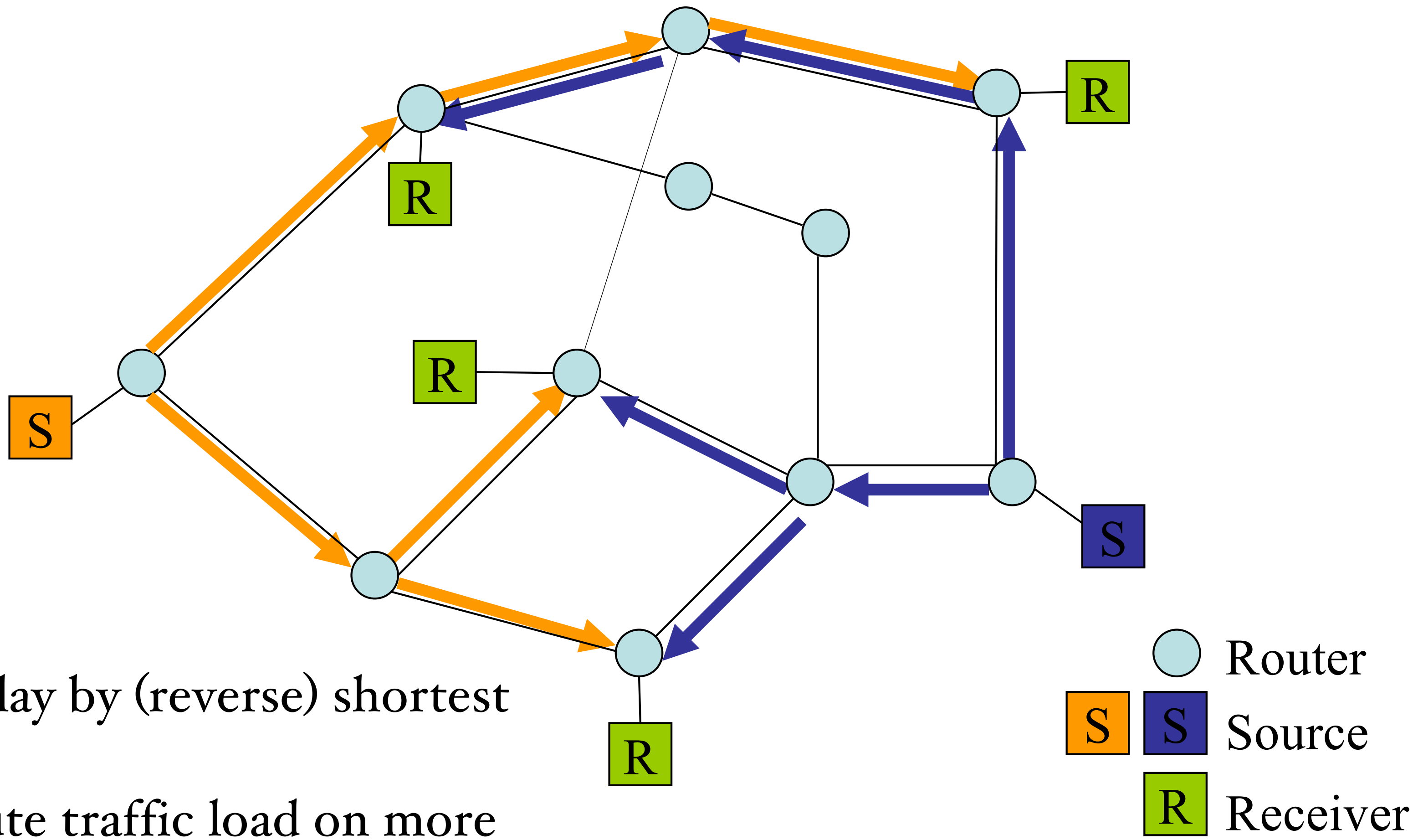
# CSC 525: Principles of Computer Networks

# Multicast Routing

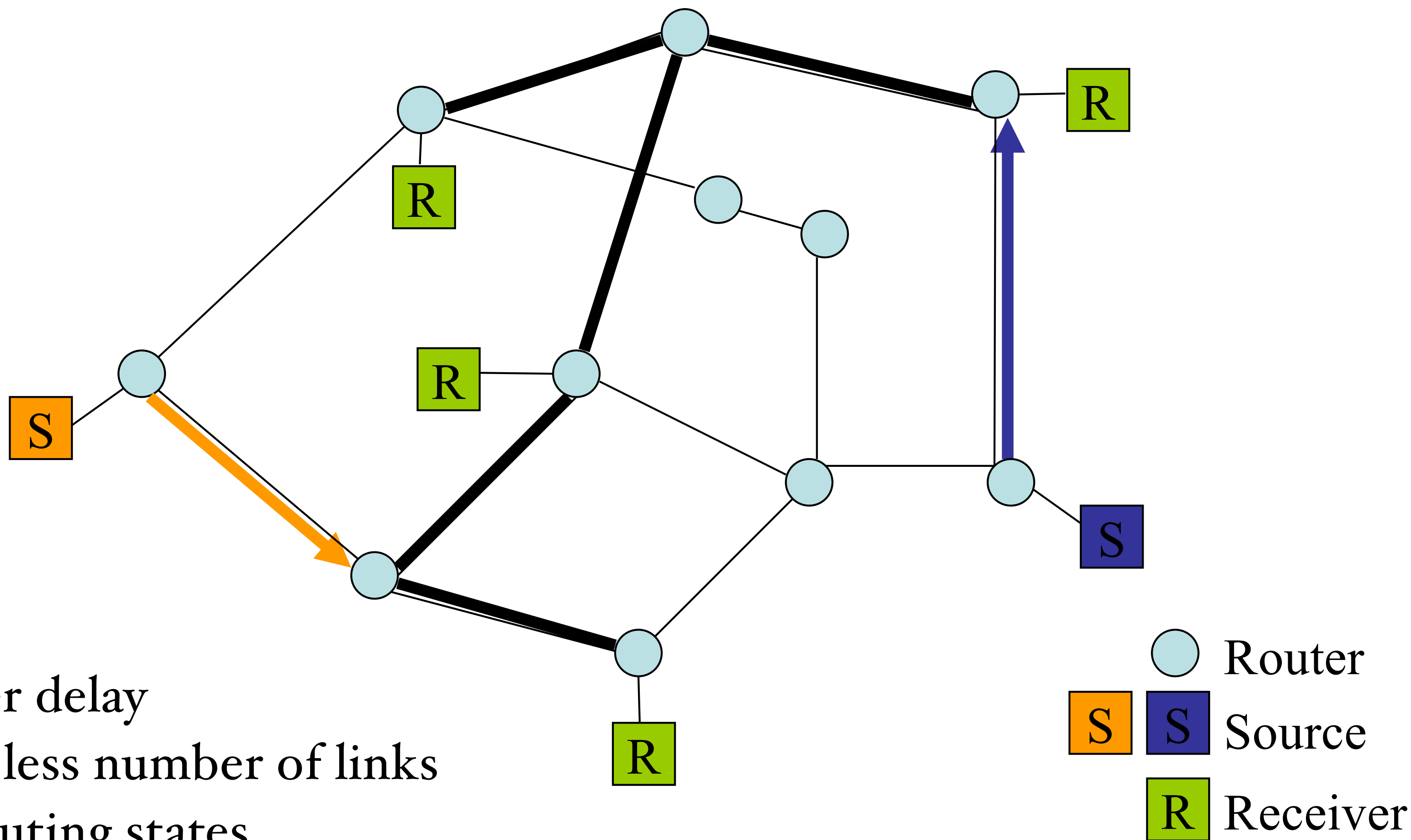
- Given a group address, build distribution trees to reach all members.
  - i.e., how senders and receivers "meet"
  - Senders broadcast data packets (DVMRP, PIM-DM)
  - Receivers broadcast group membership (MOSPF)
  - Specify a "meeting place" (CBT, PIM-SM)
- Deliver packets to group members
  - One tree per data source (DVMRP, MOSPF, PIM-DM, PIM-SM)
  - One tree per group shared by all data sources (CBT, PIM-SM)

# Source Trees




- Low delay by (reverse) shortest paths
- distribute traffic load on more links
- (S,G) routing states

# Shared Tree

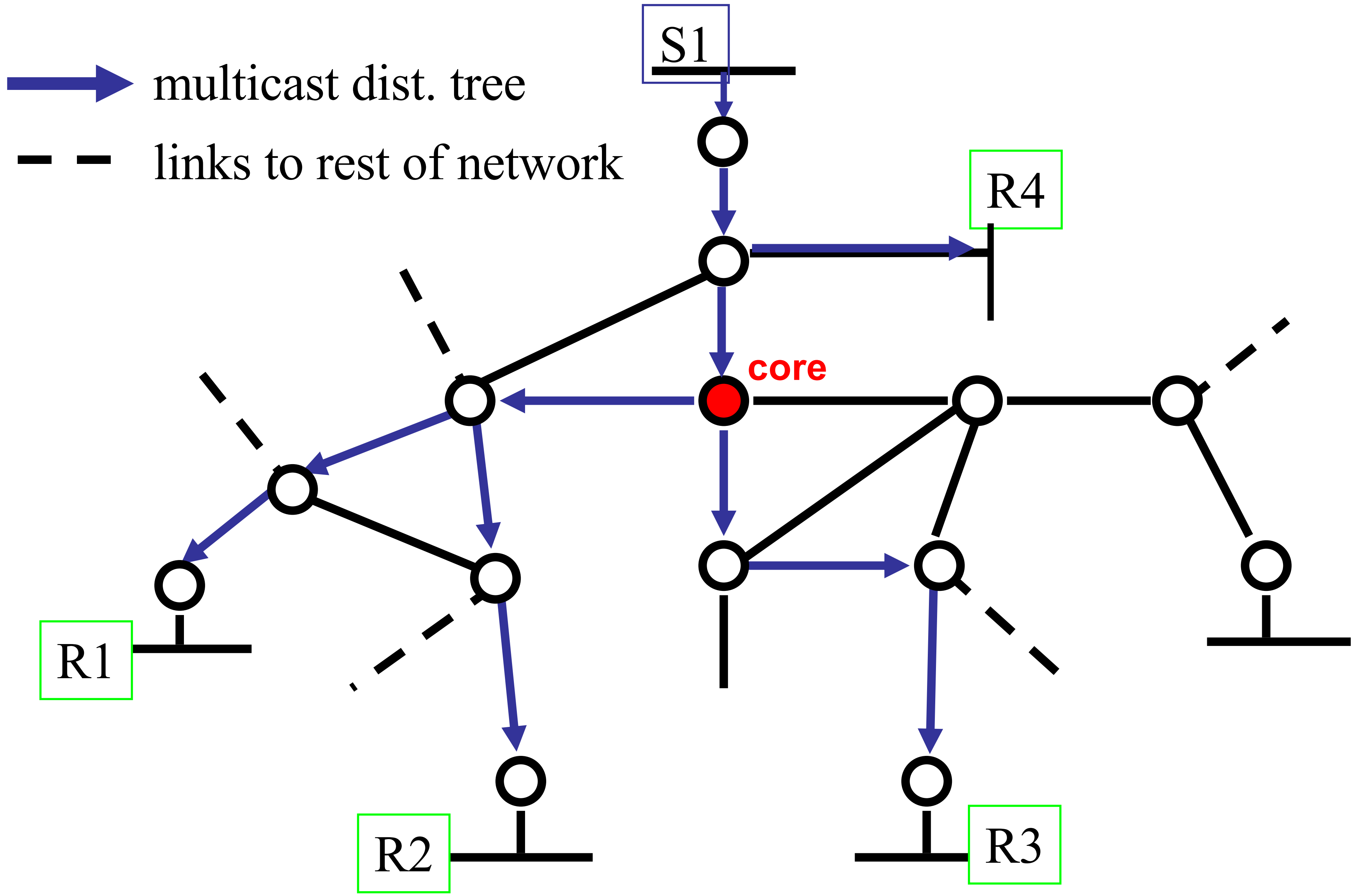



- Longer delay
- Using less number of links
- (G) routing states

# Outline

- Core-based Tree (CBT)
- Protocol-Independent Multicast (PIM)

# Core Based Tree (CBT)

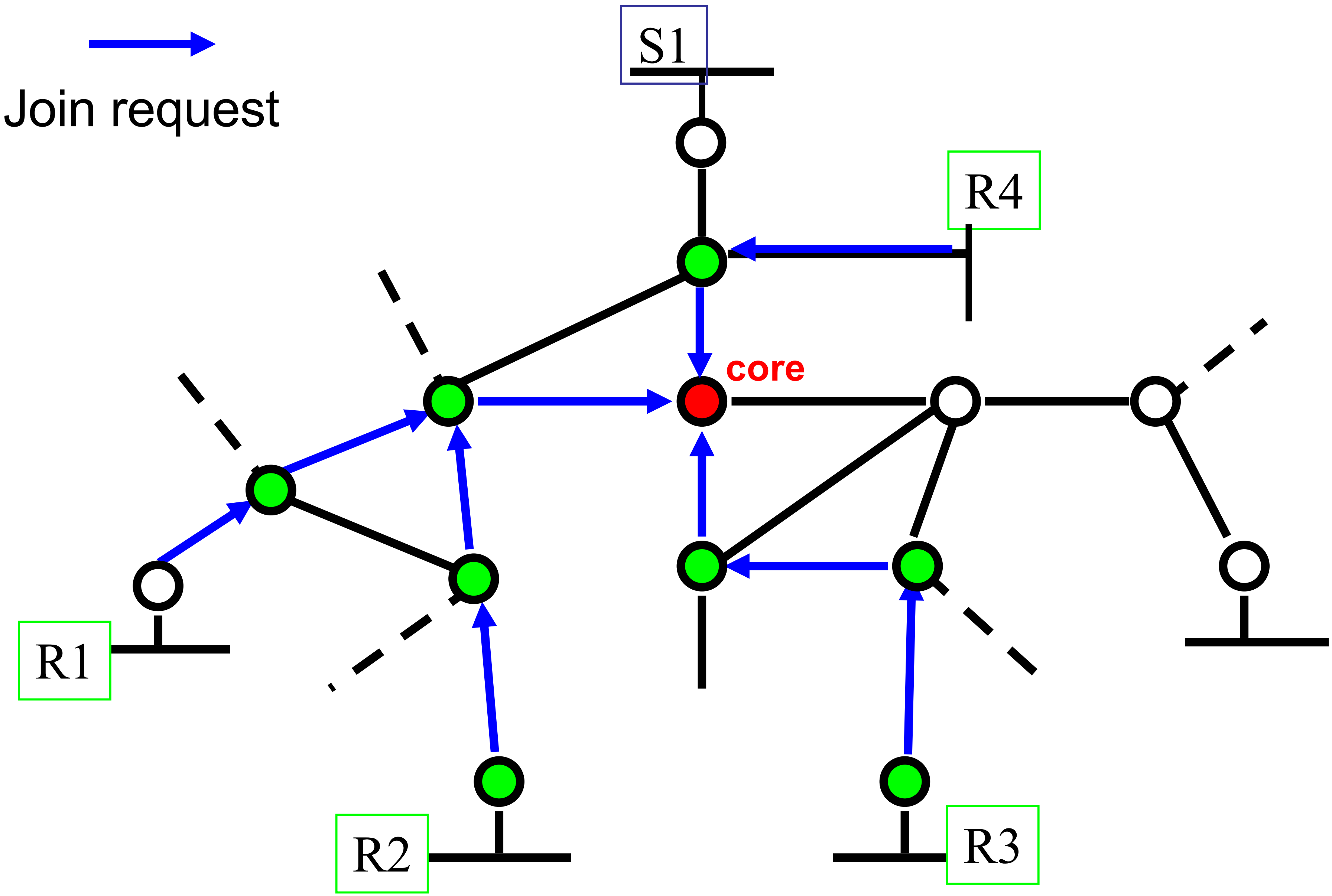


# Core-Based Tree

- Each group has both an ID (mcast address) and a core address (unicast address)
- Senders & Receivers meet at a rendezvous point (core)
  - Receivers send joining requests towards core, resulting in branches of multicast tree built
  - Sources send data packets towards the core
- Utilize underlying unicast routing.

# How CBT builds the Tree

- Add core address in IGMP membership report msg
- The router sends JOIN-REQUEST(G, C) msg towards the core (by unicast)
- Upon receiving a join request, a CBT router (which is on the unicast path to the core) checks that
  - if no routing state for group G
    - Set up (G, child-interface, parent-interface, pending)
    - forward the msg towards the core
  - if state(G) exists, adds the new child-interface
    - if G is in confirmed state, sends JOIN-ACK msg
    - if pending, wait for JOIN-ACK from parent-interface



Join request
S1
R4
core
R1
R2
R3

# How to forward multicast packets

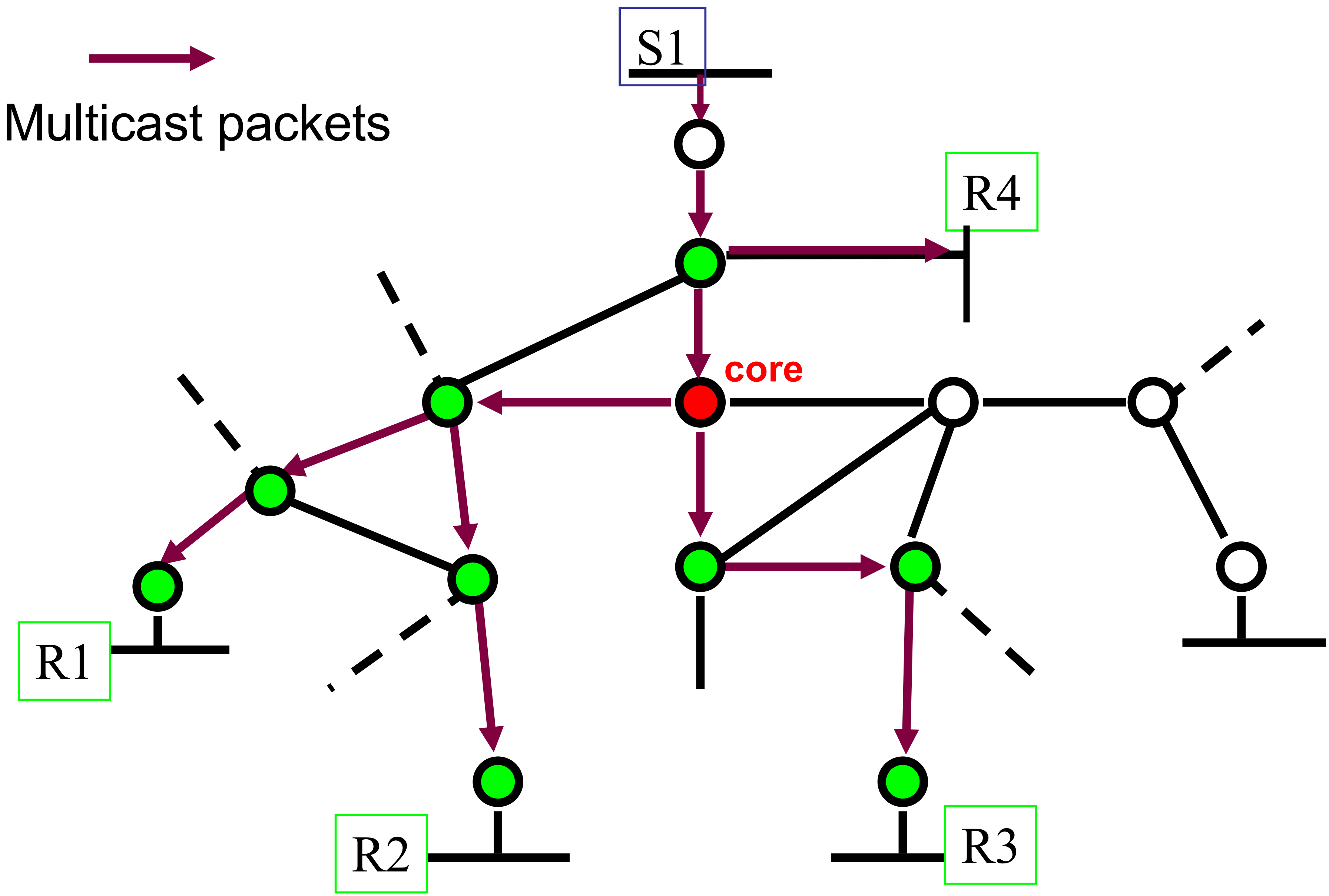
- The source unicasts data with the core address as the destination
  - group ID (mcast address) is carried in IP option field

| | | |
|---|---|---|
| IP basic header<br>dest = core | IP option,<br>multicast addr | transport header<br>and data… |

- Once the packet hits the core, or a CBT router on the established group tree, its destination address is changed to the group ID, and the packet is multicasted along the tree.

# CBT routers have a tough job

- Each CBT router R checks *every* IP packet for CBT option
  - if option=G, but R has no state for group G, forward the packet as is (unicast towards the core)
  - if R has state for G,
    - R puts G in destination address field
    - forwards a copy of the packet to the interfaces that have states for G, except the one the packet came from


Multicast packets
S1
R4
core
R1
R2
R3

# How to maintain a CBT tree

- Adjacent CBT routers exchange periodic “keepalive” msgs
  - take recovery action upon detection of failure
- Add branch: when the leaf router receives JOIN-ACK, the new branch has been attached to the tree
- Cut branch: if a router has no members of G on its subnet, and has received a QUIT from each of its child interfaces, sends QUIT-REQUEST(G) to parent
- All CBT control msgs are acknowledged.

# How to recover from failures

- if a router's path to non-core parent fails,
  - transparent recovery: resend JOIN-REQUEST msg unicast towards the core; or
  - send FLUSH-TREE msg to all child-interfaces, forcing downstream routers to rejoin the tree individually.
- if the core fails?
  - prepare an ordered list of cores to start with
    - can all participants independently choose the same one at the same time?

# Other issues in CBT design

- How to select the core?
  - the core location can make big impacts on overall performance
    - even with an optimal core location, not all data delivered over shortest paths
  - in the special case of a single sender, placing the core near the sender results in a shortest path tree
  - in general no polynomial time algorithm exists to compute optimal core

# Other issues

- multiple-core as an alternative?
  - need protocols to manage multiple cores
  - complex, failure mode not well understood
- traffic concentration on the single tree
- "hard state" does not automatically adapt to route changes
  - routes may change due to reason other than node/link failure; CBT branches do not change in absence of hard failures.

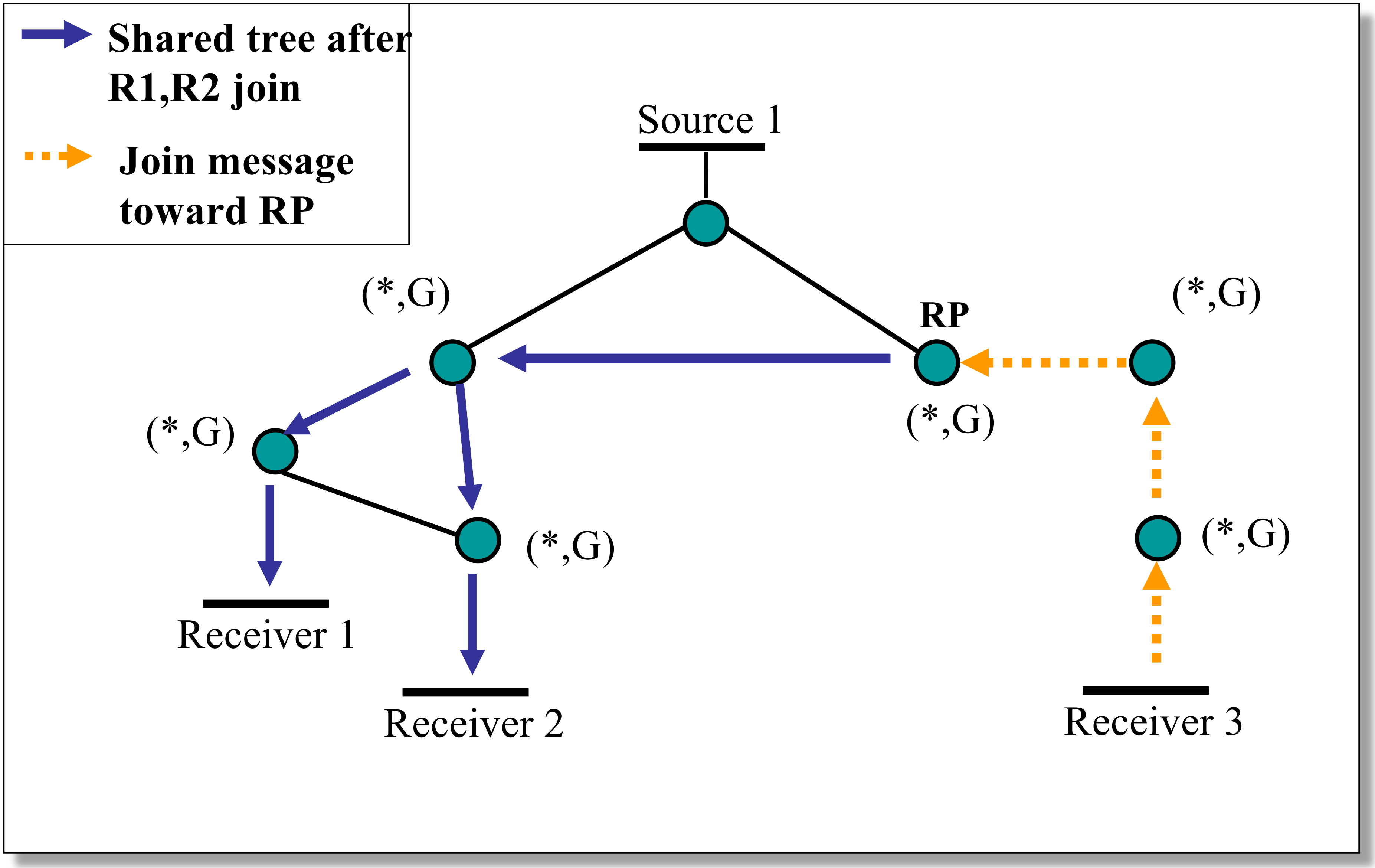
# Protocol Independent Multicast (PIM)

- The standard multicast routing protocol in Cisco and other commercial routers.
- "Protocol Independent"
  - Does not do its own routing information exchange, rely on underlying unicast routing protocol.
- PIM-DM (Dense Mode): similar to DVMRP, but
  - Without the routing information exchange part, differs in some minor details
  - For groups with dense members
- PIM-SM (Sparse Mode), or simply PIM
  - For groups with sparse members
  - *Support both shared tree and source tree.*

# PIM Basic Protocol Steps

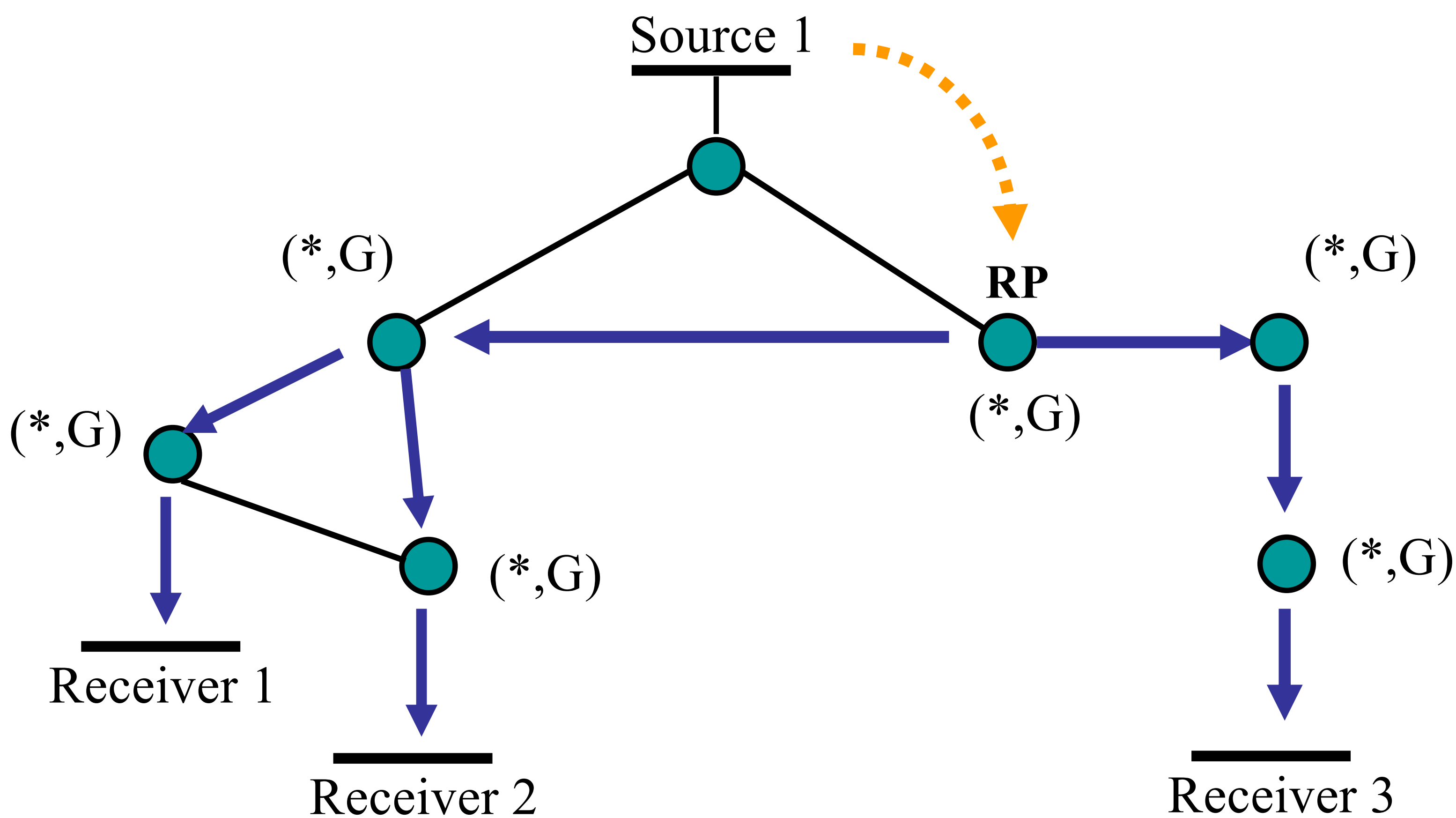
- Routers with local members *Join* toward Rendezvous Point (RP) to form the shared tree
- Routers with local sources encapsulate the data packets in *Register* messages to RP
- Routers with local members may initiate data-driven switch to source-specific trees

# Build Shared Tree



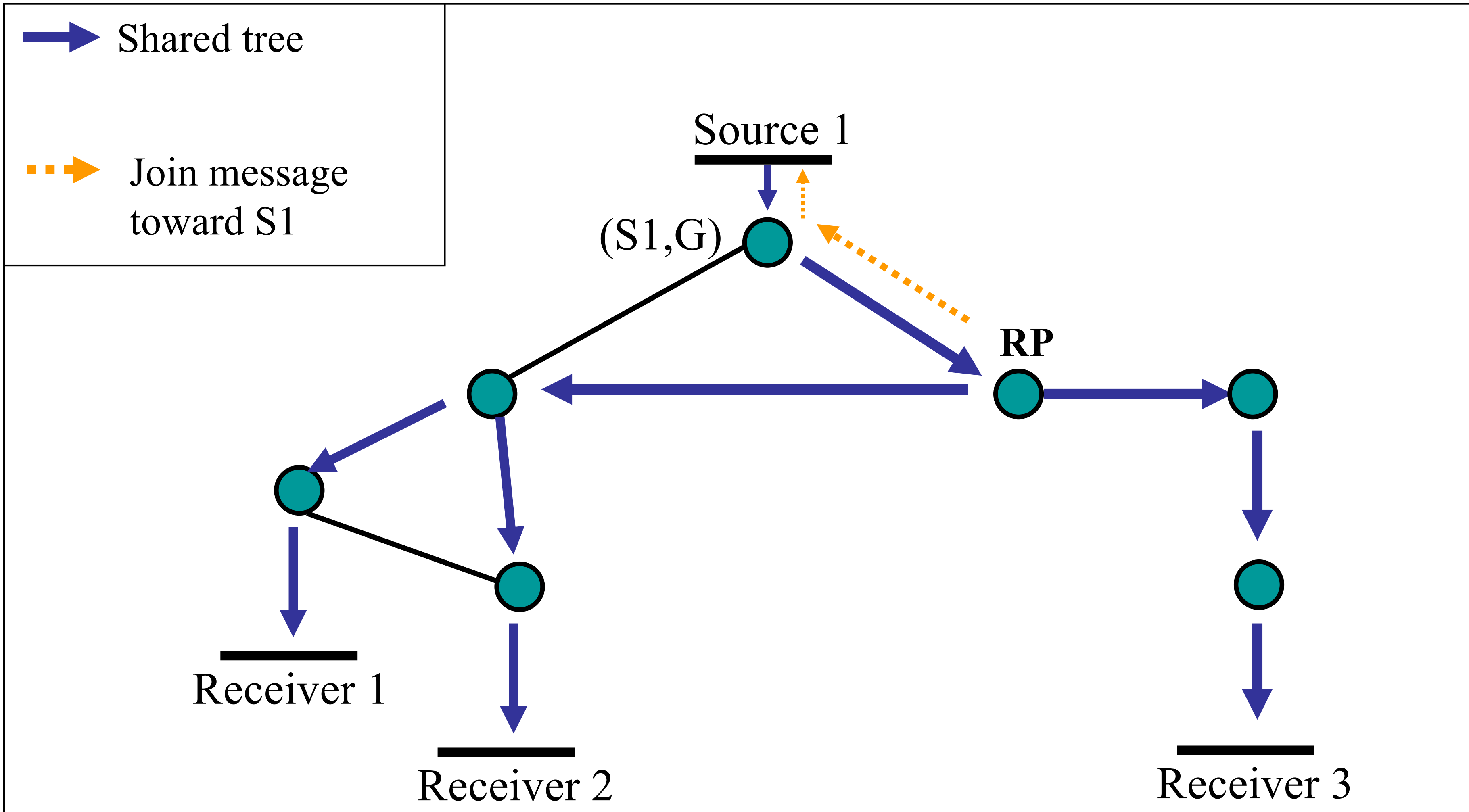
# Data Encapsulated in Register mesgs

Unicast encapsulated data packet to RP in **Register pkt**



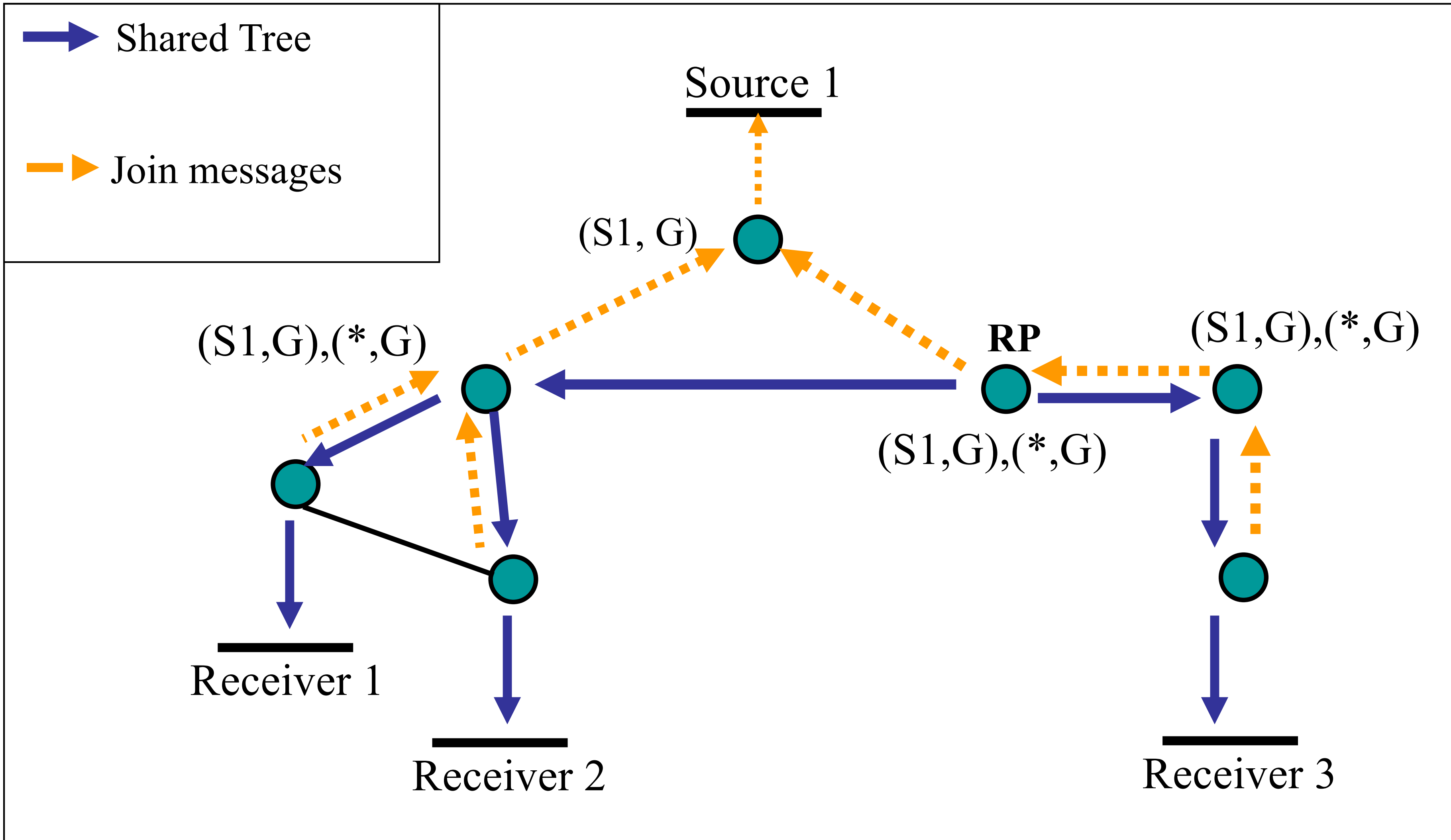

RP de-capsulates and forwards the pkt down the shared tree

# RP Sends Join to High Rate Source



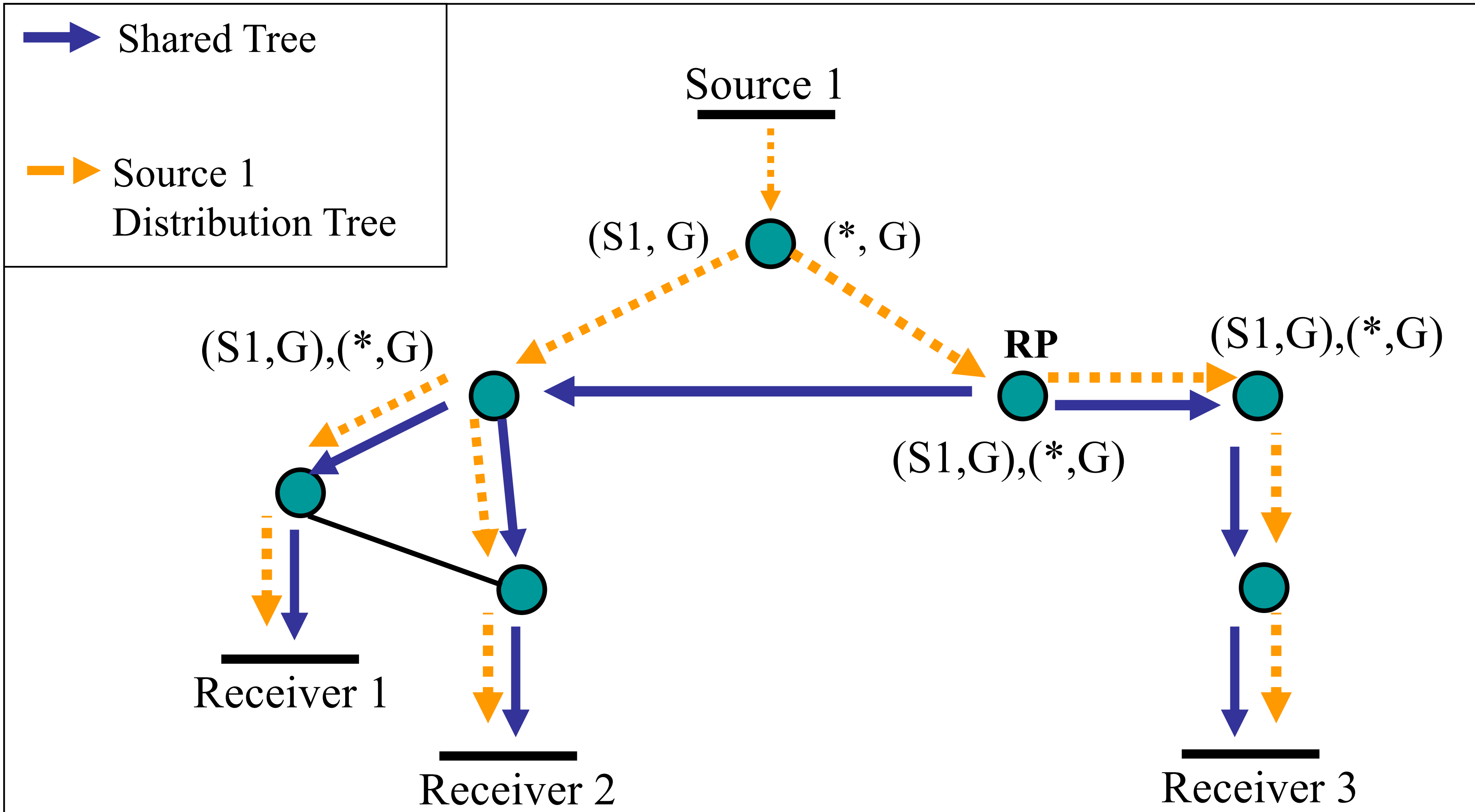

No need to encapsulate future packets

# Build Source-Specific Distribution Tree



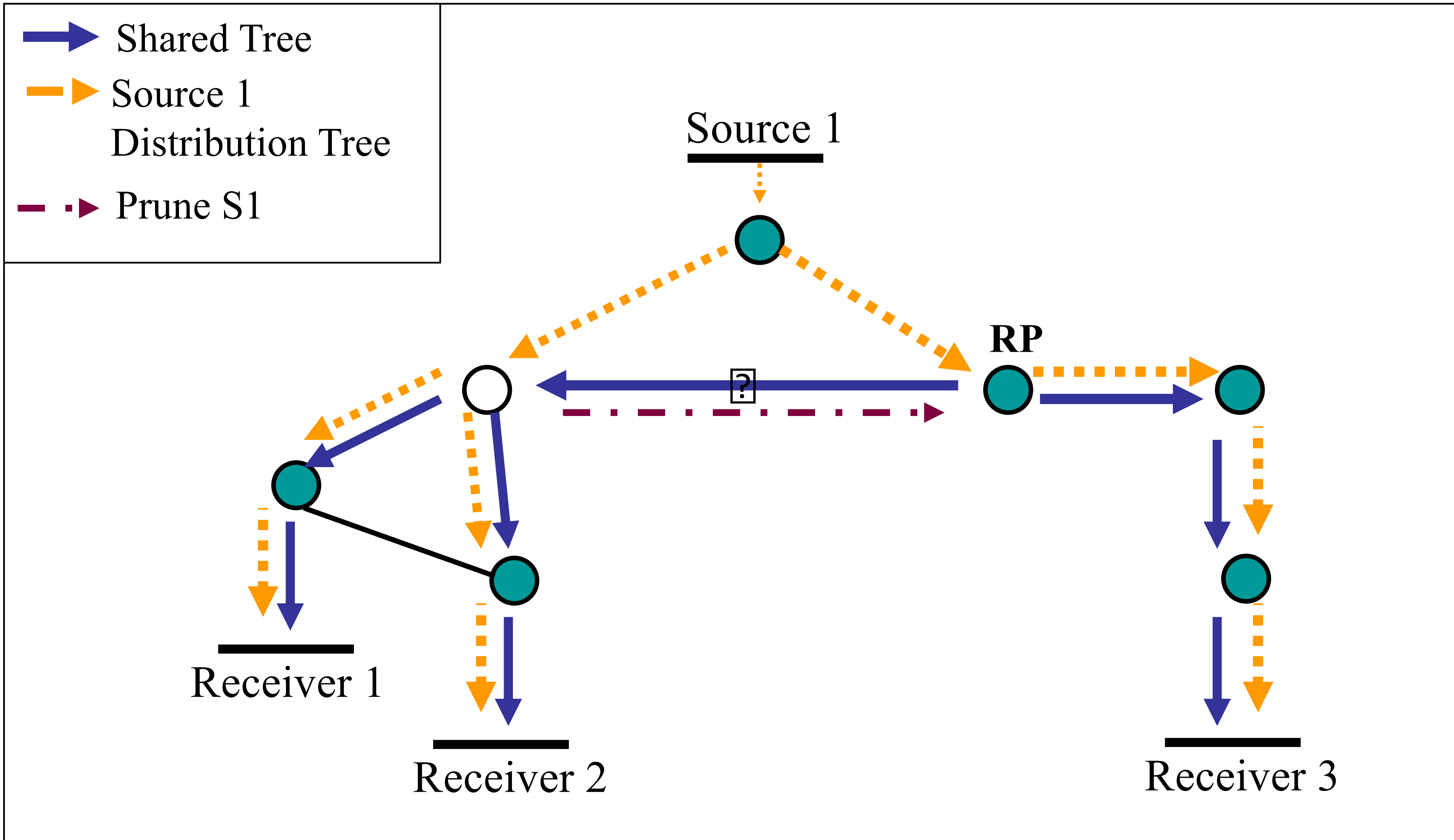

For high data rate source, receivers sends JOINs to the source

# Forward On "Longest-match" Entry




**Source-specific entry (S1, G) is "longer match" for source S1 than is shared tree entry (*, G) that can be used by any source**

# Prune S1 off Shared Tree




**Prune S1 off shared tree where S1 and RP entries differ**

# RP Mechanism

- Hosts only need multicast address to send or receive
- Routers use algorithmic mapping of Group address to RP from a known set of RPs
  - A set of available candidate RPs is flooded to PIM routers
  - Each PIM router uses the same hash function to map a particular group address to an RP.
  - Not necessarily the optimal RP location.

# IPTV services using IP Multicast

- Multicast delivery of TV content in networks
- Source Tree – minimize latency – high processing and memory requirement
- Shared Tree – less hardware resources – introduce latency