CS 6390
Advanced Computer Networks
Multicast

Computer Networks
Dr. Jorge A. Cobb

UTD

Motivation

- Original IP service model was one-to-one
  - One sender sending its data to one receiver at a time
- Recently, new apps with multi-receiver semantics
  - Audio/video conferencing, news dissemination, Internet TV, etc.
- Unicast not designed to efficiently support multi-receiver apps
- Solution
  - *Multicast support*
  - Reduces the number of “messages” in the network
  - Increases however the amount of state at routers

Multicast semantics

- A group is represented by a class D IP address (more later)
  - Zero or more receivers form a multicast group
  - Senders are not part of the group (unless they are also rcvrs)
- Open group semantic
  - Receivers can join/leave at will
  - Anyone knowing group address can send to it
- IP based best effort delivery semantics
  - Multicast supports UDP only – *no TCP !* Why?
    - If routers are reliable per-hop, this requires lots of buffer/state at each router
    - Otherwise, if source receives ack's from the receivers, we have ack implosion problem (receive N ack's)

Open group semantic

- Advantages
  - Sources do not need to know individual receivers
  - Receivers do not need to know the sources either (they simply join the group and receive data)
- Disadvantages
  - Difficult to protect from unauthorized senders/receivers

Router and Host Functions

- Basic host model
  - When sending data, normal IP-Send operation
    - IP src-addr = address of sender
    - IP dst-addr = class D address of multicast group
  - When receiving data, tell your router what group you are interested in
    - i.e., join/leave a multicast group (start/stop receiving data from **all sources** sending to the group)
    - The router will forward multicast packets to the LAN of the host
      - What are the IP source and destination addresses of these packets?
      - What about LAN addressing? (later)

Routers and Multicast

- Basic router model
  - Prepared to receive data from all multicast group addresses
  - Know when to forward or drop packets
    - Keep track of interfaces leading to receivers
    - Forward multicast packets over these interfaces
    - The packet's source and destination addresses are not changed

- Hosts are simple, is the routers that do all the "magic"

Addressing

- Multicast group addresses
  - Class D IP addresses (224.0.0.0 – 239.255.255.255)
  - Implicit scoping
    - 224.0.0.0 – 224.0.0.255: link scoped
    - 224.0.1.0 – 238.255.255.255: global scoped
    - 239.0.0.0 – 239.255.255.255: admin scoped
  - Explicit scoping
    - Use TTL value for scoping

Addressing

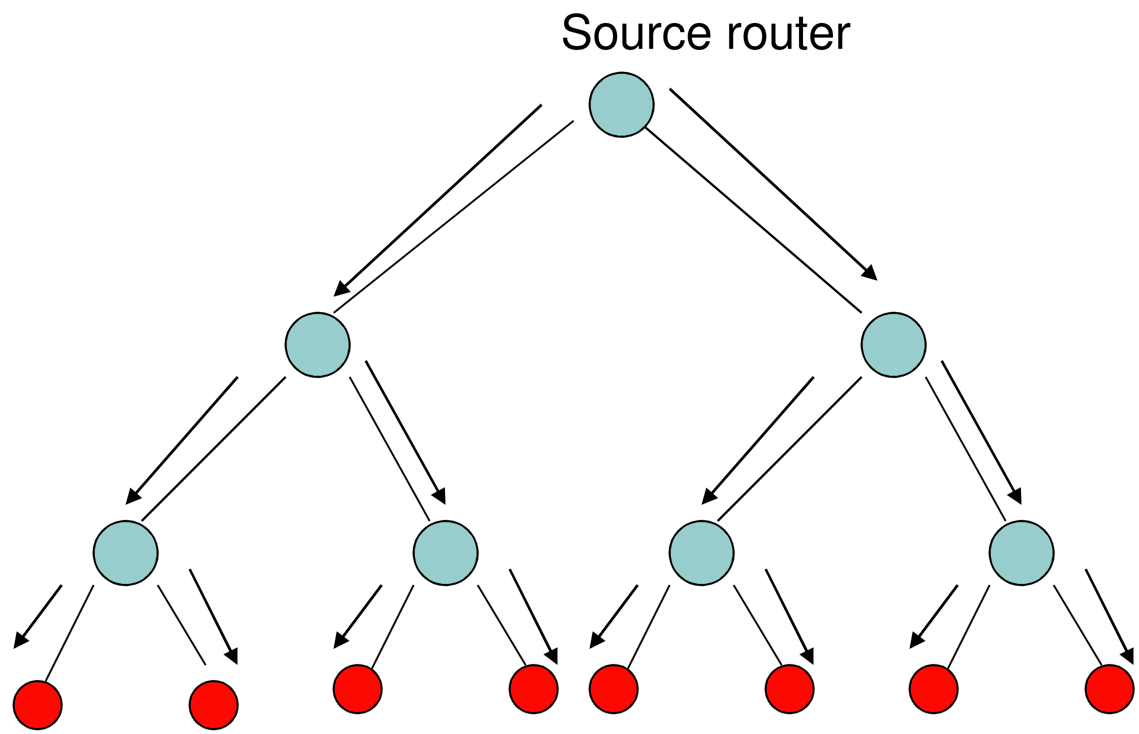
- As mentioned bfore, multicast packet format
  - Source IP field is unicast IP address of source host
  - Destination IP field is the IP multicast group address
  - In general, we show this (Source IP, Dest IP) couple as
    (S,G) where S is source IP, G is dest IP (multicast group addr)
- How to do actual delivery to a receiver host?
  - Map IP multicast address to a LAN (ethernet) multicast address
    - LAN src address: router
    - LAN dst address: multicast LAN address
    - Receiver (host) NIC is configured to receive packets destined to this LAN multicast address
  - Hence, multicast on the LAN is done automatically without IP router intervention

Addressing

- What about the source?
  - IP header?
  - LAN header?

How data is forwarded to multiple receivers?

- Routers in the network build *forwarding trees* connecting sources and receivers
  - On-tree routers keep *multicast forwarding states* for each group (more on this later)
- Source data propagates on this tree toward the receivers



N receivers

router

Single multicast message

Multicast messages = 2(N – 1)

If unicast,

- log N * N unicast msgs
- first level routers handle N/2 msgs
- source sends N msgs

Multicast forwarding states at routers

- used to determine how a multicast packet will be forwarded

- consists of several elements:
  - source address - S
  - group address - G
    - (S,G) determines the particular tree
  - incoming interface – iif (parent on tree)
  - outgoing interface list – oif list (children on tree)
  - various timers (needed to deal with aging entries out of the forwarding table)
  - Other misc. info.

Dynamic State

- Created when receivers join a multicast group and when sources send packets addressed to the group.
- Deleted after receivers leave a multicast group or senders stop sending packets addressed to the group.
- Every forwarding state has a fixed lifetime – needs refreshing
  - E.g., outgoing interface (oif) list maintenance
    - Every oif has a fixed lifetime
    - When lifetime expires for an oif, it is removed from oif list
    - Certain events can reset the timer to its max value

Big picture

- Create multicast forwarding trees
  - connects sources and receivers
  - data is sent along the tree from sources to receivers

- Three components
  - Host to router communication
  - Intra-domain routing
  - Inter-domain routing

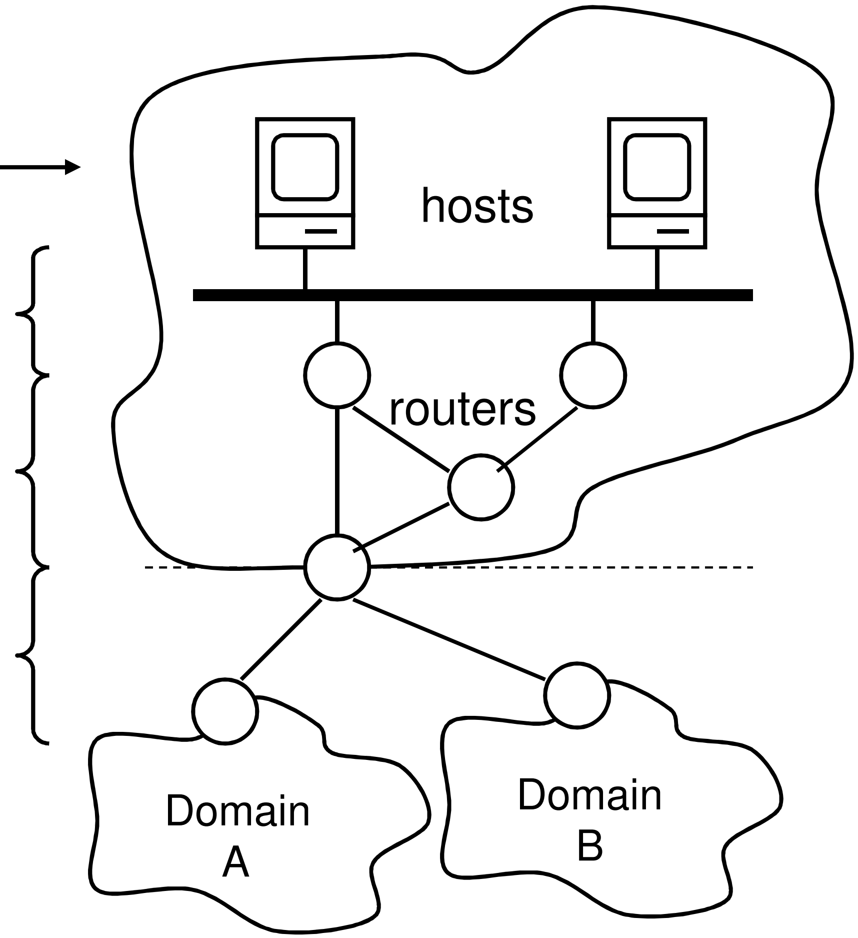
IP Multicast Architecture

application API

host-to-router (IGMP)

intra-domain routing

inter-domain routing

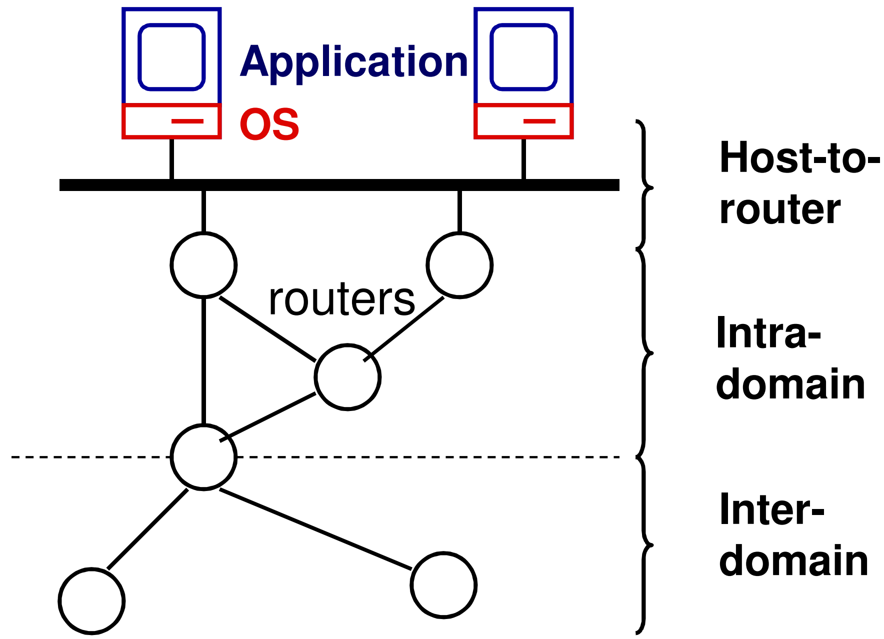


Application API

Application API has socket options:

IP_ADD_MEMBERSHIP

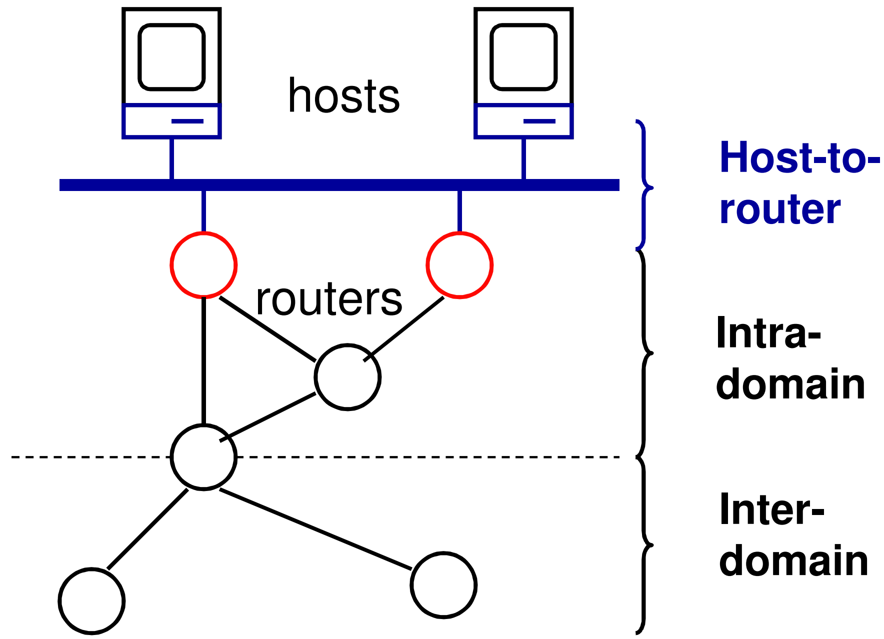
IP_DROP_MEMBERSHIP



Host-to-Router

Internet Group Management Protocol (IGMPv2, v3 standardized in 10/02)

Kernel informs router that an application wants to join a specific group G



IGMP

- used by hosts to indicate their interest in receiving packets addressed to a particular multicast group *G*.
  - IGMPv1 (RFC 1112)
    - Routers
      - *General Membership Query*
    - Hosts
      - *Membership Reports (requested by routers)*
      - *Unsolicited Group Membership Reports (sent periodically to router)*
  - IGMPv2 (RFC 2236)
    - Added explicit *Leave/Join Group (host)* and *Group Specific Membership Query (router)* messages
  - IGMPv3 (RFC 3376)
    - Added source filtering capabilities
  - IGMP messages aren't forwarded by routers

Intra-Domain Routing

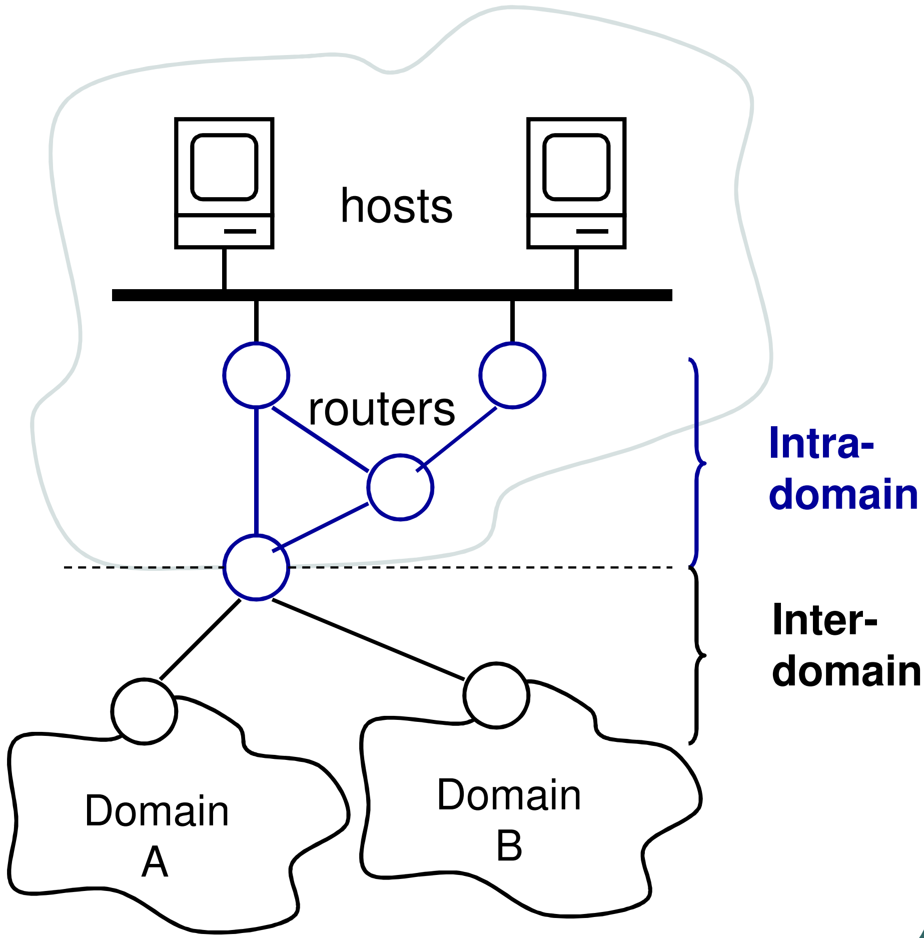
Building a forwarding tree among session sources and receivers

Dense Mode Protocols

- DVMRP, MOSPF, PIM-DM

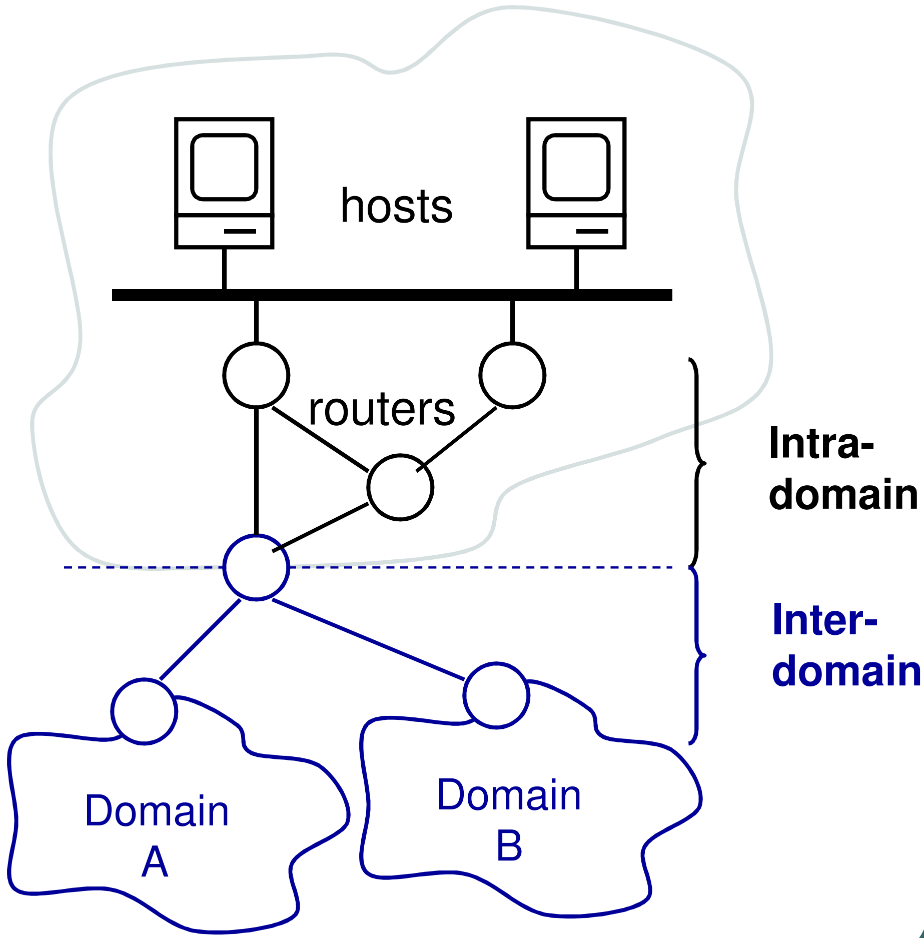
Sparse Mode Protocols

- PIM-SM, SSM



Inter-Domain Routing

Building a forwarding tree
Between ASms.



Back to Intradomain Routing

Multicast Tree construction

- Building forwarding trees between sources and receivers
- Hard due to the open service model
  - Sources do not know who/where the receivers are
  - Receivers do not know who/where the sources are

Multicast Tree construction (3 ways)

- Flood and prune
  - Begin by flooding traffic to entire network (Create a bdcast tree)
  - Prune branches with no receivers
  - Examples: DVMRP, PIM-DM
  - Disadv:
    - unwanted (prune) state where there are no receivers
    - Initial flood
- Link-state multicast protocols
  - Routers advertise groups for which they have senders/receivers to entire network
  - Compute trees on demand
  - Example: MOSPF
  - Disadv:
    - Link-state is not scalable
    - Unwanted state in routers not on the multicast tree

Multicast tree construction (contd)

- Receiver driven
  - Tree is built from the receivers up towards the tree root
  - Tree construction is based on the unicast routing tables
  - Disadvantage:
    - multicast state in routers even though there are no sources (if using a core approach see below)
    - Otherwise, none that I know of (if using the source-based tree)

Root Node approaches

- Core-based (shared-tree) protocols
  - Specify a *meeting place* or *core*
  - Receivers join the tree rooted at the core
    - A single tree is shared by all receivers
    - Sources send their packets to the core
    - Core forwards the data to the receivers
  - Requires mapping between group addresses and cores
  - Examples: CBT, PIM-SM
  - +s: less state at routers
  - -s: higher delay, traffic concentration at core

UTD

Root Node Approaches

- Source-tree protocols
  - Multicast tree is rooted at each individual source
    - Separate shortest path tree for each sender
  - Example: DVMRP, MOSPF, PIM-SM
  - +s: Low delay, better load distribution
  - -s: more state at routers (per source state), how to find out the identity of the sources?

A brief overview of multicast

The Evolution of Multicast: From the MBone to Inter-Domain Multicast to Internet2 Deployment

Kevin Almeroth

IEEE Network, 2000

(discusses interdomain multicast)