

# Multicast - 224->239 / 239.x.x.x Admin-local / 232.x.x.x PIM SSM / 233.x.x.x GLOP (AS)

- .1 - all hosts
  - .2 - all routers
  - .5 - all SFP routers
  - .6 - all SFP DRs
  - .9 - RIPv2
  - .13 - PIM
  - .18 - VRRP
  - .22 - IGMP
- 1110.xxxx.....xxx  
 28b=01.00.5e+"0"+23b MAC  
 ^ bcast bit

**RPF-check** on IGP IF towards the source  
**DENSE** - (S,G) - flood everywhere, prune where not needed (every 3 min)  
**SPARSE** - (\*,G) - use RP, flood only if needed. **RPT** = Shared Tree / \* = 0.0.0.0 in packets.

**Designated Router:** low prio or highest IP Addr; sends Join/Prune; if IGMPv1->**Querier** possibly used only in SM to send the Register-Messages

**Querier Router (IGMPv2 only):** low IP Address (sends IGMP queries every **125 sec** to .1; Host wait 0-10sec, if no other host reports, send Report; if other hosts report, shut up). Otherwise after 260 sec remove IF. (2<robustness>\*125)+(1\*10)

**Forwarder Router:** elect w/ PIM-Assert, fwd multicast on segm. (AD/cost/>IP)  
 Used in PIM BiDir to have a no-loop RPT

**IGMPv2** compat. v1 + Leave. When leave rcv, query with ResponseTime 1sec.  
**IGMPv3** adds possibility of specifying the wanted source

<p><b>IGMP v1</b></p> <ul style="list-style-type: none"> <li>- Version (1)</li> <li>- Type (1- <b>Host query</b>, 2- <b>Host report</b>(join))</li> <li>- Unused (1b)</li> <li>- Checksum</li> <li>- Group Address ("0" for query, Addr for report)</li> </ul>	<p><b>IGMP v2</b></p> <ul style="list-style-type: none"> <li>- no version field, part of Type</li> <li>- Type (11- <b>Query</b>, 12 <b>Report v1</b>, 16 <b>Report v2</b>, 17 <b>Leave</b>)</li> <li>- MaxResponseTime (default 10 sec)</li> <li>- Checksum</li> <li>- Group Address</li> </ul>
--	---

Leave ^ sent to **224.0.0.2**

**PIMv2 packet** (v1 based on IGMP)

- Version
- Type
- Reserved (all 0)
- Checksum

**Unicast Address:**  
 Family(1), Encoding(0), Addr

**Group Address:**  
 Family(1), Encoding(0), Mask, Addr

**Source Address:**  
 Family(1), Encoding(0), Bits:

- **S-bit** - sparse mode
- **W-bit** - 0-(S,G), 1-(\*,G). "1" to RPs
- **RPT-bit** - 0-> to source, 1-> to RP

- 0- Hello to 224.0.0.13
  - 1- Register (unicast to RP)
  - 2- Register-Stop (ucast to source R)
  - 3- Join/Prune (Join only in **SPARSE**)
  - 4- Bootstrap
  - 5- Assert (choose forwarding router)
  - 6- Graft (only **DENSE** ro reconnect)
  - 7- Graft-ACK (only **DENSE**)
  - 8- Candidate RP (unicast to BSR)
- Register Message:**  
**B-bit** (Border) 0:source directconnec.  
**Null-bit** normally 0 (1 for Probe)  
 Multicast Data unicast encapsulated
- Register Stop Message**  
 - Group Addr & Source Addr

**Sparse Mode:**

1. hosts join to RPT. Joins \*,G go towards RP. If Source known, RP starts forwarding traffic, all Routers install (S,G). **RPF checks RP Addr!!!**
2. when source available, its PIM router encaps. traffic & forwards unicast to RP in **Register** mess.  
 - if subscribers, decaps. + forward traffic on RPT and JOIN directly to source (thresh=0), to not use Reg packets. (**Need Tunnel-PIC**)  
 - if not, send RegisterStop back. Initial PIM router waits 60sec and try again, but with no traffic and NULL bit set continuously
3. when dest. gets Mcast last-hop router sends Join (S,G) and SPT tree built. When traffic comes over this, prune from RPT.

When RP receives prune, prunes link from shared tree. If no subscribers send RegStop. As long as source sends, RegisterNull/Stop.

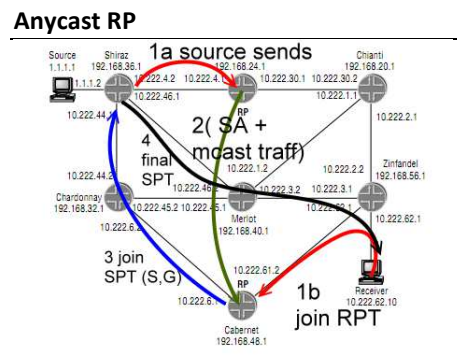
**Last-hop Routers regul. send Join/Prune** to RP to keep RPT active just-in-case even if no traffic comes through there (yet)

**Join/Prune Message:**

- Upstream Neighbor Addr
- Number of Groups
- Hold Time for PIM State (210s)

<ul style="list-style-type: none"> <li>- Multicast Group Addr</li> <li>- Number of Join Sources</li> <li>- Number of Prune Sources</li> <li>- List of Join / Prune Addresses</li> </ul>	For each <u>group</u>
---	-----------------------

Join to RP (\*,G) "source" = RP\_IPAddr



On JUNOS: **BSR > AutoRP > static**  
 (if static not "override")

**AUTO-RP**  
 224.0.1.39 (Announce) (sent by RPs)  
 224.0.1.40 (Discovery) (sent by MAs)

1. **Cand RPs** send **RP-Announce** to .39
2. **Map. Agent** chooses RP (high IP)
3. **Map. Agent** sends **RP-Discovery**

**MSDP** TCP 639. Highest IP passive, TCP start by lowest IP. When new source, -> "Source Active". Mesh-groups defined for fewer announc.

**BiDIR PIM:** no SPTs (S,G), only RPs and RPTs, all traffic goes through RP.

**BSR**

1. **Candidate BSRs** sends Bootstrap -> .13 (has Hash Mask/BSR Prio/BSR Addr/ GrpAddr/ RP Addr, Hold, Prio)
2. BSR chosen
3. **Candidate RPs** send RP-Advertis. (ucast). Contain Prio/Hold/RP Addr/Group. Can be more groups.
4. BSR collects and sends all in RP-Set

**RP choice:** 1. Most specific range of groups / 2. Highest Prio / 3. Hash Algorithm / 4. Highest IP Addr.