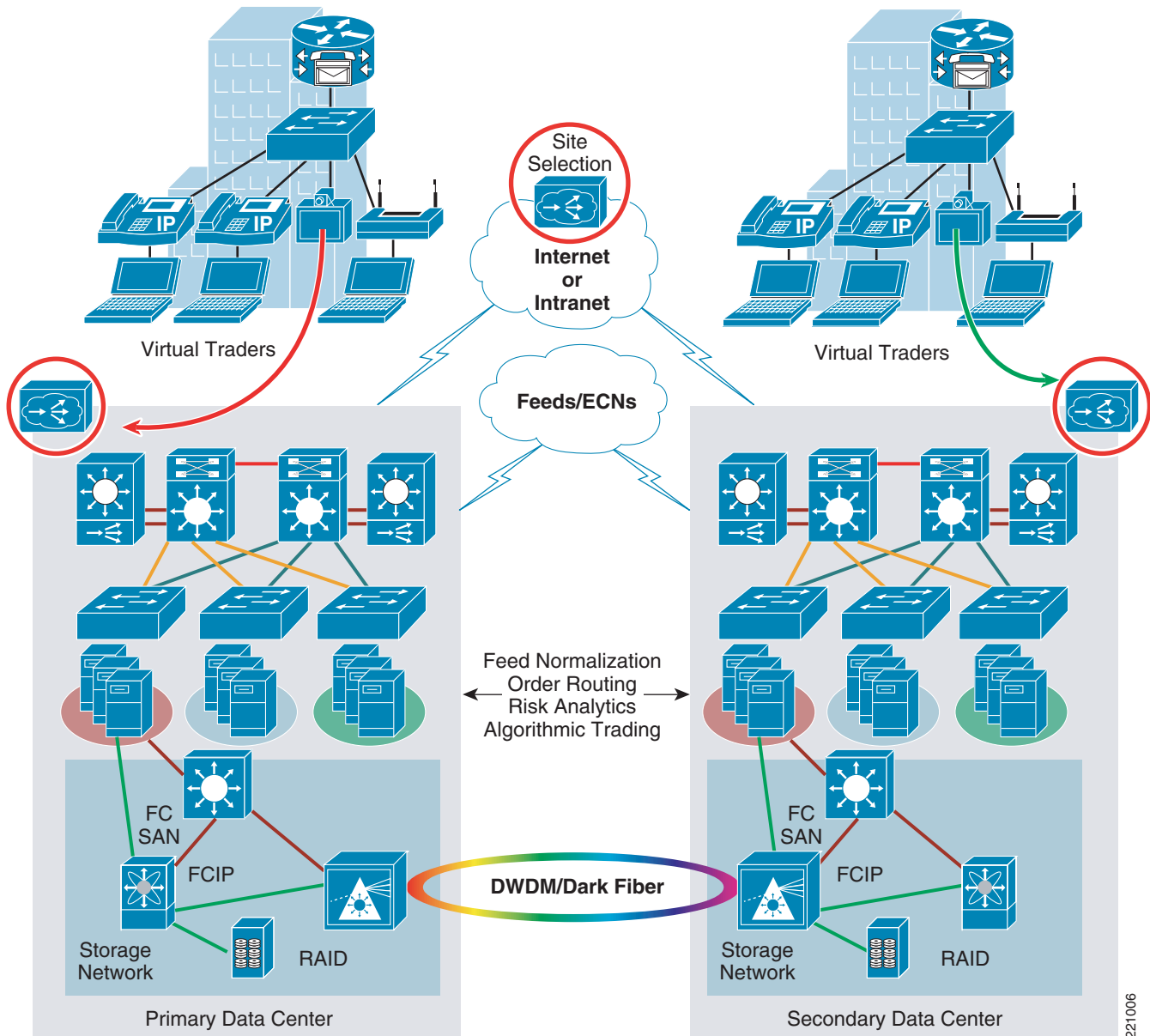
A robust server load balancing solution is required for order routing, algorithmic trading, risk analysis, and other services to offer continuous access to clients regardless of a server failure. Multiple servers encompass a "farm" and these hosts can added/removed without disruption since they reside behind a virtual IP (VIP) address which is announced in the network.

A global site load balancing solution provides remote traders the resiliency to access trading environments which are closer to their location. This minimizes latency for execution times since requests are always routed to the nearest venue.

*Figure 20*　　　　*Virtualization of Trading Environment*
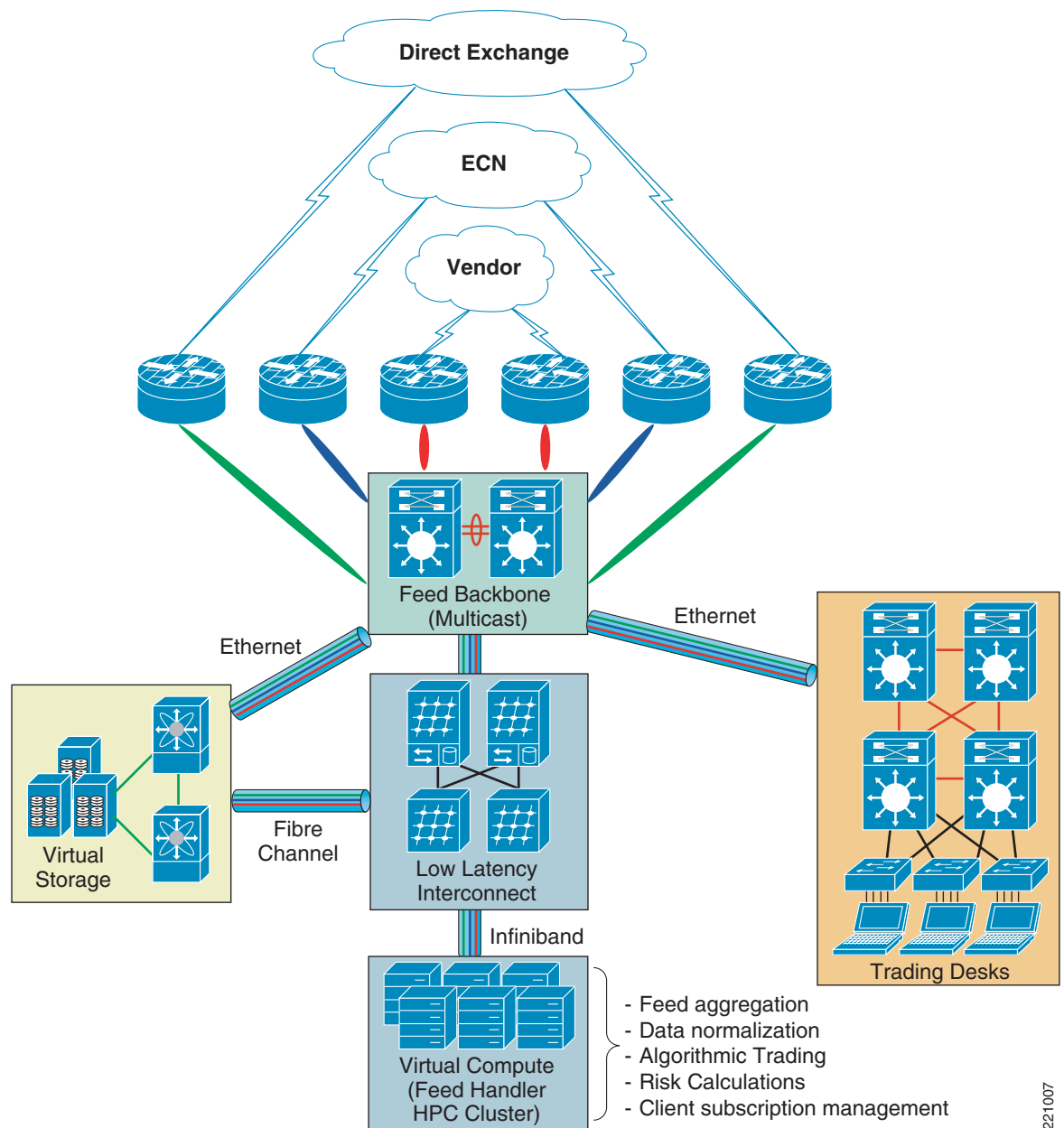
A trading environment can be virtualized to provide segmentation and resiliency in complex architectures. Figure 20 illustrates a high-level topology depicting multiple market data feeds entering the environment, whereby each vendor is assigned its own Virtual Routing and Forwarding (VRF) instance. The market data is transferred to a high-speed InfiniBand low-latency compute fabric where feed handlers, order routing systems, and algorithmic trading systems reside. All storage is accessed via a SAN and is also virtualized with VSANs, allowing further security and segmentation. The normalized data from the compute fabric is transferred to the campus trading environment where the trading desks reside.

More information on designing highly available data center clusters can be found in the ESE paper:
http://www.cisco.com/application/pdf/en/us/guest/netsol/ns500/c649/ccmigration_09186a00807c4528.pdf

Wide Area Application Services

This service provides application acceleration and optimization capabilities for traders who are located outside of the core trading floor facility/data center and working from a remote office. To consolidate servers and increase security in remote offices, file servers, NAS filers, storage arrays, and tape drives are moved to a corporate data center to increase security and regulatory compliance and facilitate centralized storage and archival management. As the traditional trading floor is becoming more virtual, wide area application services technology is being utilized to provide a "LAN-like" experience to remote traders when they access resources at the corporate site. Traders often utilize Microsoft Office applications, especially Excel in addition to Sharepoint and Exchange. Excel is used heavily for modeling and permutations where sometime only small portions of the file are changed. CIFS protocol is notoriously known to be "chatty," where several message normally traverse the WAN for a simple file operation and it is addressed by Wide Area Application Service (WAAS) technology. Bloomberg and Reuters applications are also very popular financial tools which access a centralized SAN or NAS filer to retrieve critical data which is fused together before represented to a trader's screen.

Figure 21   **Wide Area Optimization**
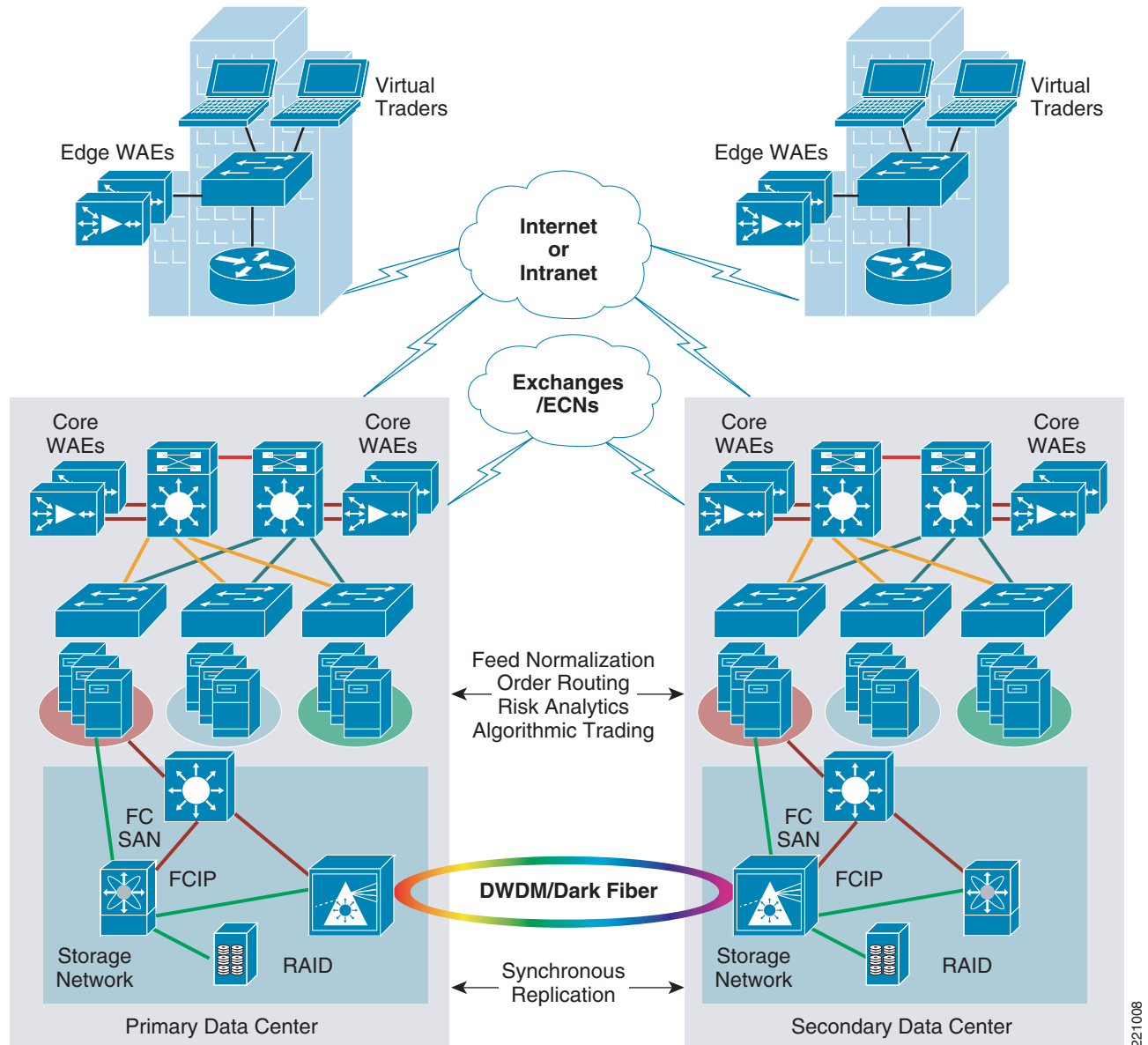
A pair of Wide Area Application Engines (WAEs) that reside in the remote office and the data center provide local object caching to increase application performance. The remote office WAEs can be a module in the ISR router or a stand-alone appliance. The data center WAE devices are load balanced behind an Application Control Engine module installed in a pair of Catalyst 6500 series switches at the aggregation layer. The WAE appliance farm is represented by a virtual IP address. The local router in each site utilizes Web Cache Communication Protocol version 2 (WCCP v2) to redirect traffic to the WAE that intercepts the traffic and determines if there is a cache hit or miss. The content is served locally from the engine if it resides in cache; otherwise the request is sent across the WAN the initial time to retrieve the object. This methodology optimizes the trader experience by removing application latency and shielding the individual from any congestion in the WAN.

WAAS uses the following technologies to provide application acceleration:

- Data Redundancy Elimination (DRE) is an advanced form of network compression which allows the WAE to maintain a history of previously-seen TCP message traffic for the purposes of reducing redundancy found in network traffic. This combined with the Lempel-Ziv (LZ) compression algorithm reduces the number of redundant packets that traverse the WAN, which improves application transaction performance and conserves bandwidth.

- Transport Flow Optimization (TFO) employs a robust TCP proxy to safely optimize TCP at the WAE device by applying TCP-compliant optimizations to shield the clients and servers from poor TCP behavior because of WAN conditions. By running a TCP proxy between the devices and leveraging an optimized TCP stack between the devices, many of the problems that occur in the WAN are completely blocked from propagating back to trader desktops. The traders experience LAN-like TCP response times and behavior because the WAE is terminating TCP locally. TFO improves reliability and throughput through increases in TCP window scaling and sizing enhancements in addition to superior congestion management.

Thin Client Service

This service provides a "thin" advanced trading desktop which delivers significant advantages to demanding trading floor environments requiring continuous growth in compute power. As financial institutions race to provide the best trade executions for their clients, traders are utilizing several simultaneous critical applications that facilitate complex transactions. It is not uncommon to find three or more workstations and monitors at a trader's desk which provide visibility into market liquidity, trading venues, news, analysis of complex portfolio simulations, and other financial tools. In addition, market dynamics continue to evolve with Direct Market Access (DMA), ECNs, alternative trading volumes, and upcoming regulation changes with Regulation National Market System (RegNMS) in the US and Markets in Financial Instruments Directive (MiFID) in Europe. At the same time, business seeks greater control, improved ROI, and additional flexibility, which creates greater demands on trading floor infrastructures.

Traders no longer require multiple workstations at their desk. Thin clients consist of keyboard, mouse, and multi-displays which provide a total trader desktop solution without compromising security. Hewlett Packard, Citrix, Desktone, Wyse, and other vendors provide thin client solutions to capitalize on the virtual desktop paradigm. Thin clients de-couple the user-facing hardware from the processing hardware, thus enabling IT to grow the processing power without changing anything on the end user side. The workstation computing power is stored in the data center on blade workstations, which provide greater scalability, increased data security, improved business continuance across multiple sites, and reduction in OPEX by removing the need to manage individual workstations on the trading floor. One blade workstation can be dedicated to a trader or shared among multiple traders depending on the requirements for computer power.

The "thin client" solution is optimized to work in a campus LAN environment, but can also extend the benefits to traders in remote locations. Latency is always a concern when there is a WAN interconnecting the blade workstation and thin client devices. The network connection needs to be sized accordingly so traffic is not dropped if saturation points exist in the WAN topology. WAN Quality of Service (QoS) should prioritize sensitive traffic. There are some guidelines which should be followed to allow for an optimized user experience. A typical highly-interactive desktop experience requires a client-to-blade round trip latency of <20ms for a 2Kb packet size. There may be a slight lag in display if network latency is between 20ms to 40ms. A typical trader desk with a four multi-display terminal requires 2-3Mbps bandwidth consumption with seamless communication with blade workstation(s) in the data center. Streaming video (800x600 at 24fps/full color) requires 9 Mbps bandwidth usage.

Figure 22 **Thin Client Architecture**
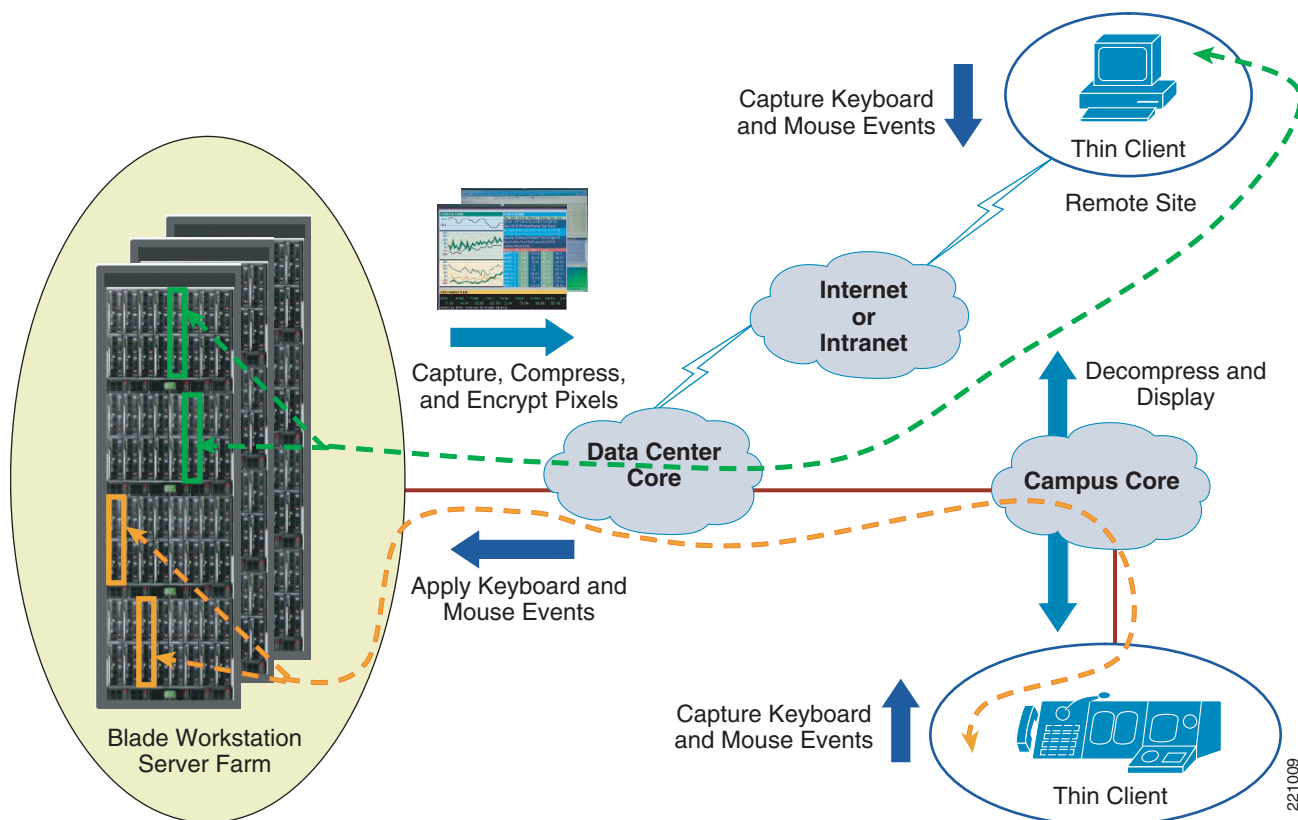
Management of a large thin client environment is simplified since a centralized IT staff manages all of the blade workstations dispersed across multiple data centers. A trader is redirected to the most available environment in the enterprise in the event of a particular site failure. High availability is a key concern in critical financial environments and the Blade Workstation design provides rapid provisioning of another blade workstation in the data center. This resiliency provides greater uptime, increases in productivity, and OpEx reduction.

Glossary

Term	Description
AES	Advanced Encryption Standard
AMQP	Advanced Message Queueing Protocol
AON	Application Oriented Networking
ARCA	The Archipelago® Integrated Web book gives investors the unique opportunity to view the entire ArcaEx and ArcaEdge books in addition to books made available by other market participants.
BRUT	ECN Order Book feed available via NASDAQ.
CAPEX	Capital Expense

Term	Description
CBOT	Chicago Board of Trade
CBWFQ	Class-Based Weighted Fair Queueing
CDP	Continuous Data Replication
CME	Chicago Mercantile Exchange is engaged in trading of futures contracts and derivatives.
CPU	Central Processing Unit
DDR	Dual Data Rate
DDTS	Distributed Defect Tracking System
DMA	Direct Market Access
DRE	Data Redundancy Elimination
DWDM	Dense Wavelength Division Multiplexing
ECN	Electronic Communication Network
ESB	Enterprise Service Bus
ESE	Enterprise Solutions Engineering
FAST	FIX Adapted for Streaming
FCIP	Fibre Channel over IP
FIX	Financial Information Exchange
FSMS	Financial Services Latency Monitoring Solution
FSP	Financial Service Provider
ILM	Information Lifecycle Management
INET	Instinet Island Book
IOS	Internetworking Operating System
ISCSI	Internet SCSI
IT	Information Technology
IVR	Inter-VSAN Routing
KVM	Keyboard Video Mouse
LLQ	Low Latency Queueing
MAN	Metro Area Network
MDS	Multilayer Director Switch
MiFID	Markets in Financial Instruments Directive
MPI	Message Passing Interface is an industry standard specifying a library of functions to enable the passing of messages between nodes within a parallel computing environment.
NAS	Network Attached Storage
NASB	Network Accelerated Serverless Backup
NIC	Network Interface Card
NQDS	Nasdaq Quotation Dissemination Service

Term	Description
NSF	Non-Stop Forwarding
OMS	Order Management System
OPEX	Operational Expense
OS	Operating System
OSI	Open Systems Interconnection
PIM	Protocol Independent Multicast
PIM-Bidir	PIM-Bidirectional
PIM-SM	PIM-Sparse Mode
PIM-SSM	PIM-Source Specific Multicast
QoS	Quality of Service
RAM	Random Access Memory
RDF	Reuters Data Feed
RDF-D	Reuters Data Feed Direct
RDMA	Remote Direct Memory Access
RegNMS	Regulation National Market System
RGS	Remote Graphics Software
RMDS	Reuters Market Data System
RMON	Remote Monitoring
RTCP	RTP Control Protocol
RTP	Real Time Protocol
RWF	Reuters Wire Format
S,G	Source, Group
SAN	Storage Area Network
SCSI	Small Computer System Interface
SDP	Sockets Direct Protocol—Given that many modern applications are written using the sockets API, SDP can intercept the sockets at the kernel level and map these socket calls to an InfiniBand transport service that uses RDMA operations to offload data movement from the CPU to the HCA hardware.
SFS	Server Fabric Switch
SFTI	Secure Financial Transaction Infrastructure network developed to provide firms with excellent communication paths to NYSE Group, AMEX, Chicago Stock Exchange, NASDAQ, and other exchanges. It is often used for order routing.
SLA	Service Level Agreement
SOA	Services Oriented Architecture
SSL	Secure Sockets Layer

Term	Description
SSM	Storage Services Module
SSO	Stateful Switch-Over
TCP/IP	Transport Control Protocol/Internet Protocol
TFO	Transport Flow Optimization
TOE	TCP Offload Engine
UDP	User Datagram Protocol
VIP	Virtual IP
VRF	Virtual Routing and Forwarding
VSAN	Virtual Storage Area Network
WAAS	Wide Area Application Service
WAE	Wide area Application Engine
WAN	Wide Area Network
WCCP	Web Cache Communication Protocol
XTP	Extreme Transaction Processing

About the Authors

Mihaela Risca is an Industry Solutions Architect for the Financial Services vertical and can be reached at mrisca@cisco.com.

Dave Malik is a Technical Leader in the Advanced Services group and can be reached at dmalik@cisco.com.

Andy Kessler is a Technical Leader in the Systems and Architecture group and can be reached at kessler@cisco.com.