Agenda

- PIM-DM Overview
- PIM-DM Protocol Mechanics
- PIM-SM Overview
- PIM-SM Protocol Mechanics

PIM Dense Mode Overview

- Uses “Push” Model
  - Traffic is initially flooded to all PIM neighbors
  - Branches that don’t want data are pruned
- Multicast forwarding state is created by the arrival of data
- If the source goes inactive, the tree is torn down

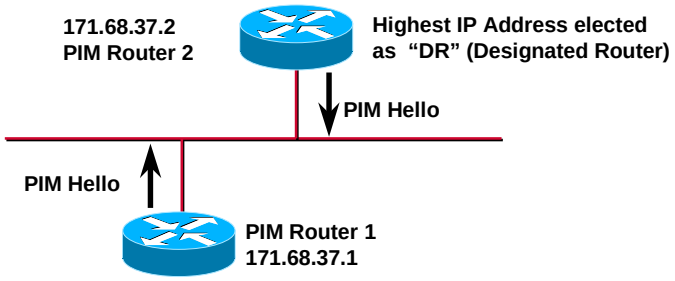
PIM Dense Mode Overview

- Grafts are used to join existing source tree
- Asserts are used to determine forwarder for multi-access LAN
- Prunes are sent on non-RPF P2P links
  - Asserts are sent on non-RPF multi-access links
- Rate-limited prunes are sent on all P2P links

PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- PIM DM Pruning
- PIM DM Grafting
- PIM Assert Mechanism

PIM Neighbor Discovery



- PIMv2 Hellos are periodically multicast to the "All-PIM-Routers" (224.0.0.13) group address. (Default = 30 seconds)
  - Note: PIMv1 multicasts PIM Query messages to the "All-Routers" (224.0.0.2) group address.
- If the "DR" times-out, a new "DR" is elected.
- The "DR" is responsible for sending all Joins and Register messages for any receivers or senders on the network.

PIM Neighbor Discovery

```
wan-gw8>show ip pim neighbor
PIM Neighbor Table
Neighbor Address  Interface          Uptime    Expires   Mode
171.68.0.70       FastEthernet0      2w1d      00:01:24  Dense
171.68.0.91       FastEthernet0      2w6d      00:01:01  Dense (DR)
171.68.0.82       FastEthernet0      7w0d      00:01:14  Dense
171.68.0.86       FastEthernet0      7w0d      00:01:13  Dense
171.68.0.80       FastEthernet0      7w0d      00:01:02  Dense
171.68.28.70      Serial2.31         22:47:11  00:01:16  Dense
171.68.28.50      Serial2.33         22:47:22  00:01:08  Dense
171.68.27.74      Serial2.36         22:47:07  00:01:21  Dense
171.68.28.170     Serial0.70         1d04h     00:01:06  Dense
171.68.27.2       Serial1.51         1w4d      00:01:25  Dense
171.68.28.110     Serial3.56         1d04h     00:01:20  Dense
171.68.28.58      Serial3.102        12:53:25  00:01:03  Dense
```

PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- PIM DM Pruning
- PIM DM Grafting
- PIM Assert Mechanism

PIM State

- Describes the “state” of the multicast distribution trees as understood by the router at this point in the network.
- Represented by entries in the multicast routing (mroute) table
  - Used to make multicast traffic forwarding decisions
  - Composed of (*, G) and (S, G) entries
  - Each entry contains RPF information
    - Incoming (i.e. RPF) interface
    - RPF Neighbor (upstream)
  - Each entry contains an Outgoing Interface List (OIL)
    - OIL may be NULL

PIM-DM State Example

```
sj-mbone> show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:00:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:10/00:00:00
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00

(128.9.160.43/32, 224.1.1.1), 00:00:10/00:02:49, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Prune/Dense, 00:00:05/00:02:55
```


PIM-DM (*,G) State Rules

- (*,G) created automatically
  - When 1st (S,G) for group is created
  - (S,G)'s always have parent (*,G)
- (*,G) reflect PIM neighbor adjacency
  - IIF = NULL
  - OIL = all interfaces
    - with PIM-DM neighbors or
    - with directly connected hosts or
    - manually configured

PIM-DM (S,G) State Rules

- (S,G) created by multicast data arrival
  - Parent (*,G) created (if doesn't exist)
  - IIF = RPF Interface in direction of source
  - OIL = Copy of OIL from (*,G) minus IIF
- Interfaces in OIL initially "Forward"
  - Go to "Pruned" state when Prune rcvd
  - "Forward" intfc timers never expire
  - "Pruned" intfc timers expire in 3 minutes

PIM-DM OIL Rules

- (*,G) OIL
  - Reflects intfc's w/PIM neighbors or
  - Locally connected members or
  - Manually configured interfaces
- (S,G) OIL
  - Copy of (*,G) OIL less IIF
- Interfaces in (S,G) OIL "pruned"
  - When appropriate Prune received
    - Prune Expiration counter (3 min) started
    - Interface marked "Prune/Dense" (not removed)
    - Returns to "Forward/Dense" when Prune expires

PIM-DM State Flags

- D = Dense Mode
- C = Directly Connected Host
- L = Local (Router is member)
- P = Pruned (All intfcs in OIL = Prune)
- T = Forwarding via SPT
  - Indicates at least one packet was forwarded
- J = Join SPT
  - Always on in (*,G) entry in PIM-DM
  - Basically meaningless in PIM-DM

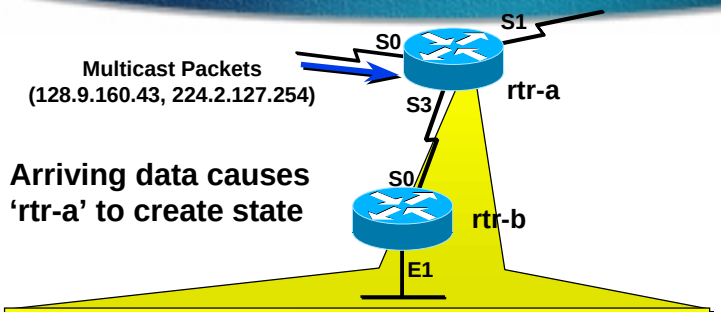
PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- PIM DM Pruning
- PIM DM Grafting
- PIM Assert Mechanism

PIM-DM Forwarding Rules

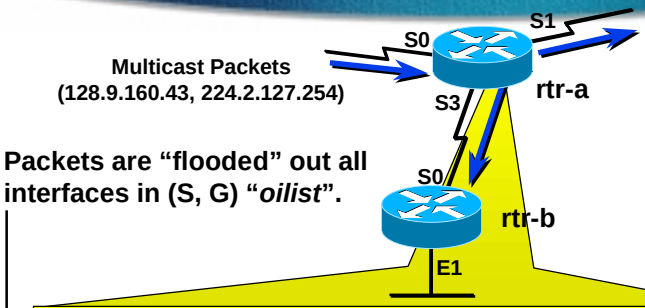
- Use longest match entry
  - Use (S, G) entry if exists
  - Otherwise, use (*, G) entry
  - Effectively, only (S,G)'s used in PIM-DM
- RPF check first
  - If Packet didn't arrive via IIF, drop it.
- Forward Packet (if RPF succeeded)
  - Send out all "unpruned" interfaces in OIL

PIM DM Forwarding




```
(*, 224.2.127.254), 00:00:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:10/00:00:00
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:10/00:02:49, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00
```

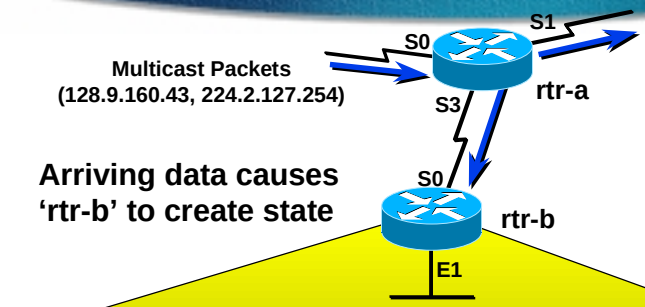
PIM DM Forwarding



Packets are “flooded” out all interfaces in (S, G) “*oilist*”.

```
(*, 224.2.127.254), 00:00:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:10/00:00:00
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:10/00:02:49, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00
```


PIM DM Forwarding



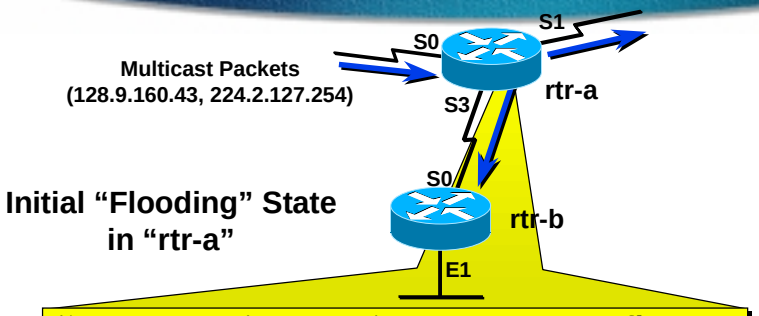
Arriving data causes ‘rtr-b’ to create state

```
(*, 224.2.127.254), 00:00:12/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:12/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:12/00:02:48, flags: PT
  Incoming interface: Serial0, RPF nbr 198.92.2.31
  Outgoing interface list: Null
```

PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- **PIM DM Pruning**
- PIM DM Grafting
- PIM Assert Mechanism

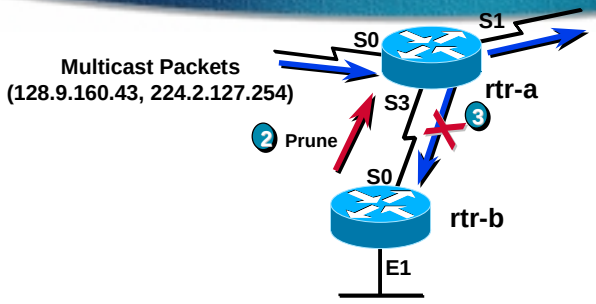
PIM DM Pruning



Initial "Flooding" State in "rtr-a"

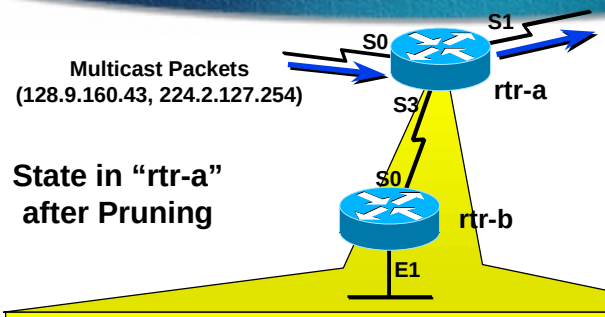
```
(*, 224.2.127.254), 00:00:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:10/00:00:00
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:10/00:02:49, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Serial1, Forward/Dense, 00:00:10/00:00:00
    Serial3, Forward/Dense, 00:00:10/00:00:00
```

PIM DM Pruning



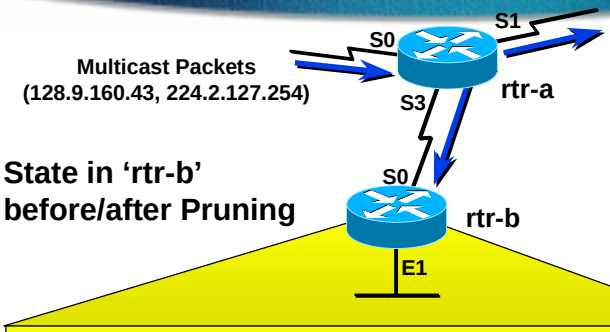
1. "rtr-a" initially floods (S, G) traffic out all interfaces in *"oilist"*.
2. "rtr-b" is a leaf node w/o receivers. Sends Prune for (S,G).
3. "rtr-a" Prunes interface for (S,G).

PIM DM Pruning



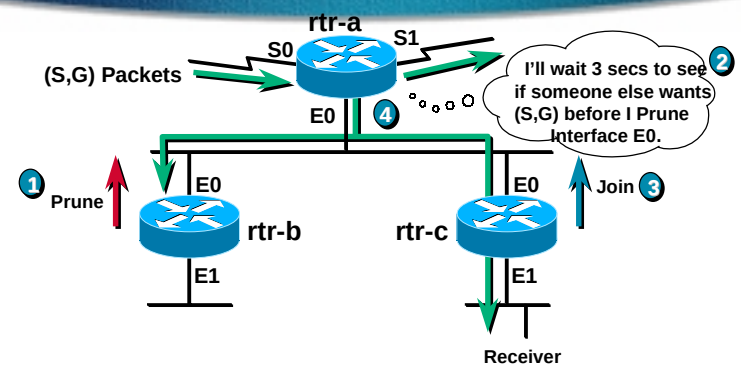

```
(*, 224.2.127.254), 00:00:12/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:12/00:00:00
    Serial1, Forward/Dense, 00:00:12/00:00:00
    Serial3, Forward/Dense, 00:00:12/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:12/00:02:48, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Serial1, Forward/Dense, 00:00:12/00:00:00
    Serial3, Prune/Dense, 00:00:12/00:02:56
```

PIM DM Pruning




```
(*, 224.2.127.254), 00:00:12/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:00:12/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:00:12/00:02:48, flags: PT
  Incoming interface: Serial0, RPF nbr 198.92.2.31
  Outgoing interface list: Null
```


Prune Delay on Multi-Access Networks

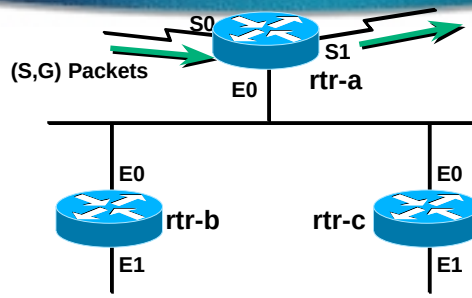


1. "rtr-b" is a leaf node w/o receivers. Sends Prune for (S,G).
2. "rtr-a" schedules a Prune for (S,G) to occur in 3 seconds.
3. "rtr-c" hears Prune from "rtr-b". Overrides with a Join.
4. "rtr-a" hears Join and cancels Prune for (S,G).

PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- PIM DM Pruning
- PIM DM Grafting
- PIM Assert Mechanism

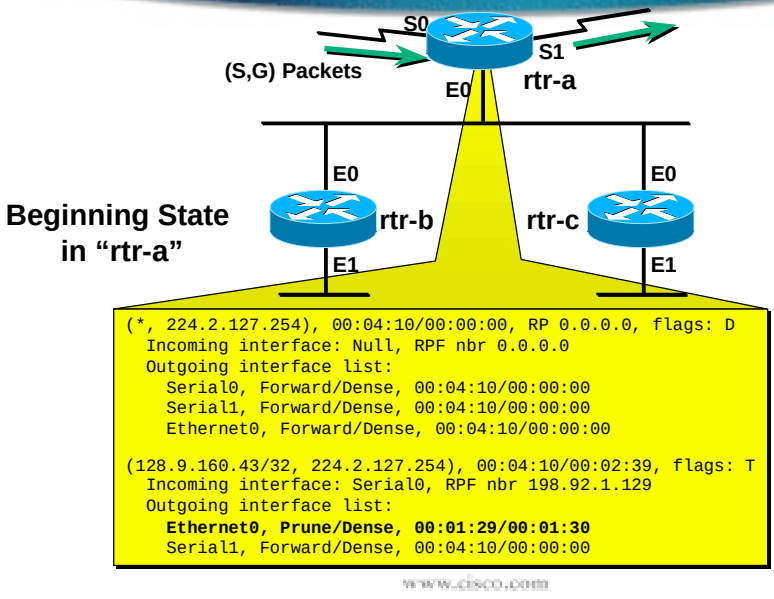
PIM DM Grafting



Beginning State

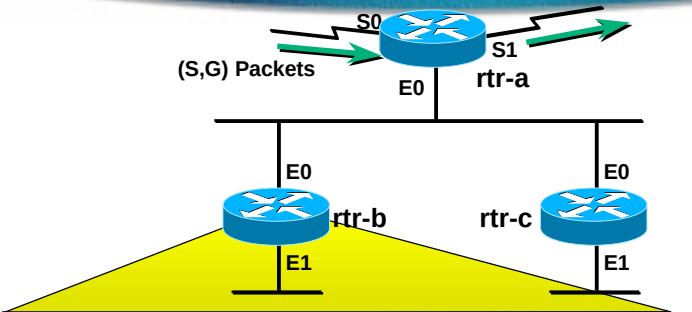
- "rtr-b" and "rtr-c" have previously Pruned (S,G) traffic.
- "rtr-a" is still forwarding traffic downstream via S1.

PIM DM Grafting




```
(*, 224.2.127.254), 00:04:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:04:10/00:00:00
    Serial1, Forward/Dense, 00:04:10/00:00:00
    Ethernet0, Forward/Dense, 00:04:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:04:10/00:02:39, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Ethernet0, Prune/Dense, 00:01:29/00:01:30
    Serial1, Forward/Dense, 00:04:10/00:00:00
```

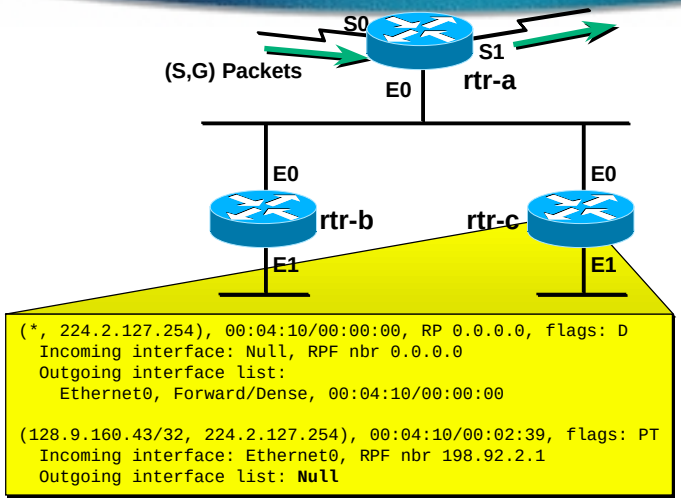

PIM DM Grafting




```
(*, 224.2.127.254), 00:04:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Ethernet0, Forward/Dense, 00:04:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:04:10/00:02:39, flags: PT
  Incoming interface: Ethernet0, RPF nbr 198.92.2.1
  Outgoing interface list: Null
```

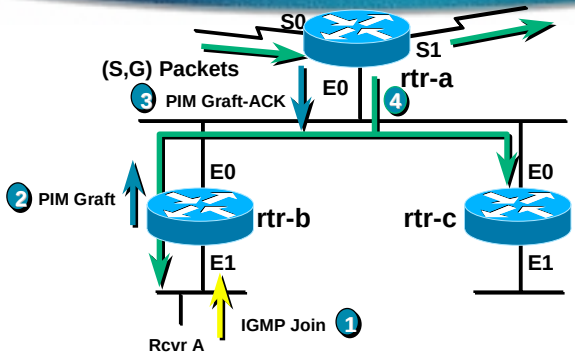
Beginning State in “rtr-b”

PIM DM Grafting



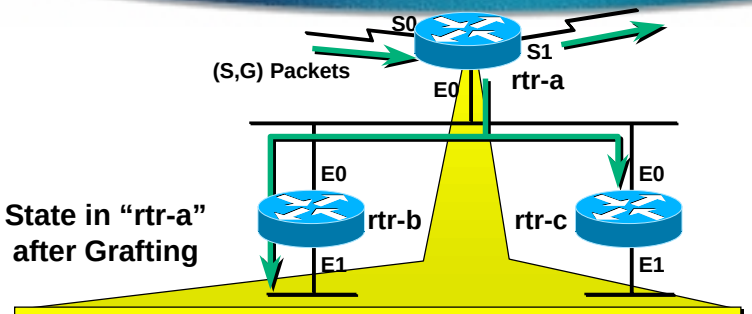
Beginning State in "rtr-c"

PIM DM Grafting



1. "Rcvr A" wishes to receive group G traffic. Sends IGMP Join for G.
2. "rtr-b" sends PIM Graft for Group (S,G).
3. "rtr-a" acknowledges with a PIM Graft-Ack.
4. "rtr-a" begins forwarding traffic for (S,G).

PIM DM Grafting

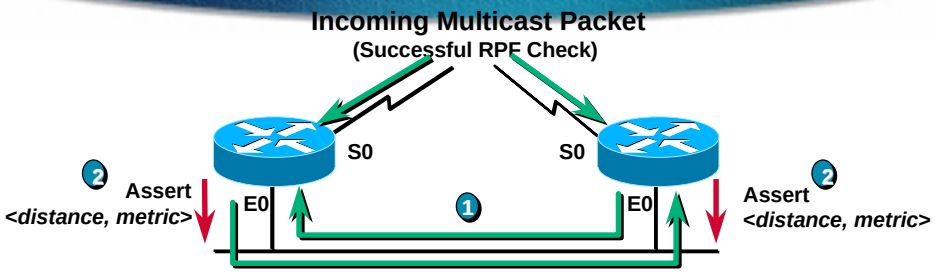



```
(*, 224.2.127.254), 00:04:10/00:00:00, RP 0.0.0.0, flags: D
  Incoming interface: Null, RPF nbr 0.0.0.0
  Outgoing interface list:
    Serial0, Forward/Dense, 00:04:10/00:00:00
    Serial1, Forward/Dense, 00:04:10/00:00:00
    Ethernet0, Forward/Dense, 00:04:10/00:00:00

(128.9.160.43/32, 224.2.127.254), 00:04:10/00:02:39, flags: T
  Incoming interface: Serial0, RPF nbr 198.92.1.129
  Outgoing interface list:
    Ethernet0, Forward/Dense, 00:00:25/00:00:00
    Serial1, Forward/Dense, 00:04:10/00:00:00
```


PIM-DM Protocol Mechanics

- PIM Neighbor Discovery
- PIM DM State
- PIM DM Forwarding
- PIM DM Pruning
- PIM DM Grafting
- PIM Assert Mechanism

PIM Assert Mechanism



1. Routers *receive* packet on an interface in their "*oilist*"!!
   - Only one router should continue sending to avoid duplicate packets.
2. Routers send "PIM Assert" messages
   - Compare *distance* and *metric* values
   - Router with best route to source wins
   - If *metric* & *distance* equal, highest IP adr wins
   - Losing router stops sending (prunes interface)

PIM-SM Overview

- **Explicit join model**
  - Receivers join to the Rendezvous Point (RP)
  - Senders register with the RP
  - Data flows down the shared tree and goes only to places that need the data from the sources
  - Last hop routers can join source tree if the data rate warrants by sending joins to the source
- **RPF check depends on tree type**
  - For shared trees, uses RP address
  - For source trees, uses Source address

PIM-SM Overview

- Only one RP is chosen for a particular group
- RP statically configured or dynamically learned (Auto-RP, PIM v2 candidate RP advertisements)
- Data forwarded based on the source state (S, G) if it exists, otherwise use the shared state (*, G)
- RFC 2326 - “PIM Sparse Mode Protocol Spec”

PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM-SM State Example

```
sj-mbone> show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
       M - MSDP created entry, X - Proxy Join Timer Running
       A - Advertised via MSDP
Timers: Uptime/Expires
Interface state: Interface, Next-Hop or VCD, State/Mode

(*, 224.1.1.1), 00:13:28/00:02:59, RP 10.1.5.1, flags: SCJ
  Incoming interface: Ethernet0, RPF nbr 10.1.2.1,
  Outgoing interface list:
    Ethernet1, Forward/Sparse, 00:13:28/00:02:32
    Serial0, Forward/Sparse, 00:4:52/00:02:08

(171.68.37.121/32, 224.1.1.1), 00:01:43/00:02:59, flags: CJT
  Incoming interface: Serial0, RPF nbr 192.10.2.1
  Outgoing interface list:
    Ethernet1, Forward/Sparse, 00:01:43/00:02:11
    Ethernet0, forward/Sparse, 00:01:43/00:02:11
```


PIM-SM (*,G) State Rules

- (*,G) creation
  - Upon receipt of a (*,G) Join or
  - Automatically if (S,G) must be created
- (*,G) reflects default group forwarding
  - IIF = RPF interface toward RP
  - OIL = interfaces
    - that received a (*,G) Join or
    - with directly connected hosts or
    - manually configured
- (*,G) deletion
  - When OIL = NULL and
  - no child (S,G) state exists

PIM-SM (S,G) State Rules

- (S,G) creation
  - By receipt of (S,G) Join or Prune or
  - By “Register” process
  - Parent (*,G) created (if doesn’t exist)
- (S,G) reflects forwarding of “S” to “G”
  - IIF = RPF Interface normally toward source
    - RPF toward RP if “RP-bit” set
  - OIL = Initially, copy of (*,G) OIL minus IIF
- (S,G) deletion
  - By normal (S,G) entry timeout

PIM-SM OIL Rules

- Interfaces in OIL added
  - By receipt of Join message
    - Intfc’s added to (*,G) are added to all (S,G)’s
- Interfaces in OIL removed
  - By receipt of Prune message
    - Intfc’s removed from (*,G) are removed from all (S,G)’s
  - Interface Expire timer counts down to zero
    - Timer reset (to 3 min.) by receipt of periodic Join or
    - By IGMP membership report

PIM-SM State Flags

- S = Sparse Mode
- C = Directly Connected Host
- L = Local (Router is member)
- P = Pruned (All intfcs in OIL = Prune)
- T = Forwarding via SPT
  - Indicates at least one packet was forwarded

PIM-SM State Flags (cont.)

- J = Join SPT
  - In (*, G) entry
    - Indicates SPT-Threshold is being exceeded
    - Next (S,G) received will trigger join of SPT
  - In (S, G) entry
    - Indicates SPT joined due to SPT-Threshold
    - If rate < SPT-Threshold, switch back to Shared Tree
- F = Register
  - In (S,G) entry
    - "S" is a directly connected source
    - Triggers the Register Process
  - In (*, G) entry
    - Set when "F" set in at least one child (S,G)

PIM-SM State Flags (cont.)

- R = RP bit
  - (S, G) entries only
  - Set by (S,G)RP-bit Prune
  - Indicates info is applicable to Shared Tree
  - Used to prune (S,G) traffic from Shared Tree
    - Initiated by Last-hop router after switch to SPT
  - Modifies (S,G) forwarding behavior
    - IIF = RPF toward RP (I.e. up the Shared Tree)
    - OIL = Pruned accordingly

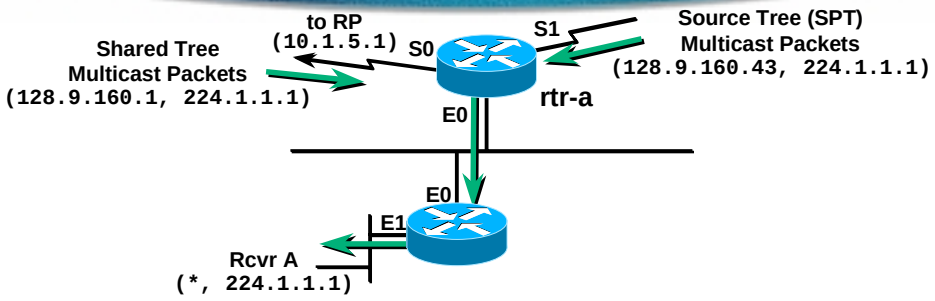
PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM-SM Forwarding Rules

- Use longest match entry
  - Use (S, G) entry if exists
  - Otherwise, use (*, G) entry
- RPF check first
  - If Packet didn't arrive via IIF, drop it.
- Forward Packet (if RPF succeeded)
  - Send out all "unpruned" interfaces in OIL

PIM SM Forwarding



- Packets are "forwarded" out all interfaces in "*oilist*".
- PIM Sparse mode interfaces are placed on the "*oilist*" for a Multicast Group IF:
  - PIM neighbor Joins the group on this interface
  - Host on this interface has joined the group
  - Interface has been manually configured to join group.

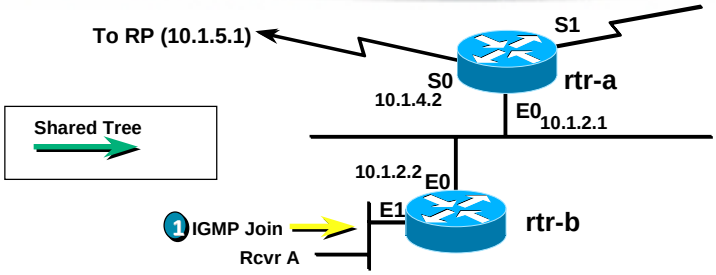
PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM SM Joining

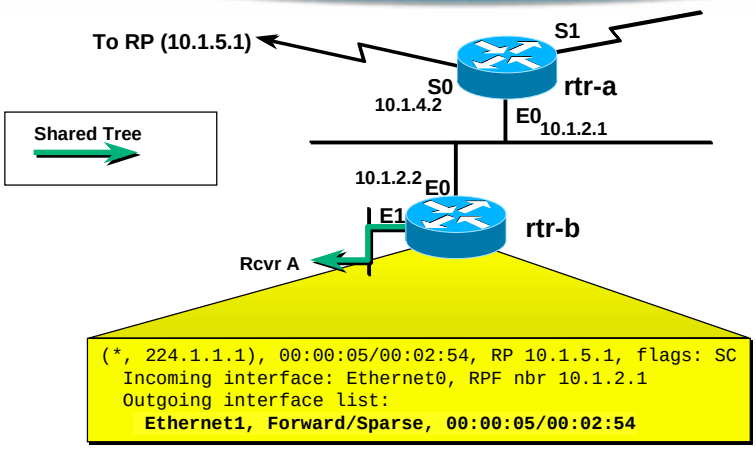
- Leaf routers send a (*,G) Join to toward RP
  - Joins sent hop-by-hop via unicast path toward RP
- Each router along path creates (*,G) state
  - IF no (*,G) state, create it & send a Join toward RP
  - ELSE Join process complete. Reached the (*,G) tree.

PIM SM Joining



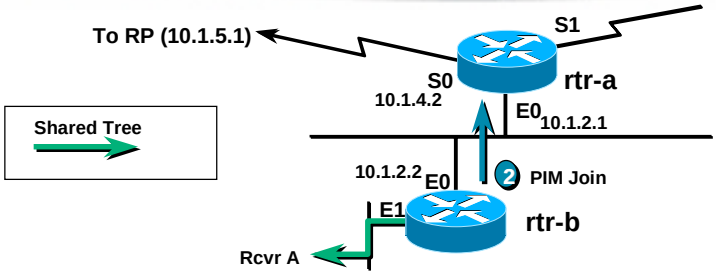
1. "Rcvr A" wishes to receive group G traffic. Sends IGMP Join for G.

PIM SM Joining



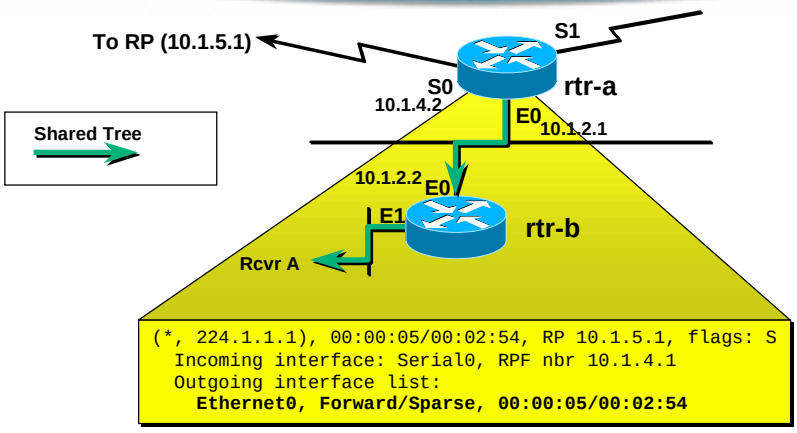
"rtr-b" creates (*, 224.1.1.1) state

PIM SM Joining



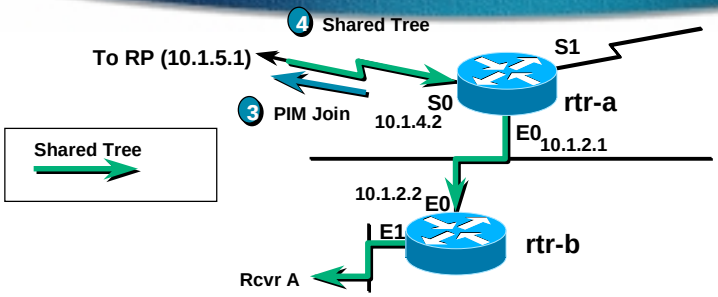
1. "Rcvr A" wishes to receive group G traffic. Sends IGMP Join for G.
2. "rtr-b" sends (*,G) Join towards RP.

PIM SM Joining



"rtr-a" creates (*, 224.1.1.1) state.

PIM SM Joining



1. "Rcvr A" wishes to receive group G traffic. Sends IGMP Join for G.
2. "rtr-b" sends (*,G) Join towards RP.
3. "rtr-a" sends (*,G) Join towards RP.
4. Shared tree is built all the way back to the RP.

PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM SM Registering

- Senders begin sourcing Multicast Traffic
  - Senders don't necessarily perform IGMP group joins.
- 1st-hop router unicasts "Registers" to RP
  - A Mcast packet is encapsulated in each Register msg
  - Registers messages follow unicast path to RP
- RP receives "Register" messages
  - De-encapsulates the Mcast packet inside Register msg
  - Forwards Mcast packet down Shared Tree
  - Sends (S,G) Join toward Source / 1st-Hop router to build an (S,G) SPT between Source and RP

PIM SM Registering

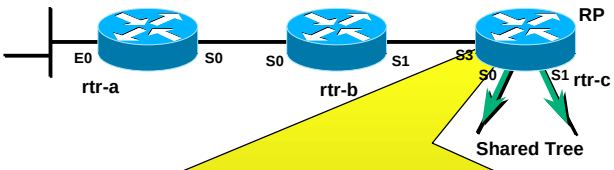
- 1st-hop router receives (S,G) Join
  - SPT between Source and RP now built.
  - Begins forwarding traffic down (S,G) SPT to RP
  - (S,G) Traffic temporarily flowing down 2 paths to RP
- RP receives traffic down native (S,G) SPT
  - Sends a "Register-Stop" msg to Source / 1st-Hop router.
- 1st-Hop router receives "Register-Stop" msg
  - Stops encapsulating traffic in "Register" messages
  - (S,G) Traffic now flowing down single SPT to RP

PIM SM Register Examples

- **Receivers Join Group First**
- **Source Registers First**
- **Receivers along the SPT**

PIM SM Registering

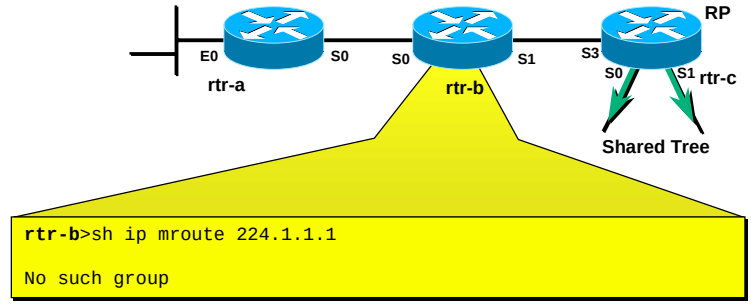
Receiver Joins Group First




```
(*, 224.1.1.1), 00:00:03/00:02:56, RP 171.68.28.140, flags:S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:03:14/00:02:59
    Serial1, Forward/Sparse, 00:03:14/00:02:59
```

State in "RP" before any source registers
(with receivers on Shared Tree)

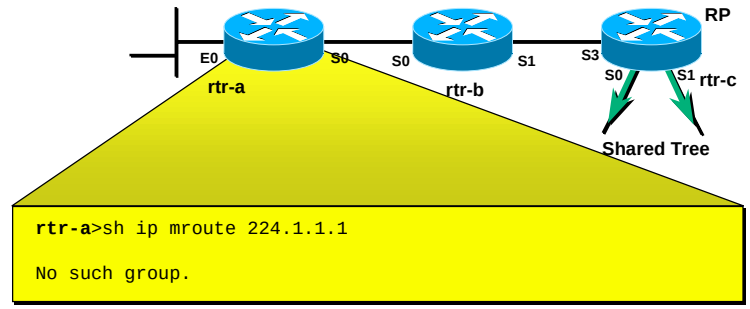
PIM SM Registering
Receiver Joins Group First




```
rtr-b>sh ip mroute 224.1.1.1

No such group
```

State in “rtr-b” before any source registers
(with receivers on Shared Tree)

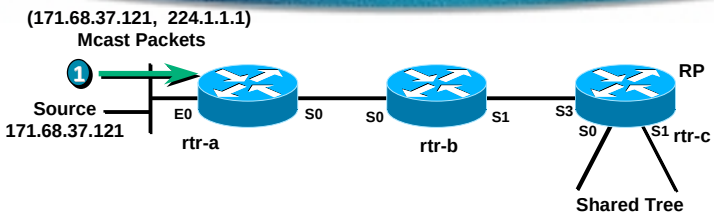
PIM SM Registering
Receiver Joins Group First




```
rtr-a>sh ip mroute 224.1.1.1

No such group.
```

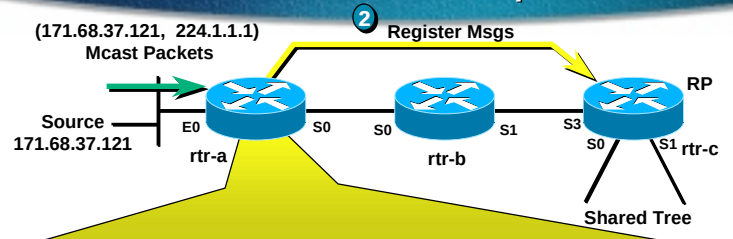
State in “rtr-a” before any source registers
(with receivers on Shared Tree)

PIM SM Registering
Receiver Joins Group First



1. "Source" begins sending group G traffic.

PIM SM Registering
Receiver Joins Group First

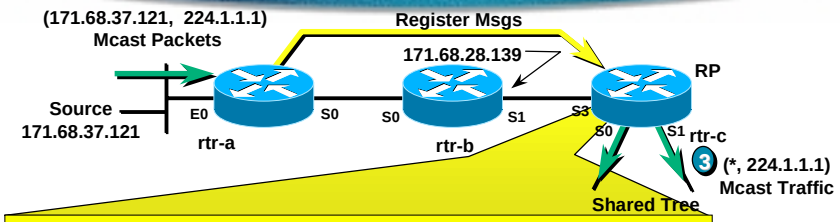



```
(*, 224.1.1.1), 00:00:03/00:02:56, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:00:03/00:02:56, flags: FPT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0, Registering
  Outgoing interface list: Null
```

"rtr-a" creates (S, G) state for source
(After automatically creating a (*, G) entry)

1. "Source" begins sending group G traffic.
2. "rtr-a" encapsulates packets in Registers; unicasts to RP.

PIM SM Registering
Receiver Joins Group First

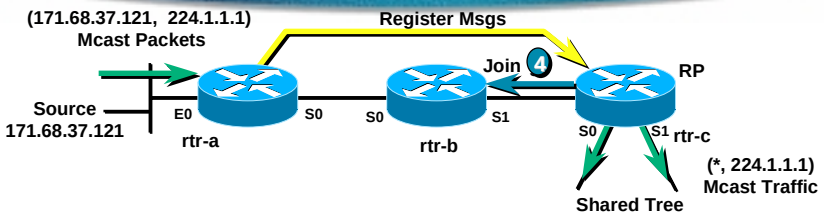



```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:09:21/00:02:38
    Serial1, Forward/Sparse, 00:03:14/00:02:46

(171.68.37.121, 224.1.1.1, 00:01:15/00:02:46, flags:
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:00:49/00:02:11
    Serial1, Forward/Sparse, 00:00:49/00:02:11
```

"RP" processes Register; creates (S, G) state

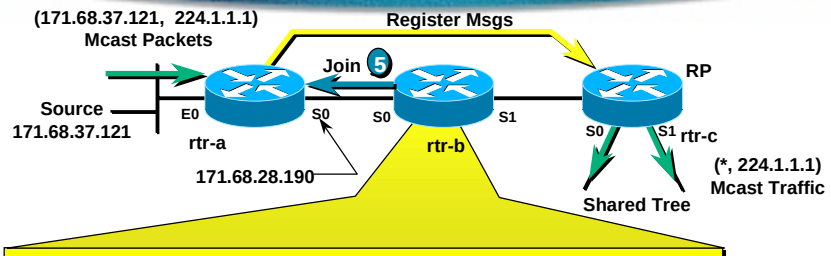
3. **"rtr-c" (RP) de-encapsulates packets; forwards down Shared tree.**

PIM SM Registering
Receiver Joins Group First



4. **RP sends (S,G) Join toward Source to build SPT.**

PIM SM Registering
Receiver Joins Group First



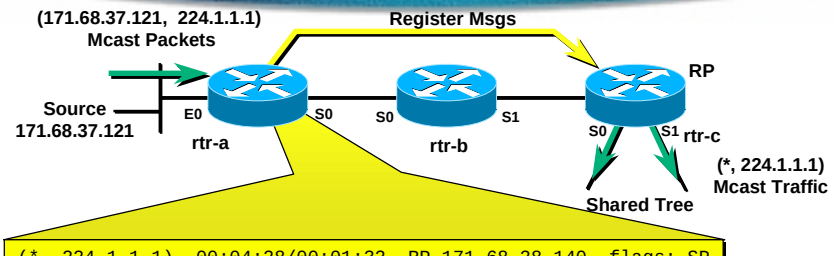

```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial1, RPF nbr 171.68.28.140,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags:
  Incoming interface: Serial0, RPF nbr 171.68.28.190
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:04:28/00:01:32
```

"rtr-b" processes Join, creates (S, G) state

(After automatically creating the (*, G) entry)

4. **RP sends (S,G) Join toward Source to build SPT.**
5. **"rtr-b" sends (S,G) Join toward Source to continue building SPT.**

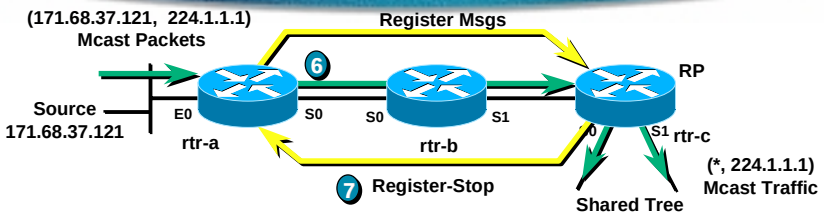
PIM SM Registering
Receiver Joins Group First




```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: FT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0, Registering
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:04:28/00:01:32
```

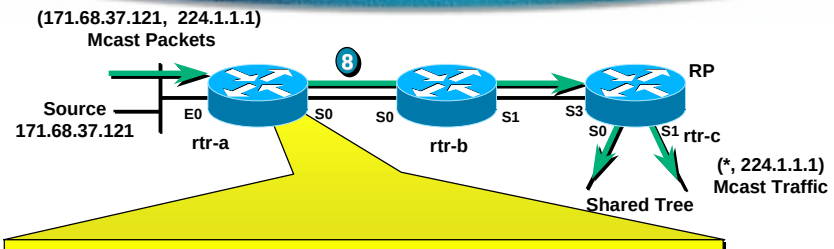
"rtr-a" processes the (S, G) Join; adds Serial0 to OIL

PIM SM Registering
Receiver Joins Group First



6. RP begins receiving (S,G) traffic down SPT.
7. RP sends “Register-Stop” to “rtr-a”.

PIM SM Registering
Receiver Joins Group First

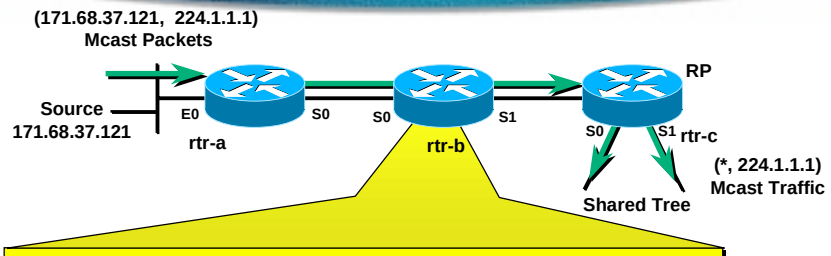



```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: FT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0,
  Outgoing interface list:
     Serial0, Forward/Sparse, 00:04:28/00:01:32
```

“rtr-a” stops sending Register messages
(Final State in “rtr-a”)

8. (S,G) Traffic now flowing down a single path (SPT) to RP.

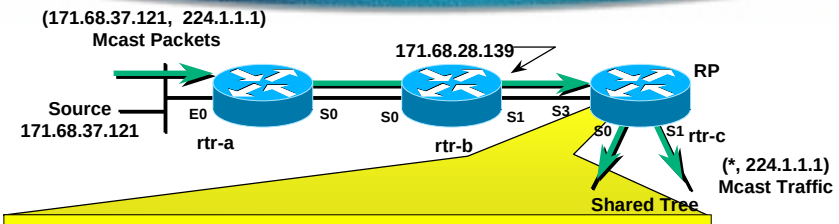
PIM SM Registering
Receiver Joins Group First




```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial1, RPF nbr 171.68.28.140,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: T
  Incoming interface: Serial0, RPF nbr 171.68.28.190
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:04:28/00:01:32
```

Final state in "rtr-b"

PIM SM Registering
Receiver Joins Group First



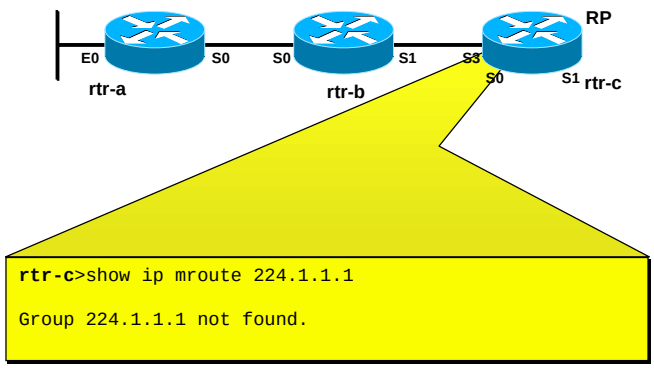

```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:09:21/00:02:38
    Serial1, Forward/Sparse, 00:03:14/00:02:46

(171.68.37.121, 224.1.1.1, 00:01:15/00:02:46, flags: T
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:00:49/00:02:11
    Serial1, Forward/Sparse, 00:00:49/00:02:11
```

Final state in the "RP"
(with receivers on Shared Tree)

PIM SM Register Examples

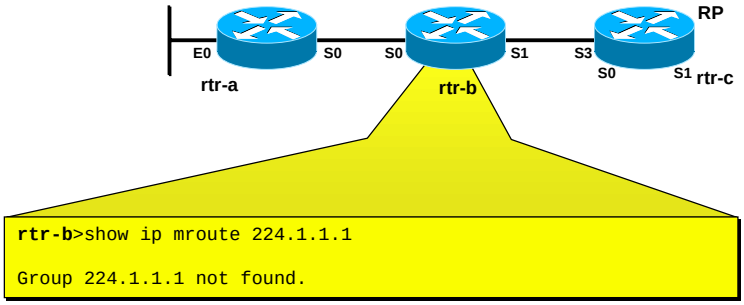
- Receivers Join Group First
- Source Registers First
- Receivers along the SPT

PIM SM Registering
Source Registers First



State in "RP" before Registering
(without receivers on Shared Tree)

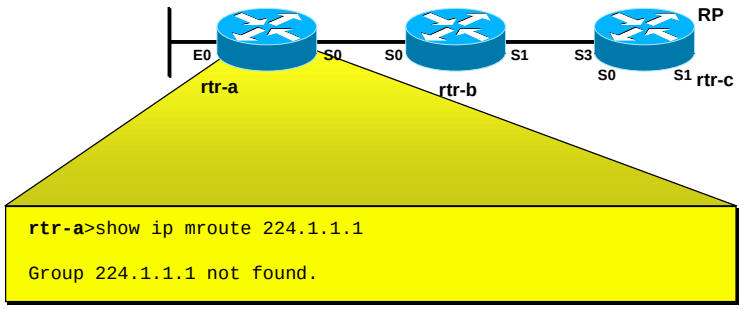
PIM SM Registering
Source Registers First




```
rtr-b>show ip mroute 224.1.1.1

Group 224.1.1.1 not found.
```

State in “rtr-b” before any source registers
(with receivers on Shared Tree)

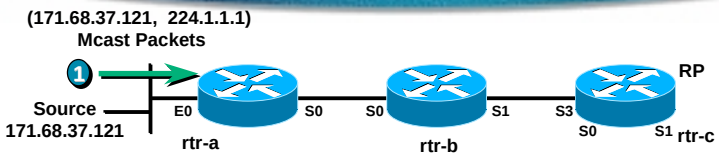
PIM SM Registering
Source Registers First




```
rtr-a>show ip mroute 224.1.1.1

Group 224.1.1.1 not found.
```

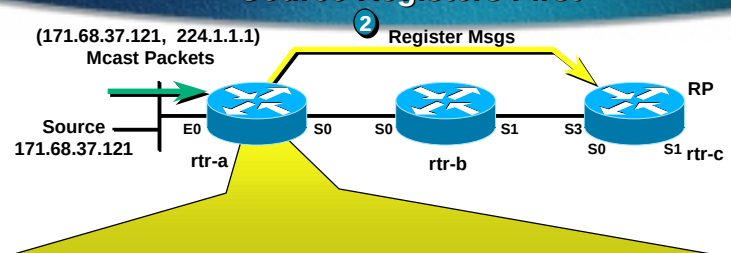
State in “rtr-a” before any source registers
(with receivers on Shared Tree)

PIM SM Registering
Source Registers First



1. "Source" begins sending group G traffic.

PIM SM Registering
Source Registers First

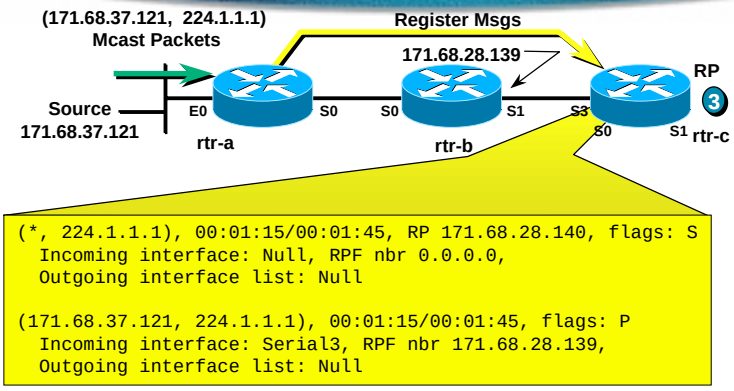



```
(*, 224.1.1.1), 00:00:03/00:02:56, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:00:03/00:02:56, flags: FPT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0, Registering
  Outgoing interface list: Null
```

"rtr-a" creates (S, G) state for source
(After automatically creating a (*, G) entry)

1. "Source" begins sending group G traffic.
2. "rtr-a" encapsulates packets in Registers; unicasts to RP.

PIM SM Registering
Source Registers First



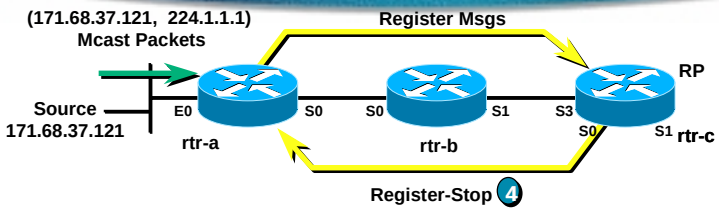

```
(*, 224.1.1.1), 00:01:15/00:01:45, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list: Null

(171.68.37.121, 224.1.1.1), 00:01:15/00:01:45, flags: P
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list: Null
```

"RP" processes Register; creates (S, G) state

(After automatically creating the (*, G) entry)

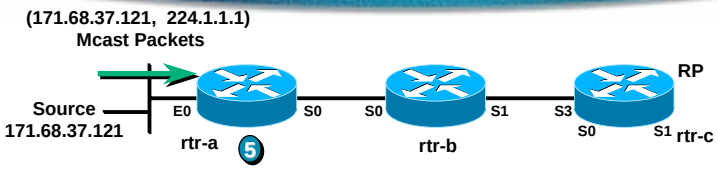
3. **"rtr-c" (RP) has no receivers on Shared Tree; discards packet.**

PIM SM Registering
Source Registers First



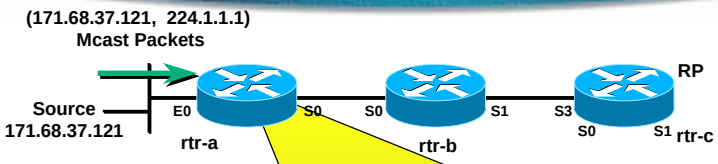
3. **"rtr-c" (RP) has no receivers on Shared Tree; discards packet.**
4. **RP sends "Register-Stop" to "rtr-a".**

PIM SM Registering
Source Registers First



3. "rtr-c" (RP) has no receivers on Shared Tree; discards packet.
4. RP sends "Register-Stop" to "rtr-a".
5. "rtr-a" stops encapsulating traffic in Register Messages; drops packets from Source.

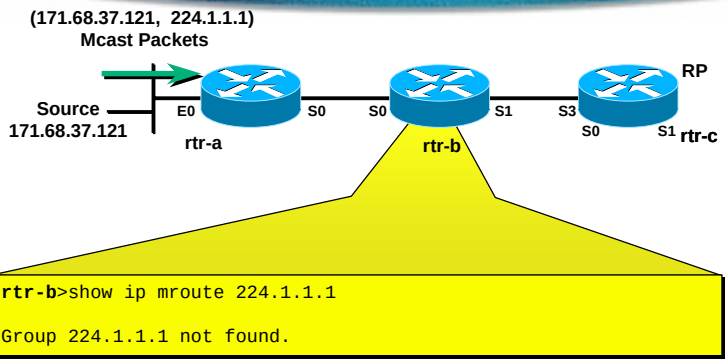
PIM SM Registering
Source Registers First




```
(*, 224.1.1.1), 00:01:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:01:28/00:01:32, flags: FPT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0
  Outgoing interface list: Null
```

State in "rtr-a" after Registering
(without receivers on Shared Tree)

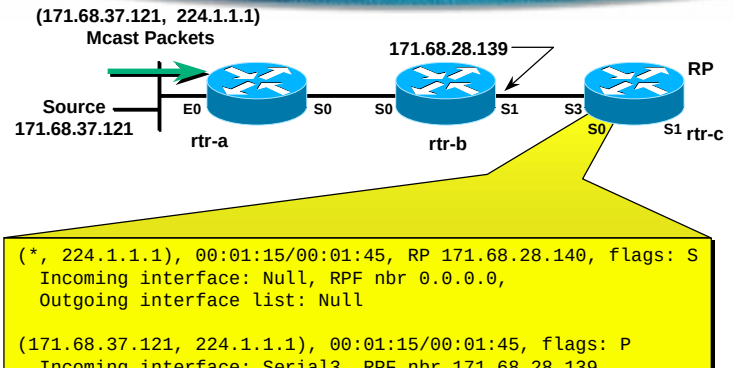
PIM SM Registering
Source Registers First




```
rtr-b>show ip mroute 224.1.1.1

Group 224.1.1.1 not found.
```

State in “rtr-b” after “rtr-a” Registers
(without receivers on Shared Tree)

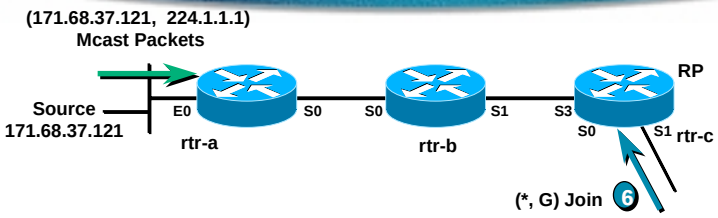
PIM SM Registering
Source Registers First




```
(*, 224.1.1.1), 00:01:15/00:01:45, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list: Null

(171.68.37.121, 224.1.1.1), 00:01:15/00:01:45, flags: P
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list: Null
```

State in “RP” after “rtr-a” Registers
(without receivers on Shared Tree)

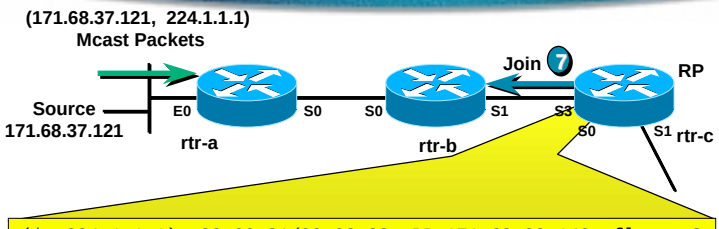
PIM SM Registering
Source Registers First



Receivers begin joining the Shared Tree

6 **RP (“rtr-c”) receives (*, G) Join from a receiver on Shared Tree.**

PIM SM Registering
Source Registers First



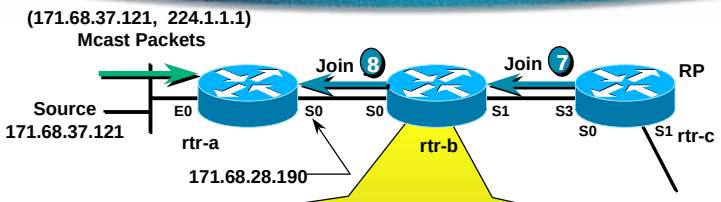

```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:00:14/00:02:46

(171.68.37.121/32, 224.1.1.1, 00:01:15/00:02:46, flags: T
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:00:14/00:02:46
```

“RP” processes (*,G) Join

(Adds Serial1 to Outgoing Interface Lists)

7 **RP sends (S,G) Joins for all known Sources in Group.**

PIM SM Registering
Source Registers First



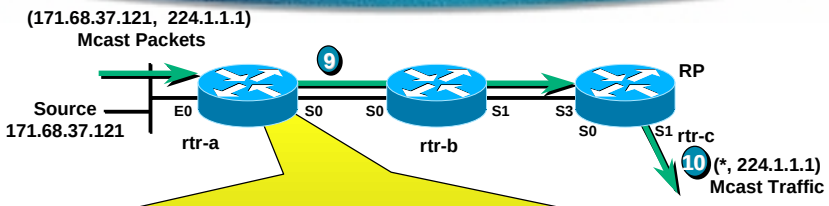

```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial1, RPF nbr 171.68.28.140,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags:
  Incoming interface: Serial0, RPF nbr 171.68.28.190
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:04:28/00:01:32
```

"rtr-b" processes Join, creates (S, G) state

(After automatically creating the (*, G) entry)

7. RP sends (S,G) Joins for all known Sources in Group.
8. "rtr-b" sends (S,G) Join toward Source to continue building SPT.

PIM SM Registering
Source Registers First

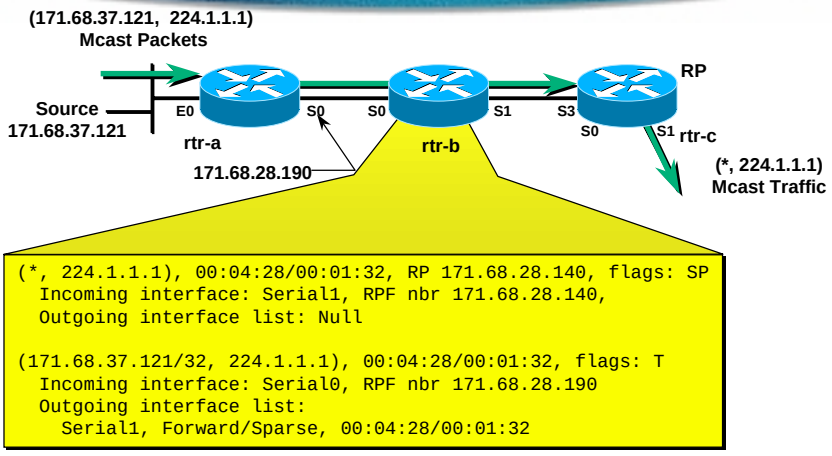



```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial0, RPF nbr 171.68.28.191,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: FT
  Incoming interface: Ethernet0, RPF nbr 0.0.0.0, Registering
  Outgoing interface list:
    Serial0, Forward/Sparse, 00:04:28/00:01:32
```

"rtr-a" processes the (S, G) Join; adds Serial0 to OIL

9. RP begins receiving (S,G) traffic down SPT.
10. RP forwards (S,G) traffic down Shared Tree to receivers.

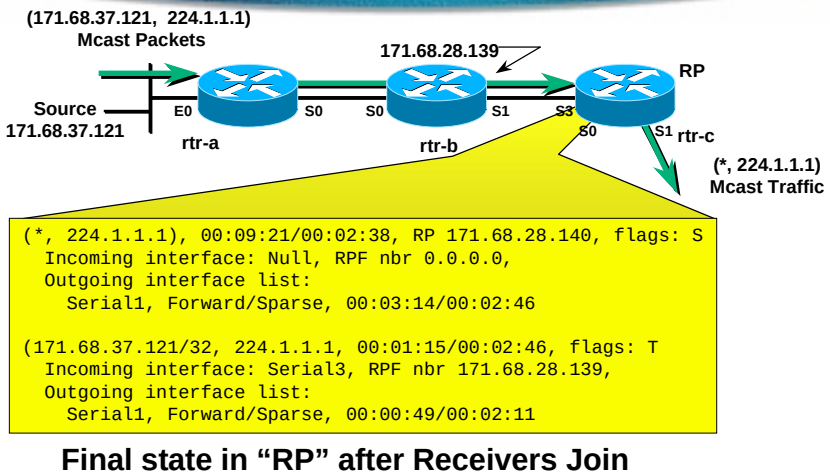
PIM SM Registering
Source Registers First




```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial1, RPF nbr 171.68.28.140,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: T
  Incoming interface: Serial0, RPF nbr 171.68.28.190
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:04:28/00:01:32
```

Final state in “rtr-b” after Receivers Join

PIM SM Registering
Source Registers First




```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:03:14/00:02:46

(171.68.37.121/32, 224.1.1.1, 00:01:15/00:02:46, flags: T
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:00:49/00:02:11
```

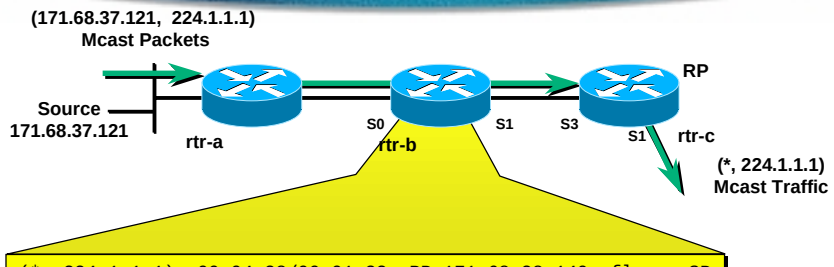
Final state in “RP” after Receivers Join

PIM SM Register Examples

- Receivers Join Group First
- Source Registers First
- Receivers along the SPT

PIM SM Registering

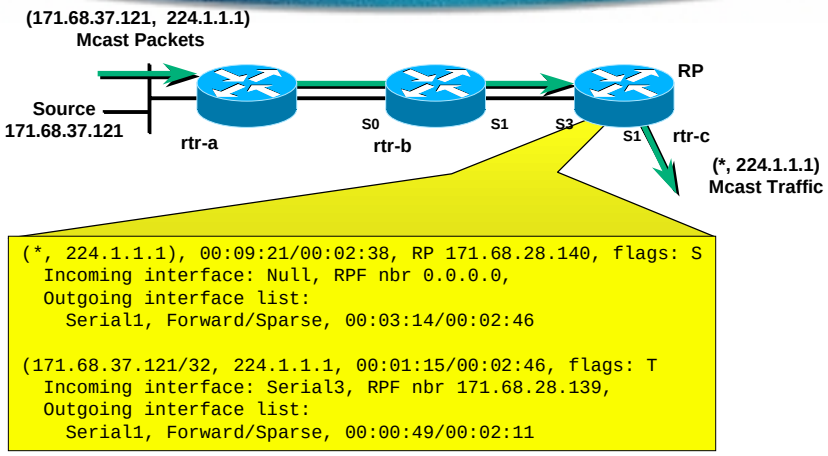
Receivers along the SPT




```
(*, 224.1.1.1), 00:04:28/00:01:32, RP 171.68.28.140, flags: SP
  Incoming interface: Serial1, RPF nbr 171.68.28.140,
  Outgoing interface list: Null

(171.68.37.121/32, 224.1.1.1), 00:04:28/00:01:32, flags: T
  Incoming interface: Serial0, RPF nbr 171.68.28.190
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:04:28/00:01:32
```

Current state in “rtr-b”

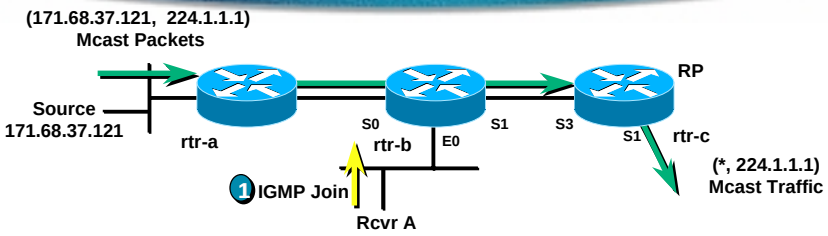
PIM SM Registering
Receivers along the SPT




```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:03:14/00:02:46

(171.68.37.121/32, 224.1.1.1, 00:01:15/00:02:46, flags: T
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:00:49/00:02:11
```

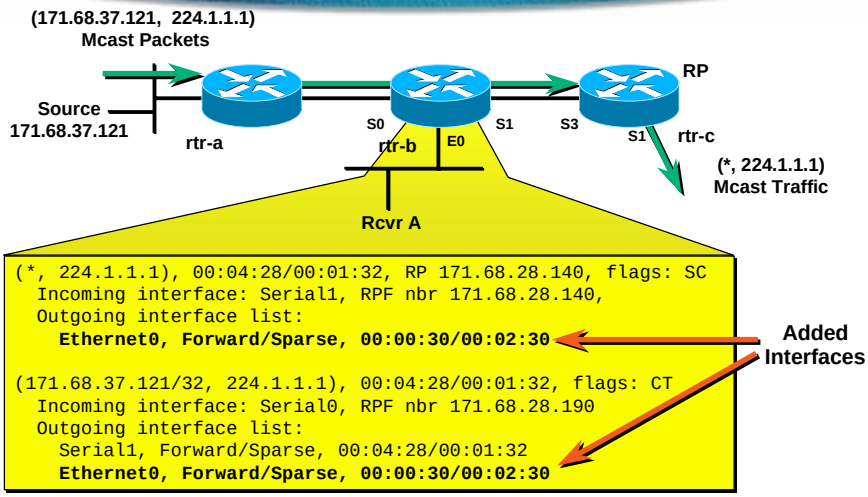
Current state in the RP

PIM SM Registering
Receivers along the SPT



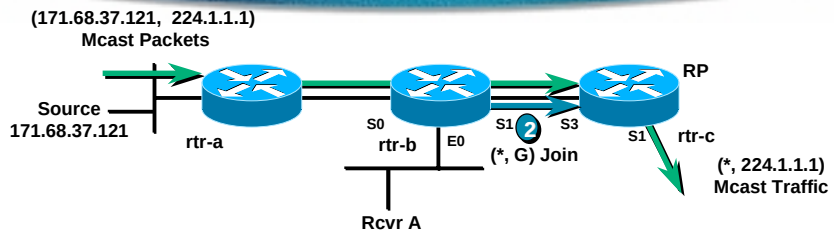
1. **"Rcvr A" wishes to receive group G traffic. Sends IGMP Join for G.**

PIM SM Registering
Receivers along the SPT



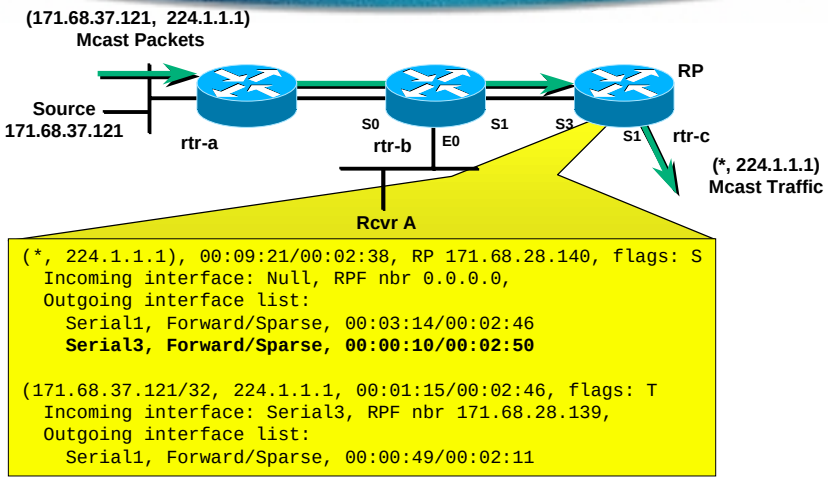
State in "rtr-b" after "Rcvr A" joins group

PIM SM Registering
Receivers along the SPT



2 "rtr-b" triggers a (*,G) Join to join the Shared Tree

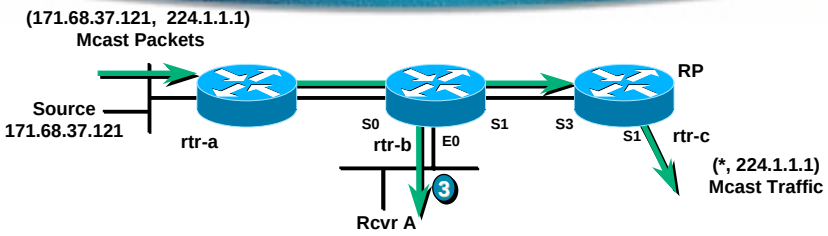
PIM SM Registering
Receivers along the SPT




```
(*, 224.1.1.1), 00:09:21/00:02:38, RP 171.68.28.140, flags: S
  Incoming interface: Null, RPF nbr 0.0.0.0,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:03:14/00:02:46
    Serial3, Forward/Sparse, 00:00:10/00:02:50

(171.68.37.121/32, 224.1.1.1, 00:01:15/00:02:46, flags: T
  Incoming interface: Serial3, RPF nbr 171.68.28.139,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:00:49/00:02:11
```

State in "RP" after "rtr-b" joins Shared Tree

PIM SM Registering
Receivers along the SPT



3. **Group G traffic begins to flow to "Rcvr A".**

   (Note: 171.68.37.121 traffic doesn't flow to RP then back down to rtr-b)

PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM SM SPT-Switchover

- SPT Thresholds may be set for any Group
  - Access Lists may be used to specify which Groups
  - Default Threshold = 0kbps (I.e. immediately join SPT)
  - Threshold = “infinity” means “never join SPT”.
- Threshold triggers Join of Source Tree
  - Sends an (S,G) Join up SPT for next “S” in “G” packet received.
- Pros
  - Reduces Network Latency
- Cons
  - More (S,G) state must be stored in the routers.

PIM SM SPT-Switchover

SPT-Switchover Mechanism

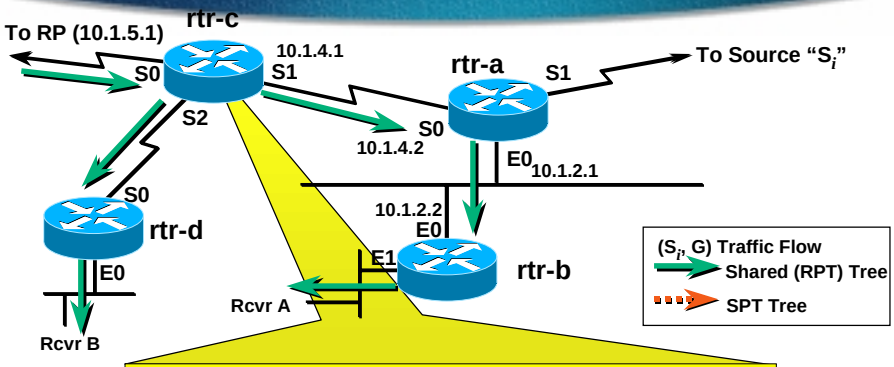
Once each second

- Compute new (*, G) traffic rate
- If threshold exceeded, set “J” flag in (*, G)

For each (S_i, G) packet received:

- If “J” flag set in (*, G)
  - Join SPT for (S_i, G)
  - Mark (S_i, G) entry with “J” flag
  - Clear “J” flag in (*,G)

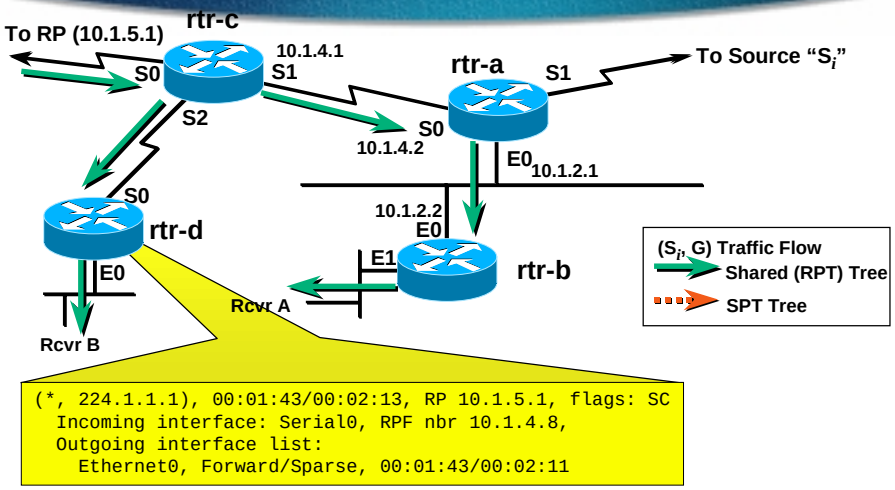
PIM SM SPT-Switchover




```
(*, 224.1.1.1), 00:01:43/00:02:13, RP 10.1.5.1, flags: S
  Incoming interface: Serial0, RPF nbr 10.1.5.1,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:01:43/00:02:11
    Serial2, Forward/Sparse, 00:00:32/00:02:28
```

State in “rtr-c” before switch

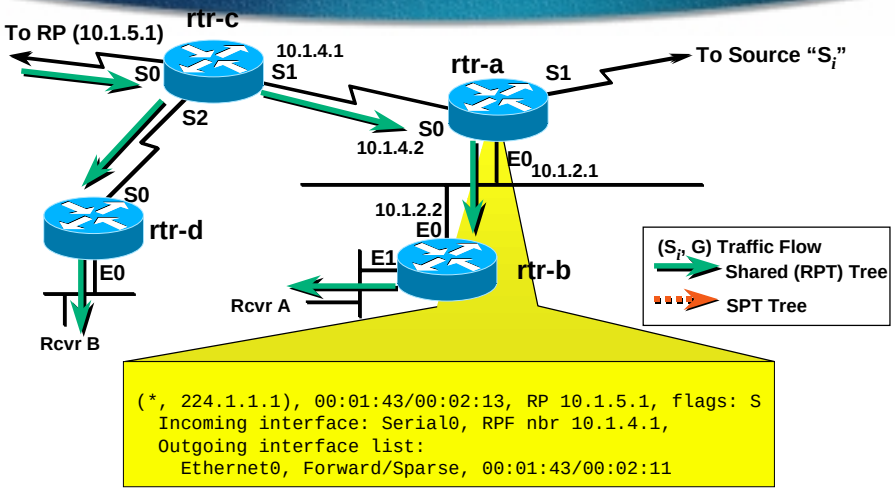
PIM SM SPT-Switchover




```
(*, 224.1.1.1), 00:01:43/00:02:13, RP 10.1.5.1, flags: SC
  Incoming interface: Serial0, RPF nbr 10.1.4.8,
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:01:43/00:02:11
```

State in "rtr-d" before switch

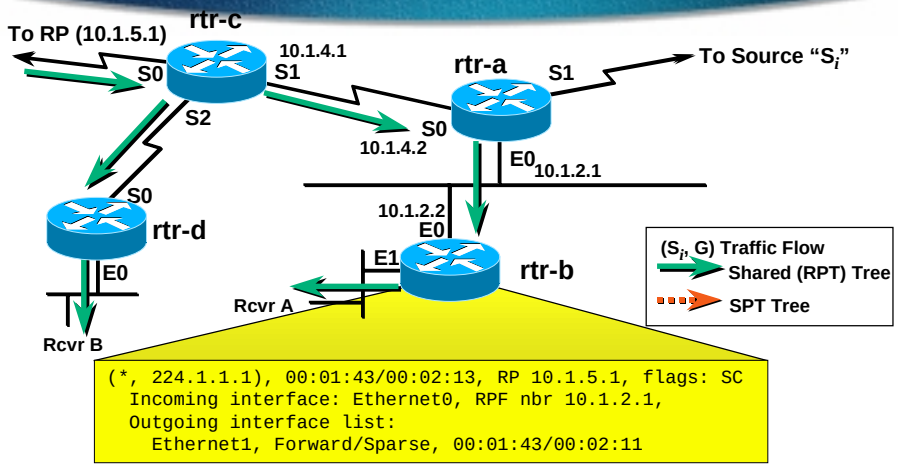
PIM SM SPT-Switchover




```
(*, 224.1.1.1), 00:01:43/00:02:13, RP 10.1.5.1, flags: S
  Incoming interface: Serial0, RPF nbr 10.1.4.1,
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:01:43/00:02:11
```

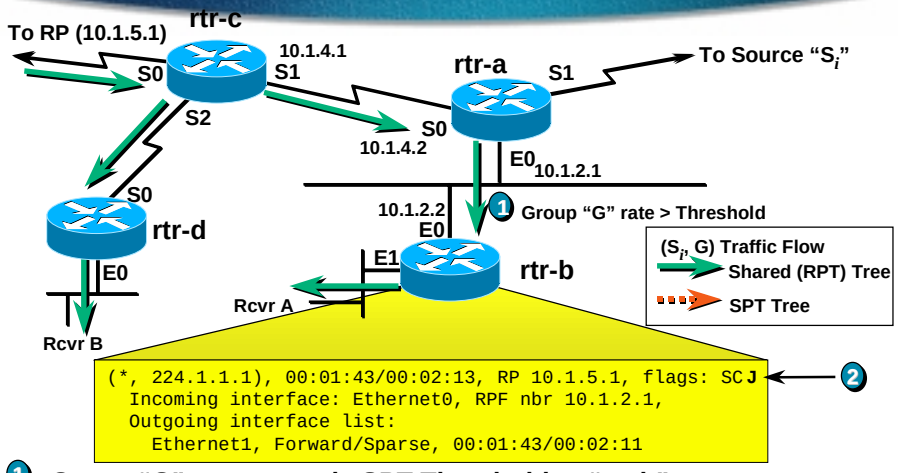
State in "rtr-a" before switch

PIM SM SPT-Switchover



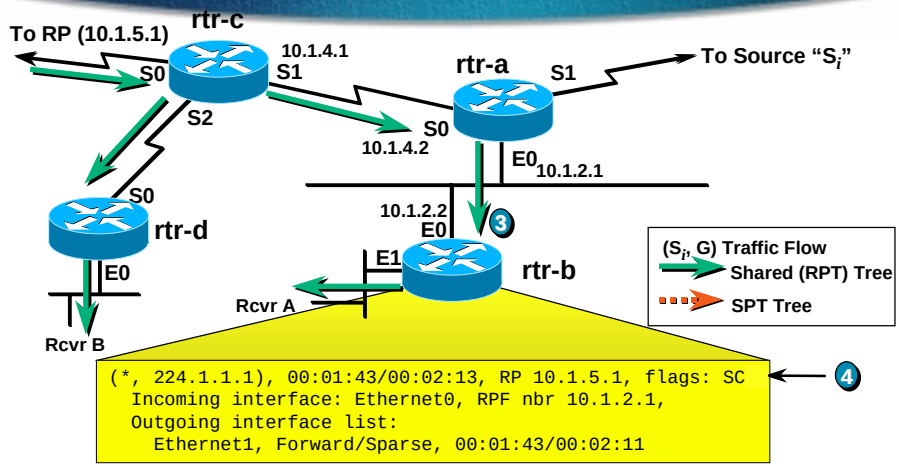
State in "rtr-b" before switch

PIM SM SPT-Switchover



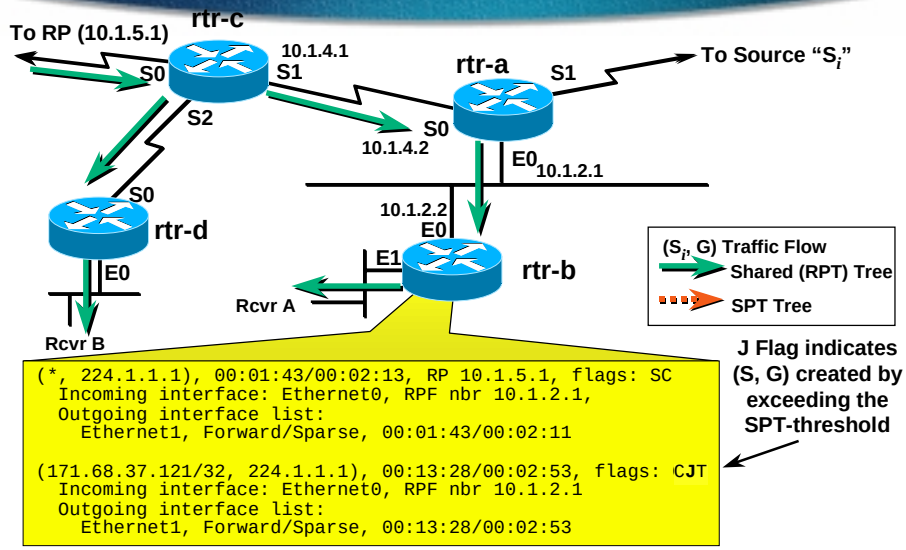
1. Group "G" rate exceeds SPT Threshold at "rtr-b";
2. Set J Flag in (*, G) and wait for next (S_i,G) packet.

PIM SM SPT-Switchover



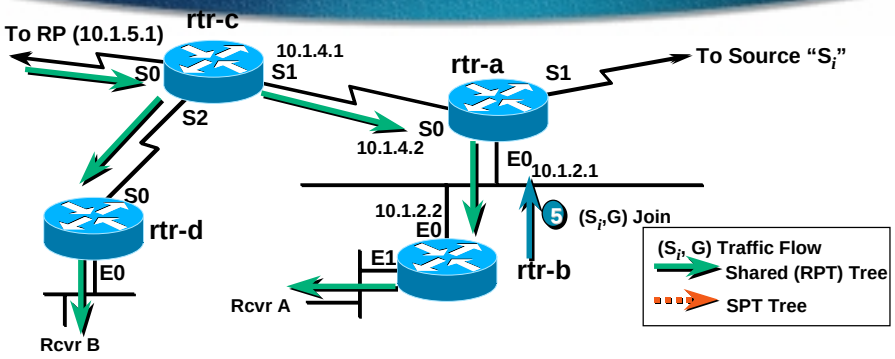
3. (S_i,G) packet arrives down Shared tree.
4. Clear J Flag in the (*,G) & create (S_i,G) state.

PIM SM SPT-Switchover



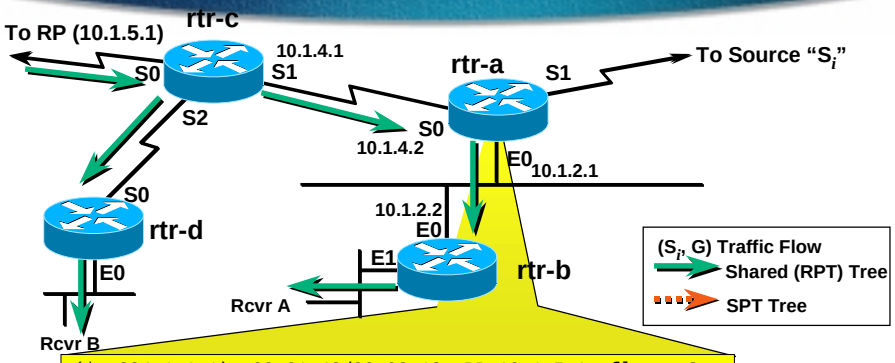
New State in "rtr-b"

PIM SM SPT-Switchover



5 Send (S_i,G) Join towards S_i .

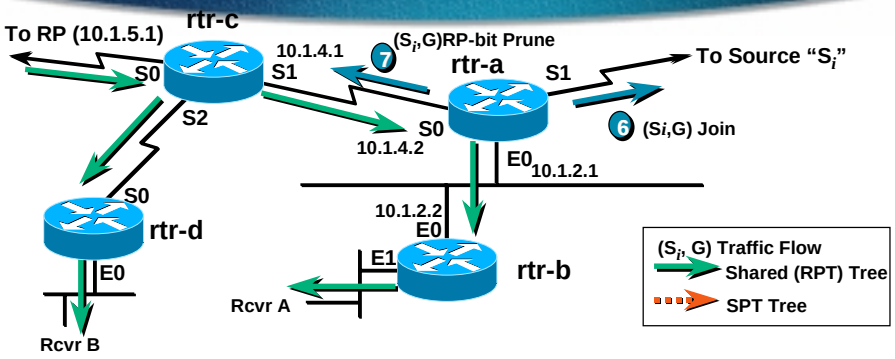
PIM SM SPT-Switchover




```
(*, 224.1.1.1), 00:01:43/00:02:13, RP 10.1.5.1, flags: S
  Incoming interface: Serial0, RPF nbr 10.1.4.1,
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:01:43/00:02:11

(171.68.37.121/32, 224.1.1.1), 00:13:28/00:02:53, flags: T
  Incoming interface: Serial1, RPF nbr 10.1.9.2
  Outgoing interface list:
    Ethernet0, Forward/Sparse, 00:13:25/00:02:30
```

New state in "rtr-a"

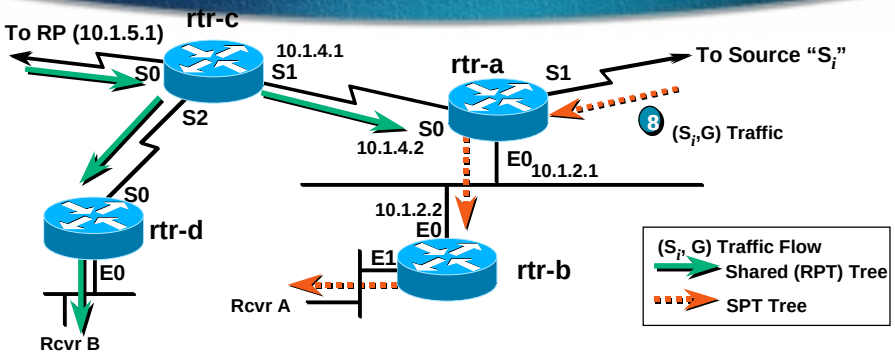
PIM SM SPT-Switchover



6 "rtr-a" forwards (S_i,G) Join toward S_i.

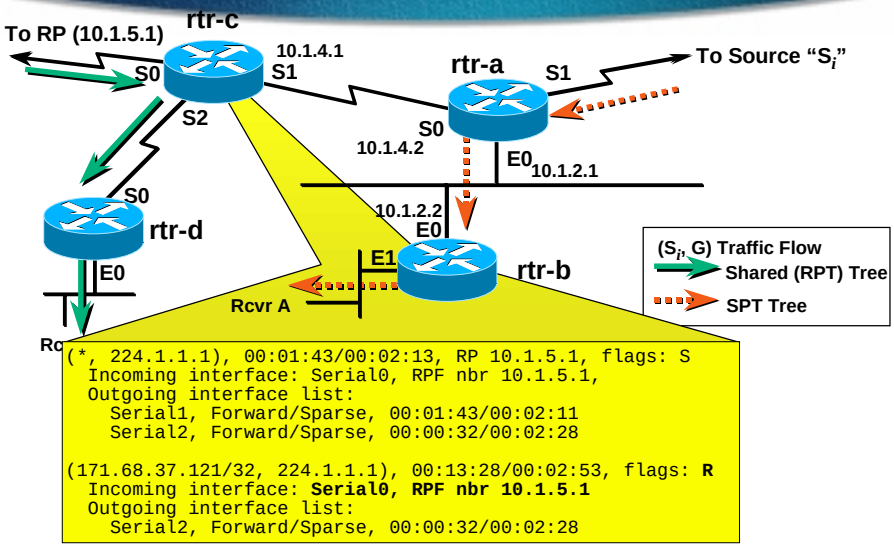
7 SPT & RPT diverge, triggering (S_i,G)RP-bit Prunes toward RP.

PIM SM SPT-Switchover



8 (S_i, G) traffic begins flowing down SPT tree.

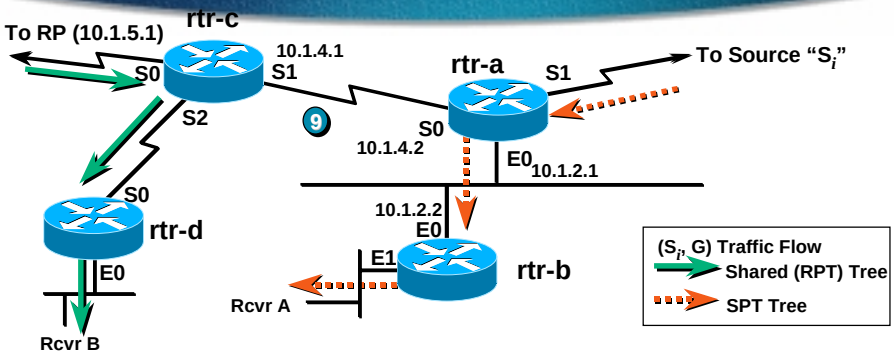
PIM SM SPT-Switchover




```
(*, 224.1.1.1), 00:01:43/00:02:13, RP 10.1.5.1, flags: S
  Incoming interface: Serial0, RPF nbr 10.1.5.1,
  Outgoing interface list:
    Serial1, Forward/Sparse, 00:01:43/00:02:11
    Serial2, Forward/Sparse, 00:00:32/00:02:28

(171.68.37.121/32, 224.1.1.1), 00:13:28/00:02:53, flags: R
  Incoming interface: Serial0, RPF nbr 10.1.5.1
  Outgoing interface list:
    Serial2, Forward/Sparse, 00:00:32/00:02:28
```

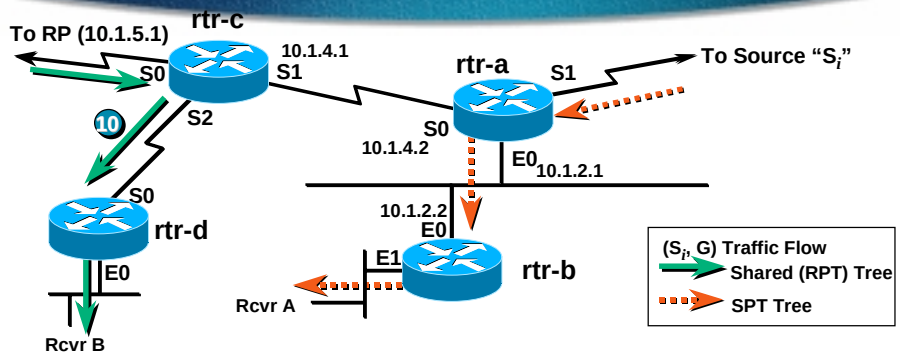
State in "rtr-c" after receiving the (S_i, G) RP-bit Prune

PIM SM SPT-Switchover



9 **Unnecessary (S_i, G) traffic is pruned from the Shared tree.**

PIM SM SPT-Switchover



9 Unnecessary (S_i, G) traffic is pruned from the Shared tree.

10 (S_i, G) traffic still flows via other branches of the Shared tree.

PIM SM SPT-Switchover

Shared Tree Switchback Mechanism

- Once each minute
  - If "J" flag set in (Si , G) entry
    - Compute new (Si , G) traffic rate
    - If rate < SPT-threshold
      - Rejoin (*, G) Tree for (Si , G) traffic
      - Send (Si , G) prune up SPT toward Si
      - Delete (Si , G) entry

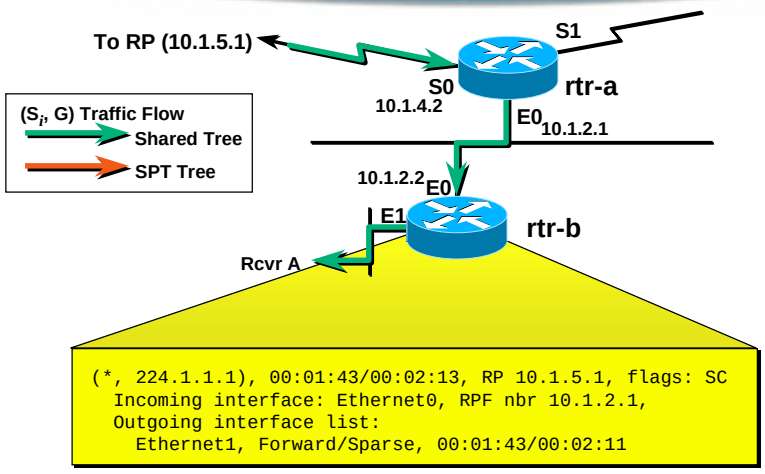
PIM-SM Protocol Mechanics

- PIM SM State
- PIM SM Forwarding
- PIM SM Joining
- PIM SM Registering
- PIM SM SPT-Switchover
- PIM SM Pruning

PIM SM Pruning

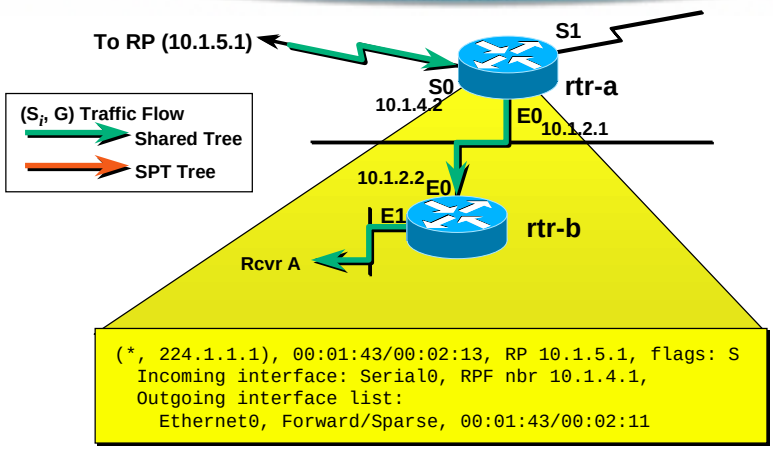
- IGMP group times out / last host sends Leave
- Interface removed from all (*,G) & (S,G) entries
  - IF all interfaces in "*oilist*" for (*,G) are pruned; THEN send Prune up shared tree toward RP
  - Any (S, G) state allowed to time-out
- Each router along path "prunes" interface
  - IF all interfaces in "*oilist*" for (*,G) are pruned; THEN send Prune up shared tree toward RP
  - Any (S, G) state allowed to time-out

PIM SM Pruning
Shared Tree Case



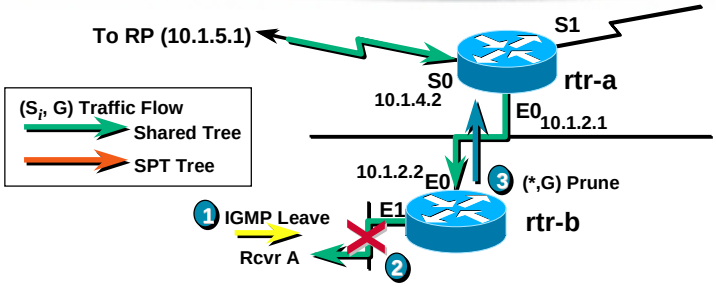
State in “rtr-b” before Pruning

PIM SM Pruning
Shared Tree Case



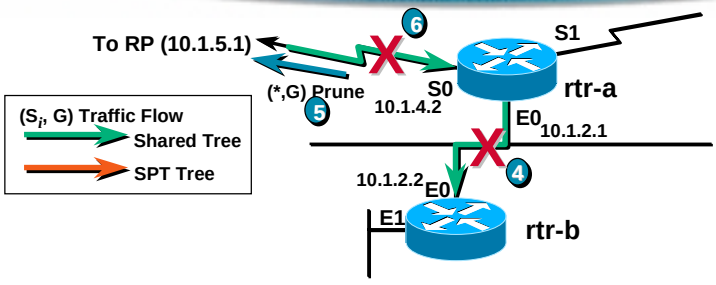
State in “rtr-a” before Pruning

PIM SM Pruning
Shared Tree Case



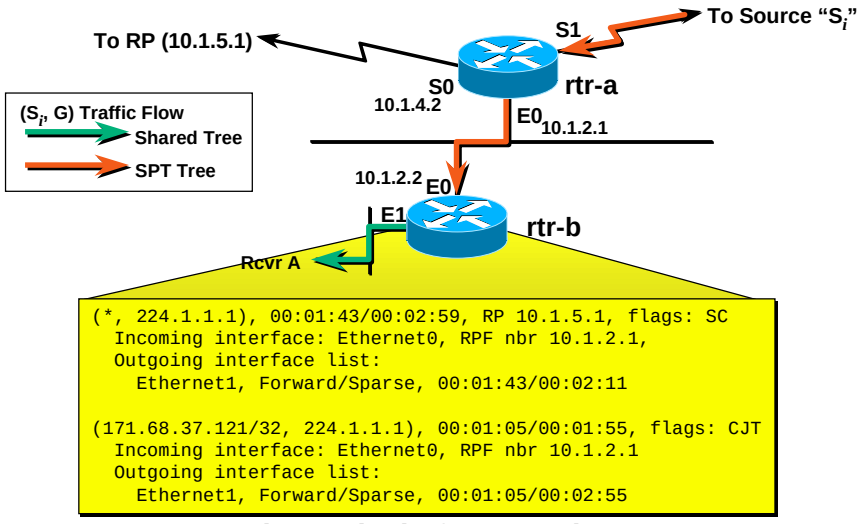
1. "rtr-b" is a Leaf router. Last host "Rcvr A", leaves group G.
2. "rtr-b" removes E1 from (*,G) and any (S$_i$,G) *"oilists"*.
3. "rtr-b" (*,G) *"oilist"* now empty; sends (*,G) Prune toward RP.

PIM SM Pruning
Shared Tree Case



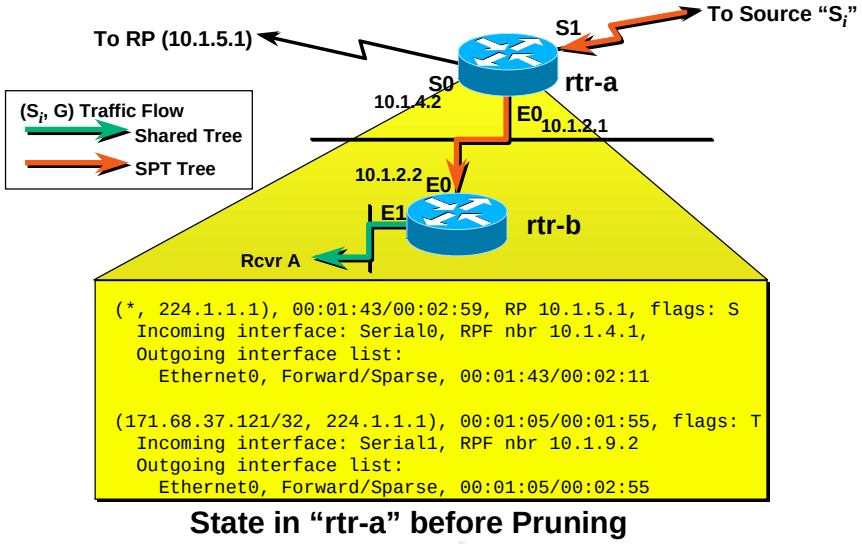
4. "rtr-a" receives Prune; removes E0 from (*,G) *"oilist"*.
   (After the 3 second Multi-access Network Prune delay.)
5. "rtr-a" (*,G) *"oilist"* now empty; send (*,G) Prune toward RP.
6. Pruning continues back toward RP.

PIM SM Pruning
Source (SPT) Case



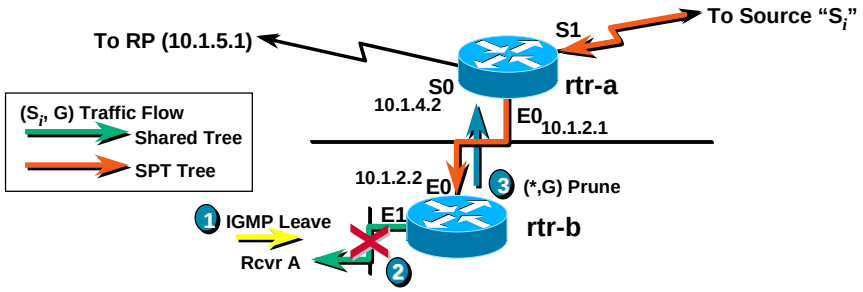
State in "rtr-b" before Pruning

PIM SM Pruning
Source (SPT) Case



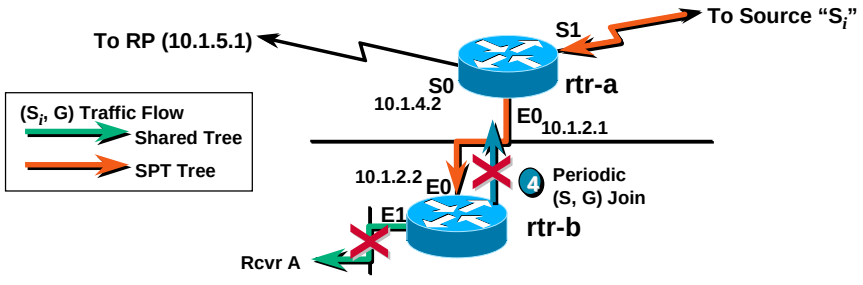
State in "rtr-a" before Pruning

PIM SM Pruning
Source (SPT) Case



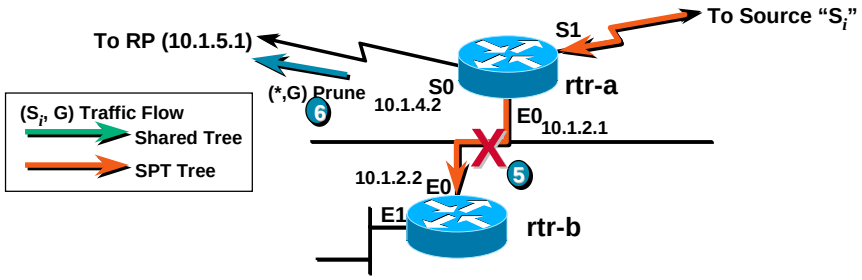
1. "rtr-b" is a Leaf router. Last host "Rcvr A", leaves group G.
2. "rtr-b" removes E1 from (*,G) and any (S_i,G) *"oilists"*.
3. "rtr-b" (*,G) *"oilist"* now empty; sends (*,G) Prune toward RP.

PIM SM Pruning
Source (SPT) Case



1. "rtr-b" is a Leaf router. Last host "Rcvr A", leaves group G.
2. "rtr-b" removes E1 from (*,G) and any (S_i,G) *"oilists"*.
3. "rtr-b" (*,G) *"oilist"* now empty; sends (*,G) Prune toward RP.
4. "rtr-b" stops sending periodic (S, G) joins.

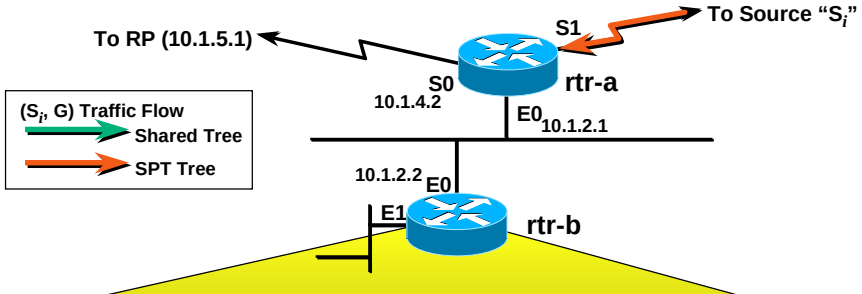
PIM SM Pruning
Source (SPT) Case



5 "rtr-a" receives Prune; removes E0 from (*,G) *"oilist"*.
(After the 3 second Multiaccess Network Prune delay.)

6 "rtr-a" (*,G) *"oilist"* now empty; sends (*,G) Prune toward RP.

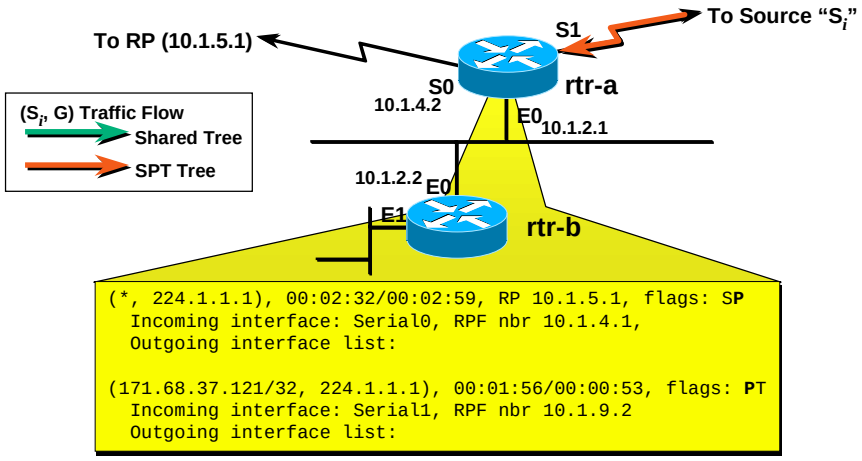
PIM SM Pruning
Source (SPT) Case




```
(*, 224.1.1.1), 00:02:32/00:02:59, RP 10.1.5.1, flags: SP
  Incoming interface: Ethernet0, RPF nbr 10.1.2.1,
  Outgoing interface list:

(171.68.37.121/32, 224.1.1.1), 00:01:56/00:00:53, flags: PT
  Incoming interface: Ethernet0, RPF nbr 10.1.2.1
  Outgoing interface list:
```

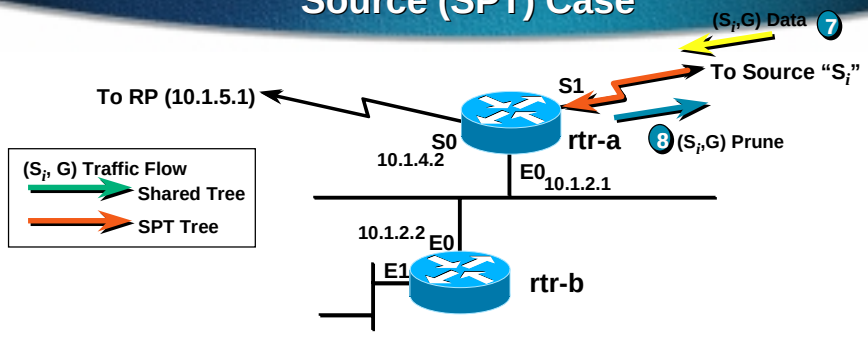
State in "rtr-b" after Pruning

PIM SM Pruning
Source (SPT) Case



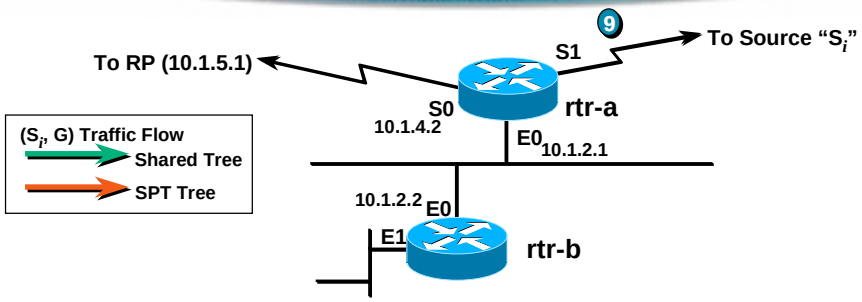
State in "rtr-a" after Pruning

PIM SM Pruning
Source (SPT) Case



7. Another (S_i,G) data packet arrives via Serial1.
8. 'rtr-a' responds by sending an (S_i,G) Prune toward source.

PIM SM Pruning
Source (SPT) Case

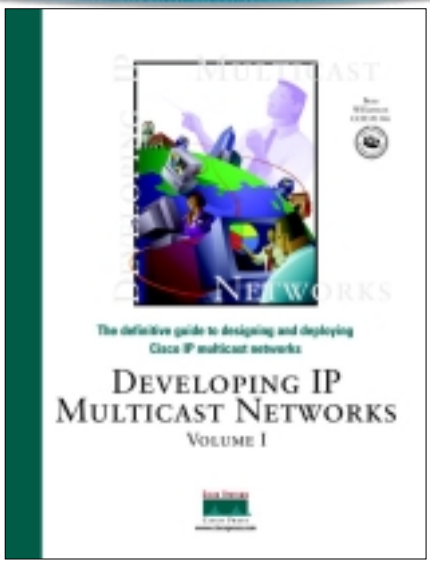


7. Another (S_i,G) data packet arrives via Serial1.
8. 'rtr-a' responds by sending an (S_i,G) Prune toward source.
9. (S_i,G) traffic ceases flowing down SPT.

Documentation and Contact Info

- **EFT/Beta Site Web Page:**
  - ftp://ftpeng.cisco.com/ipmulticast.html
- **EFT/Beta Mailing List:**
  - multicast-support@cisco.com
- **Customer Support Mailing List:**
  - cs-ipmulticast@cisco.com

If All Else Fails—RTFB[1]



[1] Read this fine book

www.cisco.com

Please Complete Your Evaluation Form

Session 306

www.cisco.com


Cisco Systems
Empowering the Internet Generation℠
www.cisco.com