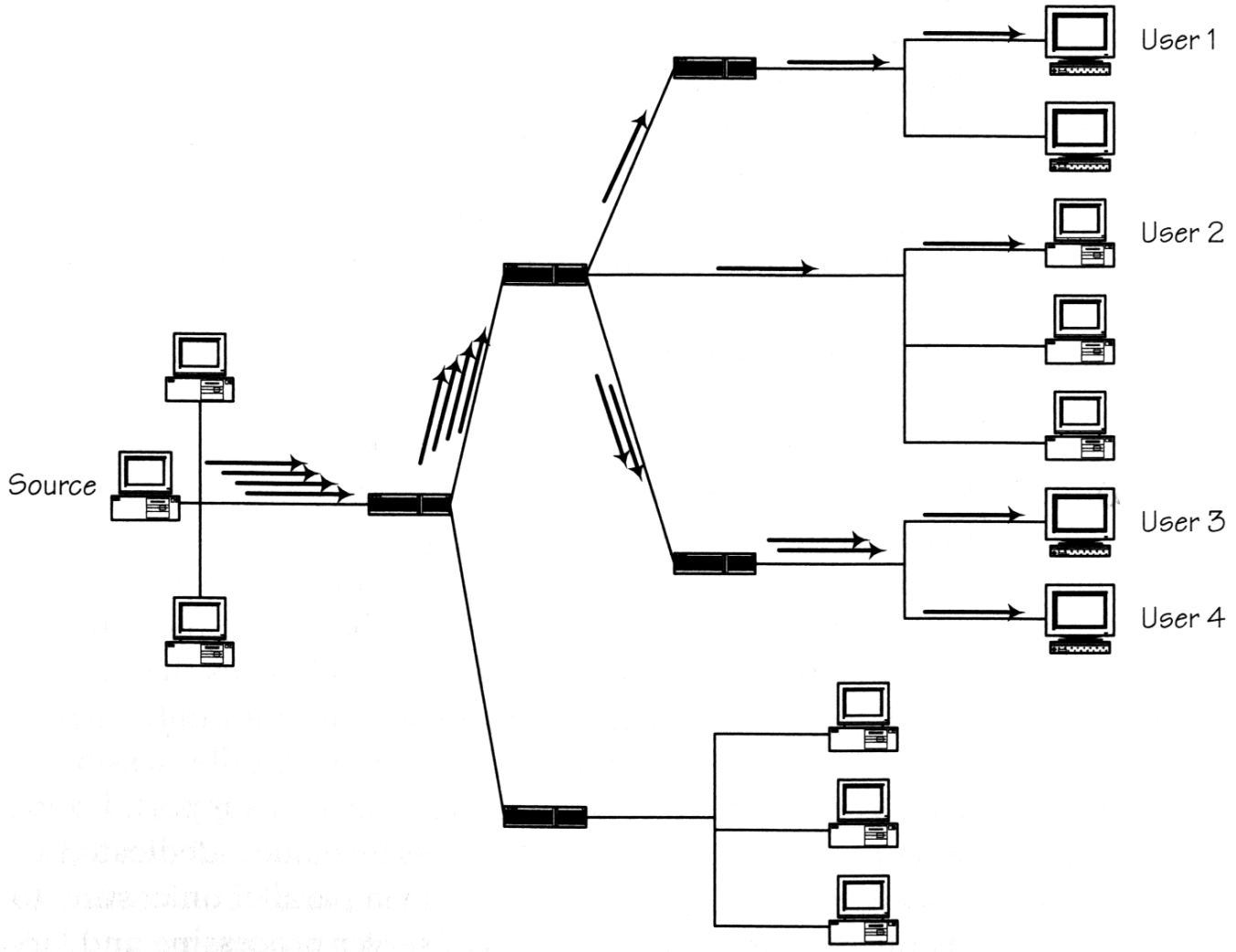
Multicast

- Introduction
- Group management
- Routing
- Real-time transfer and control protocols
- Resource reservation
- Session management
- MBone

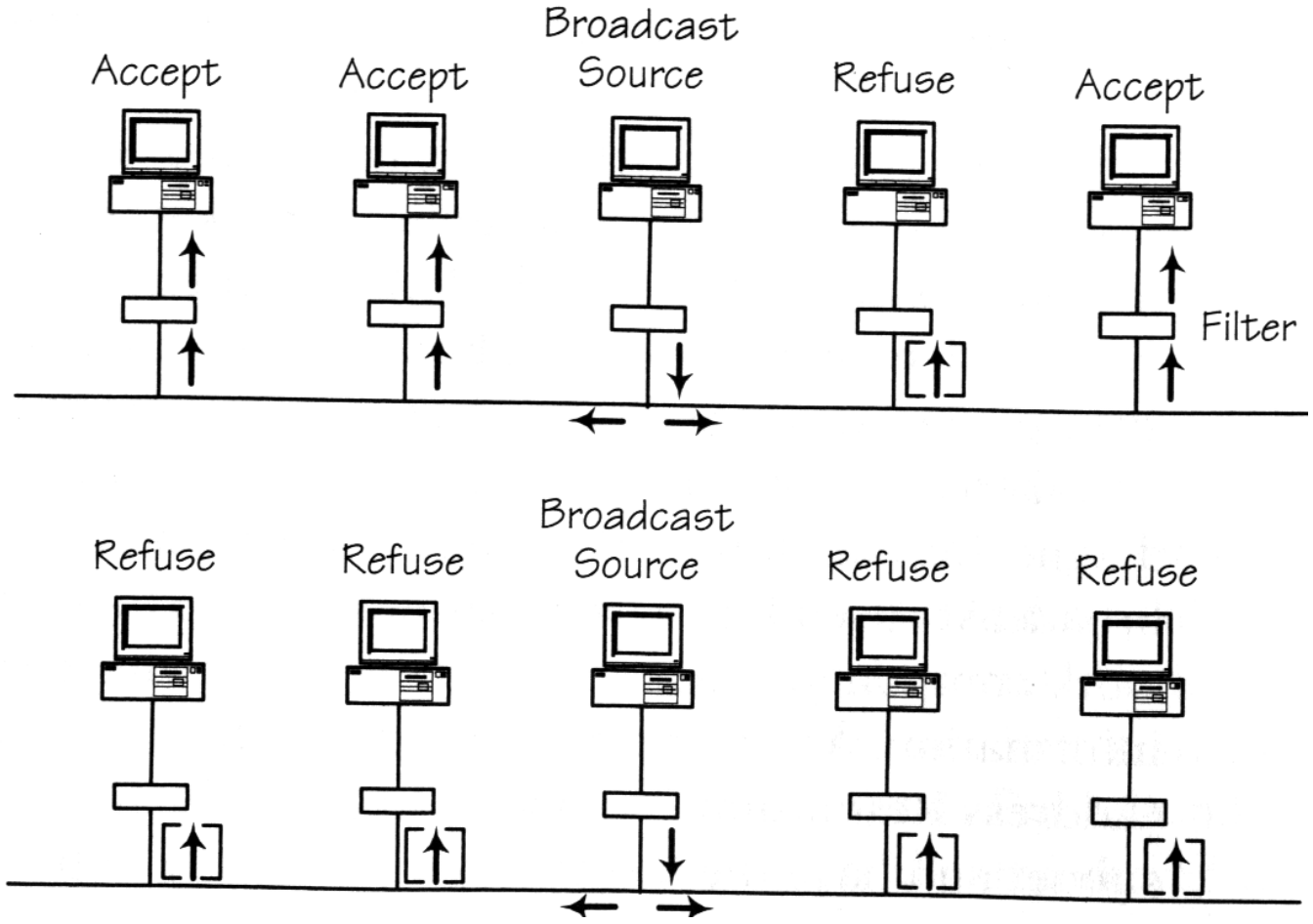
Introduction

- There are three ways to transport data in computer networks:
  + Unicast
  + Broadcast
  + Multicast
- Broadcast and multicast require special group addresses

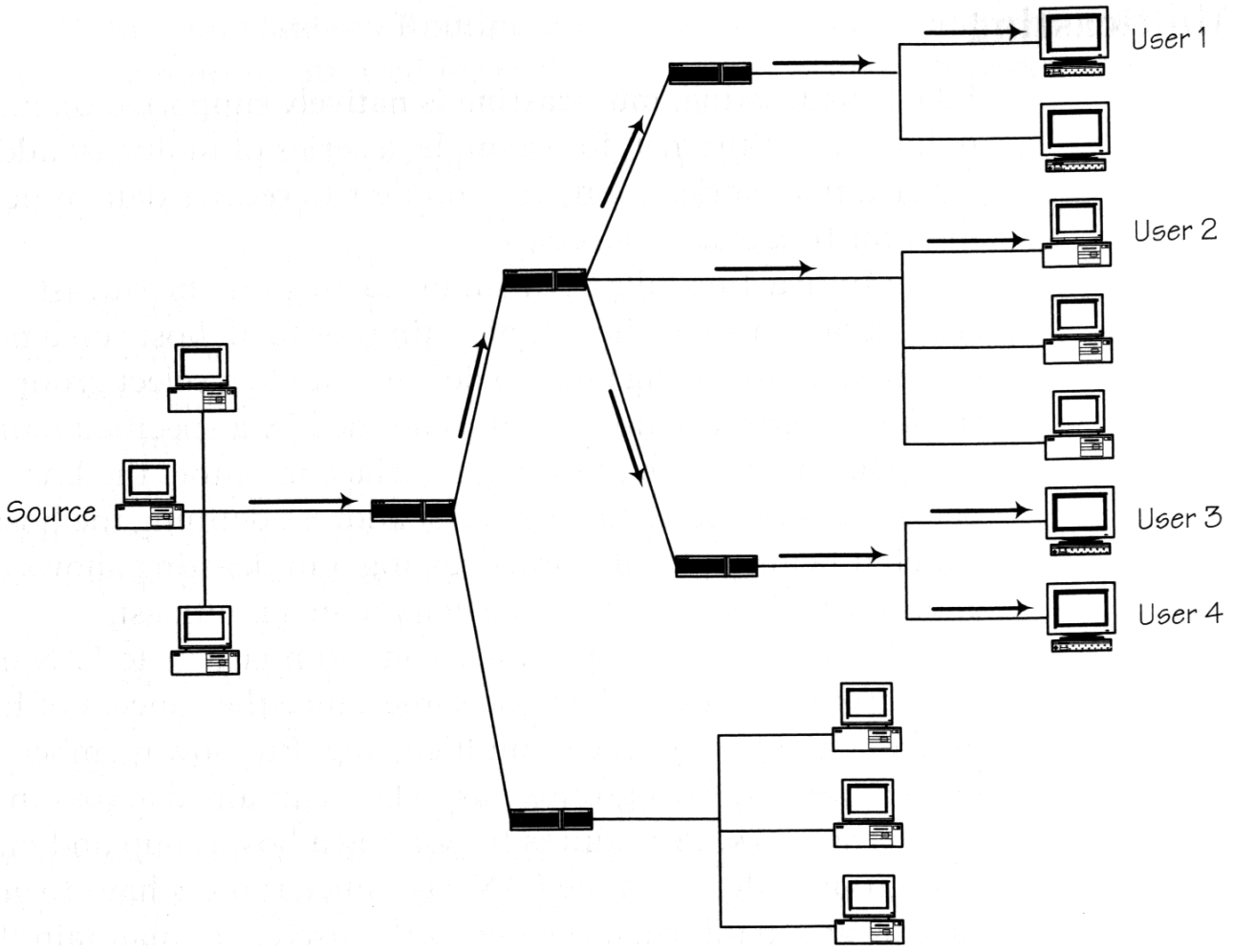
Unicast



Broadcast


Accept
Accept
Broadcast Source
Refuse
Accept
Filter
Refuse
Refuse
Broadcast Source
Refuse
Refuse

Multicast



Protocols

- Group management
- Routing
- Real-time transfer and control protocols
- Resource reservation
- Session management

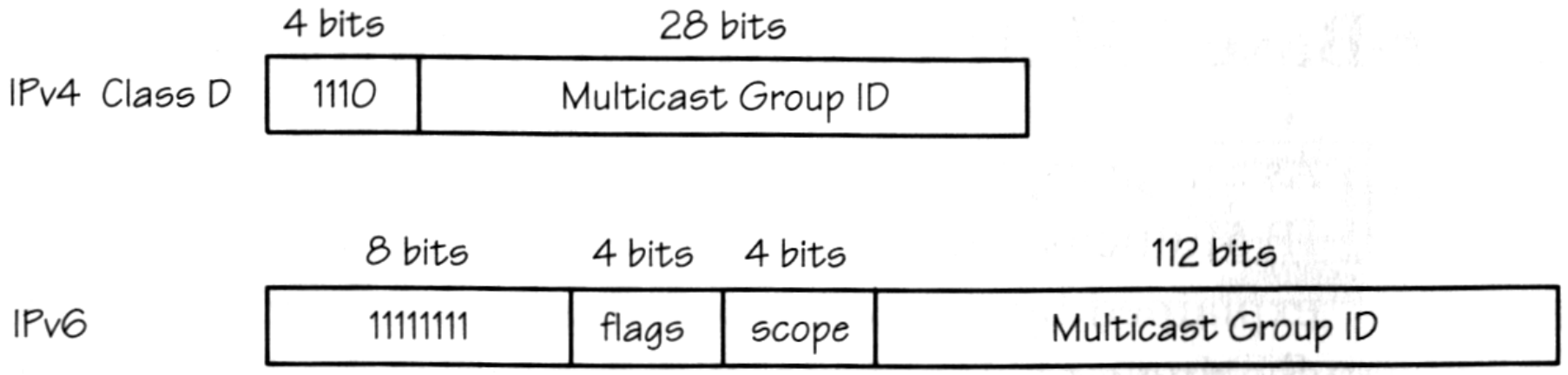
Group Management

1. Group addresses
2. Mechanism to join the groups
3. Routing protocols
4. Generation and control of the data

Group Addresses

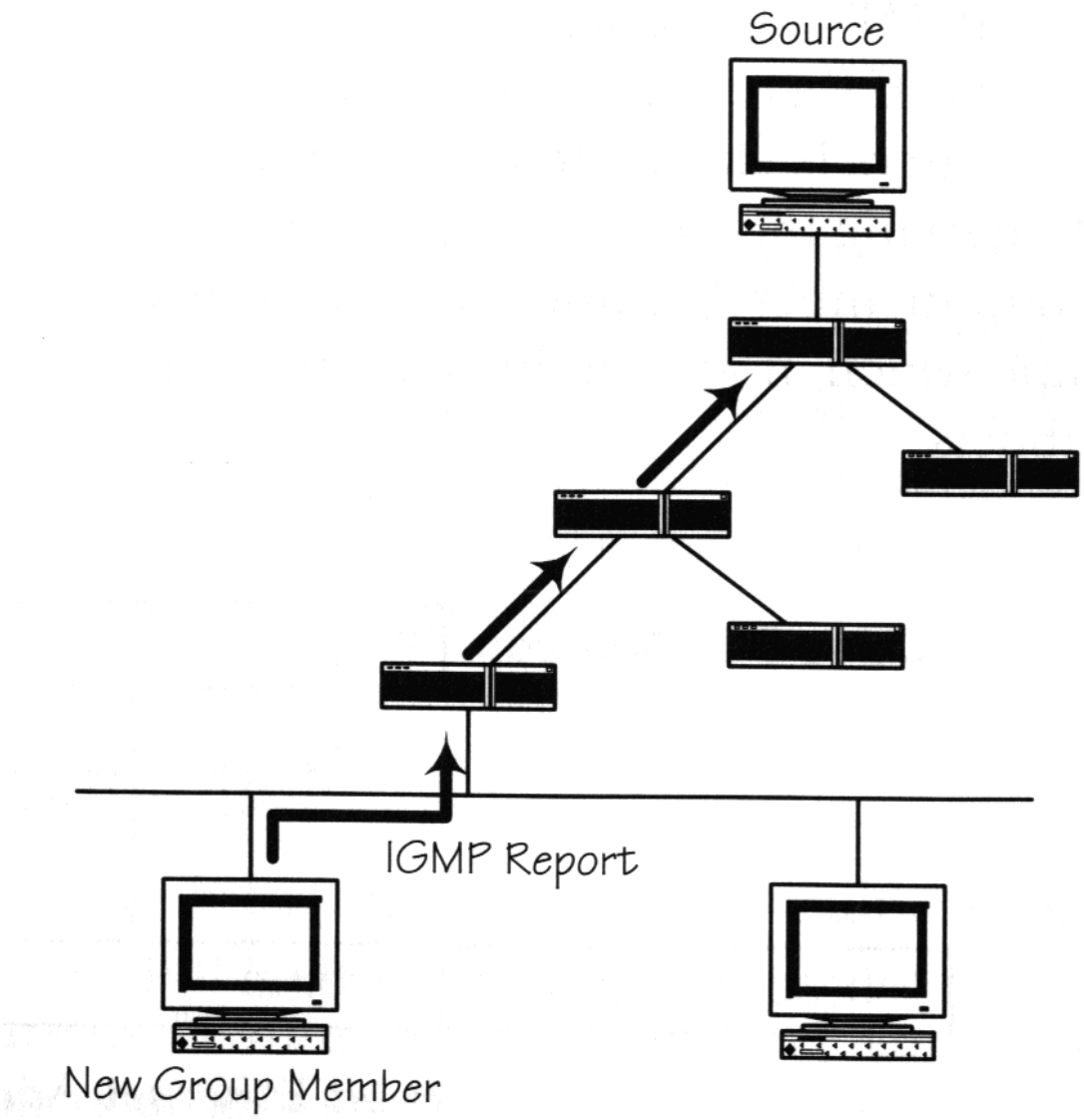
- IPv4: class D
  + addresses 224.0.0.0 - 239.255.255.255
  + addresses 244.0.0.0 - 244.255.255.255 are reserved for routing etc.
- IPv6:
  + flags: fourth bit tells whether the route is permanent
  + scope: tells how wide the group is

IPv4 vs. IPv6


4 bits
28 bits
IPv4 Class D
1110
Multicast Group ID
8 bits
4 bits
4 bits
112 bits
IPv6
11111111
flags
scope
Multicast Group ID

Joining to The Groups

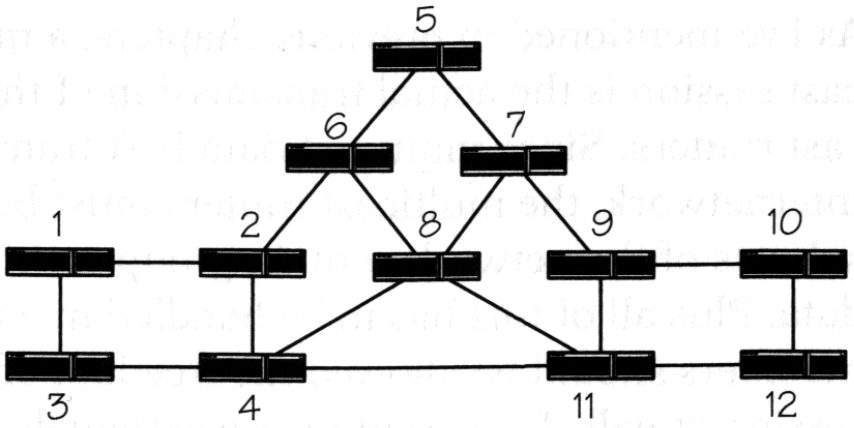
- Two alternatives:
  + A) The computer tells the router it wants to join a group
  + B) The router announces the groups and asks the hosts to join
- The latter case uses the Internet Group Management Protocol (IGMP) protocol


Source
IGMP Report
New Group Member

Routing

- The router looks the next target from a routing table
- The routers exchange and update the information in the routing tables
- Two basic methods:
  + Distance Vector
  + Link Status

Routing table



Routing Table at 1

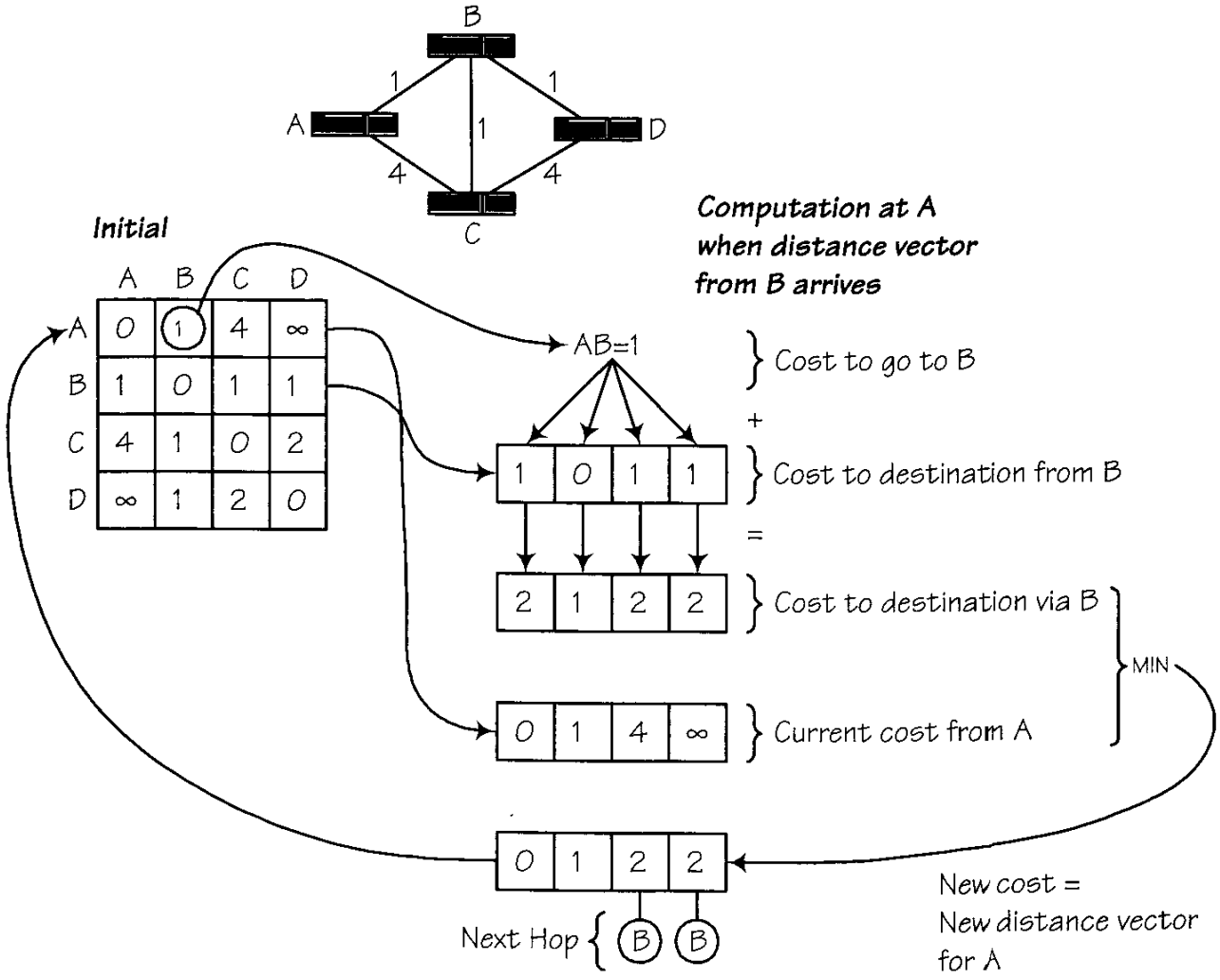
Destination	Next Hop	Destination	Next Hop
1	--	7	2
2	2	8	2
3	3	9	2
4	3	10	2
5	2	11	3
6	2	12	3

FIGURE 4.1 Example network and routing table.

Distance Vector

- Router tell its distance to other routers to its neighbors
- Easy to compute
- Does not work well, if there a often disconnections between routers
- Does not scale well

Operation


B
A
D
C
1
1
1
4
4
Initial
A B C D
A 0 1 4 ∞
B 1 0 1 1
C 4 1 0 2
D ∞ 1 2 0
Computation at A
when distance vector
from B arrives
AB=1
1 0 1 1
Cost to go to B
+
Cost to destination from B
=
2 1 2 2
Cost to destination via B
MIN
0 1 4 ∞
Current cost from A
0 1 2 2
Next Hop B B
New cost =
New distance vector
for A

Link Status

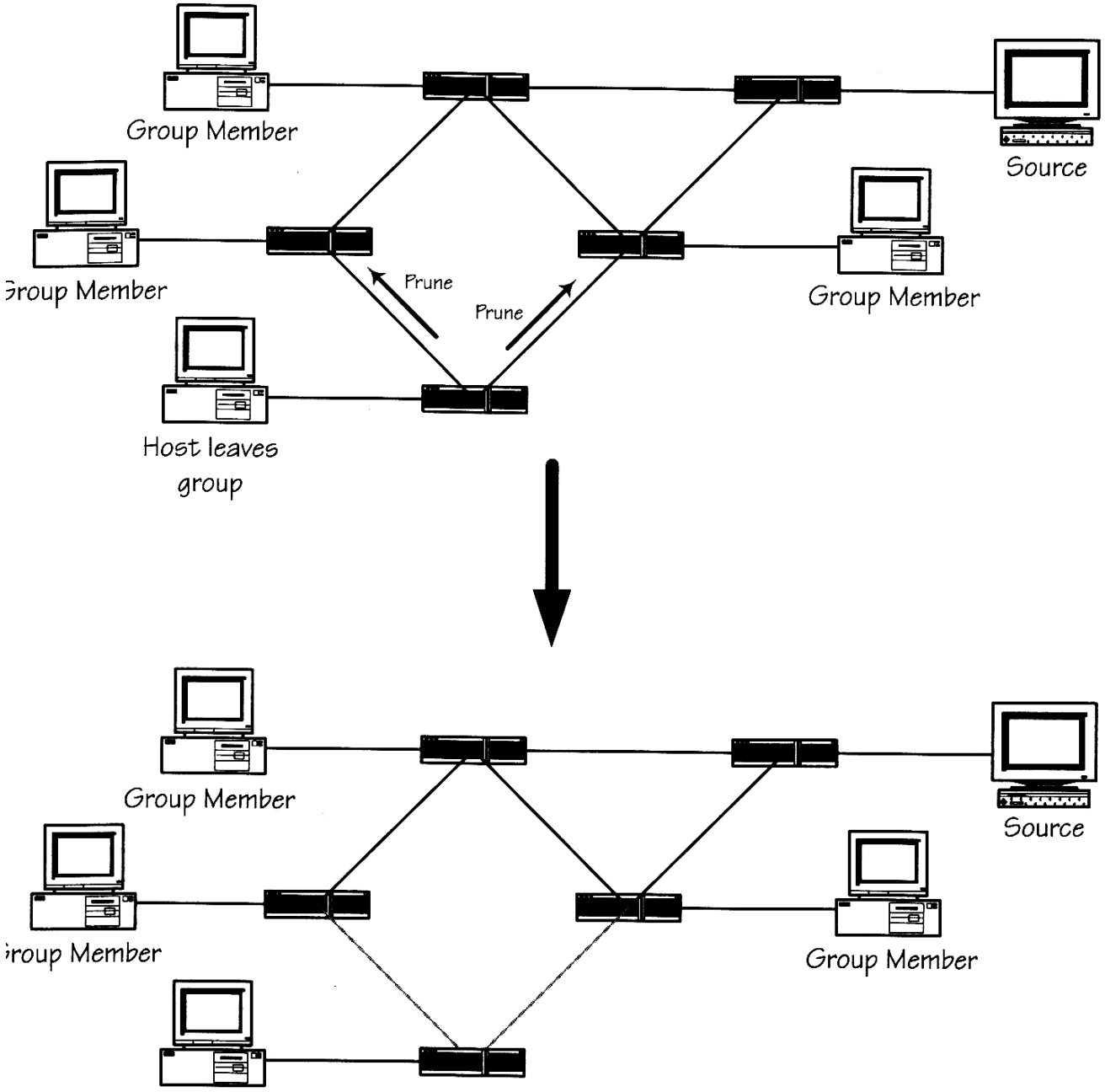
- The routers exchange information about connections instead of distances to other routers
- The receiving router updates the information about the available routers
- The routes are calculated using the Dijkstra's shortest path algorithm
- This method scales better

Multicast Routing

- Multicast routing is also based on routing tables
- In addition, the routers build a multicast tree
- The dynamic changes of the multicast tree is the problem
  + Old members leave and new join the Multicast tree
- The biggest problem is scaling

Pruning

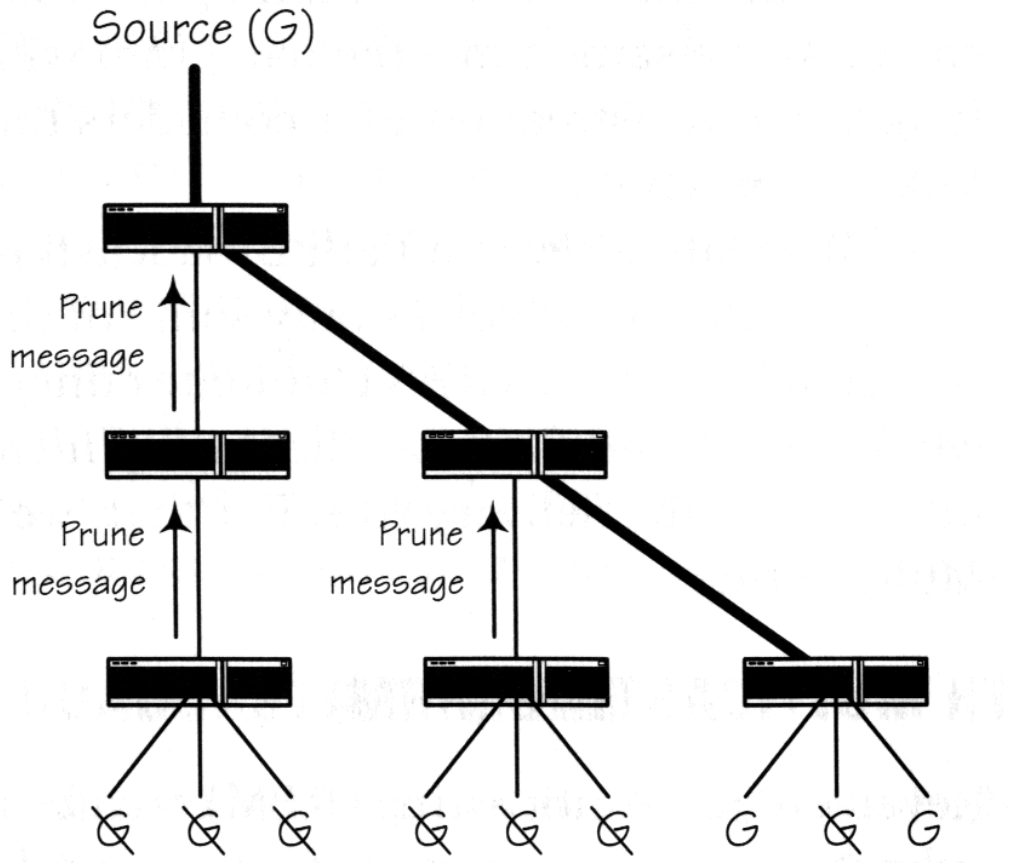
- The multicast trees can grow very big, so they have to be pruned constantly
- Branches, which do not have hosts, are removed from the multicast tree


Group Member
Source
Group Member
Prune
Prune
Group Member
Host leaves
group
Group Member
Source
Group Member
Group Member

Flooding

- Flooding is the easiest way to build the multicast trees
- Multicast packets are flooded to all output ports of the router
- A router forwards those packets which it has not seen previously
- Unnecessary branches can be pruned later on

Pruning


Source (G)
Prune message
Prune message
Prune message
G
G
G
G
G
G
G
G
G

Multicast Routing Protocols

- Distance-Vector Multicast Routing Protocol (DVMRP)
- Multicast Extension to Open Shortest Path First (MOSPF)
- Protocol Independent Multicast (PIM)

DVMRP

- Distance Vector Multicast Routing Protocol (DVMRP) is based on RPM algorithm
- Original Mbone routing protocol
- Easy to implement
- Does not scale well
- Works well only with distance vector routing protocols

MOSPF

- Multicast Extension to Open Shortest Path First (MOSPF) is based on link state method
- Multicast packets are flooded only to nearby area
- The tree is build as usual
- Then the tree is pruned to a multicast tree

Properties of the MOSPF

- Reacts fast
- Computation of the trees is heavy
- Works only with link state protocols

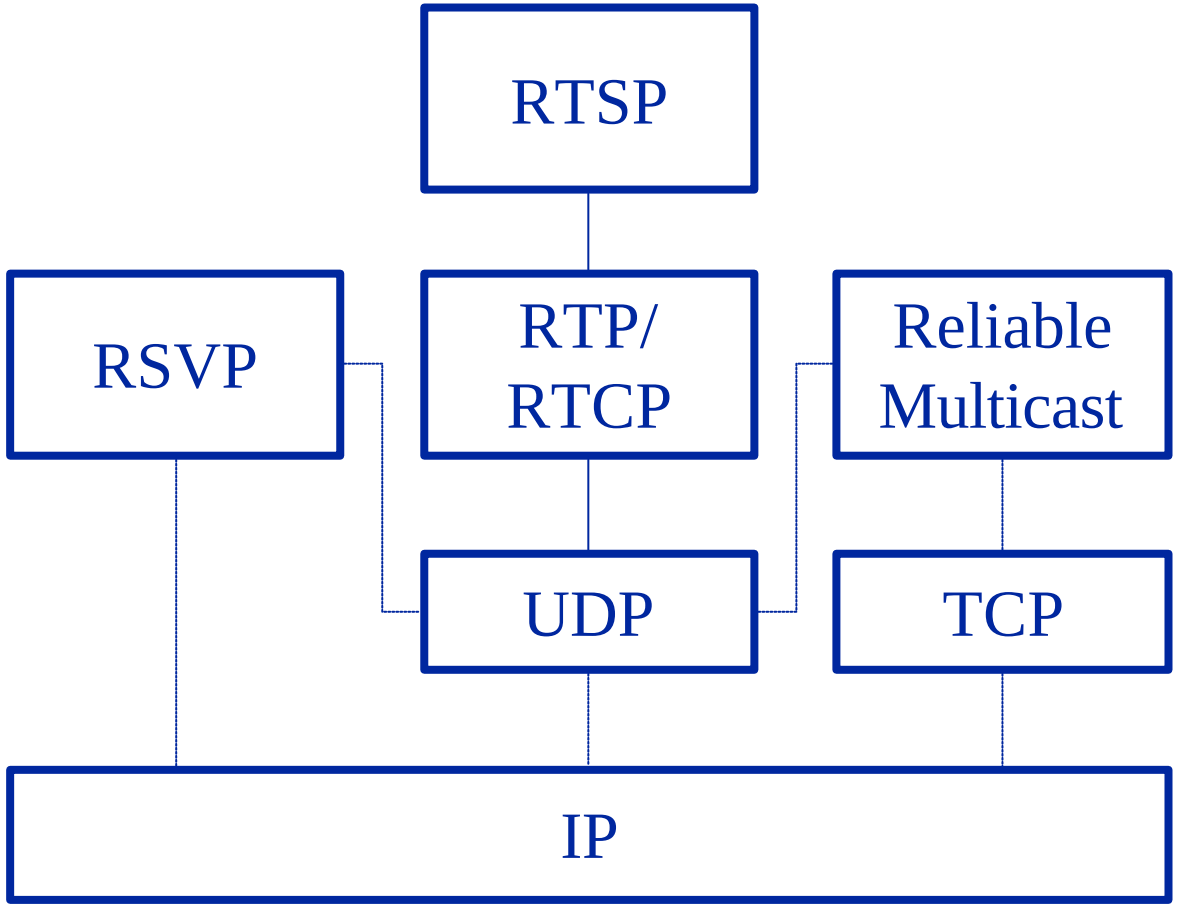
PIM

- Protocol Independent Multicast (PIM) is independent of the actual routing protocol
- Two versions:
  + Dense Mode (PIM-DM)
  + Sparse Mode (PIM-SM)

Real-time transfer protocols

- Protocol family
  + Real-Time Transport Protocol (RTP)
  + Real-Time Control Protocol (RTCP)
  + Real-Time Streaming Protocol (RTSP)
- Suitable for general continuous media transport - not just multimedia

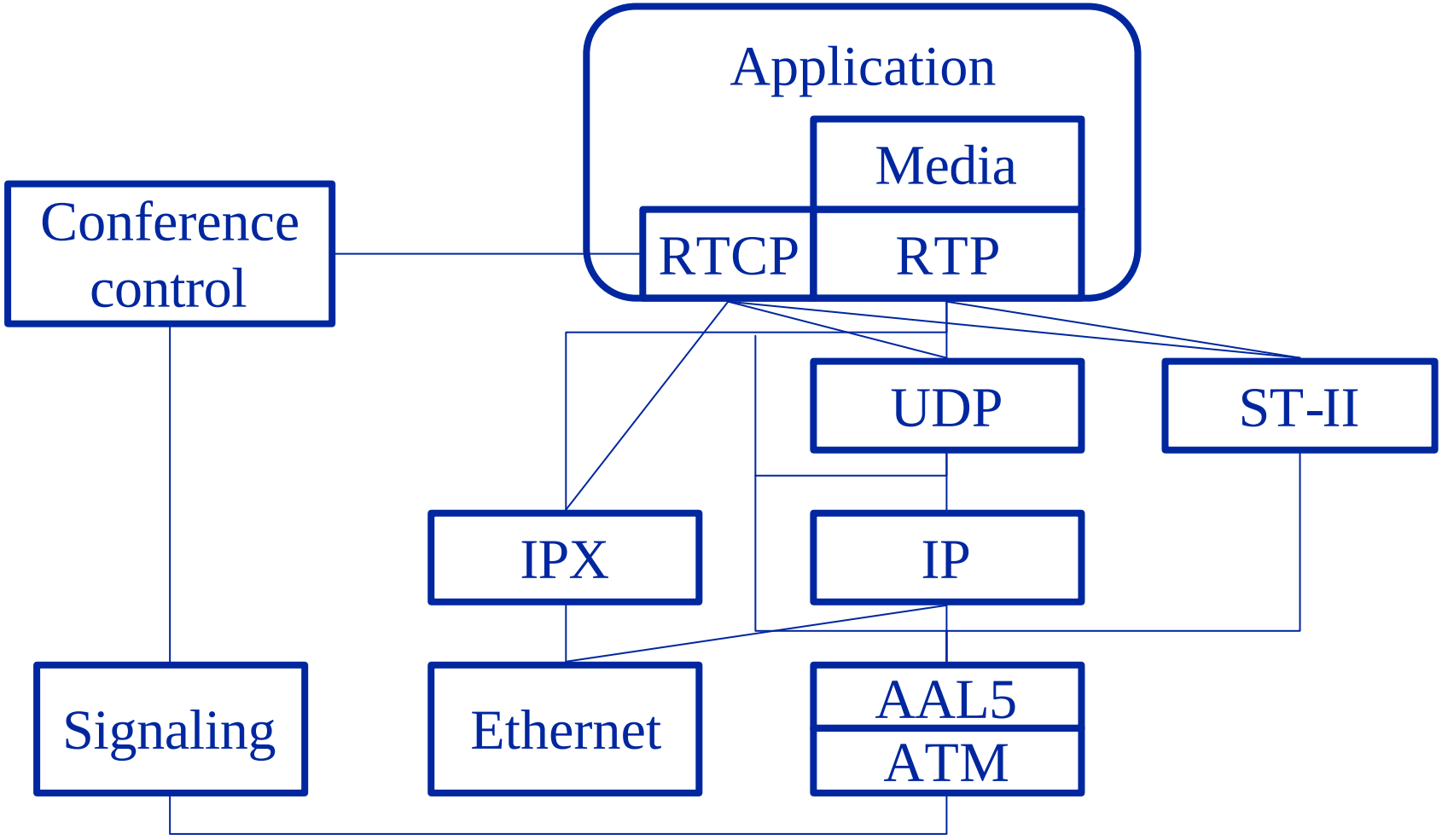
Relationships of the protocols



RTP

- Real-Time Transport Protocol (RTP)
  + sequences numbers of the packets
  + time stamps
  + identification of different payloads
- Operates usually on top of UDP
- Does not guarantee successful transport -> no QoS properties

RTP and other protocols



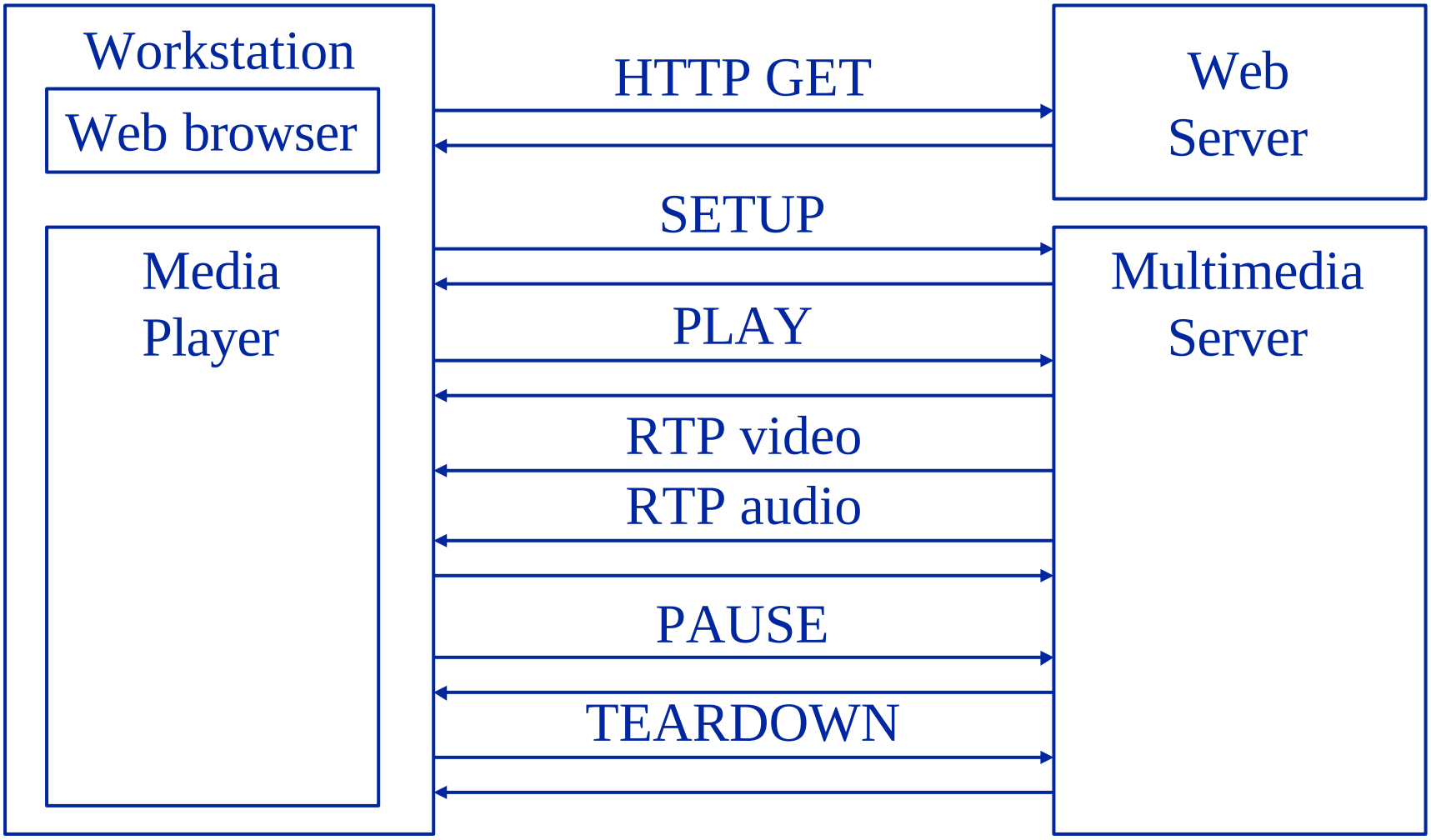
RTCP

- Real-Time Control Protocol (RTCP) controls RTP connections
- Functions:
  1. Transfers information about the RTP connection (e.g., QoS)
  2. Transfers information about source of the RTP connection
  3. Limits the amount of control information (5 %)
  4. Transfers information about the session

RTSP

- Real-Time Streaming Protocol (RTSP) builds and manages the real-time transport connections
- Works well with RTP and RTCP protocols
- Has similar functions to the HTTP protocol

RTSP - Operation



Resource reservation

- Real-time transfer protocols do not alone quarantine QoS of real-time traffic
- Required resources have to be reserved separately from all routes of the route
- For this purpose, there are special protocols
- Best known protocol is Resource ReSerVation Protocol (RSVP)

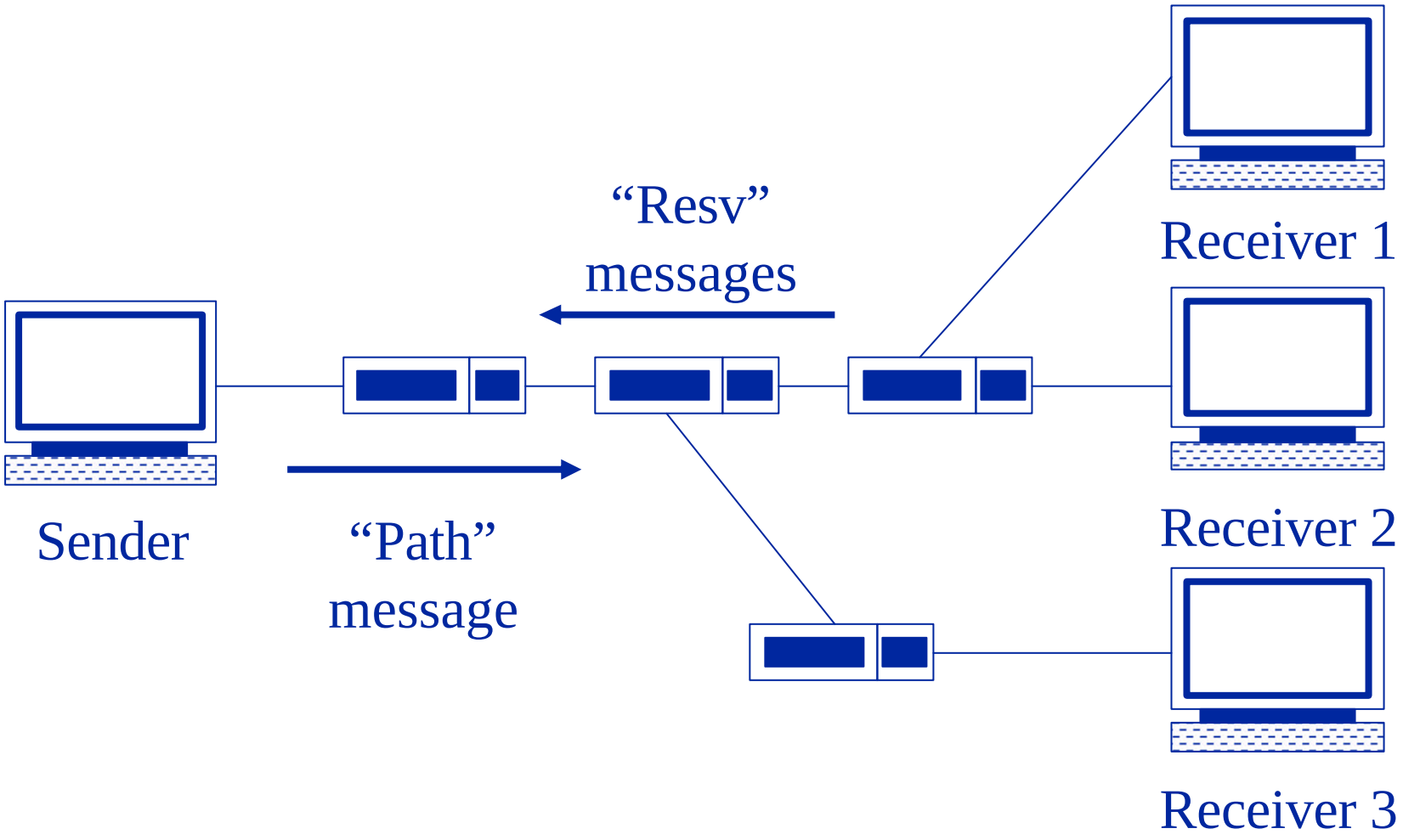
RSVP

- RSVP is based on announcements made by the receivers
- The sender sends first a “Path” message
- If necessary, the routers can send “PathErr” message

RSVP (cont.)

- Routers record the connections
  + Soft state
  + each connection has a *cleanup* and *restart* counter
- The receiver sends the “Resv” message
  + at the same time, the QoS requirements are defined
- The “Resv” messages go through the routers
  + the routers check the resources and make final reservations

RSVP messages



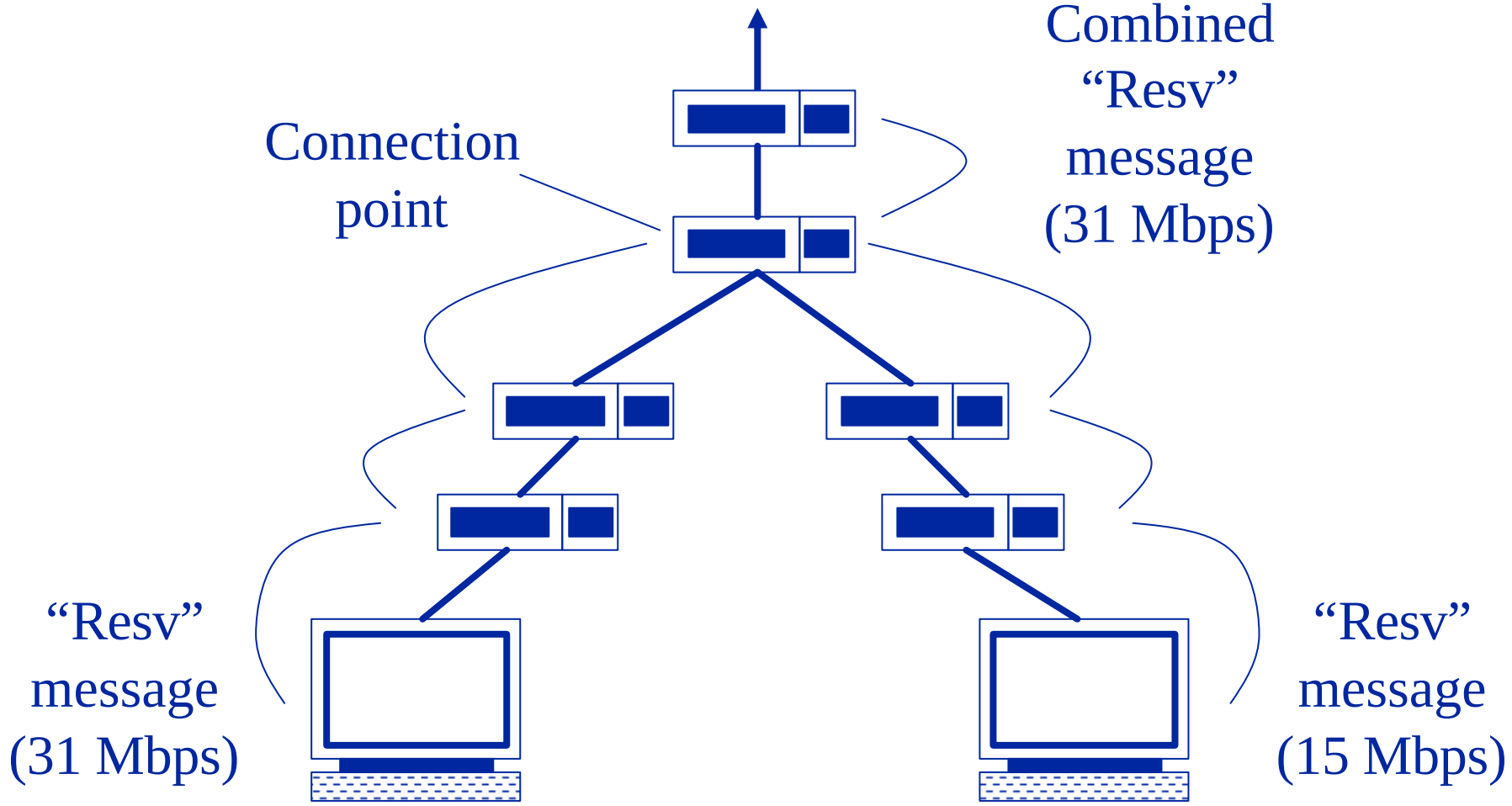
Soft state

- Each connection has to be recorded
- The information is outdated after certain time period
- That is why the state is called soft

QoS requests

- The receivers use “Resv” messages to ask for certain QoS parameters
- Each router checks whether there is enough resources
- If there is, then the connection is recorded (soft state)
- If necessary, the requests can be combined (multicast)

RSVP combination



RSVP status

- Not in wide use
- Problems
  + scaling of the method (control, connections, reestablishment of the connections)
  + good algorithm for checking resources does not exist
  + billing and bookkeeping difficult

Session management

- The available multicast sessions have to be advertised some how
- Thus directory services are needed
- Three protocols exist for this purpose: SDP, SAP, and SIP

SDP

- Session Description Protocol (SDP) distributes information about the available sessions and the attributes
- SDP is actually a format to announce the information
- Three parameter classes:
  + Session description
  + Time description
  + Media description

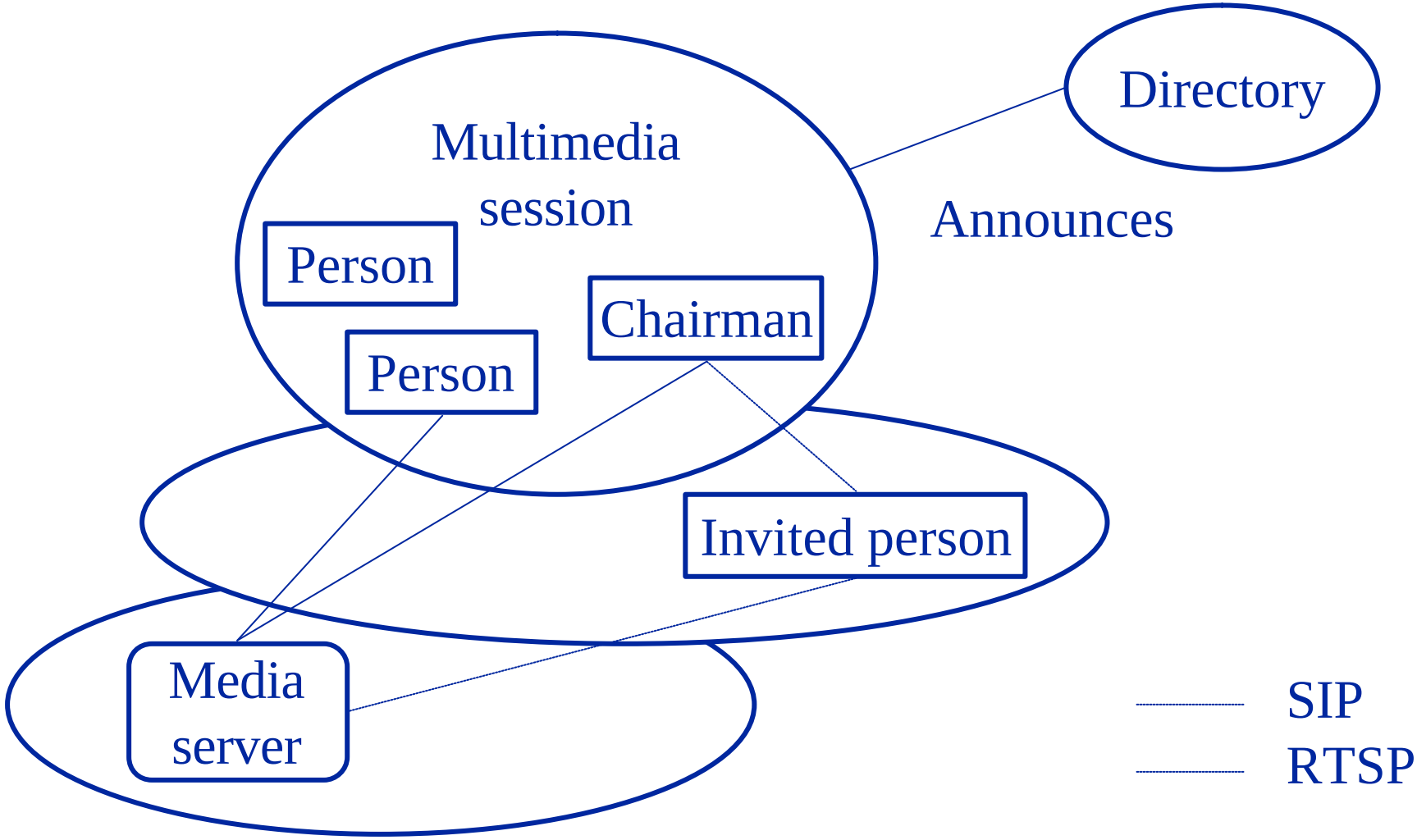
SAP

- Session Announcement Protocol (SAP) distributes the session descriptions to different directories
- Announcements are send as multicast transmissions
- Email lists and www-pages are used more often, though

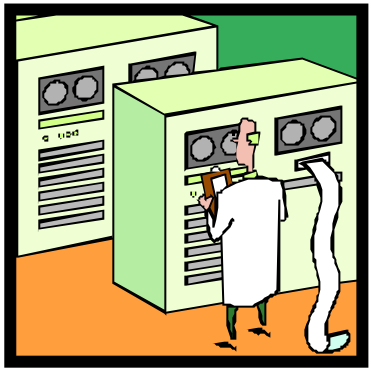
SIP

- Session Initiation Protocol (SIP) can be used, when certain participant have to be invited to the multicast session
- The participants can be persons or “robots”
- Robots are video-on-demand servers, video cameras, etc.
- SIP can use directory services when searching persons

Robots


Directory
Multimedia session
Announces
Person
Person
Chairman
Invited person
Media server
SIP
RTSP

Person search



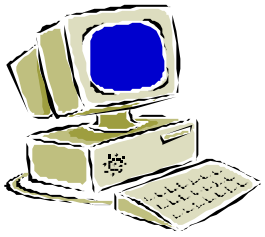
cz@cs.tu-berlin-de

2. henning

3. hgs@play

hgs@play

1. Call henning@ cs.columbia.edu



4. Call hgs@play

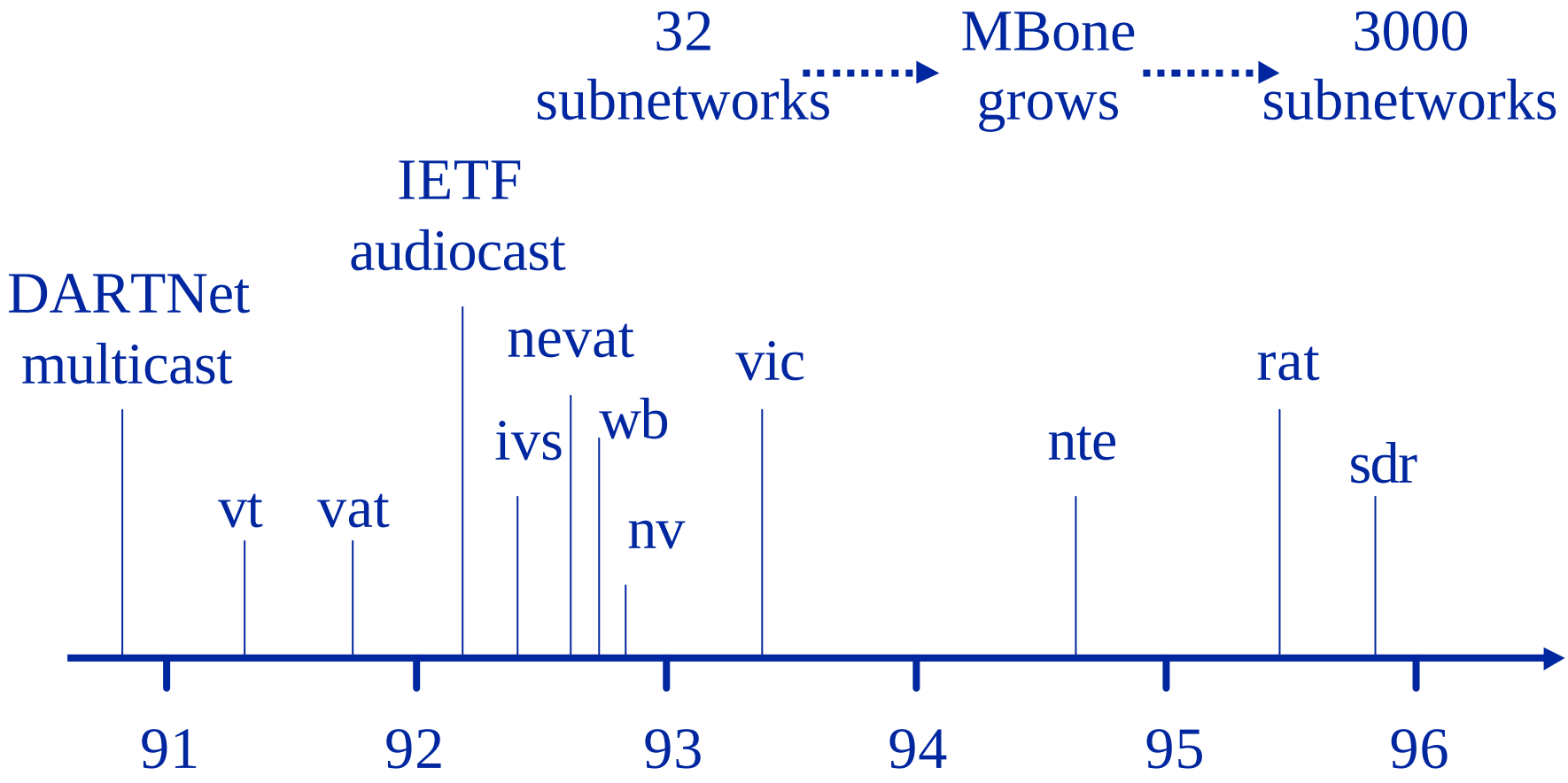
6. 200 OK

5. 200 OK

MBone

- Originally developed in research project
  + University of Southern California's Information Sciences Institute
  + Massachusetts Institute of Technology
  + Xerox Palo Alto Research Center
  + Lawrence Berkeley National Laboratory
- DARPA Research Test bed, DARTNET -90
  + Unix workstation, T1 connections

MBone - development



Mbone archives

- Merit Networks
  + www.merit.edu/~mbone/index/titles.html
- Henning Schulzrinne
  + www.cs.columbia.edu/~hgs/rtp/
- Mbone FAQ
  + www.cs.columbia.edu/~hgs/internet/mbone-faq.html