Market Data Network Architecture (MDNA) Overview

Scope

The scope of the Market Data Network Architecture (MDNA) includes the sources for market data streams (stock exchanges), the Financial Service Providers (FSP) and the final consumers of the data (Brokerage Houses).

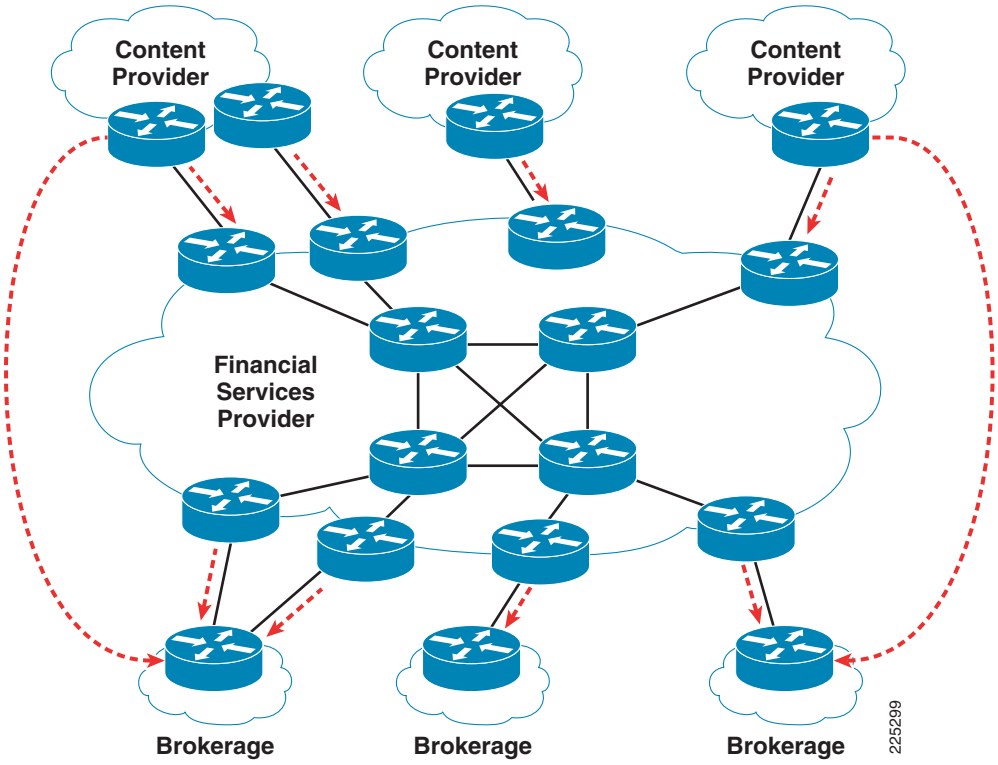
The network design and strategy for the Brokerage houses is consistent with the Trading Floor Architecture which is described in the *Trading Floor Architecture* document.

Industry Stakeholders in Financial Services

This section lists the key stakeholders that support and would implement a converged and standardized MDNA. See Figure 1.

Corporate Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Figure 1 Market Data Distribution Players



Stock Exchanges and Future/Option Exchanges

Stock exchanges and future/options exchanges are the content providers (CPs) for market data.

Financial Service Providers

Financial Service Providers (FSPs) are specialized Service Providers (SPs) that tailor their products to meet the specific needs of the financial industry. Many of them are content aggregators and offer value added services such as analytics.

Brokerage Houses

These are the ultimate consumers of the market data and the people that place the orders.

Software/Services/System Integrators

Integrators are companies that are of part of the Financial Services Ecosystem (FSE) that creates products and services that tie everything together.

Messaging/Transaction Standard Working Groups

Standards working groups are organizations working on protocols for market data and financial applications. Standardization of the industry is critical for its growth. Applicable standards are addressed in the following links:

- *Financial Information eXchange (FIX) Protocol*
- *Advanced Message Queuing Protocol (AMQP)*

Main Focus Areas

The MDNA environment primarily addressed the processes:

- Multicast Service Delivery, page 3
- Market Data Delivery, page 3
- Order Execution, page 4

Multicast Service Delivery

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 Transmission Control Protocol (TCP) or User Datagram Protocol (UDP) broadcast as the network-layer protocol, but both 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 will exhaust the resources of the servers and the network. The server side must transmit and service each of the streams individually which will require larger and larger server farms. On the network side, the required bandwidth for the application will be increasing in a linear fashion. For example, to send a one Mbps stream to 1000 recipients using TCP will require one Gbps of bandwidth.

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

Market Data Delivery

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 might normalize the data stream into a common format and then republish the data stream 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, risk modeling, best execution compliance, and so on. Many of these applications listen to the feeds and then republish their own analytical and derivative information. For example, a brokerage might compare the offered price of CSCO to the option price of CSCO on another exchange and then publish ratings that a different application might monitor to see how much they are out of sync.

The delivery of these data streams is typically over a reliable multicast transport protocol. Traditionally this has been TIBCO Rendezvous. TIBCO Rendezvous 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 subject name and receive just 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 the market data delivery. The delivery of the market data from the exchange to the brokerage is usually considered a unidirectional one-to-many application. In practice, most exchanges transmit the market data stream from several servers simultaneously which makes the service more of a few-to-many implementation.

The only exception to the unidirectional nature of the market data might be the retransmission requests that are typically sent using a unicast. However, the trading applications in the brokerage are definitely many-to-many applications and might interact with the exchanges for placing orders.

Order Execution

After the market data is received by the brokerage firms, it is used as a basis for the order execution. Below is a summary of the steps for order execution.

Order execution summary:

1. Market data is received in ticker plant—Data stream is normalized; formatted, processed, and republished.
2. Trading Applications receive data stream—Data is processed by pricing engines, algorithmic trading engines and viewed by human beings.
3. Order is triggered—Either programmatically by automated trading engine or initiated by a human trader
4. Order is sent to Order Management System (OMS)—Order is logged and passed to the Smart Routing Engine (SRE) which chooses the execution venue based on price, liquidity, latency, volume, transaction cost, and so on.
5. Order is sent to the Financial Information Exchange (FIX) engine which sends the trade to the Exchange.
6. Order is received by the Exchange FIX engine.
   a. Acknowledgement is sent to the initiating firm which gets logged in OMS.
   b. Order is sent to the matching application or Market Making Engine.
7. Market Making Engine will match sellers to buyers based on published bids and asking price.

   Seller is matched to buyer and order gets executed.
8. Exchange sends confirmation of execution to the Exchange FIX engine.
   a. Exchange FIX engine sends confirmation to brokerage firm.
   b. Order is closed.

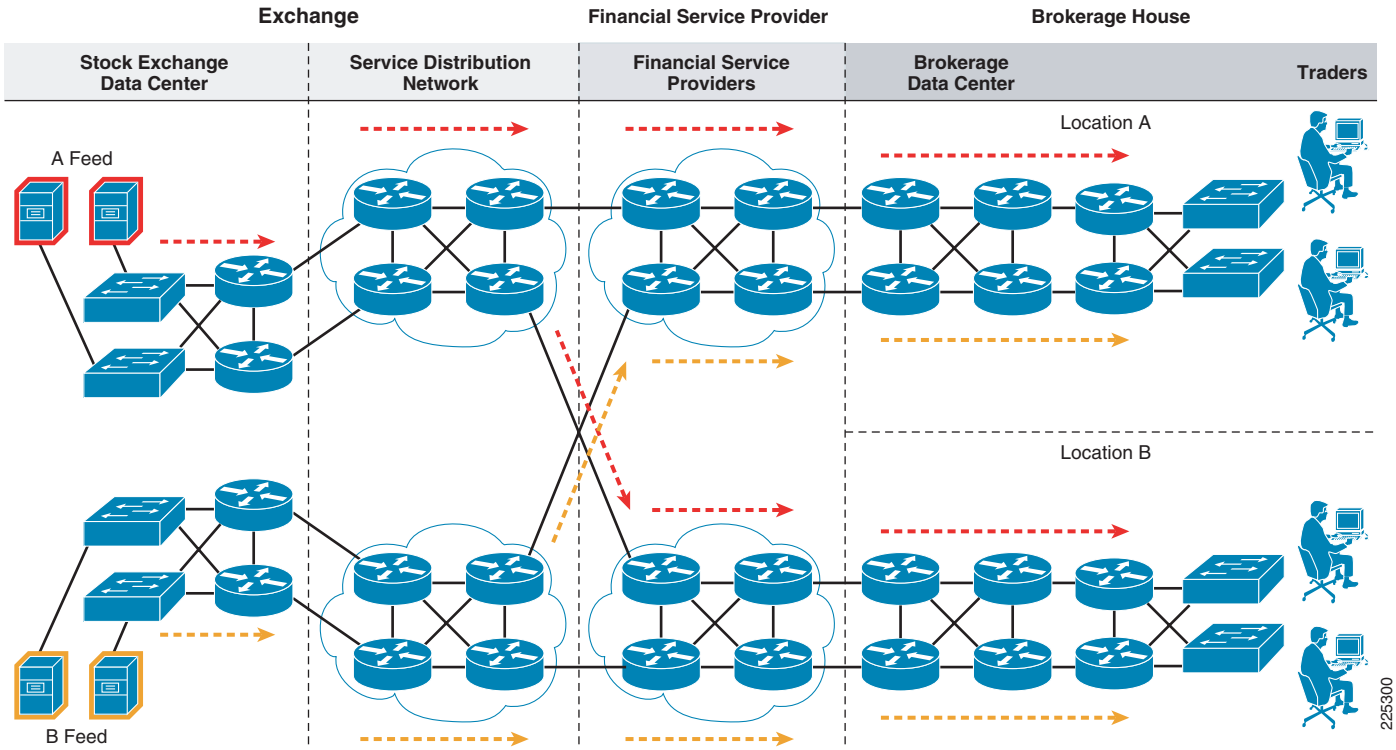
Market Data Distribution Architecture Components

In order to discuss an overall design for market data delivery we need to divide the scope of the problem into three main sections:

- The Exchange Network

  - Exchange Data Center
  - Service Distribution Network
- The Financial Service Provider
  - Provider Distribution Network
  - Service Edge
- The Brokerage Network
  - Back-Office Network
  - Front-Office Network

Figure 2 Market Data Architectural Component



The Exchange Network

The Exchange Network includes:

- The Exchange Data Center—Contains the servers which set the prices for the financial instruments and process the orders.
- The Service Distribution network—Transmits the feeds out to the service edge—which feeds the brokerages that have Direct Market Access (DMA) and the FSPs. The FSPs feed their brokerage customers and may normalize the data and add their own analytics to the data stream.

Many exchanges out-source the service distribution network to a provider so that they can focus on their core business.

The Financial Service Provider (FSP)

The FSP network includes the following:

- Provider Distribution Network—The physical network design that allows the FSP to have a regional and global service presence between the exchanges and their customers. The network design is very similar to a traditional service provider, but may contain more redundancy than is typically deployed.
- Service Edge—Includes all the infrastructure to deliver the market data streams to the consumers. This typically will include all the access technologies in their POPs and will need to support the necessary security and service policies as defined through the provisioning process.

Aggregation and Normalization

Many FSPs—such as Reuters and Bloomberg—offer feeds that have been aggregated and normalized. This allows brokerage houses to write applications that can read and parse one market data feed format instead of the many dozens of different formats from all the exchanges. Recently, due to a number of factors such as the increase in traffic volume and the use of programmed trading engines, the need to clear trades faster has become very important. Many brokerage houses are using Direct Market Access (DMA) feeds instead of the normalized streams from the FSPs. This is more work for the brokerage (to build the interfaces for the different exchanges), but it does pay off in allowing them to be more nimble and able to quickly respond to changes in the market. Some brokerages have been able to quantify each millisecond of latency to a dollar amount of lost revenue.

The normalization process in the FSPs is quite sophisticated and should not be underestimated. The raw data from the exchange can have many different types of errors which need to be cleansed. These errors include things such as a sell marked as buy, the right price on the wrong symbol, and so on. The FSPs and advanced feed handlers have developed rules over time to identify these errors and drop those particular messages. Many of the brokerages that have DMA will have to implement this type of cleansing themselves.

Business-to-Business Services

The FSPs also offer business-to-business services for their customers. This allows customers to have direct peer-to-peer networks for specific applications and form closed user groups. Typically, this is done between the large brokerage houses which engage in direct high volume transactions.

The Brokerage Network

The brokerage network includes:

- Brokerage Back-Office network—Traditionally in investment firms, the back office contains the administrative functions that support the trading of securities, including record keeping, trade confirmation, trade settlement, and regulatory compliance. In terms of market data distribution this will include the feed handlers, the OMS, the FIX Engine, the algorithmic trading engines, and all the network infrastructure to support those functions. The Back office infrastructure is typically protected from the external network connections by firewalls and strict security features.
- Brokerage Front-Office Network—The front office contains the end user trading systems, compliance and performance monitors, and all the network infrastructure to support them.

The market data feeds are brought into the data center in the back office where they are normalized and processed. The feeds then get injected into the messaging bus which delivers the data streams to the front office trading floor. This is typically done with some type of reliable multicast messaging protocol such as TIBCO Rendezvous or 29 West's LBM. A different set of multicast groups and infrastructure is used to distribute the streams to the front office applications.

Market Data Distribution Technology Components

This section describes the ideal technology components for an MDNA (see Table 1). The technologies represented here describe the forward looking vision for the industry. The actual technologies which will be selected for a particular deployment would need to reflect the requirements for those specific circumstances. This would include a specific customer's threshold for risk and drivers for innovation.

Table 1 Summary of Market Data Distribution Technology Components

Architectural Component	Platforms	Technologies	Protocols
Exchange data center	Cisco 6500, Cisco 4500/Cisco 4900, Nexus 5000, and Nexus 7000	10-Gigabit Ethernet, data center Ethernet, storage, and high availability	PIM BiDir, HSRP, and IP routing
Service distribution network	Cisco 6500 and Cisco 7600	10-Gigabit Ethernet and high availability	MPLS/MVPN, PIM SSM, PIM BiDir, and IP routing
Provider distribution network	Cisco RS-1, Cisco 7600, and GSR	10-Gigabit Ethernet and Packet over Sonet (PoS)	MPLS/MVPN, PIM SSM, and IP routing
Service edge	Cisco 7600 and Cisco 7200	Metro Ethernet and NAT	MPLS/MVPN, PIM SSM, and IP routing
Brokerage back-office network	Cisco 2800/Cisco 3800, Cisco 7200, ASR, Cisco 6500, Cisco 4500/Cisco 4900, Nexus 5000, and Nexus 7000	10-Gigabit Ethernet, data center Ethernet, storage, high availability, and NAT	PIM BiDir, HSRP, and IP routing
Brokerage front-office network	Cisco 2800/Cisco 3800, Cisco 7200, ASR, Cisco 6500, and Cisco 4500/Cisco 4900	10-Gigabit Ethernet and high availability	PIM BiDir, PIM SM, HSRP, and IP routing

IP Multicast Protocol Options

The following IP multicast options are described in this section:

- PIM Sparse Mode, page 8
- Bidirectional PIM, page 8
- Source-Specific Multicast, page 9

PIM Sparse Mode

The standard IP multicast used today for market data delivery is Protocol Independent Multicast Sparse Mode (PIM SM). 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. PIM SM works well for both many-to-many and one-to-many applications.

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

The main components of the PIM SM implementations are:

- PIM SM
- Shared Tree (spt-threshold infinity)

  A design option in the brokerage or in the exchange.
- Static RP
- Anycast RP

Details of Anycast RP and basic design can be found in the "Common Best Practices for Market Data Delivery" section on page 25 section. This whitepaper has details on a Anycast RP deployment: Anycast RP.

The classic high available design for TIBCO in the brokerage network has been documented here: Financial Services Design for High Availability.

Bidirectional PIM

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

- Better support for intermittent sources

  This is covered in the "Intermittent Sources" section on page 28 under application issues.
- No data triggered events

  One of the weaknesses of PIM SM is that the network must continually react to active data flows. This can cause non-deterministic behavior that might 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 rendezvous point (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.

  – No 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

Multicast Source Discovery Protocol (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.

PIM BiDir is ideally suited for the brokerage network and in the exchange data center. In these environments, there are many sources sending to relatively 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 of Phantom RP and basic PIM BiDir design are in the *Bidirectional PIM Deployment Guide*.

Known limitations and considerations:

- PIM BiDir RP Limitation—There is a limitation of four PIM BiDir RPs with the current Policy Feature Card 3 (PFC3) on the Cisco 6500 and Cisco 7600. This might present a network design limitation although, in many deployments, four RPs will be sufficient.

  The next version of the PFC will allow for eight Bidir RPs.

- Default Multicast Distribution Tree (MDT) Limitation—In the case in which an exchange or brokerage is using MVPN for segmentation of their market data feeds in their core there will be a limitation on the forwarding optimizations available with data MDTs.

  When data is forwarded with a *,G entry in either PIM SM or PIM BiDir, the traffic will be forwarded on the default MDT regardless of the data rate. In other words, high-bandwidth sources will not trigger a data MDT when used with PIM BiDir. This will cause all the data being sent with PIM BiDir to be delivered to all the provider edge (PE) routers that are provisioned for that VPN routing and forwarding (VRF).

  This might not be an issue—depending on the application behavior and the network design. If all the data feeds must be delivered to all the PE routers, there is no loss of optimization. However, if you need to limit some of the high bandwidth PIM BiDir groups from reaching all the PE routers, you cannot do so with the Multicast VPN (MVPN) core.

- General Risk Factor—PIM BiDir has been implemented since December 2000 on software-based Cisco routers and since April 2003 on the Cisco 6500 with hardware support with the release of the Sup720. However, there are still a limited number of actual deployments in financial services networks. Most financial customers are fairly risk adverse and slow to adopt new technologies. PIM BiDir has fallen into this category for a number of years, but testing and certification have moved forward in a number of large exchanges and brokerage houses.

Source-Specific Multicast

PIM SSM is an optimization of PIM SM for one-to-many applications. In certain environments, PIM SSM 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 a RP—there is no such concept in PIM SSM. The forwarding information in the network is completely controlled by the interest of the receivers and the route to the source.

PIM SSM is ideally suited for market data delivery in the FSP. The FSP can receive the feeds from the exchanges and then route them to the edge of their network.

Many FSPs are also implementing Multiprotocol Label Switching (MPLS) and MVPNs in their core. PIM SSM is the preferred method for transporting traffic with MVPN.

When PIM SSM is deployed all the way to end users, the receiver indicates interest in a particular S,G with Internet Group Management Protocol Version 3 (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 Internet Group Management Protocol Version 2 (IGMPv2) to participate in an PIM SSM service. This feature is called *SSM Mapping* and is documented at the following URL: http://www.cisco.com/en/US/products/sw/iosswrel/ps5207/products_feature_guide09186a00801a6d6f.html

While SSM Mapping allows an end user running IGMPv2 to join a PIM-SSM service this would require IGMP to operate at the service edge. This problem could be solved with *IGMP Mroute Proxy*, described in the "IGMP Mroute Proxy" section on page 13. A better solution would be a service called *PIM Mapping*. This service would allow a PIM *,G join to be translated into a PIM S,G join at the service edge. This is a potential new feature, that is currently being investigated, which can be implemented to create an easy method to interface between providers and their customers.

Table 2 summarizes the PIM protocol.

Table 2 PIM Protocol Summary

	PIM SM	PIM BiDir	PIM SSM
Applications	One-to many Many-to-one	Many-to-many	One-to-many
Intermittent Sources	Potential issue	No problem	No Problem
Network Deployment	Anywhere	Exchange DC or brokerage	FSP or MVPN
Number of Deployments	Most	Least	Many

Provisioning Options

FSPs and the exchanges need a method to provision services for customers. The trade-offs are administrative overhead, security and simplicity. This section describes the following options:

Static Forwarding

Static forwarding has traditionally been the first choice for provisioning market data services. The description of static forwarding provided in this publication includes the following sections:

Advantages and Disadvantages of Static Forwarding

Static forwarding has the following advantages over a dynamic forwarding:

- Straightforward Provisioning—Enabling entitlements for a customer comes down to adding the static join commands on the appropriate egress interface. This is done with the **igmp static-group** command as in the following example:

  ```
  interface Vlan6
   ip igmp static-group 224.0.2.64
  ```

  In order to simplify the configuration for dozens or hundreds of groups the **static group range** command has been added. The following is an example:

  ```
  class-map type multicast-flows market-data
   group 224.0.2.64 to 224.0.2.80
  interface Vlan6
   ip igmp static-group class-map market-data
  ```

- Minimal Coordination—The static forwarding requires very little coordination between content provider and customer. The provider is responsible for putting the packets on a wire and the customer must capture them. None of these relationships need to be negotiated:
  - PIM neighbors
  - Designated router (DR)
  - RP information
  - MSDP peering
- Clear Demarcation—There is a clear separation between the customer network and the provider network from an ownership perspective. They are essentially separate multicast domains with each responsible for their part. This separation reduces finger pointing and simplifies troubleshooting.

  Another advantage is that both customer and provider are free to choose what flavor of multicast they implement in their own domains e.g. PIM SM, PIM BiDir, PIM SSM

As for disadvantages, the main drawback for static forwarding is that the customer is unable to dynamically control subscriptions and bandwidth usage for the last mile. As the data rates for market data from the exchanges continue to climb month-by-month this becomes more of an issue.

Virtual RP with Static Forwarding

Statically forwarding market data streams from the providers multicast domain into the customers multicast domain presents a number of small challenges. In an ideal situation, the provider pushes packets onto an Ethernet segment and it is the customer's responsibility to pull the data stream through its infrastructure.

Since the provider and customer have separate administrative domains, this implies there are different RPs. However, with static forwarding, there is no interdomain multicast protocol such as MSDP. Furthermore, there is no control plane protocol such as IGMP or PIM to create multicast state.

The problem is that multicast state must exist on a router in order for it to forward multicast traffic. If there is no state on the router then it will just drop the incoming packets.

There must be a way create (*,G) state on the edge customer router and pull the data down from the edge of the network into the customer network.

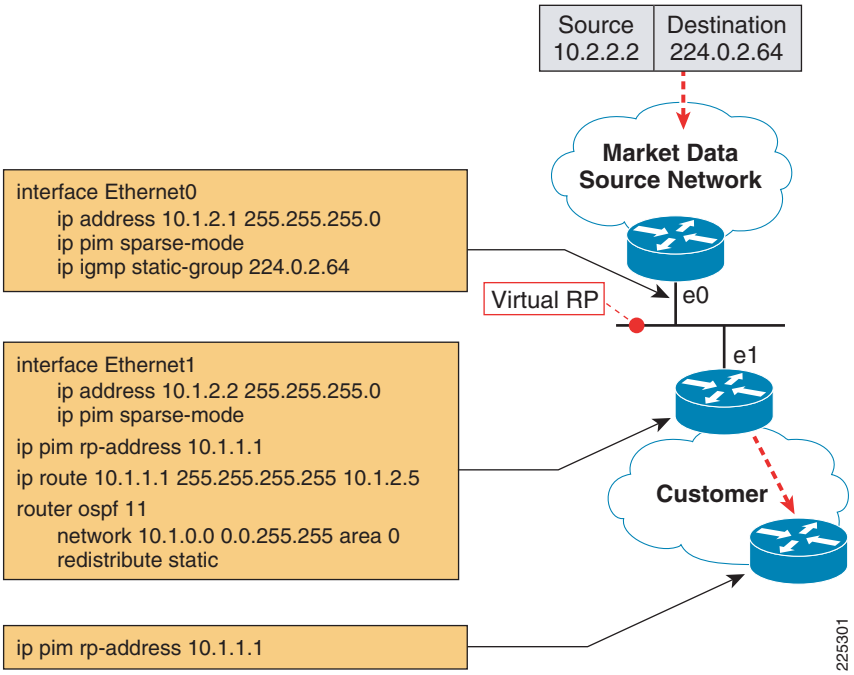
The main trick is to point the customer routers to an RP with an address that is out the upstream interface. IGMP from the downstream receivers will trigger PIM (*,G) messages from the last hop router (LHR) toward the virtual RP. This will create (*,G) state on the customer edge router with an incoming interface pointing to the provider and an outgoing interface toward the receivers. The ingress traffic will be forwarded out the egress interface on the (*,G) forwarding tree.

This approach is used by many customers and applies equally well to PIM BiDir or PIM SM.

A benefit of using the virtual RP is that the RP address can be carved out of the customer's address range. This does not require injecting the provider's address range into the customer network.

In Figure 3, the virtual RP address is injected into the customer network with a static route that is redistributed into the interior gateway protocol (IGP).

Figure 3 MD Distribution—Virtual RP



Another advantage of the virtual RP is that because traffic will be forwarded on the *,G entry, the packets will pass an RPF check according to the virtual RP address—not the source address of the packet. Again, this saves the customer from having to redistribute the source address range in their network. The customer can add the command **ip pim spt-threshold infinity** on all of the LHRs if desired, in order to prevent the LHRs from sending PIM (S,G) joins and creating (S,G) state. The Reverse Path Forwarding (RPF) check and the PIM (S,G) joins will not be issues if the customer uses PIM BiDir.

Static Service Levels—Cable Model

Many content providers have a limited set of different service levels for their products. Borrowing an analogy from the cable TV world, this would be described as silver, gold, or platinum service. In the cable model, each level of service includes more channels and therefore more bandwidth. Every customer that receives platinum service would have the same access and entitlement.

Market data content providers can adopt the same service model for their customers and would reduce the static provisioning issues down to a small set of configuration options.

Dynamic Forwarding

The data rates for market data have been steadily increasing at an alarming rate. OPRA (www.opradata.com) data rates are the most challenging in the industry—today they peak at over 500,000 messages per second and that number is expected to increase steadily for the forseeable future.

In the past, customers would receive the entire data stream and just process the subset that they need. The increased data rates have driven up the transmission costs to the point that it is now economically desirable to limit the amount of traffic by dynamically requesting a smaller portion of the available data streams.

Dynamic subscriptions give the subscriber the ability to request individual multicast streams. This gives the customer the ability to manage the bandwidth usage on the last mile.

Another driver for dynamic subscription requests is the move to 24-hour trading. Customers need to archive all the market prices throughout the day to analyze trends for long term forecasting. Many exchanges retransmit the entire trading day after the market closes so that customers can capture any missed packets. As the exchanges more closer to 24-hour operation, they will not be able to retransmit the entire trading day. Customers will need to dynamically request a portion of the data stream.

The description of dynamic forwarding provided in this publication includes the following sections:

- IGMP Membership Reports for Dynamic Subscription, page 13
- *,G PIM Joins or PIM SM/PIM BiDir Joins, page 15
- PIM SSM Joins or S,G Joins, page 15

IGMP Membership Reports for Dynamic Subscription

IGMP membership reports have some advantages for controlling the dynamic subscriptions. The coordination at the interface is at a minimum—there is no need for a PIM neighbor relationship and RP information need not be exchanged.

IGMP presents some challenges at the service interface. The main issue is that it strains the model of IGMP being used for hosts and PIM being used for routers. No one wants to require that the hosts be located in a DMZ facing the provider. We need to jump through some hoops to make this work in a standard routed environment. However, this does remain a popular way to receive market data streams because of the separation of Layer-3 multicast domains. The common way this is accomplished is with the IGMP Mroute Proxy service described in the following section.

CME uses IGMP today to subscribe to retransmission groups. A new service offering is planned and will be using the same method for dynamic subscriptions. Other exchanges are also considering using IGMP.

IGMP Mroute Proxy

Using IGMP at the service interface creates a number of basic problems. The main problem that the receivers are typically not directly connected to edge router. When the hosts send IGMP reports to the LHR, the LHR will create a (*,G) state and then send PIM (*,G) join toward the RP.

Since, the provider network is expecting IGMP not PIM, a way is needed to covert the PIM joins back to IGMP membership reports.

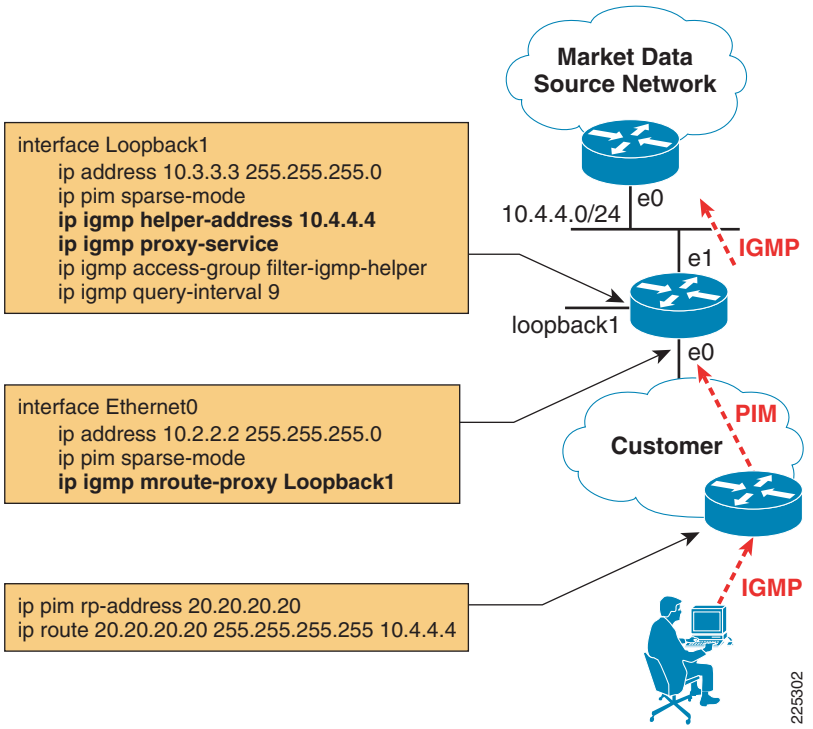
In many ways, the problem is similar to connecting a network to a digital subscriber line (DSL) service that provides a host-based service with Dynamic Host Configuration Protocol (DHCP and Domain Name System (DNS). If you want to connect a whole network to that service, the router on the customer edge

must act like a host to the provider and participate in the DNS/DHCP exchange and then provide that information down to the hosts. With IP multicast, the process is the similar, but in reverse. The router must proxy the IGMP messages from the hosts up to the service interface.

Many customers like this approach because it provides a clean interface to divide the domains. The customer network just looks like a host to the provider—not an internetwork.

A combination of **igmp proxy** commands can make this conversion possible. Figure 4 shows a typical config example.

Figure 4 MD Distribution—IGMP Mroute Proxy



All the interesting configuration is placed on the customer edge router. The rest of the routers in the customer network are configured with standard PIM SM and point to a RP in the provider network. This RP is not require—only a route to the address pointing out the upstream interface is needed. An example using a virtual RP with static forwarding is discussed in the "Virtual RP with Static Forwarding" section on page 11.

The steps to make IGMP mroute proxy process work are as follows:

1. The host application triggers an IGMP membership report which is sent to the LHR.
2. (*,G) state is created on the LHR and triggers a PIM (*,G) join message to be sent toward the RP.
3. The PIM join message filters up through the network toward the provider service edge.
4. The PIM (*,G) join is received on downstream interface of the customer edge router. This causes (*,G) multicast state to be created on the edge router.
5. The creation of the *,G state and the presence of the **mroute-proxy** command on the downstream interface triggers an unsolicited IGMP membership report to be generated on the loopback interface.
6. The IGMP helper address on the loopback interface redirects the IGMP membership report to the upstream interface toward the market data service.

7. The market data feed is pulled down into the customer network.
8. As long as the mroute state is active on the customer edge router, IGMP membership reports will continue to be sent to the provider network.

Enhancements in IOS are underway to make this type of service compatible with the IGMP Proxy described in RFC 4605.

*,G PIM Joins or PIM SM/PIM BiDir Joins

PIM joins would be the preferred method for dynamic joins between multicast domains. With PIM SM or PIM BiDir, this implies *,G join messages. In the past, many FSPs preferred to minimize the coordination and risk associated with a complicated interface with their customers. Today, many of these concerns can be managed with minimal risk. The following notes relate to these concerns:

- PIM Neighbor Relationships—The provider edge must recognize the customer routers as neighbors so that they will accept the PIM join messages. Potential additional security options are PIM neighbor filters and IP Security (IPSec) authentication for the neighbors. Both of these methods have a trade off with security versus additional maintenance.
- RP Info—The provider will need to share their RP address with the customer. This would be an Anycast or PriorityCast RP address following the multicast best practices. Some providers are dynamically sharing this info with customers using AutoRP.
- MSDP—This is the standard interdomain PIM SM solution and can be used for market data delivery. It requires a peering relationship between the provider RP and the customer RP. Depending on the number of customers served by the provider there may be scaling considerations. A multi-layer peering hierarchy would most likely be required.
- Redundancy Issues—Using PIM SM and PIM BiDir will leave the source redundancy up to the server side of the application. The standby server can monitor the stream from the primary server by subscribing to the multicast group. If the stream stops for longer than a defined time the secondary server can start transmitting. The traffic will be forwarded to all the receivers on the *,G (shared tree) without any additional control plane activity. Alternatively, the standby server can send periodic keepalives to maintain the S,G state when it is in standby mode so that an S,G mroute will already be established. PIM BiDir does not have this issue; every packet will be delivered to the receivers without any control plane involvement. There is more on this issue in the "Intermittent Sources" section on page 28.

PIM SSM Joins or S,G Joins

PIM SSM would be the ideal market data delivery solution provided when the following conditions are met:

- Service is unidirectional—The multicast service is unidirectional in nature. This is typically the case since the retransmission requests are sent using unicast.
- Ability to discover source addresses—The application has an in-band or out-of-band service to discover the source addresses so that the client can join toward the source.
- Global source addresses—The provider uses a global source address that can be delivered end-to-end. If the provider cannot use global source addresses, then there might be an address collision and some type of network address translation (NAT) will need to be used which might reduce performance and latency.
- Support for IGMPv3—The application, the client OS, and the network must support IGMPv3. There are a number of transitional features such as SSM Mapping, but these might be problematic to implement as part of the service.

Redundancy Issues

PIM SSM places the source redundancy requirements up to the network or the receiver side of the application.

- Redundancy on the host side—Hosts join primary and secondary servers.

  Market data consumers can join to both the primary and secondary A and B feeds which are sent with different S,G mroute entries. The host then would be responsible for arbitrating between the primary and secondary data streams.

- Redundancy on the network/server side—Anycast source.

  This method would allow the receiver of the market data stream to just join one channel—one S,G—and the redundancy will be handled by the network. The idea is that the primary and backup server are both transmitting the same data stream in the network at the same time using the same S,G.

  The servers are then directly connected to two different routers in the network that are configured with the same source subnet address. One of the data streams would be forwarded down to the end users and the other one would be dropped due to the lower preference of the route—thereby failing RPF checks. If one of the servers was to fail, the carrier line would drop and the source route of the primary server would be removed from the routing tables. Then the standby server would have the better route and the standby stream would pass the RPF checks and be forwarded to the end users.

  An alternate way Anycast Source can be implemented is to connect the sources to dedicated streaming devices. These devices would run a scaled down version of the RIP protocol and would advertise a 32-bit host-route for the source address. If the source fails or stops sending for a period of time, then the streaming device would withdraw the source route and the standby server would then have a better route metric and be the preferred route.

  The Anycast source method has been proven to work well in a video broadcast environment. It might not be ideal for market data due to number of streams (usually hundreds and sometimes thousands) and the fact that many streams are intermittent in nature. It would be difficult to gauge when to fail over to the standby server if the source periodically stops for long periods of time. However, if the characteristics of the market data application require that the streams are always running, it might be a viable option.

Considerations with PIM Joins—Provisioning and Security

Several exchanges have expressed concerns about denial of service (DoS) attacks with PIM joins. These joins would need to be processed on the service edge and could create a high CPU load on the edge routers. A combination of control plane policing (CoPP), multicast boundaries, and firewalls can minimize this risk.

A related issue is the enforcement of entitlements on the service edge. There needs to be some way to provision the individual services on a per customer basis and ensure that customers cannot access other services.

Hybrid Design—Static for Key Groups/Rest Dynamic

A compromise approach might be the best solution. The core market data groups would be nailed up statically and every customer would receive these data streams. The rest of the groups, including the retransmission groups, would be dynamic.

This hybrid approach could also be combined with the cable model to allow for cookie cutter configuration options.

Market Data Distribution Design

Today's market data distribution design must be resilient, redundant, reliable and efficient to the degree that not a single packet will be lost during a link or router failure. All of this must be performed in a secure fashion that delivers the data streams with the lowest possible latency.

The designs that follow for the exchange, FSP and brokerage environments are built with those requirements in mind and meet these goals with specific design principles—such as live-live/hot-hot data streams and physical path diversity.

Each of market data distribution component is typically a separate administrative domain and can use an independent method for multicast delivery. For example, the exchange can use PIM BiDir to deliver the market feeds to the edge of their network and a brokerage can receive the feeds using PIM SM.

Market data feeds are typically few-to-many applications. In other words there are several sources sending to each multicast group. The service model is that the brokers can subscribe to a particular multicast group and receive a defined subset of the instruments. The exchanges divide up those instruments over a few servers. This allows the exchange to have the flexibility to move instruments from one server to another and still be delivered with the same multicast group. Additionally, this system can be used for redundancy. When a server fails, the brokerage can immediately bring up a new server in a way that is transparent to the end user.

The design principles described in this document are consistent with the *Financial Services Task Force Report* that was produced by the National Security Telecommunications Advisory Committee (NSTAC). The report offers recommendations to the financial services industry to improve business continuity plans. The report can be found at the following link:

http://www.ncs.gov/nstac/reports/2004/Financial%20Service%20Task%20Force%20Report%20(April%202004).pdf

The Exchange Network

Exchanges produce two live redundant data feeds, called *A* and *B* feeds. Each of these feeds contains the same payload, but each is transported using different source and multicast group addresses.

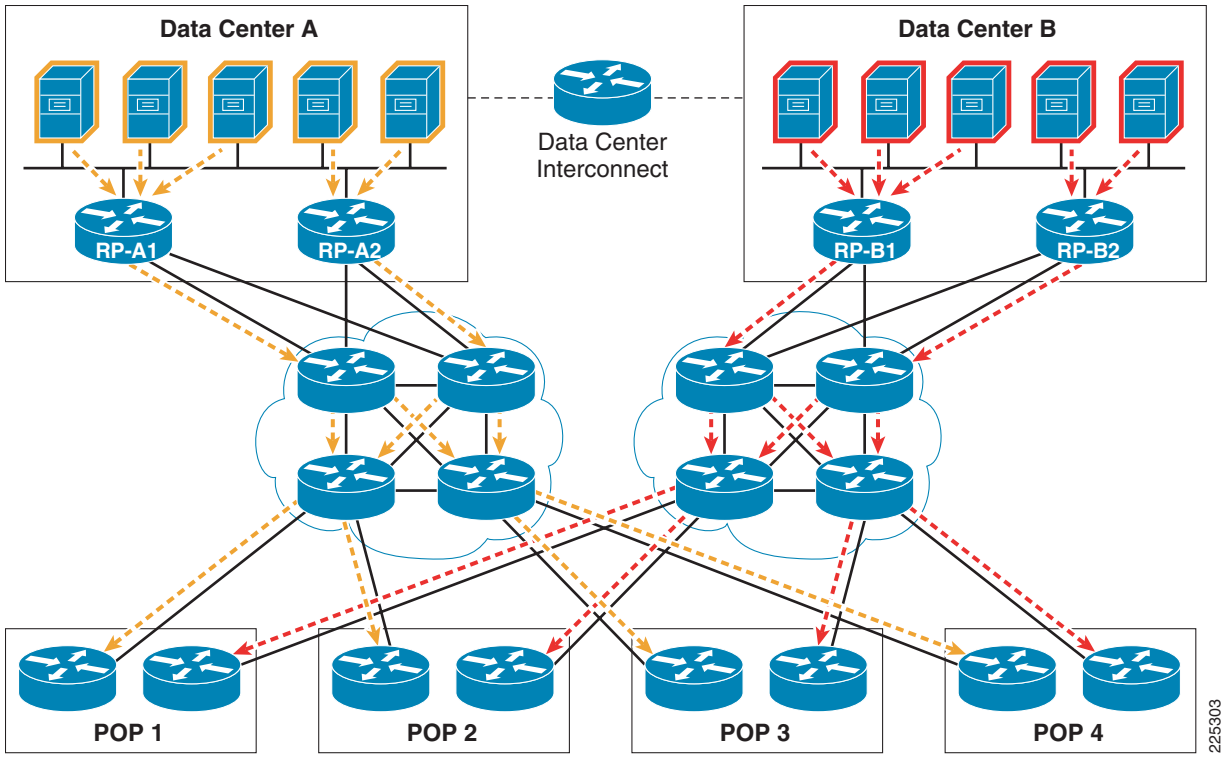
The A and B feeds are typically created from two data centers connected with a backend interconnection. The two data centers can be logically separated in the same room or can be in a nearby building. They cannot be physically separated by a large distance due to latency requirements.

In order to ensure that both of these feeds do not fail at the same time, the exchanges use physical path diversity in their distribution network. Ideally, this includes separate infrastructure for each of the data paths. The physical separation implies a separate control plane which also adds reliability.

The exchange might have several points of presence (PoP) located throughout the region—as well as the major global financial centers to deliver their products. Path diversity is extended to these PoPs when it is feasible and economically sound.

Figure 5 illustrates the high-level design of the exchange data center and the distribution network.

Figure 5 Exchange Data Center and Distribution Network



The market data feeds can be delivered either through a native IP unicast/multicast design or an MPLS configuration. The MPLS configuration is typically used to achieve service separation with MPLS-VPN and MVPNs.

The larger exchanges offer many different services. They are essentially several different exchanges co-located together. The exchanges need a way to provision any of these services for their customers—while keeping customers separated and distinct. Service separation is a key method to handle this issue.

Native Unicast and Multicast Design

This section contains a detailed illustration of a stock exchange configuration using the recommended design principles.

Data Centers A and B are configured from different address spaces. This makes route manipulation for path diversity a straight forward process and simplifies troubleshooting.

The network is physically configured so that traffic to Data Center A and Data Center B will always follow different paths to the customer edge routers.

For IP multicast routing this can be accomplished with routes that point to back to the RPs for each feed. PIM BiDir is recommended for this type of market data distribution as discussed in other sections of this document.

In Figure 5, RP-A1 and RP-A2 (not explicitly shown) are the RPs for Data Center A. RP-B1 and RP-B2 are the RPs for Data Center B.

The IP multicast address range for our example is as follows:

- 233.255.255.0 to 233.255.255.127 for the A feeds

- 233.255.255.128 to 233.255.255.255 for the B feeds
- RP-A1 and RP-A2 will be the RPs for the A feeds
- RP-B1 and RP-B2 will be the RPs for the B feeds

We represent the RP address of the A feeds as *RP-A-addr* and the B feeds as *RP-B-addr*.

Each pair of RPs are configured as Phantom RPs. This type of configuration is described in the *Bidirectional PIM Deployment Guide*. There is a link to the deployment guide and further discussion about PIM BiDir in the "IP Multicast Protocol Options" section on page 7.

Each PoP would have routes that point back to the RPs in Data Center A or Data Center B using distinct paths. The RPs would be defined as follows:

```
ip pim rp-address RP-A-addr 1 override bidir
ip pim rp-address RP-B-addr 2 override bidir

access-list 1 permit 233.255.255.0 0.0.0.127
access-list 2 permit 233.255.255.128 0.0.0.127
```

This design enables the exchange to deliver multicast market data services without any of the potential control plane issues associated with PIM SM. However, there might be brokerages that are unable to initially support PIM BiDir end-to-end and that require a transitional strategy.

The **bidir-neighbor-filter** command can be used on the customer facing interfaces of the exchange to ensure that designated forwarder (DF) election will occur even if the downstream router does not support PIM BiDir. It also offers some control plane security since it will demand that the exchange router will always be the DF.

```
interface GigabitEthernet3/15
 ip pim bidir-neighbor-filter 9
access-list 9 deny any
```

MPLS-VPN and MVPN Design

An alternate design for the exchange would be to have the core of the distribution networks running MPLS. Service separation for unicast would be achieved with MPLS-VPN and multicast would be accomplished with MVPNs.

The physical network would be designed using the same topology. Many of the same features would be used as in the native IP multicast design. For example, PIM BiDir would still be used inside the VRF.

The main difference is that there would now be PE routers in the data center and at the PoP. There would be one VRF for the A feeds and another VRF for the B feeds.

A combination of physical network separation and route manipulation will pull the traffic for one VRF to the A-PoP router or the B-PoP routers.

MVPN uses MDTs which encapsulate the multicast within the VRF inside native multicast in the core. The standard flavor of multicast today for MVPN MDTs is PIM SSM, although PIM BiDir can be considered if it is supported by the router platform.

The key method used to engineer the path of an individual VRF is to specify a different loopback address for each VRF. This can be accomplished with the **bgp next hop** command:

```
ip vrf RED  bgp next-hop loopback 2
```

This causes the source address for the default and data MDTs of VRF RED to be the IP address on the loopback 2 interface.

Now that the source addresses for all the MDTs associated with this VRF are different from other VRFs the PEs at the bottom of the network can be configured to prefer one data path over another through normal unicast routing methods such as floating static routes. This can be used to direct one set of streams to flow down the left side of the network illustrated in Figure 5 and the rest down the right side of the network.

Provisioning feeds from the exchange can be done in one of two ways.

- Static Provisioning—This option is for FSPs that require the feeds all the time. These FSPs have no hosts locally to subscribe to the feeds and the FSPs must distribute the data streams to their customers without having to rely on end-to-end dynamic joins.
- Dynamic Provisioning—For brokerages that want to manage their own bandwidth consumption. The recommended method for dynamic provisioning is PIM joins which is described in detail in the "Dynamic Forwarding" section on page 13.

This design ensures that the exchange will dependably deliver the multicast market data feeds with the typical behaviors seen today in the data streams—such as intermittent sources.

The customer facing interfaces should be configured with the appropriate safeguards as discussed in the "Edge Security" section of the *IP Multicast Best Practices for Enterprise Customers* document.

The Financial Services Provider

The design requirements for a FSP are similar to a standard SP, except FSPs have higher reliability and more strict latency requirements. The prevailing trend in SP networks today is MPLS. For an integrated unicast and multicast solution, this means MPLS-VPN and MVPN.

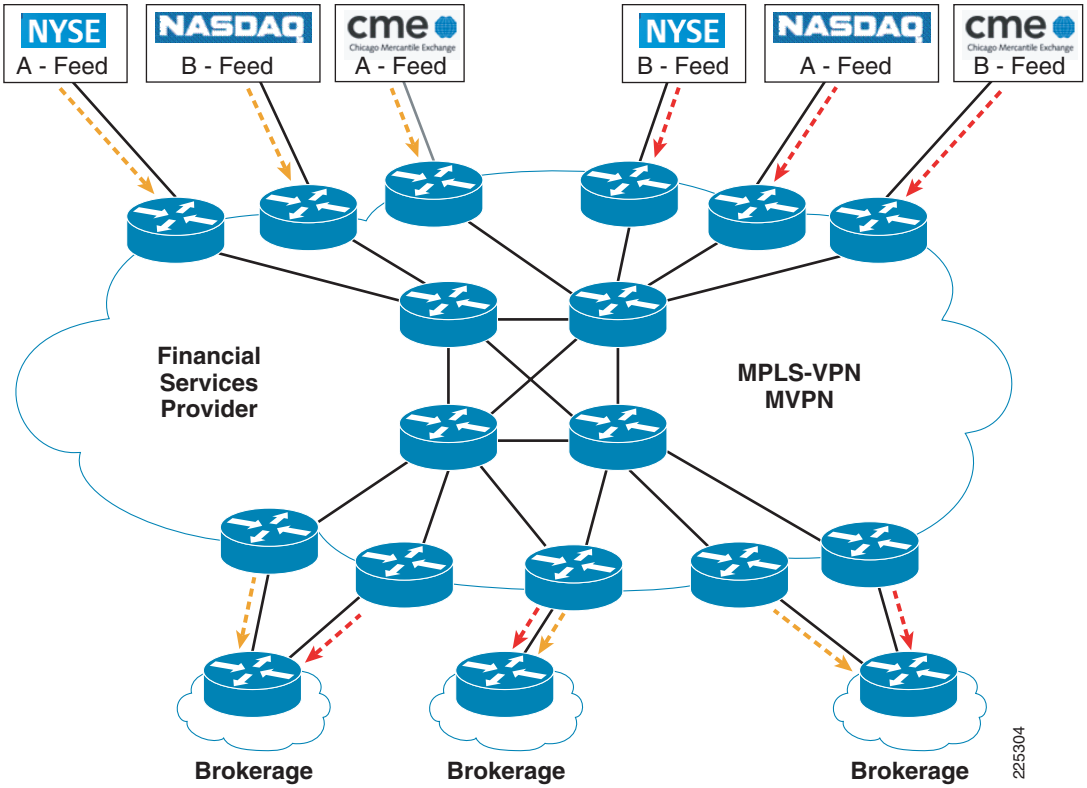
The market data streams will typically be forwarded statically into the FSP as noted immediately preceding description of exchange provisioning.

There are two main approaches for the FSP network design: a converged network and separate cores.

Converged Network

In the converged core design (see Figure 6), traffic shares a single physical topology. Path separation is accomplished by some type of traffic engineering. In terms of native multicast traffic this generally means route manipulation.

Figure 6 Financial Services Provider—Converge Network

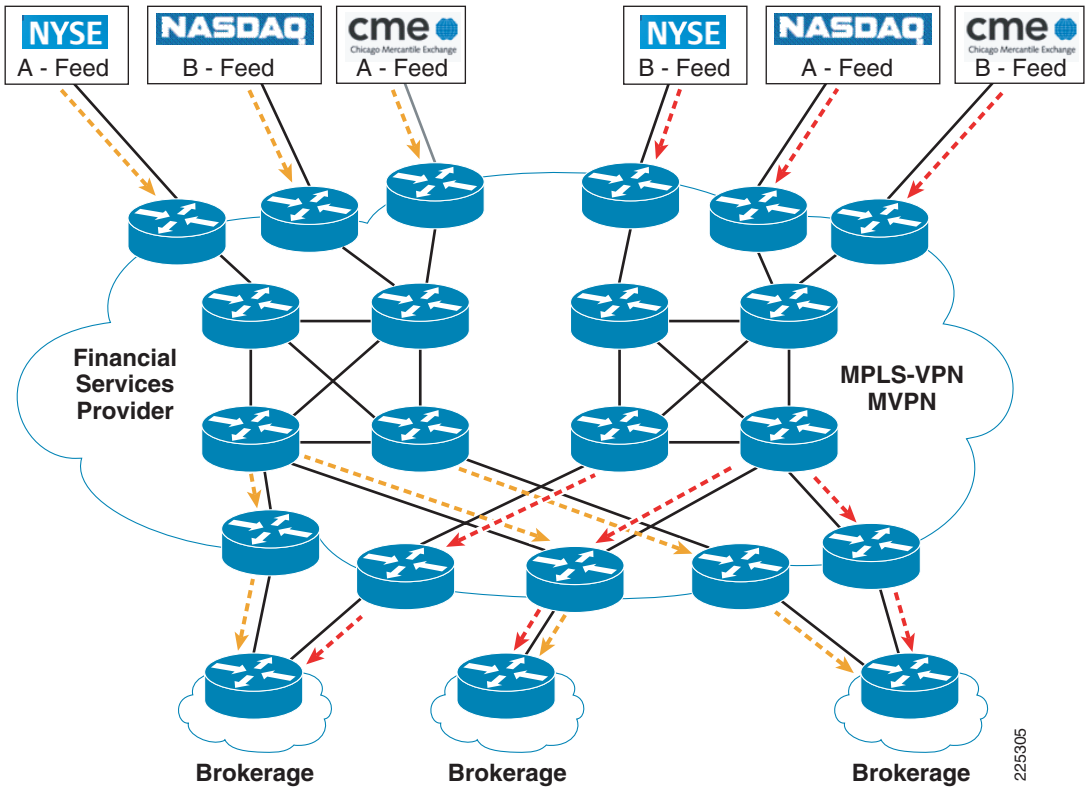


In the future, the traffic engineering for multicast might be done with Label Switched Multicast (LSM) and point-to-multipoint traffic engineering (P2MP TE).

Separate Cores

In the separate core design (see Figure 7) there will be no requirement for traffic engineering. The path separation will be accomplished by physical network separation.

Figure 7 Financial Services Provider—Separate Cores Design



The provisioning method in the FSP can be either static or dynamic as is the case for the exchange environment. The dynamic subscription model is either IGMP or PIM joins—both are described in detail in the "Dynamic Forwarding" section on page 13.

The customer facing interfaces should be configured with the appropriate safeguards as discussed in the "Edge Security" section of the *IP Multicast Best Practices for Enterprise Customers* document.

The Brokerage Network

The brokerage network is traditionally divided into two parts: the back-office network and the front-office network. The back office network includes the feed handlers, the OMS, the FIX Engine, the Algorithmic Trading engines, and all the network infrastructure to support those functions. The back-office infrastructure is typically protected from the external network connections by firewalls and strict security features. The front-office network contains the end user trading systems, compliance and performance monitors, and all the network infrastructure to support those components.

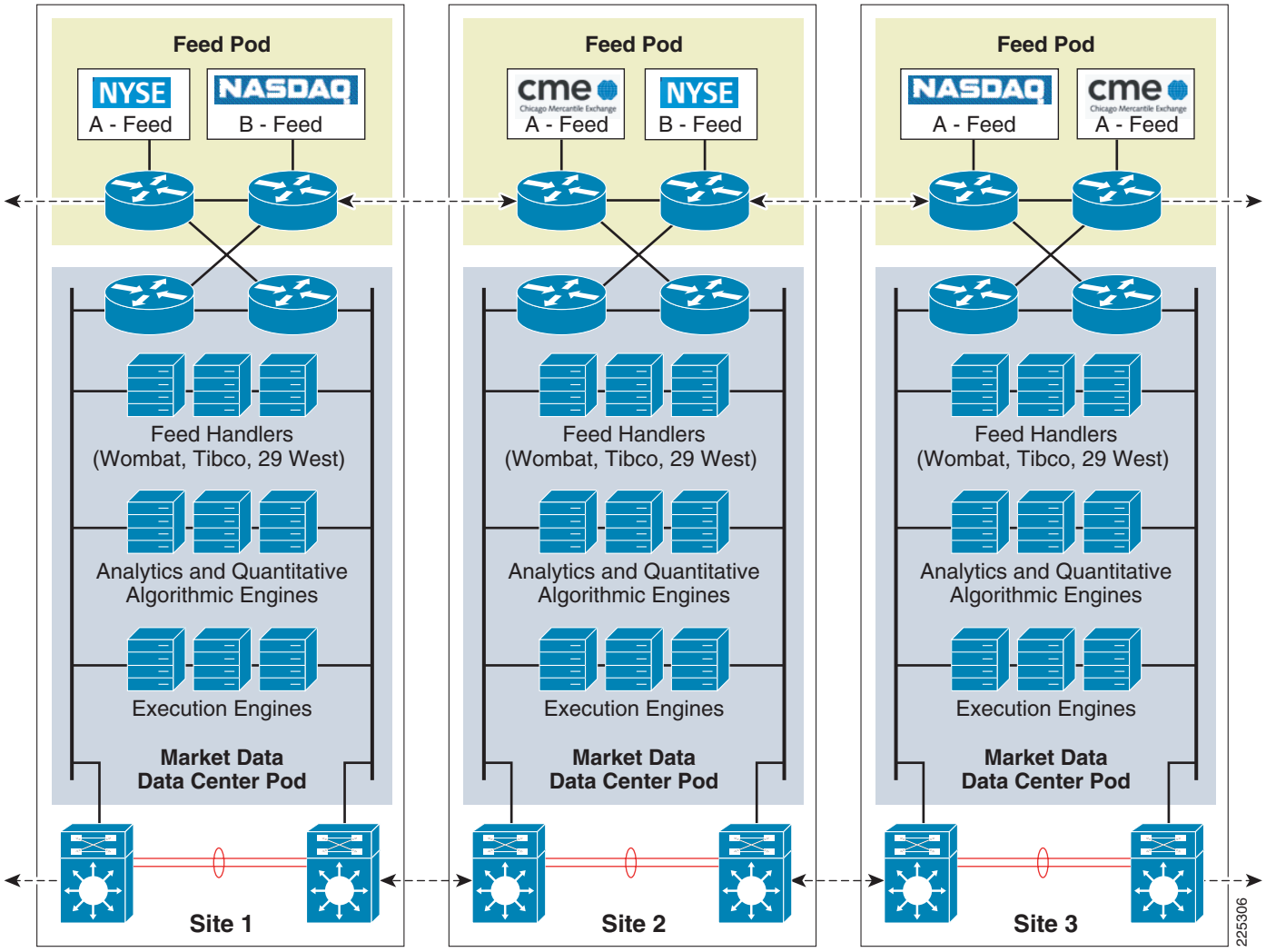
Back-Office Network

The large brokerages houses are typically spread across multiple sites for many reasons, including real estate space, power limitations, business continuance design, and so on.

The design presented in Figure 8 leverages the use of multiple sites to create greater efficiencies in space, power, and business continuance. Each remote site has a *feed pod*. These feed pods are independent and receive a different subset of the total market data feeds which allow for source diversity.

The feed pods are connected with Dense Wavelength Division Multiplexing (DWDM) or 10-Gigabit Ethernet so that each pod has access to all the feeds with the minimum latency penalty.

Figure 8 Brokerage Market Data Distribution



The recommended method of multicast delivery for these feeds is PIM BiDir. The routers in each pod are the RPs for those feeds. This is a very straight forward and elegant configuration since a different multicast address range is received in each pod.

The feeds are then pulled down into the market data DC pods, which process data and execute the trades. Each of the market data DC pods has access to all the feeds in the firm. This design allows the firm to position people and servers at any of the three locations—whichever makes sense for space and power consumption purposes. There is no penalty for having the servers in one location or another.

There are also connections between sites on the front office side of the market data DC pods. This allows specialized analytics to be shared by trading applications throughout the firm.

The failure of any single link causes a minimal interruption of the data flow.

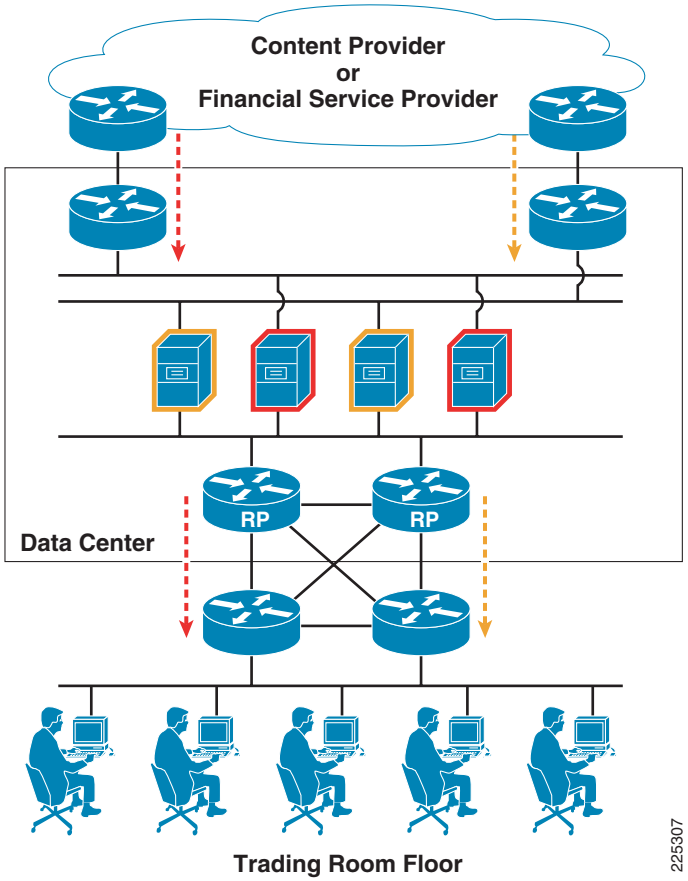
Front-Office Network

In each site, there is a front office network to distribute the analytics and market data to the various trading applications and human traders. See Figure 9.

The data distribution is typically handled by a separate infrastructure than in the back office network.

In terms of multicast, this means a different set of RPs and usually a different set of multicast addresses. The servers in the market data DC pods read in the raw feeds and then republish with a different set of sources and groups.

Figure 9 Brokerage Data Center and Trading Floor



If PIM BiDir is being used, then path diversity is achieved with two sets of RPs—each responsible for half the multicast groups. This is the same technique used in the exchange data center described previously.

If PIM SM is being used for multicast forwarding, then path diversity and reliability is achieved with a combination of alternating designated router (DR) priority and dedicated networks. An example of this approach is explained in the "Alternating DR Priority" section of the *IP Multicast Best Practices for Enterprise Customers* document.

Common Best Practices for Market Data Delivery

A reliable network to deliver market data using multicast is only as reliable as the underlying Layer-2 and Layer-3 unicast network. A description of the best practices for a reliable network IP multicast design for enterprise and financial customers is explained in detail in the following document:

- *IP Multicast Best Practices for Enterprise Customers*

This document provides the best methods for optimizing multicast delivery by focusing on the following design goals:

- Resiliency
  - Path diversity
  - Redundancy
  - Load sharing or splitting
- Latency
- Security

Topics covered in the *IP Multicast Best Practices for Enterprise Customers* include:

- Using Point-to-Point links in the core
- Tuning at Access Layer edge
- IGP tuning
- IGMP Snooping
- Choosing the right multicast groups
- PIM query-interval tuning
- Register rate limits
- MSDP timers
- Multicast Stub Recommendation
- Static RP vs. AutoRP Listener
- Anycast RP for PIM SM
- Phantom RP for PIM BiDir
- Alternating DR Priority
- Multicast multipath for Load Splitting
- Edge Security

Design Issues for Latency

There are many factors that affect latency in receiving market data and executing transactions. Some of these factors are caused by physical and geographical limitations and might be beyond the scope of this publication's focus, but they need to be understood and minimized.

The main areas that can increase or decrease latency in market data delivery are as follows:

- Propagation delay
- Processing and serialization delay

- Smaller packets—Lower bandwidth and compression
- Queuing delay
- Transport layer and TCP/IP stack
- Middleware characteristics
- Application architecture
- Server/OS architecture
- Security and compliance

These topics are discussed in detail in the following whitepaper:

- *Design Best Practices for Latency Optimization*

The approach to minimize latency must be a holistic effort that reviews the entire market data system from end-to-end and focuses on reducing latency throughout the design.

Note The areas with the most room for latency improvement and that can have the greatest impact are the application and middleware components.

Application Issues

This section addresses the following application considerations:

- "Number of Groups and Channels to Use" section on page 26
- "Intermittent Sources" section on page 28
- "RTCP Feedback" section on page 29
- "TIBCO Heartbeats" section on page 29
- "Fast Producers and Slow Consumers" section on page 29
- "Network Design with Reuters Market Data System" section on page 29

Number of Groups and Channels to Use

Many application developers consider using thousands 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. Typically, several messages are sent in each packet that is 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 will quickly reach the point of diminishing returns—there will be more overhead sent than actual data.

Additionally, there is a practical limit to the number of groups to which a receiver can subscribe. Previously, the limit that the NIC MAC filtering could support was 50 groups. Today, it might be higher, but, nonetheless, after a point the NIC card goes into promiscuous mode and all the filtering would be done at the kernel. This might not be as efficient as dropping the packets at the driver level.

If IGMP snooping is configured on the receiver ports, then only the data from groups to which the receiver has subscribed will be delivered to that port. Cisco switches can filter several thousand groups on each switchport but there are practical limitations.

Perhaps the biggest limitation is the IGMP stack on the host. The host needs to respond to queries for each group at least once per minute. When you reach thousands of groups, this is a limitation—especially when the host receives a general query and needs to respond to each group to which it has subscribed. If there are many hosts connected to a single switch, processing the thousands of reports from all the hosts will be a limitation.

The application developers need to find a reasonable compromise between the number of groups and breaking up their products into logical buckets.

Consider the NASDAQ Quotation Dissemination Service (NQDS) for example. The instruments are broken up alphabetically as follows:

- 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, which breaks down as illustrated in Table 3.

Table 3 Breakdown of NASDAQ TotalView Service

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 does allow for straight-forward network/application management, but does not necessarily allow for an optimized bandwidth utilization for most users. A user of NQDS that is interested in technology stocks, and that would like to subscribe only to CSCO and INTL, would need to pull down all the data for the first two groups of NQDS. Understanding the way the users will be pulling down the data and then organizing that into the appropriate logical groups will optimize the bandwidth for each user.

In many market data applications, optimizing the data organization would be of limited value. Typically customers will bring in all data into a few machines and filter the instruments. Using more groups is just more overhead for the stack and will 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 multicast streams are forwarded based on destination address, but the UDP ports can be used to aid in filtering the traffic.

The other extreme would be to use just one multicast group for the entire application and then have the end user filter the data. One multicast group may be sufficient for cases in which all hosts would be receiving the majority of the financial instruments.

Intermittent Sources

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

There are a few scenarios in which the outage can be more severe. One case would be if the source starts sending again right around the 3.5 minute mark. At that point state has started to time out in some of the routers along the data path and there might be inconsistent states in the network. This could create a situation in which data from the source would be dropped for as long as a minute until state clears out and then is created again on the intermediate routers.

On the Cisco 6500 and Cisco 7600 there are some additional platform-specific issues with intermittent sources. Multicast flows are forwarded by hardware shortcuts on the Policy Feature Card (PFC) or Distributed Forwarding Card (DFC). The statistics from these flows are maintained on the PFC/DFC and are periodically updated to the Multilayer Switch Feature Card (MSFC). By default this update happens every 90 seconds, but can be lowered to every 10 seconds by lowering the **mls ip multicast flow-stat-timer** value to *1*. Due to this delay in receiving the latest flow statistics for individual multicast streams, it is possible that a source could go quiet for three minutes and then start transmitting again; the mroute state will still be removed for no activity. This could cause an outage of an active stream for one-to-two minutes, depending on the state of the network.

The following are the best solutions to deal with intermittent sources: PIM BiDir or PIM SSM; null packets; periodic keepalives or heartbeats; and, S,G expiry timer. Each is described briefly in the short discussions that follow.

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.

Neither of these optimizations of the PIM protocol have any data driven events in creating forwarding state. That means that as long as the receivers are subscribed to the streams, the network will have 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 used to ensure that a forwarding state is created is to send a burst of null packets to the multicast group before the actual data stream. The application needs to effectively ignore these null data packets so they do not affect performance. The sources only need to send the burst of packets if they have been silent for more than three minutes. A good practice would be to send the burst if the source was silent for more than one minute.

Many financial applications send out an initial burst of traffic in the morning and then all well-behaved sources will not have a problem.

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 will never expire. A typical timer for the heartbeat message is 60 seconds.

S,G Expiry Timer

Cisco has made a modification to the operation of the S,G expiry timer in Cisco IOS. There is now a CLI command option to allow the state for a S,G to stay alive for hours without any traffic being sent. This fix was in response to a customer request in a PIM SM environment to maintain the state and not fall back to *,G forwarding. The following is the relevant command: **ip pim sparse sg-expiry-timer**

It is described in the associated command reference:
http://www.cisco.com/en/US/docs/ios/ipmulti/command/reference/imc_04.html#wp1018443

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 Real-time Transport Protocol (RTP) is the use of Real-Time Control Protocol (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.

Receivers can be implemented and configured to send RTCP feedback using unicast. This has the advantage of allowing the server to still receive the feedback, but not create all the multicast state.

TIBCO Heartbeats

TIBCO Rendezvous has had the ability to use IP multicast for the heartbeat between the TIBCO Information Caches (TICs) for many years. However, there are some brokerage houses that are still using very old versions of TIBCO Rendezvous 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 TIBCO Rendezvous should be phased out in favor of the IP multicast-supported versions.

Fast Producers and Slow Consumers

Many servers providing market data are attached at Gigabit Ethernet or 10-Gigabit Ethernet speeds, while the receivers are attached at different speeds—usually 100 Mbps or 1 Gbps. This creates the potential for receivers to drop packets and request retransmissions. The result is increased traffic that the slowest consumers cannot handle—exacerbating the cycle of congestion.

The solution is some type of access control in the application that will limit the amount of data that one host can request. Quality-of-service (QoS) and other network functions can mitigate the problem, but ultimately the subscriptions must be managed in the application.

Network Design with Reuters Market Data System

Reuters Market Data System (RMDS) applications may use a combination of unicast, broadcast, and multicast traffic. For example, information concerning a particular financial instrument could be spread across all three delivery mechanisms. These packets might follow different paths through the network and can potentially arrive out of order. Ideally, this situation must be avoided because some RMDS applications are sensitive to out-of-order packets.

The path through the network needs to be convergent or polarized for unicast, broadcast, and multicast traffic. The following features below can affect the forwarding paths:

- Multicast multipath—This feature is used to load balance multicast traffic between equal-cost neighbors. Normally, PIM joins are forwarded to the PIM neighbor with the highest IP address—if there are multiple equal-cost alternatives. When this command is enabled the PIM neighbor will be selected pseudo-randomly from the available equal-cost neighbors, resulting in load-splitting of traffic from different sources.

  Multicast multipath should be disabled to guarantee that multicast traffic will follow the equal-cost path with the highest IP address. This feature is not enabled by default.

- Cisco Express Forwarding (CEF) per-destination mode—Unicast routing can use a number of different methods to forward traffic. The unicast forwarding method must be verified as being compatible with the multicast forwarding path.

  For example, CEF can be configured with per-destination or per-packet forwarding modes. The per-destination mode guarantees that all packets for a given destination are forwarded along the same path. In most cases, the per-destination option is the better choice. The per-destination mode is the default with CEF.

- Port channel hashing—Port channeling is used to combine multiple physical channels together into one logical channel. The physical path that any one traffic stream will take is dependent on a hashing algorithm.

  The options available for the hashing algorithm are different depending on the switch platform and software version, but a common load-balancing policy for the hash is a combination of the source and destination IP address of the traffic stream.

  Since RMDS traffic for each financial instrument is sent from one source address to destination address (unicast, broadcast, and multicast addresses) it is possible that different hashes will be selected for each packet stream.

  The number of different paths chosen for a particular source can be minimized by choosing a hashing algorithm that only uses the source address. Therefore, a hash that takes only the source address into consideration would work best with RMDS. This can be configured globally in Cisco IOS with the following command: **port-channel load-balance src-ip**

Common Best Practices for Multicast Delivery

The following multicast delivery best-practice guidelines are addressed in this section:

Live-Live or Hot-Hot

The term *Live-Live* (also referred to as *Hot-Hot*) refers to the method of sending redundant data streams through the network using path separation and dedicated infrastructure. For example, an A copy of the streams would be sent to one set of multicast groups and a B set of streams will be sent using a second set of multicast groups. Each of these groups will typically be delivered using a parallel, but separate, set of equipment to the end user with complete physical path separation.

One of the main justifications for Live-Live is the requirement to not lose a single packet in the data stream. When Live-Live is implemented with full physical path separation and redundant server infrastructure for the A and B streams, it can provide resiliency for a failure in a servers or in the network. Live-Live can also be implemented in a converged network (no physical path separation) and there will still be resiliency for the servers, but not necessarily for a network failure.

Financial services have been doing this for many years. Live-Live is usually preferred over a reliable multicast solution such as TIBCO Rendezvous, 29West, or Pragmatic General Multicast (PGM). One of the limitations with a reliable multicast solution is that the retransmissions and overhead introduce latency and delays. In the finance world today, there is an arms race to reduce latency. Every millisecond is worth money and financial services organizations want reliability, but not at the expense of latency. The Live-Live approach will allow for the minimum possible latency without the need for retransmissions.

A-B In C Out

Many brokerages receive both the A and B streams in their data centers and then feed handlers are used to arbitrate, normalize, and clean the data stream. The post-processed data stream is then injected into a messaging bus that feeds the core infrastructure of the trading applications. The message bus typically uses a reliable multicast transport protocol, such as TIBCO Rendezvous or 29West.

Some brokerages position algorithmic trading engines in parallel with the feed handlers and process the raw feeds directly. This requires the trading engines to be able to parse the raw streams and clean the inherent transmission errors. This is a high-maintenance procedure and is usually only performed by the largest brokerages.

A-B In A-B Out

In this flavor of Live-Live, the A and B streams are first brought into the data center, arbitrated, normalized, and cleaned and are then republished using a completely different set of A and B streams.

- Usually, these new A and B streams are not Live-Live in that the end users do not have to perform arbitration, but rather they are both forwarded using a reliable messaging bus and the infrastructure is completely separate. Usually half the end users would receive the A stream through the left side of the infrastructure and the other half would receive the B stream through the right half of the infrastructure. This is the same design described in the “Alternating DR Priority” section of the *IP Multicast Best Practices for Enterprise Customers* document.

If this design is implemented properly, the data will continue to operate with a failure in the provider at the same time as a failure in the brokerage. For example, even with a failure in the provider with the original A feed and a failure in the brokerage with the new B feed, half the end users should still be able to receive the new A feed. Alternatively, with the A-B in C out strategy, if the C feed fails everything stops.

Multisite

Some brokerages combine the benefits of Live-Live with the added strategy of using multiple sites. For example, some brokerages receive the A feed in New York and the B feed in New Jersey. These streams are then processed with one of the above methods and then republished.

Brokerages usually cannot justify receiving the A-B in New York and then A-B in New Jersey again. When this can be justified, it leads to some interesting republishing schemes.

There are several ways to handle republishing the stream in this situation. One way would be a hot-standby method. The New York stream would be transmitted everywhere under normal circumstances. The servers in New Jersey would listen to the stream and then only start publishing if then server in New York fails. There are more complicated schemes that have been considered in which each site receives the local stream all the time. When there is a failure then the servers switch to the stream from the other site and republish that data.

Considerations with Using PIM BiDir

A and B streams are typically published to different groups to maintain path and infrastructure separation. This requirement means that separate sets of Phantom RPs are needed for redundancy and path separation.

One set of RPs is for the A groups and routing will be configured to direct the PIM *,G joins to the RPs up the left hand side of the network. The other set will be for the B groups and the unicast routing will direct the PIM joins up the right hand side of the network.

The only route that must be manipulated with PIM BiDir is the route to the Phantom RP.

There is a limit of four RPs with the current hardware design on the Catalyst 6500. This should be enough to implement Live-Live, but the limitation needs to be considered.

Considerations with Using PIM SSM

PIM SSM has no concept of an RP. The PIM S,G joins are sent toward the source address. Therefore the routes to the source subnet must be manipulated to send the joins to the A sources or the B sources. The routes to the A sources must be directed up the left side of the network in Figure 5 and the B sources will be directed up the right side.

Considerations with Using PIM SM

Live-Live implemented with standard PIM SM has both the route engineering requirements of PIM BiDir and PIM SSM. It requires two sets of Anycast RPs for the A and B streams and S,G mroutes for the source streams. The routes for the RPs are configured the same as with PIM BiDir. The routes for the source subnets will be preferred in the same manner as PIM SSM.

Alternatively, if PIM SM is implemented with shortest-path first (SPT) threshold infinity, there might be no need for the source routes—there will only be *,G PIM joins sent from the designated routers toward the RP. This configuration can be tricky across multicast domains. If the provider does not have control over the designated routers, then there is no way to enforce that SPT threshold infinity is configured and that only PIM S,G joins will be sent. Implementing SPT threshold infinity in the enterprise will require the configuration on every LHR.

Retransmission Models

IP multicast uses UDP which is an unreliable protocol at the network layer. In order to guarantee the delivery of every message, there must be a method higher up in the stack. This section discusses the following retransmission options:

Reliable Multicast

Reliable multicast protocols have some type of retransmission scheme to deal with the cases of dropped packets. PGM and TIBCO Rendezvous (via TIBCO Reliable Data Protocol or TRDP) have similar retransmission schemes which involve using negative acknowledgements (NAKs) and report suppression.

PGM and TRDP have the ability to cache the stream and therefore limit the retransmissions to a local area. PGM uses a Data Local Repairer (DLR) and TIBCO Rendezvous has this ability in the RVRD function.

Reliable multicast schemes are typically deployed in the brokerage and not in the FSP or the exchange.

Live-Live

Live-Live does not generally need a retransmission scheme. The redundant streams are what guarantee delivery. However, there are still times when gaps exist in the data stream at the brokerage and retransmissions are then needed for part of the data stream. In those situations, methods described in the sections that follow are generally used.

Unicast Retransmission Request

Most exchanges that use Live-Live have the facility to accept a unicast retransmission request for a range of sequence numbers in the market data stream. The retransmitted packets are typically sent on a different set of multicast groups. Some exchanges offer the capability for these groups to be dynamically joined by the customers when they need a retransmission.

Replay of Entire Trading Day

The brokerages have applications, such as long-term trending analysis, that require a copy of every trading message. Many exchanges offer a replay of the entire trading day after the market close to meet this requirement. Typically, the replay begins at a predetermined time after the market close.

Replay for the entire day is becoming more of a problem as the markets are moving toward a 24-hour trading day. For example, CME is already trading 23.5 hours per day for five days a week. Replay of the entire trading day is not possible. CME uses unicast retransmission requests and sends the replay traffic to dynamically joined multicast groups.

Plain Old Telephone Service

There are several exchanges that are not setup for electronic requests for retransmission. The brokers must pick up a telephone and call the exchange to request a specific retransmission when there has been a gap in the data stream. The retransmitted packets can be sent using the same groups as the original data or on special retransmission channels. If the same groups are used then the duplicate packets are identified by their sequence numbers and timestamps.

Reliable Multicast vs. Live-Live

Reliable multicast and the Live-Live delivery model has several distinct advantages and disadvantages. The following notes summarize related considerations:

- Latency—Reliable multicast will introduce latency for overhead and retransmissions when compared to Live-Live.
- Licensing—The licensing fees for the messaging protocols which use reliable multicast can add up quickly with a large deployment. This might change in the future with the development of the open source message protocol AMQP.
- Live-Live—Ideal for unidirectional market data in which all the receivers will receive the full feed. A message bus with reliable multicast is ideal for complex applications with many hosts sharing information in a many-to-many situation. They fit different application environments, but there is some overlap.
- Content filtering—Reliable multicast messaging usually has the built in ability for some type of subject-based filtering which can limit the traffic forwarded to individual branches.

Service Segmentation

Service segmentation is a method by which network connectivity and application reachability can be divided into separate virtual silos. The network services in terms of control plane and data plane reachability are completely separate from one silo to another.

The requirements for network segmentation are applied equally to unicast and multicast. Many market data products have both unicast and multicast components and the segmentation would need to apply to both.

There are many reasons why financial organizations are implementing some type of service segmentation The key reasons are as follows:

- Path separation—Financial organizations that want to deliver redundant services in a Live-Live delivery model could benefit from service segmentation. It would allow them to bundle a collection of streams together and insure that they will flow down different paths in the network. The job of manipulating the services through the network becomes a much easier task when they are bundled together in their own block.

- Service provisioning—FSPs need the ability to provision whole groups of services to individual customers in an incremental fashion. Service segmentation will allow the providers to enable certain services and limit the customer to those services.
- Fault isolation—Service segmentation can help with reliability and troubleshooting.
- Closed user group extranet services—Many providers are offering extranet services to their customers. This arrangement allows a subset of customers to conduct business-to-business operations in a private environment.
- Partner networks—Many FSPs resell their services to partners. There are requirements to keep those networks separate from their other production services.

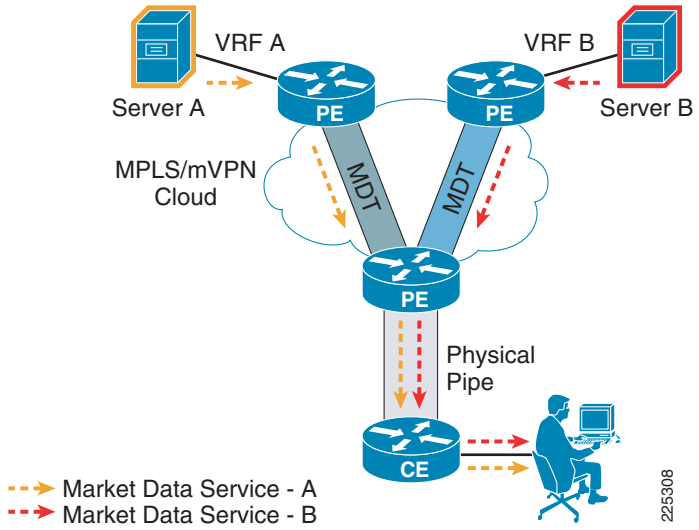
Multicast VPN

Multicast VPN (MVPN) is an approach for delivering segmented services for multicast feeds that is gaining traction in the financial community. It has already been implemented by the some exchanges and several brokerage firms are looking at it.

One method of provisioning segmented services is with 802.1Q and multiple VRFs. See Figure 10. The interface between the provider and the customer is divided into a number of subinterfaces with each one being in a different VRF. This is not considered MVPN because it does not require Border Gateway Protocol (BGP) and the traffic is not encapsulated.

Figure 10 Service Separation in Converged Network (Extranet)

f



The 802.1Q/multi-VRF approach allows the provider to offer multiple services and closed user groups without the use of an extranet. An extranet implementation would be an alternate option to offer the same type of services without using 802.1Q.

Service Translation

FSPs and brokerages often require transforming market data streams to different multicast address ranges. This section describes the main benefits and existing solutions to accomplish that goal.

Multicast Destination NAT

Multicast destination NAT has the following advantages:

- Address Collision—Many CPs and FSPs offer overlapping services in the administratively scoped multicast range (RFC 2365).
- Domain separation—Multicast destination NAT is a key tool used to forward data streams between two distinct multicast domains. Ideally, the customer edge router can appear like a receiver/host in one domain and a source in the second domain.
- Redundancy—NATing the multicast address allows the customer to create an A and B stream. The original stream can be forwarded through the network and a new copy of the stream with a different group address can also be forwarded.

Existing Solutions

The following existing solutions are available for FSPs and brokerages for transforming market data streams to different multicast address ranges:

- Multicast NAT—The feature that is called Multicast NAT today only has the ability to modify the unicast source addresses and it is not supported on hardware based platforms.

  It does, however, translate the unicast address in PIM control packets including joins and registers. A summary of the functionality for Multicast NAT can be found at the following link:

  http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a008009474d.shtml

  That link provides content describing how Multicast NAT works on Cisco routers.

  Limitations: Multicast NAT does not support translation of MSDP messages.
- Multicast helper-map—A destination NAT of multicast traffic is an unsupported side effect that is not recommended. This functionality might be removed in new Cisco IOS versions.
- Multicast service reflection—Multicast service reflection is a feature that was added to the software-based platforms. It is recommended in cases with moderate performance requirements.

  Relevant Cisco-provided docs can be found at the following location:

  - http://www.cisco.com/en/US/products/ps6441/products_feature_guide09186a008073f291.html

  Service reflection benefits include:

  - Source and destination NAT
  - Ability to create multiple copies of a stream
  - Ability to convert multicast streams to unicast or vice versa

  Limitations:

  - No **show** commands to see active translations
  - No management information base (MIB) support for translation information
  - Only supported on software-based platforms

  - Does not modify any multicast control plane traffic (PIM, AutoRP)

- Cisco Firewall Service Module (FWSM) on Cisco 6500/Cisco 7600—The NAT functionality on the Cisco FWSM includes the ability to translate the source and destination address of multicast streams.

  It is recommended for applications that require higher performance than multicast service reflection can provide.

  Known limitations:

  - Cisco FWSM does not NAT any control plane packets. This makes for some interesting use cases and troubleshooting.
  - No support for the multicast boundary command which could be used to filter the control plane packets.
  - No ability to make two copies of one stream.
  - No ability to convert unicast stream to multicast.

Messaging and Transaction Protocol Standards

Three protocols are applicable to the MDNA environment and are described briefly in the subsections that follow:

- Financial Information eXchange Protocol, page 37
- FIX Adapted for STreaming Protocol, page 37
- Advanced Message Queuing Protocol, page 38

Financial Information eXchange Protocol

The Financial Information eXchange (FIX) Protocol is a messaging standard developed specifically for the real-time electronic exchange of securities transactions. FIX is a public-domain specification that is the standard for financial transactions. More information can be found at the following resource:

http://www.fixprotocol.org/

FIX Adapted for STreaming Protocol

The FIX Adapted for STreaming (FAST) Protocol is an open specification that can be used for streaming market data. CME has started using FAST as their new data format.

The industry needs to work together to standardize on a common data format for market data. This will create huge economies of scale for exchanges and brokerage houses. FAST might be the best choice for a common data format. More information can be found at the following resource:

http://www.fixprotocol.org/fast

Advanced Message Queuing Protocol

The Advanced Message Queuing Protocol (AMQP) is an open standard for messaging middleware. By complying to the AMQP standard, middleware products written for different platforms and in different languages can send messages to one another.

AMQP has support from a number of key players, including Cisco Systems, Credit Suisse, Deutsche Borse Systems, Goldman Sachs, JPMorgan Chase Bank, Red Hat and 29West. More information can be found at the following resource:

http://www.amqp.org/