Internet Engineering Task Force PIM WG
INTERNET-DRAFT
draft-ietf-pim-bidir-05.ps
Mark Handley/ICIR
Isidor Kouvelas/Cisco
Tony Speakman/Cisco
Lorenzo Vicisano/Cisco
26 June 2002
Expires: December 2002

Bi-directional Protocol Independent Multicast (BIDIR-PIM)

Status of this Document

This document is a product of the IETF PIM WG. Comments should be addressed to the authors, or the WG's mailing list at pim@catarina.usc.edu.

Abstract

This document discusses Bi-directional PIM, a variant of PIM Sparse-Mode [9] that builds bi-directional shared trees connecting multicast sources and receivers. Bi-directional trees are built using a fail-safe Designated Forwarder (DF) election mechanism operating on each link of a multicast topology. With the assistance of the DF, multicast data is natively forwarded from sources to the Rendezvous-Point and hence along the shared tree to receivers without requiring source-specific state. The DF election takes place at RP discovery time and provides a default route to the RP thus eliminating the requirement for data-driven protocol events.

Note on BIDIR-PIM status

The differences between this version of the BIDIR-PIM specification and draft-ietf-pim-bidir-new-00.txt are mostly in the format of the information presented. As BIDIR-PIM has many similarities in operation to Sparse-Mode PIM, the earlier version of this spec relied heavily on the now obsolete PIM-SM [11] specification. This revision removes this dependency and instead references the new Sparse-Mode documentation [9] where necessary. In addition the method in which the protocol specification is presented has been updated to follow the format of [9].

Table of Contents

1. Introduction

This document specifies Bi-directional PIM, a variant of PIM Sparse-Mode (PIM-SM) [9] that builds bi-directional shared trees connecting multicast sources and receivers.

PIM-SM constructs uni-directional shared trees that are used to forward data from senders to receivers of a multicast group. PIM-SM also allows the construction of source specific trees, but this capability is not related to the protocol described in this document.

The shared tree for each multicast group is rooted at a multicast router called the Rendezvous Point (RP). Different multicast group ranges can use separate RPs within a PIM domain.

In unidirectional PIM-SM, there are two possible methods for distributing data packets on the shared tree. These differ in the way packets are forwarded from a source to the RP:

- Initially when a source starts transmitting, its first hop router encapsulates data packets in special control messages (Registers) which are unicast to the RP. After reaching the RP the packets are decapsulated and distributed on the shared tree.
- A transition from the above distribution mode can be made at a later stage. This is achieved by building source specific state on all routers along the path between the source and the RP. This state is then used to natively forward packets from that source.

Both these mechanisms suffer from problems. Encapsulation results in significant processing, bandwidth and delay overheads. Forwarding using source specific state has additional protocol and memory requirements.

Bi-directional PIM dispenses with both encapsulation and source state by allowing packets to be natively forwarded from a source to the RP using shared tree state. For a complete discussion of the pros and cons of Bi-directional PIM consult appendix C.

2. Terminology

In this document, the key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119 and indicate requirement levels for compliant PIM-SM implementations.

2.1. Definitions

This specification uses a number of terms to refer to the roles of routers participating in BIDIR-PIM. The following terms have special significance for BIDIR-PIM:

MRIB

Multicast Routing Information Base. This is the multicast topology table, which is typically derived from the unicast routing table, or routing protocols such as MBGP that carry multicast-specific topology information. It is used by PIM for establishing the RPF interface (used in the forwarding rules). In PIM-SM the MRIB is also used to make decisions regarding where to forward Join/Prune messages whereas in BIDIR-PIM it is used as a source for routing metrics for the DF election process.

Rendezvous Point (RP):

An RP is a router that has been configured to be used as the root of the distribution tree for a range of multicast groups. Join messages from receivers for a group are sent towards the RP.

Upstream

Towards the root (Rendezvous-Point) of the tree. The direction used by packets traveling from sources to the RP.

Downstream
: Away from the root of the tree. The direction on which packets travel from the RP to receivers.

Designated Forwarder (DF):
: The protocol presented in this document is largely based on the concept of a Designated Forwarder (DF). A single DF exists for each RP on every link within a BIDIR-PIM domain (this includes both multi-access and point-to-point links). The DF is the router on the link with the best unicast route to the RP. A DF for a given RP is in charge of forwarding downstream traffic onto the link, and forwarding upstream traffic from the link towards the RP. It does this for all the bi-directional groups served by the RP. The DF on a link is also responsible for interpreting IGMP information from local receivers and processing Join messages from other routers on the link.

RPF Interface
: RPF stands for "Reverse Path Forwarding". The RPF Interface of a router with respect to an address is the interface that the MRIB indicates should be used to forward packets to that address. In the case of a BIDIR-PIM multicast group, the RPF interface is the interface that would be used to send packets to the RP for the group.

RPF Neighbor
: The RPF Neighbor of a router with respect to an address is the neighbor that the MRIB indicates should be used to forward packets to that address. Note that in BIDIR-PIM, the RPF neighbor for a group is not necessarily the router on the RPF interface that Join messages for that group would be directed to (Join messages are directed to the DF on the RPF interface for the group).

TIB
: Tree Information Base. This is the collection of state at a PIM router that has been created by receiving PIM Join/Prune messages, PIM DF election messages and IGMP information from local hosts. It essentially stores the state of all multicast distribution trees at that router.

MFIB
: Multicast Forwarding Information Base. The TIB holds all the state that is necessary to forward multicast packets at a router. However, although this specification defines forwarding in terms of the TIB, to actually forward packets using the TIB is very inefficient. Instead a real router implementation will normally build an efficient MFIB from the TIB state to perform forwarding. How this is done is implementation-specific, and is not discussed in this document.

2.2. Pseudocode Notation

We use set notation in several places in this specification.

`A (+) B`
: is the union of two sets A and B.

`A (-) B`
: is the elements of set A that are not in set B.

`NULL`
: is the empty set or list.

In addition we use C-like syntax:

= denotes assignment of a variable.

== denotes a comparison for equality.

!= denotes a comparison for inequality.

Braces { and } are used for grouping.

3. Protocol Specification

The specification of BIDIR-PIM is broken into several parts:

- Section 3.1 details the protocol state stored.
- Section 3.3 specifies the data packet forwarding rules.
- Section 3.4 specifies the BIDIR-PIM Join/Prune generation and processing rules.
- Designated Forwarder (DF) election is specified in Section 3.5.
- PIM packet formats are specified in Section 3.7.
- A summary of BIDIR-PIM timers and their default values is given in Section 3.6.

3.1. BIDIR-PIM Protocol State

This section specifies all the protocol state that a BIDIR-PIM implementation should maintain in order to function correctly. We term this state the *Tree Information Base* or TIB, as it holds the state of all the multicast distribution trees at this router. In this specification we define PIM mechanisms in terms of the TIB. However, only a very simple implementation would actually implement packet forwarding operations in terms of this state. Most implementations will use this state to build a multicast forwarding table, which would then be updated when the relevant state in the TIB changes.

Although we specify precisely the state to be kept, this does not mean that an implementation of PIM-SM needs to hold the state in this form. This is actually an abstract state definition, which is needed in order to specify the router's behavior. A BIDIR-PIM implementation is free to hold whatever internal state it requires, and will still be conformant with this specification so long as it results in the same externally visible protocol behavior as an abstract router that holds the following state.

We divide TIB state into two sections:

RP state
: State that maintains the DF election information for each RP.

Group state
: State that maintains a group-specific tree for groups that map to a given RP.

The state that should be kept is described below. Of course, implementations will only maintain state when it is relevant to forwarding operations - for example, the "NoInfo" state might be assumed from the lack of other state information, rather than being held explicitly.

3.1.1. General Purpose State

A router holds the following state that is not specific to a RP or group:

Neighbor State:

For each neighbor:

- Information from neighbor's Hello
- Neighbor's Gen ID.
- Neighbor liveness timer (NLT)

3.1.2. RP State

A router maintains a multicast-group to RP mapping which is built through static configuration or by using an automatic RP discovery mechanism like BSR or AUTO-RP (see section 4). For each BIDIR-PIM RP a router holds the following state:

- RP address

Designated Forwarder (DF) State:

For each router interface:

Acting DF information:

- DF IP Address
- DF metric

Election information:

- Election State
- DF Election-Timer (DFT)
- Offer-Count (OC)

Current best offer:

- IP address of best offering router
- Best offering router metric

Designated Forwarder state is described in section 3.5.

3.1.3. Group State

For every group G a router keeps the following state:

Group state:

For each interface:

Local Membership:

- State: One of {"NoInfo", "Include"}

PIM Join/Prune State:

- State: One of {"NoInfo" (NI), "Join" (J), "PrunePending" (PP)}
- Prune Pending Timer (PPT)
- Join/Prune Expiry Timer (ET)

Not interface specific:

- Upstream Join/Prune Timer (JT)
- Last RP Used

Local membership is the result of the local membership mechanism (such as IGMP) running on that interface. This information is used by the `pim_include(*,G)` macro described in section 3.1.4.

PIM Join/Prune state is the result of receiving PIM (*,G) Join/Prune messages on this interface, and is specified in section 3.4.1. The state is used by the macros that calculate the outgoing interface list in section 3.1.4, and in the `JoinDesired(G)` macro (defined in section 3.4.2) that is used in deciding whether a Join(*,G) should be sent upstream.

The upstream Join/Prune timer is used to send out periodic Join(*,G) messages, and to override Prune(*,G) messages from peers on an upstream LAN interface.

The last RP used must be stored because if the RP Set changes [9] then state must be torn down and rebuilt for groups whose RP changes.

3.1.4. State Summarization Macros

Using this state, we define the following "macro" definitions which we will use in the descriptions of the state machines and pseudocode in the following sections.

```
olist(G) =
    RPF_interface(RP(G)) (+) joins(G) (+) pim_include(G)
```

`RPF_interface(RP)` is the interface the MRIB indicates would be used to route packets to RP. The olist(G) is the list of interfaces on which packets to group G must be forwarded.

The macro pim_include(G) indicates the interfaces to which traffic might be forwarded because of hosts that are local members on that interface.

```
pim_include(G) =
    { all interfaces I such that:
      I_am_DF(RP(G),I) AND  local_receiver_include(G,I) }
```

The clause "I_am_DF(RP,I)" is TRUE if the router is in the `Win` or `Backoff` states in the DF election state machine for interface I (described in section 3.5). Otherwise it is FALSE.

The clause "local_receiver_include(G,I)" is true if the IGMP module or other local membership mechanism has determined that there are local members on interface I that desire to receive traffic sent to group G.

The set "joins(G)" is the set of all interfaces on which the router has received (*,G) Joins:

```
joins(G) =
    { all interfaces I such that
      I_am_DF(RP(G),I) AND
      DownstreamJPState(G,I) is either Joined or PrunePending }
```

DownstreamJPState(G,I) is the state of the finite state machine in section 3.4.1.

`RPF_DF(RP)` is the neighbor that Join messages must be sent to in order to reach the RP. This is the Designated-Forwarder on the `RPF_interface(RP)`.

3.2. PIM Neighbor Discovery

PIM routers exchange PIM-Hello messages with their neighboring PIM routers. These messages are used to update the `Neighbor State` described in section 3.1. The procedures for generating and processing received Hello messages as well as maintaining `Neighbor State` are specified in the PIM-SM [9] documentation.

Bidir PIM introduces the Bidir_Capable PIM-Hello option that MUST be included in all Hello messages from a Bidir-PIM capable router. The Bidir_Capable option advertises the router's ability to participate in the Bidir-PIM protocol. The format of the Bidir_Capable option is described in section 3.7.

3.3. Data Packet Forwarding Rules

For groups mapping to a given RP, the following responsibilities are uniquely assigned to the DF for that RP on each link:

- The DF is the only router that forwards packets traveling downstream onto the link.
- The DF is the only router that picks-up upstream traveling packets off the link to forward towards the RP.

Non-DF routers on a link, that use that link as their RPF interface to reach the RP, may perform the following forwarding actions for bidirectional groups:

- Forward packets from the link towards downstream receivers.
- Forward packets from downstream sources onto the link (provided they are the DF for the downstream link from which the packet was picked-up).

The BIDIR-PIM packet forwarding rules are defined below in pseudocode.

`iif` is the incoming interface of the packet.
`G` is the destination address of the packet (group address).
`RP` is the address of the Rendezvous Point for this group.

First we check to see whether the packet should be accepted based on TIB state and the interface that the packet arrived on. A packet is accepted if it arrives on the RPF_interface to reach the RP (downstream traveling packet) or if the router is the DF on the interface the packet arrives (upstream traveling packet).

If the packet should be forwarded we build an outgoing interface list for the packet.

Finally we remove the incoming interface from the outgoing interface list we've created, and if the resulting outgoing interface list is not empty, we forward the packet out of those interfaces.

On receipt on a data to G on interface iif:

```
if( iif == RPF_interface(RP) || I_am_DF(RP,I) ) {
    oiflist = olist(G) (-) iif
    forward packet on all interfaces in oiflist
}
```

Note: A major advantage of using a Designated Forwarder in BIDIR-PIM compared to PIM-SM is that special treatment is no longer required for sources that are directly connected to a router. Data from such sources does not need to be differentiated from other multicast traffic and will automatically be picked up by the DF. This removes the need for performing a directly-connected-source check for data to groups that do not have existing state.

3.3.1. Source-Only Branches

Source-only branches of the distribution tree for a group G are branches which do not lead to any receivers, but which are used to forward packets traveling upstream from sources towards the RP. Routers along source-only branches only have the RPF_interface to the RP in their olist for G and hence do not need to maintain any group specific state. Upstream forwarding can be performed using RP state. An implementation may decide to maintain group state for source-only branches for accounting or performance reasons.

3.4. PIM Join/Prune Messages

A BIDIR-PIM Join/Prune message consists of a list of Joined and Pruned Groups. When processing a received Join/Prune message, each Joined or Pruned Group is effectively considered individually by applying the following state machines. When considering a Join/Prune message whose PIM Destination field addresses this router, (*,G) Joins and Prunes can affect the downstream state machine. When considering a Join/Prune message whose PIM Destination field addresses another router, most Join or Prune entries could affect the upstream state machine.

3.4.1. Receiving (*,G) Join/Prune Messages

When a router receives a Join(*,G) or Prune(*,G) it must first check to see whether the RP in the message matches RP(G) (the router's idea of who the RP is). If the RP in the message does not match RP(G) the Join or Prune MUST be silently dropped. In addition a router MUST NOT process Join(*,G) messages targeted to itself if it is not the DF for RP(G) on the interface on which the message was received.

The per-interface state-machine for receiving (*,G) Join/Prune Messages is given below. There are three states:

NoInfo (NI)
The interface has no (*,G) Join state and no timers running.

Join (J)
The interface has (*,G) Join state which will cause us to forward packets destined for G from this interface.

PrunePending (PP)
The router has received a Prune(*,G) on this interface from a downstream neighbor and is waiting to see whether the prune will be overridden by another downstream router. For forwarding purposes, the PrunePending state functions exactly like the Join state.

In addition the state-machine uses two timers:

ExpiryTimer (ET)
This timer is restarted when a valid Join(*,G) is received. Expiry of the ExpiryTimer causes the interface state to revert to NoInfo for this group.

PrunePendingTimer (PPT)
This timer is set when a valid Prune(*,G) is received. Expiry of the PrunePendingTimer causes the interface state to revert to NoInfo for this group.

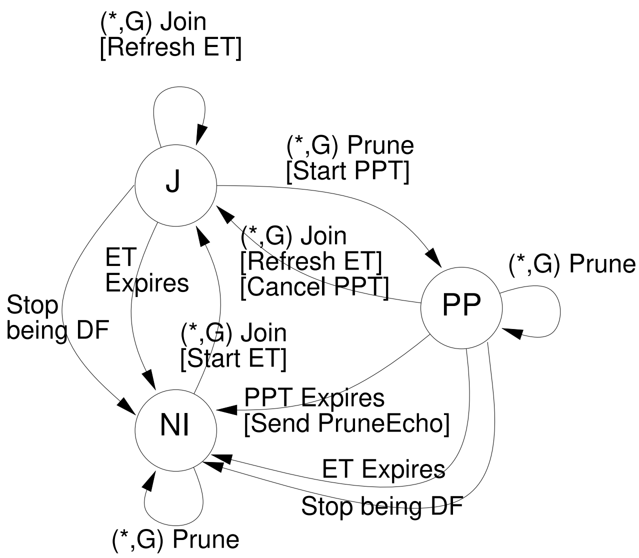

Figure 1: Downstream group per-interface state-machine

In tabular form, the group per-interface state-machine is:

Prev State	Event				
	Receive Join(*,G)	**Receive Prune(*,G)**	**Prune Pending Timer Expires**	**Expiry Timer Expires**	**Stop Being DF on I**
NoInfo (NI)	-> J state start Expiry Timer	-> NI state	-	-	-
Join (J)	-> J state restart Expiry Timer	-> PP state start Prune Pending Timer	-	-> NI state	-> NI state
Prune Pending (PP)	-> J state restart Expiry Timer; stop Prune Pending Timer	-> PP state	-> NI state Send Prune-Echo(*,G)	-> NI state	-> NI state

The transition events "Receive Join(*,G)" and "Receive Prune(*,G)" imply receiving a Join or Prune targeted to this router's address on the received interface. If the destination address is not correct, these state transitions in this state machine must not occur, although seeing such a packet may cause state transitions in other state machines.

On unnumbered interfaces on point-to-point links, the router's address should be the same as the source address it chose for the hello packet it sent over that interface. However on point-to-point links we also recommend that PIM messages with a 0.0.0.0 destination address are also accepted.

The transition event "Stop being DF" implies a DF re-election taking place on this router interface and the router changing status from being the active DF to being a non-DF router (the value of the `I_am_DF` macro changing to FALSE).

When ExpiryTimer is started or restarted, it is set to the HoldTime from the triggering Join/Prune message.

When PrunePendingTimer is started, it is set to the J/P_Override_Interval if the router has more than one neighbor on that interface; otherwise it is set to zero causing it to expire immediately.

The action "Send PruneEcho(*,G)" is triggered when the router stops forwarding on an interface as a result of a prune. A PruneEcho(*,G) is simply a Prune(*,G) message sent by the upstream router to itself on a LAN. Its purpose is to add additional reliability so that if a Prune that should have been overridden by another router is lost locally on the LAN, then the PruneEcho may be received and cause the override to happen. A PruneEcho(*,G) need not be sent on a point-to-point interface.

3.4.2. Sending Join/Prune Messages

The downstream per-interface state-machines described above hold join state from downstream PIM routers. This state then determines whether a router needs to propagate a Join(*,G) upstream towards the RP. Such Join(*,G) messages are sent on the RPF_interface towards the RP and are targeted at the DF on that interface.

If a router wishes to propagate a Join(*,G) upstream, it must also watch for messages on its upstream interface from other routers on that subnet, and these may modify its behavior. If it sees a Join(*,G) to the correct upstream neighbor, it should suppress its own Join(*,G). If it sees a Prune(*,G) to the correct upstream neighbor, it should be prepared to override that prune by sending a Join(*,G) almost immediately. Finally, if it sees the Generation ID (see PIM-SM specification [9]) of the correct upstream neighbor change, it knows that the upstream neighbor has lost state, and it should be prepared to refresh the state by sending a Join(*,G) almost immediately.

In addition changes in the next hop towards the RP trigger a prune off from the old next hop, and join towards the new next hop. Such a change can be cause by the following two reasons:

- The MRIB indicates that the RPF_interface towards the RP has changed.
- There is a DF re-election on the RPF_interface and a new router emerges as the DF.

The upstream (*,G) state-machine only contains two states:

Not Joined
The downstream state-machines indicate that the router does not need to join the RP tree for this group.

Joined
The downstream state-machines indicate that the router would like to join the RP tree for this group.

In addition, one timer JT(G) is kept which is used to trigger the sending of a Join(*,G) to the upstream next hop towards the RP (the DF on the RPF_interface for RP(G)).

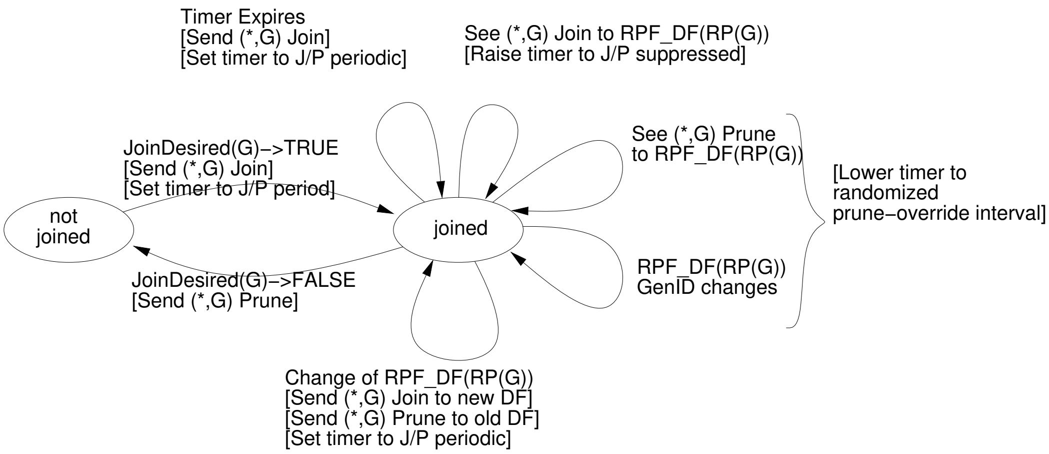

Figure 2: Upstream group state-machine

In tabular form, the state machine is:

Prev State	**Event**	
	JoinDesired(G) ->True	**JoinDesired(G) ->False**
NotJoined (NJ)	-> J state Send Join(*,G); Set Timer to t_periodic	-
Joined (J)	-	-> NJ state Send Prune(*,G)

In addition, we have the following transitions which occur within the Joined state:

In Joined (J) State			
Timer Expires	**See Join(*,G) to RPF_DF(RP(G))**	**See Prune(*,G) to RPF_DF(RP(G))**	**RPF_DF(RP(G)) changes**
Send Join(*,G); Set Timer to t_periodic	Increase Timer to t_suppressed	Decrease Timer to t_override	Decrease Timer to t_override

In Joined (J) State	
Change of RPF_DF(RP(G))	**RPF_DF(RP(G)) GenID changes**
Send Join(*,G) to new DF; Send Prune(*,G) to old DF; set Timer to t_periodic	Decrease Timer to t_override

This state machine uses the following macro:

```
bool JoinDesired(G) {
   if (olist(G) (-) RPF_interface(RP(G))) != NULL
       return TRUE
   else
       return FALSE
}
```

3.5. Designated Forwarder (DF) Election

This section presents a fail-safe mechanism for electing a per-RP designated router on each link in a BIDIR-PIM domain. We call this router the Designated Forwarder (DF).

3.5.1. DF Requirements

The DF election chooses the best router on a link to assume the responsibility of forwarding traffic between the RP and the link for the range of multicast groups served by the RP. Different multicast groups that share a common RP must use the same bi-directional tree for data forwarding. Hence, the election of an upstream forwarder on each link does not have to be a group specific decision but instead can be RP-specific. As the number of RPs is typically small, the number of elections that have to be performed is significantly reduced by this observation.

To optimise tree creation, it is desirable that the winner of the election process should be the router on the link with the "best" unicast routing metric to the RP (as reported by the MRIB). When comparing metrics from different unicast routing protocols, we use the same comparison rules used by the PIM-SM assert process [9].

The election process needs to take place when information on a new RP initially becomes available, and can be re-used as new bidir groups for the same RP are encountered. There are however some conditions where an update to the election is required:

- There is a change in unicast metric to reach the RP for any of the routers on the link.
- The interface on which the RP is reachable changes to an interface for which the router was previously the DF.
- A new PIM neighbor starts up on a link.
- The elected DF dies.

The election process has to be robust enough to ensure with very high probability that all routers on the link have a consistent view of the DF. This is because with the forwarding rules described in section 3.3 if multiple routers end-up thinking that they should be responsible for forwarding, loops may result. To reduce the possibility of this occurrence to a minimum, the election algorithm has been biased towards discarding DF information and suspending forwarding during periods of ambiguity.

3.5.2. DF Election description

This section does not provide the definitive specification for the DF election process. If any discrepancy exists between section 3.5.3 and this section, the specification in section 3.5.3 is to be assumed correct.

To perform the election of the DF for a particular RP, routers on a link need to exchange their unicast routing metric information (as reported by the MRIB) for reaching the RP.

In the election protocol described below, many message exchanges are repeated `Election_Robustness` times for reliability. In all those cases the message retransmissions are spaced in time by a small random interval.

3.5.2.1. Bootstrap Election

Initially when no DF has been elected, routers finding out about a new RP start participating in the election by sending Offer messages. Offer messages include the router's metric to reach the RP. Offers are periodically retransmitted with a period of `Offer_Interval`.

If a router hears a better offer than its own from a neighbor, it stops participating in the election for a period of `Election_Robustness * Offer_Interval`. If during this period no winner is elected, then the router restarts the election from the beginning. If a router receives an offer with worse metrics than its own, then it restarts the election from the beginning.

The result should be that all routers except the best candidate stop advertising their offers.

A router assumes the role of the DF after having advertised its metrics `Election_Robustness` times without receiving any offer from any other neighbor. At that point it transmits a Winner message which declares to every other router on the link the identity of the winner and the metrics it is using.

Routers hearing a winner message stop participating in the election and record the identity and metrics of the winner. If the local metrics are better than those of the winner then the router records the identity of the winner but reinitiates the election.

3.5.2.2. Loser Metric Changes

Whenever the unicast metric to a RP changes for a non-DF router to a value that is better than that previously advertised by the acting DF, the router with the new metric should take action to eventually assume forwarding responsibility. After the metric change is detected, the non-DF router with the now better metric restarts the DF election process by sending Offer messages with this new metric. If no response is received after `Election_Robustness` retransmissions, the router assumes the role of the DF following the usual Winner announcement procedure.

Upon receipt of an offer that is worse than its current metric, the DF will respond with a Winner message declaring its status and advertising its metric. Upon receiving this message, the originator of the Offer records the identity of the DF and aborts the election.

Upon receipt of an offer that is better than its current metric, the DF records the identity and metrics of the offering router and responds with a Backoff message. This instructs the offering router to hold off for a short period of time while the unicast routing stabilises. The Backoff message includes the offering router's new metric and address. All routers on the link who have pending offers with metrics worse than those in the backoff message (including the original offering router) will hold further offers for a period of time defined in the Backoff message.

If during the `Backoff_Period`, a third router sends a new better offer, the Backoff message is repeated for the new offer and the `Backoff_Period` restarted.

Before the `Backoff_Period` expires, the acting DF nominates the router having made the best offer as the new DF using a Pass message. This message includes the IDs and metrics of both the old and new DFs. The old DF stops performing its tasks as soon as the transmission is made. The new DF assumes the role of the DF as soon as it receives the Pass message. All other routers on the link take note of the new DF and its metric.

3.5.2.3. Winner Metric Changes

If the DF's routing metric to reach the RP changes to a worse value, it sends a set of `Election_Robustness` randomly spaced Winner messages on the link, advertising the new metric. Routers who receive this announcement but have a better metric may respond with an Offer message which results in the same handoff procedure described above. All routers assume the DF has not changed until they see a Pass or Winner message indicating the change.

There is no pressure to make this handoff quickly if the acting DF still has a path to the RP. The old path may now be suboptimal but it will still work while the re-election is in progress.

If the routing metric at the DF changes to a better value, a single Winner message is sent advertising the new metric.

3.5.2.4. Winner Loses Path

If a router's RPF_interface to the RP switches to be on a link for which it is acting as the DF, then it can no longer provide forwarding services for that link. It therefore immediately stops being the DF and restarts the election. As its path to the RP is through the link, an infinite metric is used in the Offer message it sends.

Note: At this stage the old DF will have a new RPF neighbor on the link (indicated by unicast routing) which it could use in a Pass message but this adds unnecessary complication to the election process.

3.5.2.5. Late Router Starting Up

A late router starting up after the DF election process has completed will have no immediate knowledge of the election outcome. As a result, it will start advertising its metric in Offer messages. As soon as this happens, the currently elected DF will respond with a Winner message if its metric is better than the metric in the Offer message, or with a Backoff message if its metric worse than the metric in the Offer message.

3.5.2.6. Winner Dies

Whenever the DF dies, a new DF has to be elected. The speed at which this can be achieved depends on whether there are any downstream routers on the link.

If there are downstream routers, typically their RPF_neighbor as reported by the MRIB before the DF dies will be the DF itself. They will therefore notice either a change in the metric for the route to the RP or a change in RPF_neighbor away from the DF and will restart the election by transmitting Offer messages. If according to the MRIB the RP is now reachable through the same link via another upstream router, an infinite metric will be used in the Offer.

If no downstream routers are present, the only way for other upstream routers to detect a DF failure is by the timeout of the PIM neighbor information, which will take significantly longer.

3.5.3. Election Protocol Specification

This section provides the definitive specification for the DF election process. If any discrepancy exists between section 3.5.2 and this section, the specification in this section is to be assumed correct.

3.5.3.1. Election State

The DF election state is maintained per RP for each multicast enabled interface on the router as introduced in section 3.1:

The state machine has the following four states:

`Offer`
: Initial election state. When in the Offer state a router thinks it can eventually become the winner and periodically generates Offer messages.

`Lose`
: In this state the router knows that there either is a different election winner or that no router on the link has a path to the RP.

`Winner`
: The router is the acting DF without any contest.

`Backoff`
: The router is the acting DF but another router has made a bid to take over.

In the state machine a router is considered to be an acting DF if it is in the `Win` or `Backoff` states.

The operation of the election protocol makes use of the variables and timers described below:

`Acting DF information`
: Used to store the election winner who is the currently acting DF.

`Election-Timer (DFT)`
: Used to schedule transmission of Offer, Winner and Pass messages.

`Offer-Count (OC)`
: Used to maintain the number of times an Offer or Winner message has been transmitted.

`Best-Offer`
: Used by the DF to record who has made the last offer for sending the Pass message.

3.5.3.2. Election Messages

The election process uses the following PIM control messages the packet format of which is described in section 3.7:

`Offer (OfferingID, Metric)`
: Sent by routers that believe they have a better metric to the RP than the metric that has been on offer so far.

`Winner (DF-ID, DF-Metric)`
: Sent by a router when assuming the role of the DF or when re-asserting in response to worse offers.

`Backoff (DF-ID, DF-Metric, OfferingID, OfferMetric, BackoffInterval)`
: Used by the DF to acknowledge better offers. It instructs other routers with equal or worse offers to wait till the DF passes responsibility to the sender of the offer.

`Pass (Old-DF-ID, Old-DF-Metric, New-DF-ID, New-DF-Metric)`
: Used by the old DF to pass forwarding responsibility to a router that has previously made an offer. The Old-DF-Metric is the current metric of the DF at the time the pass is sent.

3.5.3.3. Election Events

During protocol operation, in addition to the expiration of the `Election-Timer` and the reception of the four control messages, the following events can take place:

- Discovery of new RP
- Metric reported by the MRIB to reach the RP changes
- DF loses path to RP
- Detection of DF failure

3.5.3.4. Election Notation

The DF election state machine description uses the following notation in addition to the pseudocode notation described earlier in this spec.

?= denotes the operation of lowering a timer to a new value. If the timer is not running then it is started using the new value. If the timer is running with an expiration lower than the new value, then the timer is not altered.

When a control message is received and actions are specified on a condition that metrics are Better or Worse the comparison must be performed as follows:

- On receipt of an Offer or Winner message compare our current metrics for the DF with the metrics advertised for the sender of the message.
- On receipt of a Backoff or Pass message compare our current metrics for the DF with the metrics advertised for the target of the message.

When an action of "set DF to Sender or Target" is encountered during receipt of a Winner, Pass or Backoff message it means the following:

- On receipt of a Winner message set the DF to be the originator of the message and record its metrics.
- On receipt of a Pass message set the DF to be the target of the message and record its metrics.
- On receipt of a Backoff message set the DF to be the originator of the message and record its metrics.

3.5.3.5. Election State Transitions

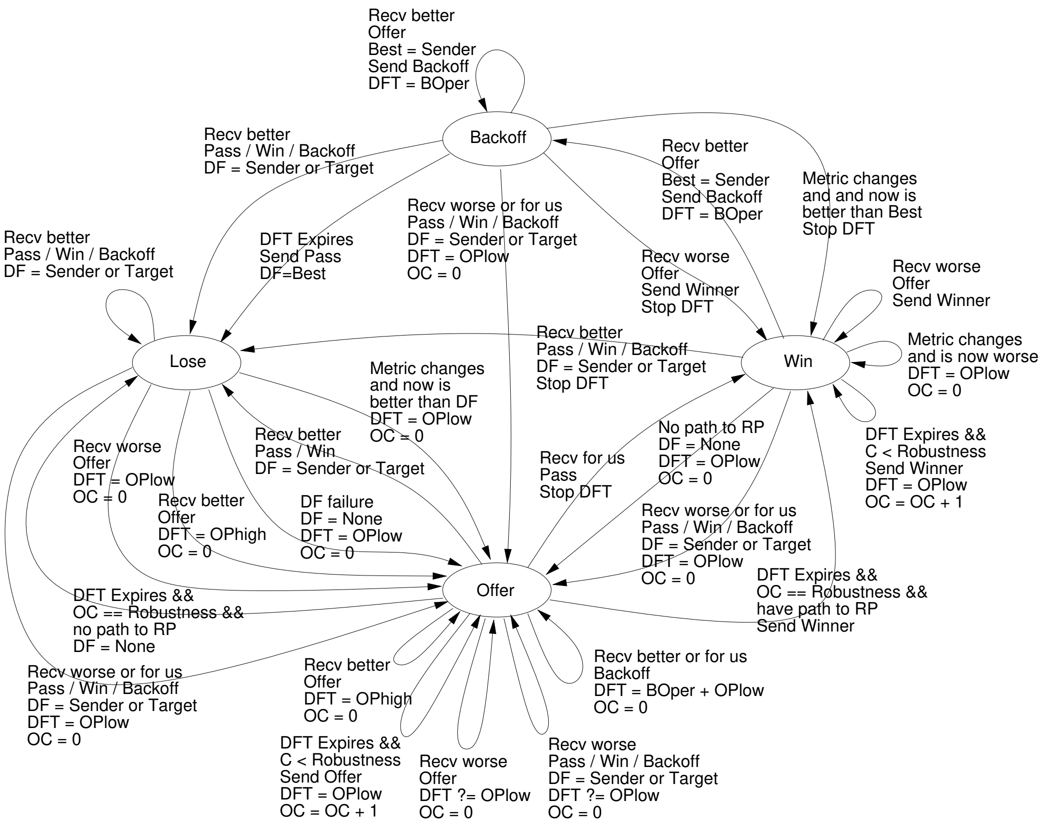

Figure 3: Designated Forwarder election state-machine

In tabular form, the state machine is:

Prev State	Event: Recv better Pass / Win	Event: Recv better Backoff	Event: Recv better Offer
Offer	-> Lose DF = Sender or Target; Stop DFT	- DFT = BOperiod + OPlow; OC = 0	- DFT = OPhigh; OC = 0
Lose	- DF = Sender or Target	- DF = Sender	-> Offer DFT = OPhigh; OC = 0
Win	-> Lose DF = Sender or Target; Stop DFT	-> Lose DF = Sender; Stop DFT	-> Backoff Set Best to Sender; Send Backoff; DFT = BOperiod
Backoff	-> Lose DF = Sender or Target; Stop DFT	-> Lose DF = Sender; Stop DFT	- Set Best to Sender; Send Backoff; DFT = BOperiod

Prev State	Event			
	Recv Backoff for us	**Recv Pass for us**	**Recv Worse Pass / Win / Backoff**	**Recv worse Offer**
Offer	- DFT = BOperiod + OPlow; OC = 0	-> Win Stop DFT	- Set DF to Sender or Target; DFT ?= OPlow; OC = 0	- DFT ?= OPlow; OC = 0
Lose	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender or Target; DFT = OPlow; OC = 0	-> Offer DFT = OPlow; OC = 0
Win	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender or Target; DFT = OPlow; OC = 0	- Send Winner
Backoff	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender; DFT = OPlow; OC = 0	-> Offer DF = Sender or Target; DFT = OPlow; OC = 0	-> Win Send Winner; Stop DFT

In Offer State		
DFT Expires and OC is less than Robustness	**DFT Expires and OC is equal to Robustness and we have path to RP**	**DFT Expires and OC is equal to Robustness and there is no path to RP**
- Send Offer; DFT = OPlow; OC = OC + 1	-> Win Send Winner	-> Lose Set DF to None

In Lose State	
Detect DF Failure	**Metric changes and now is better than DF**
-> Offer DF = None; DFT = OPlow_int; OC = 0	-> Offer DFT = OPlow_int; OC = 0

In Win State		
Metric changes and is now worse	**Timer Expires and Count is less than Robustness**	**No path to RP**
- DFT = OPlow; OC = 0	- Send Winner; DFT = OPlow; OC = OC + 1	-> Offer Set DF to None; DFT = OPlow; OC = 0

In Backoff State	
Metric changes and is now better than Best	**Timer Expires**
-> Win Stop Timer	-> Lose Send Pass; Set DF to stored Best

3.6. Timers and Constants

BIDIR-PIM maintains the following timers, as discussed in section 3.1. All timers are countdown timers - they are set to a value and count down to zero, at which point they typically trigger an action. Of course they can just as easily be implemented as count-up timers, where the absolute expiry time is stored and compared against a real-time clock, but the language in this specification assumes that they count downwards to zero.

Per Rendezvous-Point (RP):

Per interface (I):

DF Election Timer: DFT(RP,I)

Per Group (G):

Upstream Join Timer: JT(G)

Per interface (I):

Join Expiry Timer: ET(G,I)

PrunePending Timer: PPT(G,I)

When timers are started or restarted, they are set to default values. This section summarizes those default values.

Timer Name: DF Election Timer (DFT)

Value Name	Value	Explanation
Offer_Period	100 ms	Interval to wait between repeated Offer and Winner messages.
Backoff_Period	1 sec	Period that acting DF waits between receiving a better Offer and sending the Pass message to transfer DF responsibility.
OPLow	rand(0.5, 1) * Offer_Period	Range of actual randomised value used between repeated messages.
OPHigh	Election_Robustness * Offer_Period	Interval to wait in order to give a chance to a router with a better Offer to become the DF.

Timer Names: Join Expiry Timer (ET(G,I))

Value Name	Value	Explanation
J/P HoldTime	from message	Hold Time from Join/Prune Message

Timer Names: Prune Pending Timer (PPT(G,I))

Value Name	Value	Explanation
J/P Override Interval	Default: 3 secs	Short period after a join or prune to allow other routers on the LAN to override the join or prune

Timer Names: Upstream Join Timer (JT(G))

Value Name	Value	Explanation
t_periodic	Default: 60 secs	Period between Join/Prune Messages
t_suppressed	rand(1.1 * t_periodic, 1.4 * t_periodic)	Suppression period when someone else sends a J/P message so we don't need to do so.
t_override	rand(0, 0.9 * J/P Override Interval)	Randomized delay to prevent response implosion when sending a join message to override someone else's prune message.

For more information about these values refer to the PIM-SM [9] documentation.

Constant Name: DF Election Robustness

Constant Name	Value	Explanation
Election_Robustness	Default: 3	Minimum number of election messages that must be lost in order for election to fail.

3.7. BIDIR PIM Packet Formats

This section describes the details of the packet formats for BIDIR-PIM control messages. BIDIR-PIM shares a number of control messages in common with PIM-SM [9] well as the format for the Encoded-Unicast address. For details on the format of these packets please refer to the PIM-SM documentation. Here we will only define the additional packets that are introduced by BIDIR-PIM. These are the packets used in the DF election process as well as the Bidir_Capable PIM-Hello option.

3.7.1. DF Election Packet Formats

All PIM control messages have IP protocol number 103.

BIDIR-PIM messages are multicast with TTL 1 to the 'ALL-PIM-ROUTERS' group '224.0.0.13'.

All DF election BIDIR-PIM control messages share the common header below:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|PIM Ver| Type  |Subtype| Rsvd  |           Checksum            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Encoded-Unicast-RP-Address                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                    Sender Metric Preference                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                          Sender Metric                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

PIM Ver
PIM Version number is 2.

Type
All DF-Election PIM control messages share the PIM message Type of 10.

Subtype
Subtypes for DF election messages are:

1 = Offer
2 = Winner
3 = Backoff
4 = Pass

Rsvd
Set to zero on transmission. Ignored upon receipt.

Checksum
The checksum is standard IP checksum, i.e. the 16-bit one's complement of the one's complement sum of the entire PIM message. For computing the checksum, the checksum field is zeroed.

RP-Address
The address of the bidir RP for which the election is taking place (note that the length of this field is more than 32 bits).

Sender Metric Preference
Preference value assigned to the unicast routing protocol that the message sender used to obtain the route to the RP-address.

Sender Metric
The unicast routing table metric used by the message sender to reach the RP. The metric is in units applicable to the unicast routing protocol used.

In addition to the fields defined above the Backoff and Pass messages have the extra fields described below.

3.7.2. Backoff Message

The Backoff message uses the following fields in addition to the common election message format described above.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|               Encoded-Unicast-Offering-Address                |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                  Offering Metric Preference                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Offering Metric                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|            Interval           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Offering Address
The address of the router that made the last (best) Offer (note that the length of this field is

more than 32 bits).

Offering Metric Preference
Preference value assigned to the unicast routing protocol that the offering router used to obtain the route to RP-address.

Offering Metric
The unicast routing table metric used by the offering router to reach the RP. The metric is in units applicable to the unicast routing protocol used.

Interval
The backoff interval in milliseconds to be used by routers with worse metrics than the offering router.

3.7.3. Pass Message

The Pass message uses the following fields in addition to the common election fields described above.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|              Encoded-Unicast-New-Winner-Address               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                 New Winner Metric Preference                  |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                      New Winner Metric                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

New Winner Address
The address of the router that made the last (best) Offer (note that the length of this field is more than 32 bits).

New Winner Metric Preference
Preference value assigned to the unicast routing protocol that the offering router used to obtain the route to RP-address.

New Winner Metric
The unicast routing table metric used by the offering router to reach the RP. The metric is in units applicable to the unicast routing protocol used.

3.7.4. Bidir Capable PIM-Hello Option

BIDIR-PIM introduces one new PIM-Hello option.

- OptionType 22: Bidir Capable

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Type = 22            |          Length = 0           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4. RP Discovery

Routers discover that a range of multicast group addresses operates in bi-directional mode and the address of the Rendezvous-Point serving the group range either through static configuration or using an automatic RP discovery mechanism like the PIM Bootsrtap mechanism (BSR). [12].

By default the BSR protocol advertises RPs that operate the PIM-SM protocol. In order to identify a RP as operating in BIDIR mode, the Encoded-Group Address field in Bootstrap and Candidate-RP Advertisement messages has been extended by adding the BIDIR bit (B-bit) as specified below:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
| Addr Family   | Encoding Type |B|    Reserved   |  Mask Len     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                Group Multicast Address                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

B-bit

When the Bidir-bit is set, all BIDIR capable PIM routers will operate the protocol described in this document for the specified group range.

5. Security Considerations

All PIM control messages MAY use IPsec [6] to address security concerns. The authentication methods are addressed in a companion document [7]. Keys may be distributed as described in [8].

5.1. Appendix A: Election Reliability Enhancements

For the correct operation of bi-directional PIM it is very important to avoid situations where two routers consider themselves to be Designated Forwarders for the same link. The two precautions below are not required for correct operation but can help diagnose anomalies and correct them.

5.1.1. A.1 Missing Pass

After a DF has been elected, a router whose metrics change to become better than the DF will attempt to take over. If during the re-election the acting DF has a condition that causes it to lose all of the election messages (like a CPU overload), the new candidate will transmit three offers and assume the role of the forwarder resulting in two DFs on the link. This situation is pathological and should be corrected by fixing the overloaded router. It is desirable that such an event can be detected by a network administrator.

When a router becomes the DF for a link without receiving a Pass message from the known old DF, the PIM neighbor information for the old DF can be marked to this effect. Upon receiving the next PIM Hello message from the old DF, the router can retransmit Winner messages for all the RPs for which it acting as the DF. The anomaly may also be logged by the router to alert the operator.

5.1.2. A.2 Periodic Winner Announcement

An additional degree of safety can be achieved by having the DF for each RP periodically announce its status in a Winner message. Transmission of the periodic Winner message can be restricted to occur only for RPs which have active groups, thus avoiding the periodic control traffic in areas of the network without senders or receivers for a particular RP.

5.2. Appendix B: Interoperability with legacy code

The rules provided in [10] for interoperating between legacy PIM-SM routers and new bi-directional capable routers change only slightly to support this new proposal. The only difference is in the definition of a boundary between a bi-directional capable area and a legacy area of the network. In [10], a bidir capable router forwarding upstream, register encapsulates the data packet to the RP if its RPF neighbor is not bidir capable.

In our proposal, since all the routers on a link need to co-operate to elect the Designated Forwarder, if even one of the routers on the link is a legacy router, the election cannot take place. As a result register encapsulation is necessary if one or more routers on the RPF interface are not bi-directional capable.

As in [10], a Hello option must be used to differentiate between bi-directional capable and legacy routers, and (S,G) state must be created on the router doing the register encapsulation to prevent loops.

5.3. Appendix C: Comparison with PIM-SM

This section describes the main differences between Bidir PIM and sparse-mode PIM:

- Bidir PIM uses a single shared tree for distributing the data for all the sources of a multicast group. The use of a single tree significantly reduces state requirements on a router. The drawback is that it may produce suboptimal paths from sources to receivers possibly resulting in higher network latency and less efficient bandwidth utilisation.
- In Bidir PIM, packets traveling from a source to the RP, are natively forwarded on the shared tree. In contrast sparse-mode PIM uses unicast encapsulation or source-specific state.
- In Bidir PIM, sender-only branches do not need to keep group state. Data from the source can be natively forwarded towards the RP using RP-specific forwarding state.
- The Bidir Designated Forwarder (DF) assumes all the responsibilities of the sparse-mode DR. In a multi-access link, the DF responds to IGMP notifications. Downstream routers on the link use the DF as their upstream neighbor and direct all Join/Prune messages towards it.
- To enforce a single forwarder on multi-access links, sparse-mode PIM uses the Assert mechanism which requires data-packets to trigger protocol events. In Bidir PIM, data-driven events are completely eliminated as a correct route is always available at packet forwarding time.

  The DF election problem is easier than the assert problem because there is a small number of RPs and the per RP DF election can be done in advance. With the assert mechanism, in addition to each RP, a forwarder has to be elected for each possible source to a group. This can not be done before data is available.
- With sparse-mode PIM, when forwarding packets using shared-tree (*,G) state, a directly-connected-source check has to be made on every packet. This is done to determine if the packet was originated by a source which is directly connected to the router. For a connected source, source-specific state has to be created to register packets to the RP and prune the source off the shared tree.

  With Bidir PIM directly connected sources do not need any special handling. The DF for the RP of the group the source is sending to, seamlessly picks-up and forwards upstream traveling packets.

6. Todo list...

- Update legacy interoperability section to remove dependency on 10.
- Incorporate BSR mods into BSR spec.
- In the state machine, perhaps add an arc from NI to NI, labelled "(*,G) Join but not DF" just to make it really clear.

7. Authors' Addresses

Mark Handley
ICIR/ICSI
1947 Center St, Suite 600
Berkeley, CA 94708
mjh@icir.org

Isidor Kouvelas
Cisco Systems
kouvelas@cisco.com

Tony Speakman
Cisco Systems
speakman@cisco.com

Lorenzo Vicisano
Cisco Systems
lorenzo@cisco.com

8. Acknowledgments

The bidir proposal in this draft is heavily based on the ideas and text presented by Estrin and Farinacci in [10]. The main difference between the two proposals is in the method chosen for upstream forwarding.

We would also like to thank Deborah Estrin at ISI/USC as well as Nidhi Bhaskar, Yiqun Cai, Apoorva Karan, Rajitha Sumanasekera and Beau Williamson at cisco for their contributions and comments to this draft.

9. References

[1] T. Bates , R. Chandra , D. Katz , Y. Rekhter, "Multiprotocol Extensions for BGP-4", RFC 2283

[2] S.E. Deering, "Host extensions for IP multicasting", RFC 1112, Aug 1989.

[3] W. Fenner, "Internet Group Management Protocol, Version 2", RFC 2236.

[4] IANA, "Address Family Numbers", linked from http://www.iana.org/numbers.html

[5] T. Narten , H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 2434.

[6] S. Kent, R. Atkinson, "Security Architecture for the Internet Protocol.", RFC 2401.

[7] L. Wei, "Authenticating PIM version 2 messages", draft-ietf-pim-v2-auth-01.txt, work in progress.

[8] T. Hardjono, B. Cain, "Simple Key Management Protocol for PIM", draft-ietf-pim-simplekmp-01.txt, work in progress.

[9] B. Fenner, M. Handley, H. Holbrook, I. Kouvelas "Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised)", Work In Progress, <draft-ietf-pim-sm-v2-new-05.txt>, 2002.

[10] D. Estrin, D. Farinacci, "Bi-directional Shared Trees in PIM-SM", Work In Progress, <draft-farinacci-bidir-pim-01.txt>, May 1999.

[11] D. Estrin et al, "Protocol Independent Multicast-Sparse Mode (PIM-SM): Protocol Specification", RFC 2362, Nov 1999.

[12] W. Fenner, M. Handley, R. Kermode and D. Thaler, "Bootstrap Router (BSR) Mechanism for PIM Sparse Mode", draft-ietf-pim-sm-bsr-00.txt, work in progress.

10. Index