

MULTICAST ARCHITECTURES IN CROSSBAR-BASED ROUTERS

Replication Choices and Their Impact on Real-World HDTV

Table of Contents

Executive Summary	1
Introduction	1
General Packet Replication Process	1
Local Packet Replication	2
Distributed Packet Replication	2
Distributed Packet Replication Modes	4
Distributed Ingress Replication (DIR)	4
Distributed Fabric Replication (DFR)	4
Evolution of DFR	6
HDTV and Media Convergence	6
Distributed Tree Replication (DTR)	6
Conclusion: Pros and Cons of Competing Approaches	9
Appendix	11
Background	11
Software References	11
About Juniper Networks	11

Table of Figures

Figure 1: Multicast delivery tree	2
Figure 2: Generic distributed system	3
Figure 3: Distributed packet replication process	3
Figure 4: Distributed fabric replication	5
Figure 5: DTR Step 1 – Packet arrives at ingress	7
Figure 6: DTR Step 2 – Packet is replicated to two egress PFEs	7
Figure 7: DTR Step 3 – Process in step 2 is repeated	8
Figure 8: DTR Step 4 - Full multicast packet flow through the device	8

Executive Summary

IP multicast has a number of uses for Internet applications that are one-to-many or many-to-many in nature.

Low-bandwidth multicast has long been used in financial environments, and the ability to replicate packets with high speed is a necessity in business and residential broadband networks.

Of particular note is the fact that IP multicast is the most efficient way to distribute HDTV content over cable, digital subscriber line (DSL), or fiber to the home (FTTH) deployments. Growing demand for bandwidth and services has made multicast a required feature for virtually all modern router/switch platform designs, and created a new challenge for hardware engineers.

Since multicast traffic can enter and exit a router or switch through any port, it may need to be replicated within a line card at the port level, and also between line cards at the fabric level. Such a replication scheme has to be fast and quality of service (QoS)-aware—requirements that often seem to be contradictory.

In this paper, we take a closer look at system architectural choices and their advantages and tradeoffs. We then provide a comprehensive list of features in possible solutions, as well as “best-practice” applications (especially HDTV) suitable for them.

Introduction

Network equipment is designed to increase enterprise productivity and/or bring revenue to service providers. Therefore, a successful router/switch design is one that combines performance with robust quality of service (QoS) and a sustainable user experience—properties that are attributable to both unicast and multicast payloads.

It is worth noting that optimizing multicast performance is a very difficult task with larger, denser packet platforms.

Smaller designs can use one or two Packet Forwarding Engines (PFEs), where a single packet instance can be read several times from the same memory location, thus achieving high-speed multicast replication. The same option is not available in large carrier-class devices, where every slot may house several PFEs interconnected with a crossbar or switching fabric. While every PFE has its own packet memory, a multicast datagram can be effectively replicated between ports that belong to the same PFE. But when the multicast stream crosses the fabric, externalized packet replication is considered necessary to make sure that every receiver PFE gets a separate copy.

Since several PFEs can be part of the same multicast stream, multiple packet copies may need to cross the fabric. They also have to share the crossbar with unicast packets at different priority levels, which can be higher or lower than multicast.

For example, unicast voice may have a higher priority than multicast video, which is, in turn, a priority application over the best-effort unicast traffic. Such priority schemes must be strictly observed in router/switch designs in order to make them operationally tenable. Failure to do so could effectively mean that some application classes cannot be provisioned and this may hamper the ability to build efficient and profitable data networks.

Multicast replication performance is another dimension by which all commercial router/switch implementations are benchmarked.

While it is rarely required to run multicast at 100 percent line rate, a real-life mix of unicast and multicast packets must be delivered with sustainable speed and quality, without wrongful loss or excessive performance restrictions. The exact requirements are unique to each carrier, but nevertheless can be reasonably gauged by analyzing current and future multicast applications.

In the following sections, we will concentrate on achieving the right mix of multicast performance and QoS in router/switch designs.

General Packet Replication Process

Fundamentally, multicast is different from unicast in the transmission model: multicast is used in a one-to-many delivery scheme whereas unicast is a one-to-one communication. This makes multicast useful for situations like delivering content to IPTV subscribers where an operator will wish to save bandwidth when delivering massive amounts of identical data to many users, or for applications such as trade-floor operations where synchronizing single-sourced data streams across multiple receivers is critical. Internet multicast flow is often described as an (S, G) tuple, where S denotes the source, and G denotes the IP multicast group of interest to listeners (receivers). All receivers of a particular multicast flow (S,G) form a dynamic set R with cardinality |R| equal to the number of active

listeners. When crossing the network, the multicast stream (S,G) forms a graph, where multiple receivers may share common branches. Only a single copy of the packet is delivered along these branches thus eliminating the need for extra bandwidth in case of unicast delivery (Figure 1).

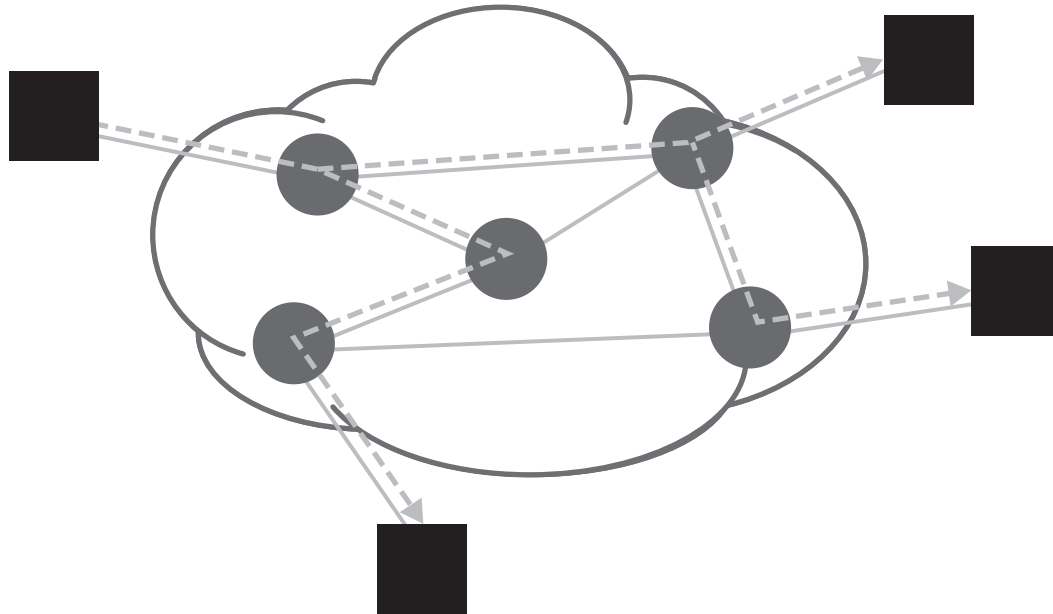


Figure 1: Multicast delivery tree

While general-purpose multicast operates on top of a sophisticated control plane for building and optimizing the packet delivery trees, in this paper we concentrate on aspects specific to router/switch hardware that forms the packet path. In such a path, multicast tree build-up is achieved by sending two or more packet instances down the branch links, thus physically replicating the packet received from the upstream source. Packet replication makes physical copies of the original packet and sends them to two or more locations, with this process repeating itself at every branch point until all members of receiver set R get their own instance. In the absence of shared media, this means that each packet that belongs to original stream (S,G) has to be replicated exactly $(|R|-1)$ times.

Local Packet Replication

At the router/switch Packet Forwarding Engine (PFE) level, multicast operation can be defined for a number of Layer 2 and 3 protocols, including IPv4, IPv6, Ethernet and MPLS. In their transmission model, they all equate to sending the incoming multicast packets to a list of the egress ports. If a network device sits on the branching node of a multicast delivery tree, the list contains two or more destination ports and copies have to be made. Regardless of the system architecture, the part that deals with packet replication forms an important part of the PFE.

When an ingress port and all egress ports reside on the same PFE, the replication process immediately achieves the goal of building the multicast tree via a process called "Local Replication." Since object replication within the same memory bank is a relatively low cost function, local packet replication is a fast and effective operation. It achieves the goal of sending multicast traffic to the locally connected receivers in one step.

Distributed Packet Replication

High capacity network devices contain multiple packet forwarding engines (PFEs) interconnected with a switch fabric and managed via a Routing Engine. Packet forwarding engines normally reside on dedicated boards (line cards) that can host one or more PFEs [Figure 2]. In turn, line cards are mapped to physical interfaces providing the revenue-generating connectivity to external networks (LAN/WAN connections).

Since every PFE may be receiving traffic from any other PFE (many-to-one pattern), this creates a potential bottleneck. Contemporary distributed designs solve this problem of egress PFE congestion (in the "from-fab" direction) by implementing a backpressure mechanism from the congested receiver PFE toward the source PFEs. When source PFEs are throttled via backpressure, the jam point shifts accordingly into their "to-fab" queues,

where an elaborate queuing mechanism performs intelligent, priority- and protocol-aware congestion control. This mechanism is variously referred to as virtual output queuing (VOQ) or simply "fabric queuing." A typical fabric queuing engine provides a separate set of priority queues for every destination PFE, and manages fill level and selective packet discard according to a QoS policy; in the event of congestion, priority traffic is properly buffered and preserved.

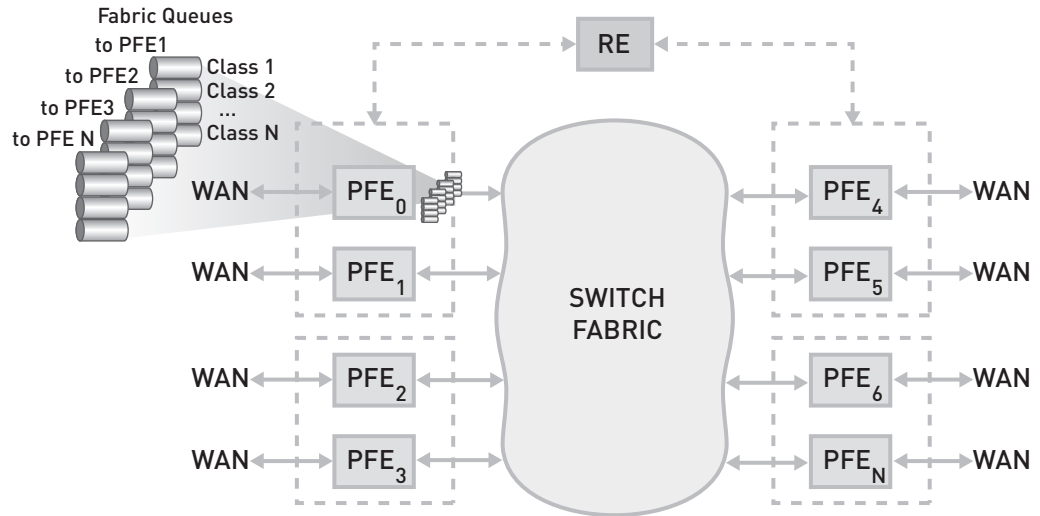


Figure 2: Generic distributed system

It is important to note, that in the aforementioned distributed system architecture, relying solely on the ingress card's local packet replication is no longer feasible. This happens because the overall system capacity can be well in excess of terabit speeds, which is way beyond the replication capabilities of any single PFE.

Instead, in the large packet platforms, the distributed packet replication process takes care of scaling the multicast performance.

In this process, every PFE that serves receiver ports performs local packet replication just for them [Figure 3]. Therefore, it is sufficient for only one multicast stream instance to reach every PFE that serves one or more receiver ports. All line cards can run their application engines in parallel, and multicast capacity can scale proportionally to the number of line cards installed in the system. This forms the basis for contemporary multicast architectures.

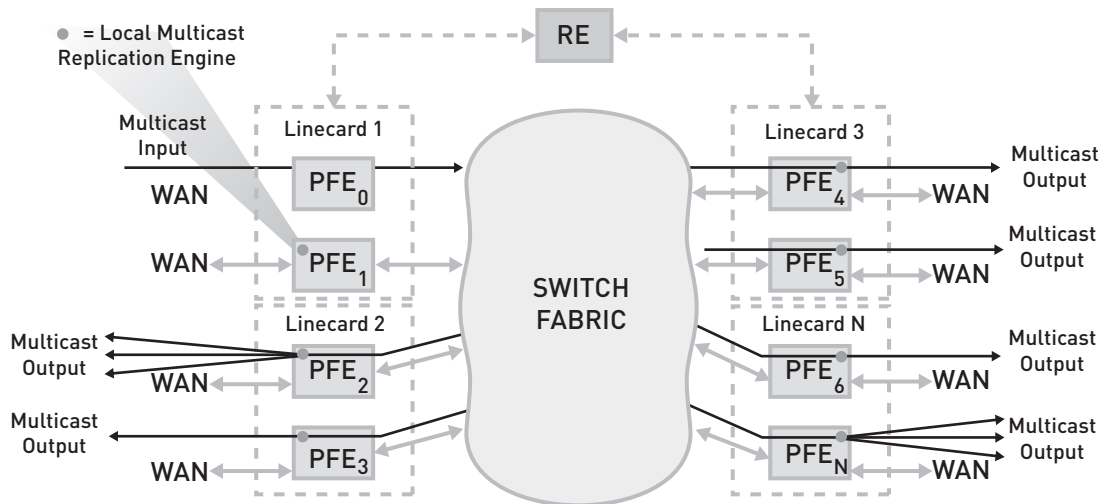


Figure 3: Distributed packet replication process

Distributed Packet Replication Modes

When speaking about distributed packet replication, we have decisively omitted the mechanism that carries packet copies over fabric. As a reminder, in Figure 3 above, every destination PFE has to receive its own copy of each multicast stream prior to forwarding or replicating to connected receivers.

How and when is this done?

The actual implementation of this mechanism is a critical part of system design, and determines many pivotal scaling and QoS limits. This area has led to much research and experimentation over the course of the last ten years.

In the next sections, we will describe some of the most popular choices.

Distributed Ingress Replication (DIR)

The Distributed Ingress Replication algorithm, at a high level, can be summarized as follows:

1. Packet arrives at an ingress interface (ingress PFE)
2. It is replicated to the locally connected receivers (if needed).
3. If there are remote receivers (connected to remote PFEs), one copy is made for each remote PFE.
4. Copies made at Step 3 are sent into a fabric queuing engine according to their QoS settings.
5. Remote PFEs receive their copies and replicate for local receivers (if needed).

The DIR process is straightforward and relatively easy to implement. It uses a PFE replication engine as the only multicast-aware entity in the entire packet path, leaving the fabric queuing engine and the crossbar completely multicast-agnostic. Therefore, it automatically achieves QoS levels for multicast in the fabric that are identical to those established for unicast packets.

However, DIR is no longer actively used in packet platform designs. The reason is simple—it does not scale very well. As the number of destination PFEs grows, DIR requires the ingress PFE to send more and more packets into fabric, putting significant stress on the ingress PFE's replication engine. More importantly, the DIR algorithm requires excess bandwidth to be provisioned from every PFE into the fabric, and this accounts for the extra packet copies. For instance, a 100 Gbps PFE replicating a 4 Gbps multicast stream into 24 destination PFEs, in the worst case scenario, will need 192 Gbps of effective bandwidth (96 Gbps unicast plus 96 Gbps multicast) into the fabric to do the work.

Since fabric bandwidth at the forefront of platform development is a very expensive resource, ingress packet replication is rarely seen in production except in old and legacy platforms.

Distributed Fabric Replication (DFR)

By the middle 1990s, the drawbacks of DIR were well known and the algorithm became a limiting factor to building faster packet platforms. At this time, a new idea was conceived—to complement the local packet replication engine with a second stage replicator in the switch fabric. This required significant changes to the fabric interface on many levels (as multicast traffic could no longer be treated equally to unicast), but it brought the promise of better scaling.

Indeed, a fabric-level packet replicator can take care of all packet copies required for remote PFEs, and therefore can dramatically increase system multicast performance; it does this by making the fabric (or at least some fabric stages) more complex [Figure 4].

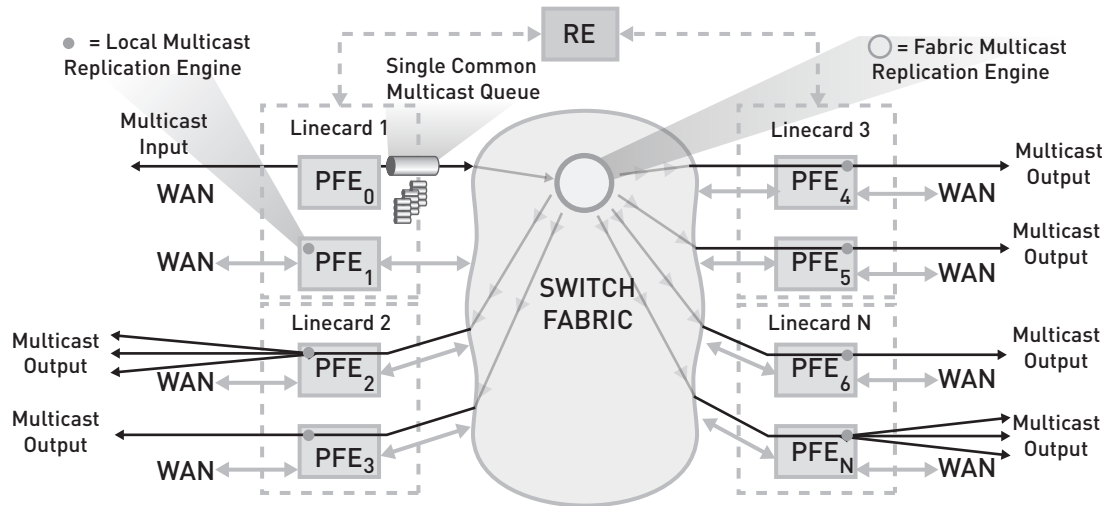


Figure 4: Distributed fabric replication

Distributed Fabric Replication can be summarized in the following steps:

1. Packet arrives at an ingress interface.
2. It is replicated to the locally connected receivers (if needed).
3. If there are remote receivers (connected to other PFEs), one packet copy is dispatched to the fabric.
4. A single packet copy from Step 3 is placed into a special “multicast” queue in the fabric queuing engine.
5. Once the multicast packet enters the fabric, extra copies are being made for all destination PFEs within fabric itself.
6. Remote PFEs receive their copies and replicate for local receivers (if needed).

DFR scales linearly as the number of destination PFEs grows; if a 100 Gbps ingress PFE receives a 1Gbps multicast stream to 24 remote PFEs, in the worst case scenario it will send exactly 100 Gbps worth of traffic into fabric (99 Gbps unicast plus 1 Gbps multicast), thus making efficient use of fabric bandwidth. This efficiency has made DFR a popular choice for packet platforms designed from the early 2000s on.

However, practical use of DFR revealed that it has a very serious design flaw.

With the advent of HDTV and bandwidth-hungry multicast implementations, it quickly became obvious that proper QoS is the key for commercial success in modern markets. The customer experience has to be smooth and predictable—no matter where the bottleneck points might be. The classical DFR algorithm fell short of filling these requirements.

With DFR, QoS can easily become a problem by virtue of the fact that DFR-driven multicast bypasses normal queuing/backpressure mechanisms built into the fabric. For unicast traffic, whenever a destination PFE gets congested, it signals backpressure to source PFEs, which start queuing traffic in their respective fabric queues bound for a congested destination. Sophisticated classification and congestion avoidance mechanisms are applied to provide fairness and to preserve critical traffic, such as voice and video.

DFR effectively breaks this scheme—every source PFE has only one fabric-bound queue for all multicast input, and it cannot be “back-pressed” at destination granularity. Consequently, in the event of congestion, the egress PFE continues receiving multicast at full throttle, thus creating an additional congestion point inside the fabric (typically at the final switching stages). Since crossbar switching elements lack deep buffers and the sophisticated capabilities of fabric queuing engines, they can only make fairly crude drop decisions based on packet priority levels—creating a significant chance for damaging priority traffic in both unicast and multicast domains.

Such an undesired effect can easily be observed in virtually all DFR implementations, even under modest congestion conditions.

Evolution of DFR

The issues experienced by early DFR adopters in triple-play and IPTV implementations has spurred significant R&D in the area of packet replication in distributed routers and switches.

Several options to improve DFR, such as over engineering the fabric data path and merging fabric switches with extensive buffers and drop engines, were considered and dismissed as impractical; it became apparent that a simple “any-to-any” approach does not work and some restriction must be added into a multicast flow model.

A detailed analysis of multicast applications reveals that a typical multicast topology is, in fact, restricted in the amount of ingress bandwidth. For example, a core router in the middle of a triple-play backbone may have only one or two MPEG program envelope streams entering from head stations, but it can effectively replicate them into hundreds of streams exiting towards regional and metropolitan distribution networks, sometimes up to the line rate of the entire system. Therefore, ingress and egress multicast patterns are typically asymmetrical.

This prompted equipment suppliers to modify the DFR scheme to be restricted on bandwidth received from ingress PFEs. The resulting “rDFR” algorithm allows for ingress multicast bandwidth in an amount that is limited according to speed-up in the entire fabric packet path and the egress PFE data path. This ensures that multicast traffic can always access the pre-allocated resources.

This consequential bandwidth restriction cannot be easily lifted as it would result in significant over engineering; however, rDFR does achieve zero-loss operation in IPTV environments and is in use today. As a practical trade-off between the cost of implementation and performance, rDFR typically allows for 2.5 - 5 Gbps of multicast input into fabric, which is well within the sensible limits of SD programming.

HDTV and Media Convergence

The start of the new millenium brought new challenges to service providers. Increased competition, rising energy costs and slowing economy growth has challenged the models of overlaid or physically separate media and data networks, making a clear case for convergence that should last well into this century.

As new interactive services are continued to be created, it becomes clear that prospective media streams cannot fit into SD payload and will require higher bandwidth; more and more global and local programs are being converted into HDTV format.

Recognizing this trend very early, the Juniper Networks® Engineering organization has made a goal to support multicast at HDTV-type speeds on packet platforms at commercial-grade quality. This goal quickly challenged the limits of DFR and rDFR implementation, and a new replication scheme has been designed in-house.

This is how a new replication scheme—Distributed Tree Replication (DTR)—was born.

Distributed Tree Replication (DTR)

As the result of significant R&D efforts, Juniper has invented and patented a distributed N-way tree replication process (DTR), combining the best of DIR and DFR algorithms. With Juniper DTR, large fan-out configurations may co-exist with strict QoS policies and uniform treatment is assured for all packet types. First proven on Juniper Networks T Series Core Routers, DTR was quickly found to be successful and was eventually extended into Juniper Networks M Series Multiservice Edge Routers and MX Series Ethernet Services Routers.

The current generation of DTR-enabled Juniper hardware runs in 100 Gbps/slot realm at industry’s lowest power consumption level².

Distributed Tree Replication can be seen as a hybrid between DIR and DFR schemes. Like DIR, DTR treats multicast and unicast packets equally and does not have a special data path for multicast. Unlike DFR, DTR provides distributed packet replication without a second stage in the fabric; it does this by spreading the fabric-level packet replication between multiple destination PFEs linked into a chain.

The entire N-way tree replication (for N=2) can be summarized in four steps.

¹US Patent 7,263,099 “Multicast packet replication” Juniper Networks, Inc. Woo; Hsien-Chung, Ferguson; Dennis, Hui; Lawrence

²As of 07/2008, based on vendor’s data for commercially available 1-Tbps routing platforms

Step One. Consider the following diagram showing the parts of the router that might be used in replication. In this diagram, it shows multicast input arriving on the input interface from the WAN.

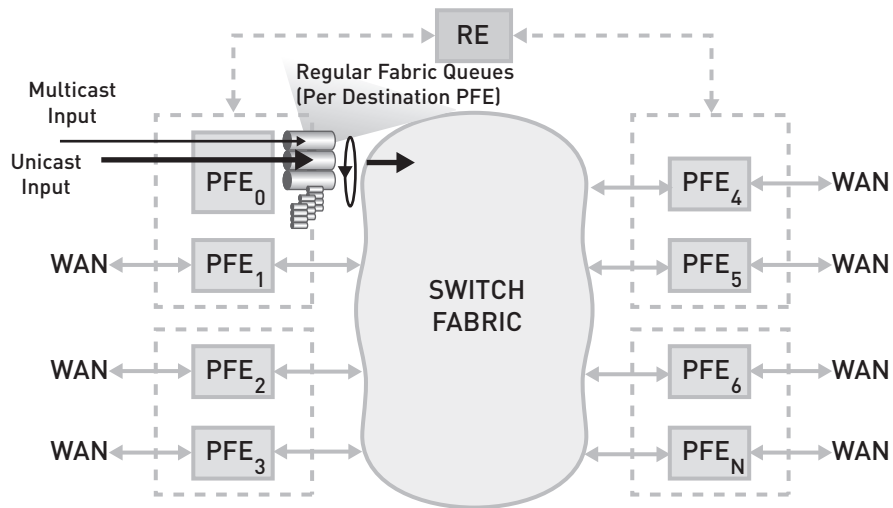


Figure 5: DTR Step 1 – Packet arrives at ingress

Note the use of queues between the packet forwarding engines and the switch fabric. These queues are used for prioritization of traffic, and treat multicast and unicast input equally. Packets of all types are queued per class and destination PFE and intelligently dropped in the event of congestion according to their priority. The ability to handle prioritization in this way is paramount to triple-play networks.

Step Two. In the next step, the packet is switched through the ingress PFE and sent to two egress PFEs (local replicators). Ingress replicator and PFE data path are over-engineered for twice the maximum amount of bandwidth that could be required for HDTV applications.

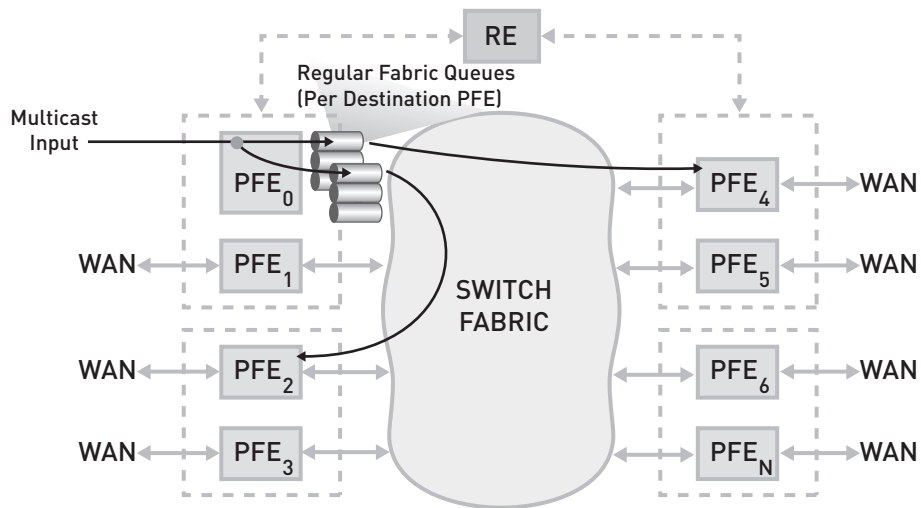


Figure 6: DTR Step 2 – Packet is replicated to two egress PFEs

Step Three. Egress PFEs will replicate and send packets to two other egress PFEs, repeating until each egress PFE with an outgoing interface has received packets.

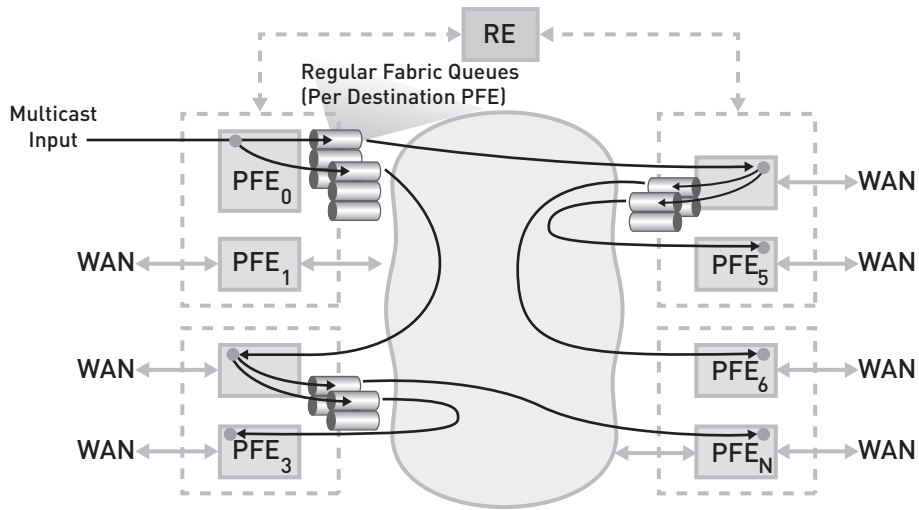


Figure 7: DTR Step 3 – Process in step 2 is repeated

Step Four. Egress PFEs with multiple outgoing interfaces will replicate locally, and send copies to outgoing WAN interfaces (Figure 3). This is the final step in the process. At this point, the participating N receiver PFEs are organized into a tree with a maximum depth proportional to $\log_2 N$.

This methodology can allow large fanout—many multicast receivers—with little or no impact on performance. No single PFE replicates more than twice.

The DTR algorithm is specifically designed for triple-play applications; it significantly improves QoS levels compared to DFR and trumps rDFR in the amount of ingress multicast bandwidth that a single PFE may serve, thus paving the way to a true HDTV experience.

The tradeoff in DTR is the fact that a replication tree depends on the mid-level PFEs to operate properly; although hardware failures are statistically rare, they have the potential of disrupting traffic flow on dependent PFEs, thus affecting some portion of the active subscribers for a timeframe of approximately 800 ms—the time it takes to rebuild a tree around a failed PFE. Normal Internet Group Management Protocol (IGMP) or Protocol Independent Multicast (PIM) join/leave events do not affect tree build-out, because inactive PFEs are dynamically excluded from the tree with a “make-before-break” truncation algorithm.

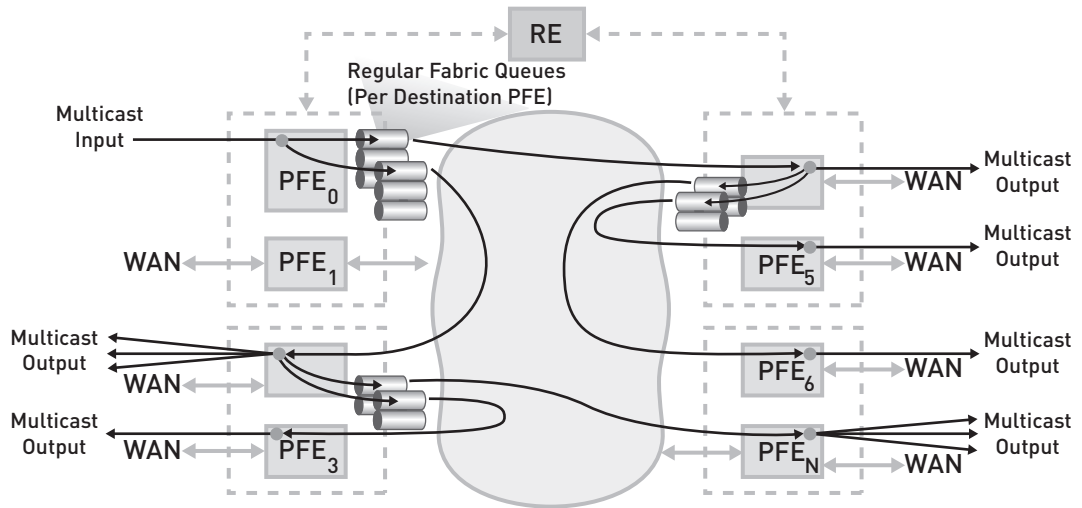


Figure 8: DTR Step 4 - Full multicast packet flow through the device

Conclusion: Pros and Cons of Competing Approaches

As the router/switch market remains very competitive, and DTR, DFR and rDFR platforms are widely available to enterprises and service providers today, choosing the right platform is very important for a successful triple-play or HDTV service rollout, as mistakes can be very hard to correct later.

The following two tables summarize the pros and cons of each approach.

Table 1: DTR Versus DFR

FEATURE	DTR	DFR
QoS (general)	Always guaranteed as multicast goes through fabric prioritization queues; backpressure notifications and random early detection (RED) ensure good QoS for all traffic.	No guarantee for multicast as it bypasses the fabric queues.
Latency	Multicast does not change unicast latency.	Latency may change in the presence of multicast traffic.
Side effects	None.	Unexpected wrongful drops in unicast can be seen if multicast is present.
HDTV-ready	Yes (200-400 channels per system).	No, due to practical QoS limits (unless all egress PFEs are significantly undersubscribed).
Total multicast traffic supported (ingress)	4-10 Gbps per system, no restrictions per PFE.	Up to line rate (Tbps+ speeds).
Total multicast traffic supported (egress)	Up to line rate (Tbps+ speeds).	Up to line rate (Tbps+ speeds).
Failure handling	If a PFE with multicast subscribers fails, it may temporarily affect other PFEs if it was in the middle of the replication tree.	No special failure modes.
Practical tradeoffs	Lower amount of ingress multicast traffic supported; PFE failures may affect multicast delivery.	Adverse effects on voice and video when running close to PFE capacity

As can be seen from Table 1, DFR has better multicast scaling capabilities assuming strict QoS is not required—an assumption that can be valid in certain environments outside commercial TV deployments. Another option for DFR platforms is to underprovision PFEs, but this is rarely used due to extremely ineffective resource utilization (25 percent or more of practical capacity loss).

Table 2: DTR Versus rDFR

FEATURE	DTR	rDFR
QoS (general)	Always guaranteed as multicast goes through fabric prioritization queues; backpressure notifications and RED ensures good QoS for all traffic.	Always guaranteed as data path is over engineered for limited multicast input.
Latency	Multicast does not change unicast latency.	Multicast does not change unicast latency.
Side effects	None.	None.
HDTV-ready	Yes (200-400 channels per system).	Mostly suitable for regular IPTV.
Total multicast traffic supported (ingress)	4-10 Gbps per system, no restrictions per PFE	2-4 Gbps per PFE, no restrictions per system
Total multicast traffic supported (egress)	Up to line rate (Tbps+ speeds)	Up to line rate (Tbps+ speeds).
Failure handling	If a PFE with multicast subscribers fails, it may temporarily affect other PFEs if it was in the middle of the replication tree.	No special failure modes.
Practical tradeoffs	Lower amount of total (system) ingress multicast traffic supported; PFE failures may affect multicast delivery.	Relatively low bandwidth allowance per individual PFE significantly restricts multicast topology (i.e. placement of the TV head stations in the network)

In real-world scenarios of HDTV rollouts, there are about 4 Gbps of video traffic required to handle approximately 200 channels. This teeters on the edge of rDFR capabilities, thus making DTR a smarter choice for forward looking networks. Although rDFR can compensate for this limit by engineering of multicast stream inputs (place ingress streams on different PFEs), in practice this is complex and rarely done; such solution would put serious limits on production TV locations and traffic affinity.

Although every solution has its own advantages and disadvantages, DTR is currently the only practical solution to HDTV stream distribution requirements, which highlights the advantages of in-house software and hardware development programs. Proven and widely deployed, Juniper Networks packet platforms deliver unicast and multicast payloads around the world.

Appendix

The following additional materials are available for additional reading.

Background

Interdomain Multicast Routing: Practical Juniper Networks and Cisco Systems Solutions

Brian M. Edwards, Leonard A. Giuliano, Brian R. Wright, Addison-Wesley Professional (2002), ISBN 0201746123

Software References

[1] JUNOS 9.1 Multicast Protocols Configuration Guide:

<http://www.juniper.net/techpubs/software/junos/junos91/swconfig-multicast/swconfig-multicast.pdf>

About Juniper Networks

Juniper Networks, Inc. is the leader in high-performance networking. Juniper offers a high-performance network infrastructure that creates a responsive and trusted environment for accelerating the deployment of services and applications over a single network. This fuels high-performance businesses. Additional information can be found at www.juniper.net.

Corporate And Sales Headquarters

Juniper Networks, Inc.
1194 North Mathilda Avenue
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER
(888.586.4737)
or 408.745.2000
Fax: 408.745.2100

APAC Headquarters

Juniper Networks (Hong Kong)
26/F, Cityplaza One
1111 King's Road
Taikoo Shing, Hong Kong
Phone: 852.2332.3636
Fax: 852.2574.7803

EMEA Headquarters

Juniper Networks Ireland
Airside Business Park
Swords, County Dublin,
Ireland
Phone: 35.31.8903.600
Fax: 35.31.8903.601

Copyright 2009 Juniper Networks, Inc. All rights reserved. Juniper Networks, the Juniper Networks logo, JUNOS, NetScreen, and ScreenOS are registered trademarks of Juniper Networks, Inc. in the United States and other countries. "Engineered for the network ahead" and JUNOSe are trademarks of Juniper Networks, Inc. All other trademarks, service marks, registered marks, or registered service marks are the property of their respective owners. Juniper Networks assumes no responsibility for any inaccuracies in this document. Juniper Networks reserves the right to change, modify, transfer, or otherwise revise this publication without notice.

To purchase Juniper Networks solutions, please contact your Juniper Networks representative at 1-866-298-6428 or authorized reseller.

