Internet Protocol: Forwarding IP Datagrams

- Basic unit of transfer in TCP/IP internet is the datagram
  - **Datagram format characterizes the static aspects of IP**
  - **Forwarding characterizes the operational aspects**

Forwarding In An Internet

- Forwarding refers to choosing a path for packets
  - **Term *routing* was previously used**
  - **Popular term now is *forwarding***
- Router is the computer making the choice
- WAN
  - **Has multiple physical connections between packet switches**
  - **Network itself routes packets while in net**
  - **Internal routing; self-contained in the WAN**

- Goal of IP: Virtual network
- Focus on IP forwarding
  - **Also called internet routing or IP routing**
  - **Information used to make decisions called IP forwarding information**
- Similar to forwarding in single physical network:
  - **IP forwarding chooses path for datagram**
- Difference
  - **Chooses routes across multiple physical networks**

- Forwarding can be difficult
  - Ideally, should be based on:
    - Network load
    - Datagram length
    - Type of service specified
  - Most software is less sophisticated
    - Selects on fixed assumptions about shortest paths
- Both hosts and routers participate
  - Host may make initial forwarding decision if has access to multiple routers (even if singly-homed)
  - Also, multi-homed hosts & general machines
  - For now, treat hosts and routers separate

Direct and Indirect Delivery

- Can divide forwarding into two forms:
  - Direct Delivery
    - Datagram from one machine to another
    - Across single physical network
    - Both must be attached to same physical network
  - Indirect Delivery
    - Destination not on a directly attached network
    - Router must become involved

- Delivery over a single network
  - IP datagram encapsulated into frame
  - IP address mapped to physical address
  - Network hardware delivers the datagram
  - To tell if destination directly connected:
    - Easy based on prefix and suffix of IP address
    - Sender extracts network prefix of destination
    - If same as its network portion: directly connected
  - Special case of general purpose forwarding
    - Final router always directly connected to destination

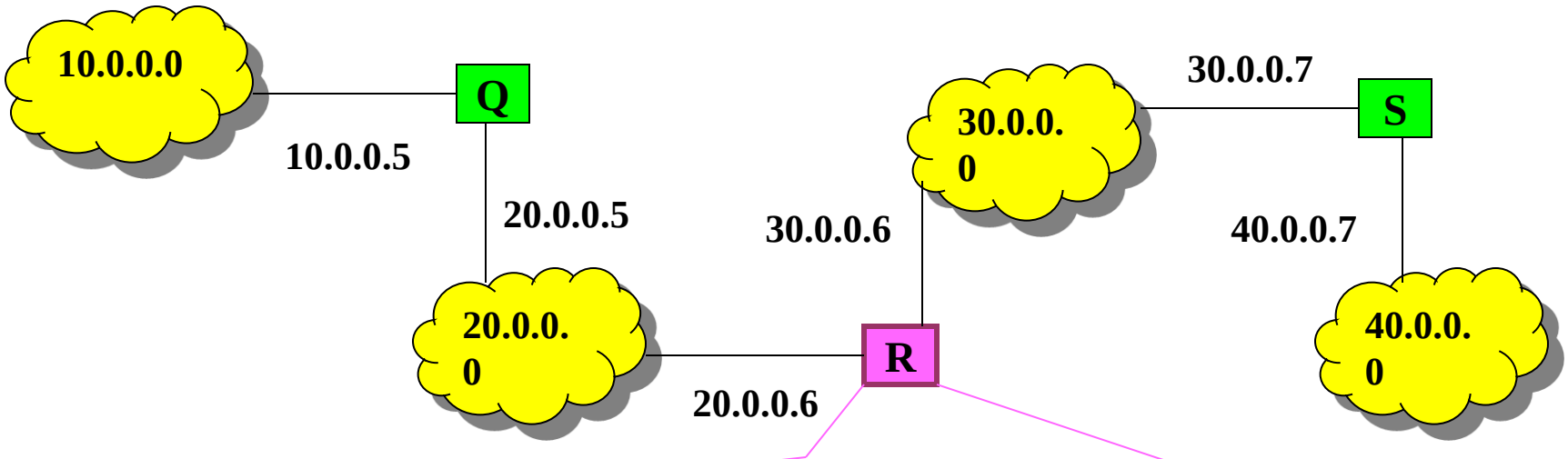
- Indirect delivery
  - Sender must specify a router
  - Router must then forward toward destination
  - Datagram passes from router to router until it reaches a router directly connected to destination
  - Questions:
    - How does a router know where to send a datagram?
    - How can a host know which router to use?
  - Consider basic table-driven forwarding now
    - Later talk about routers learning new routes

Table-Driven IP Routing

- Uses IP routing table
  - **Info about destinations & how to reach them**
  - **Both hosts and routers have**
- What info to keep?
  - **Can't do all possible destinations**
    - **Couldn't keep current; too much storage space**
  - **IP address scheme helps**
    - **Direct delivery test is efficient**
    - **Routing tables only need network prefixes**

Next-Hop Forwarding

- Routing table has pairs (*N*,*R*)
  - *N* is the IP address of a destination *network*
  - *R* is the IP address of the "next" router along the path to *N*
    - *R* is the next hop
    - Using this table is next-hop forwarding
    - Don't know whole path, only one step
  - All routers in machine *M*'s table must be directly connected to *M*



To reach hosts on network	Route to this address
20.0.0.0	Deliver Directly
30.0.0.0	Deliver Directly
10.0.0.0	20.0.0.5
40.0.0.0	30.0.0.7

- Routing table size depends on number of networks
  - **Size & contents independent of number of hosts**
- Consequences of choosing routes based on destination network ID alone:
  - Traffic for same network usually take same path
    - **Multiple paths may not be used concurrently**
    - **All traffic types follow same path regardless of delay or throughput**
  - Only final router knows if hosts exists or is operational
    - **Must report delivery problems**
  - Traffic from A–B may have different path than B–A

Default Routes

- If no route in table, use default route
  - **Keeps tables small**
  - **Consolidates multiple entries into default case**
  - **Useful when only one connection to internet**
    - **Test if local net destination**
    - **If not, only can go through one router anyway**

Host-Specific Routes

- Forwarding based on networks, not individual hosts
- IP forwarding software allows per-host routes
  - **Gives local network admin more control over network use**
  - **Permits testing**
  - **Controls access for security purposes**

IP Forwarding Algorithm

ForwardDatagram(Datagram,RoutingTable)

Extract destination IP address, D from datagram;
If the table contains a host-specific route for D
    send datagram to next-hop specified in the table and quit;
Compute N, the network prefix of address D;
If N matches any directly connected network address
    deliver datagram to destination D over that network;
    (Involves resolving D to a physical address, encapsulating the datagram, and sending the frame.)
Else if the table contains a route for network N
    send datagram to next-hop specified in table;
Else if the table contains a default route
    send datagram to the default router specified in table;
Else declare a forwarding error

Forwarding With IP Addresses

- IP forwarding does not alter original datagram
  - **Except for decrementing TTL and recomputing the checksum**
  - **Source and destination addresses unchanged**
  - **Router must pick next-hop IP address**
    - **Where does this address get stored?**
    - **There is no place in the datagram for it**

- Does not store the next-hop address at all
  - **IP passes datagram & next-hop address to the network interface**
  - Network interface software:
    - **Binds next-hop address to physical address**
    - **Forms a frame using the physical address**
    - **Places datagram in data portion of frame**
    - **Sends the result**
    - **Discards next-hop address**

- Why not use physical addresses when storing and computing routes?
  - Routing table provides clean interface between IP forwarding SW and high-level SW
    - Look at routing table to debug forwarding problems
    - IP addresses makes it easy to do
  - Point of IP is to build abstraction
    - Communication software can be written to use internet addresses
    - Only a few low-level routines need to know and interface with the physical addresses

Handling Incoming Datagrams

- When datagram arrives at a *host*:
  - **Network interface SW delivers incoming datagrams to IP module**
    - If destination address matches host's:
      - IP software accepts the datagram
      - Passes it on to higher-level protocol software
    - If destination address does not match host's:
      - Datagram must be discarded
      - Hosts forbidden from trying to fix routing problem

- When datagram arrives at a *router*:
  - **Delivered to IP module**
    - If destination IP address matches the routers:
      - Datagram passed to higher-level protocol software
      - Usually destined for router if testing or sending commands
    - If datagram is not at final destination:
      - TTL field is decremented
        - If TTL = 0, datagram is discarded
        - If TTL > 0, computes new checksum
      - IP forwards the datagram using the forwarding algorithm

- Determining if at final destination is not trivial
  - Host may have multiple physical connections
    - Destination IP address must be compared with that of each connection
  - Also, may be a broadcast datagram
    - Have to see if IP address matches the limited or directed broadcast IP address
  - Classless, subnet, and multicast addresses make it even more complex
    - Will see in later chapters….

- Why forbid hosts from forwarding functions?
  - Host receiving datagram not for it indicates a problem
    - Won't be revealed if host takes corrective action
  - Forwarding steals time from legitimate uses of the host
  - Simple errors can cause chaos
    - Some host accidentally broadcasts datagram for a host
    - Every local host receive a copy; all send to recipient
  - Routers do more than merely route traffic
    - Report errors
    - Propagate forwarding information

Establishing Routing Tables

- How do systems initialize routing tables?
- How are routing tables updated?
  - **Later chapters discuss the protocols that do this**
  - For now:
    - **Understand IP software uses routing tables to decide how to forward a datagram**
    - **Changing routing tables will change the paths**

Summary

- IP software forwards datagrams
  - Decides where to send based on destination IP address
- Direct delivery used if the destination machine is on the same network as the sender
  - **Otherwise, sender must go through a router**
  - **Datagrams travel from router to router until they can be delivered directly**
- IP software produces IP address of the next hop
  - **Network interface SW encapsulates datagram, maps next-hop address to physical address, and sends**

- Internet forwarding algorithm:
  - **Is table-driven**
  - **Uses only IP addresses**
- Possible to have host-specific destination addresses
  - **Most routing tables only have network addresses**
- Using default routes helps keep tables small
  - **Especially for hosts that can access only one router**