

# **CS423: Lecture 21, Routing: Addressing and Neighbor Discovery**

**George Varghese**

November 11, 1997

## **Routing (4 to 5 lectures)**

- Connectionless (Post Office model: IP, OSI)
  - Addressing
  - Finding Neighbors
  - Finding Routes. Distance Vector (e.g., IP's RIP) and Link State (OSI's LSP Protocol)
- Connection Oriented (Telephone Model: X.25, ATM) Networks and Virtual Circuits.

## Connectionless Routing

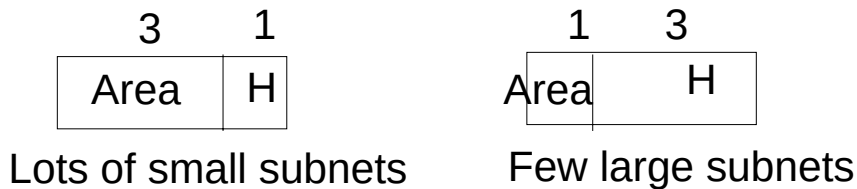
- 2 main problems: finding out routes and forwarding packets along the computed routes.
- However, the structure of routing addresses deeply influences routing (telephone numbers versus social security numbers), and so we begin with them.
- Also in the problem of computing routes, we separate out the problem of finding out who one's direct neighbors are. Allows problem to have 2 parts: i) find neighbors ii) Compute routes to neighbors.

## Endnodes versus Routers

- **Endnodes** are hosts, user workstations. Large numbers (e.g., 150,000), dedicated to computing. Thus endnode routing protocol should be simple.
- *Routers* (IMPs, Gateways). Smaller numbers (e.g., 200). Dedicated to communication. Makes sense for them to bear major load of routing calculation

Thus makes sense to have endnode routing, forwarding, and neighbor discovery to be as simple as possible.

## Addressing



- The most important thing about addresses is hierarchy. If addresses are hierarchical, then the routing protocol can be hierarchical.
- How many levels of hierarchy? How many bytes in each level address. Unlike telephones, we want addressing structure that fits a number of different networks and needs. Need *flexible hierarchies*. IP offers some flexibility. OSI offers more.

## IP addresses

|   |     |      |  |
|---|-----|------|--|
| 1 |     | 3    |  |
| 0 | Net | Host |  |

Class A

|    |     |      |  |
|----|-----|------|--|
| 2  |     | 2    |  |
| 10 | Net | Host |  |

Class B

|     |     |      |  |
|-----|-----|------|--|
| 3   |     | 1    |  |
| 110 | Net | Host |  |

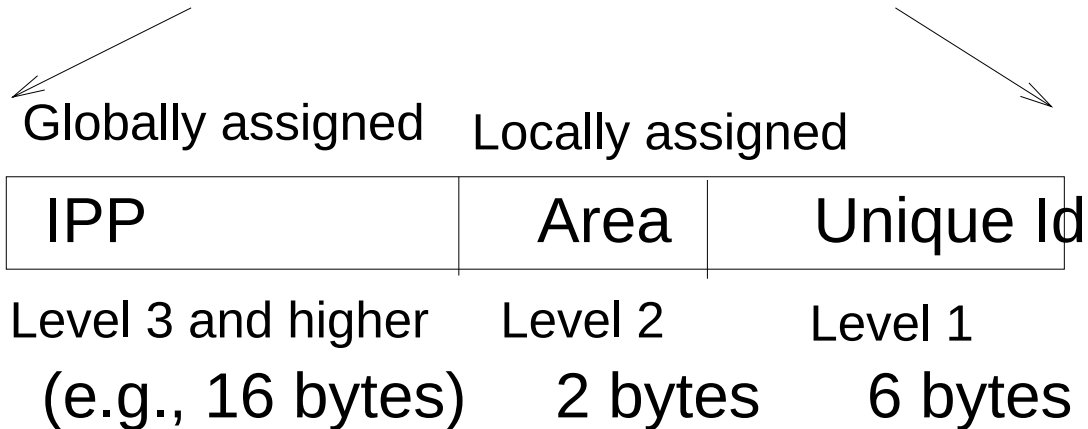
Class C

- In IP every internetwork gets a net number. An internetwork could be a full-scale network (e.g., ARPANET) or a LAN. All LANs have separate net numbers.
- Class A,B, C provide moveable boundary between host and net numbers. Allows 3 positions of partition. Generalize via masks.

## Names and addresses

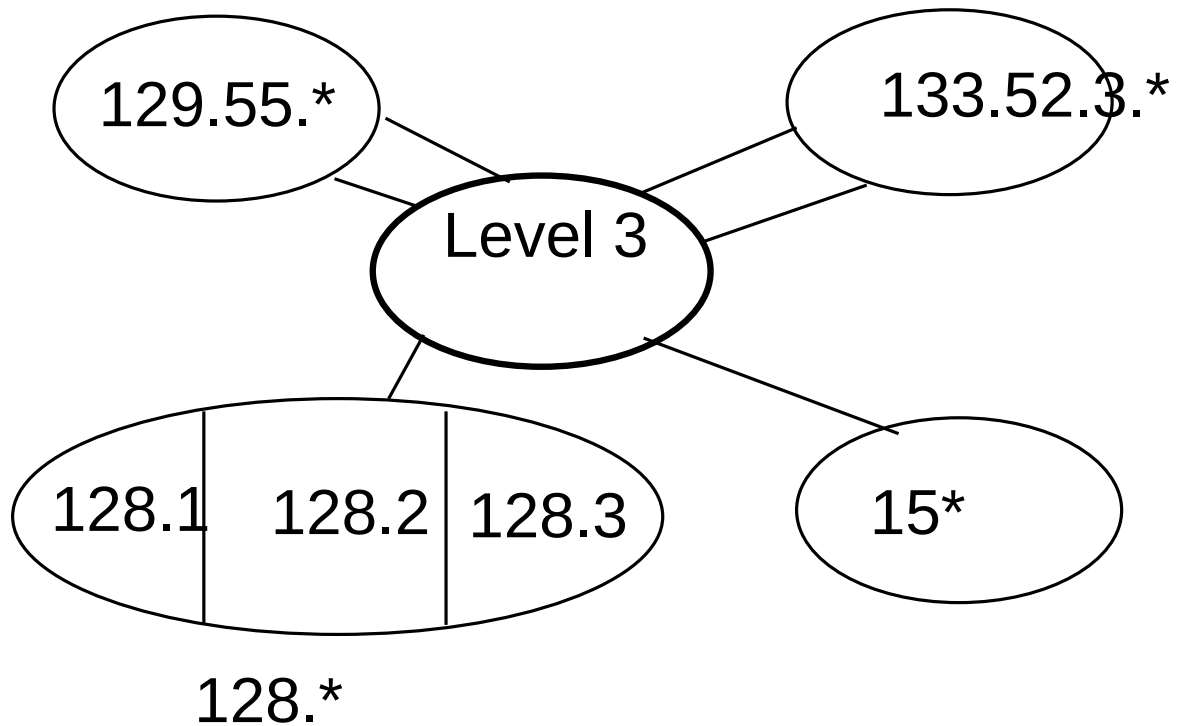
- When you want to connect to a host you tend to know the user-friendly host name (e.g., dworkin).
- Your local host translates this into a host ID by consulting a telephone book (etc/hosts) or by asking an operator (domain name service).

## OSI addresses



- 3 or more levels of hierarchy, variable length IPP.  
Quite flexible. Too flexible??
- Third level of hierarchy (IPP part) was intended to be telephone numbers which have their own levels of hierarchy.

## Large OSI Networks



- Level 3 subnetwork could be the telephone network. Each Level 2 network could have one or more telephone numbers (919.202.3132). Thus the Level 3 address could have more levels of hierarchy in it.
- In each Level 2 network, a local authority could divide it into Level 1 networks using the 2 byte Area field.

*NEIGHBOR GREETING*  
*(Step 1 and Step 3 for Endnodes)*

1) ROUTERS NEED TO FIND NEIGHBORS: ROUTING

-- ROUTING ADDRESSES FOR ROUTE COMPUTATION

-- DATA LINK ADDRESS FOR LAST HOP ON LANS.

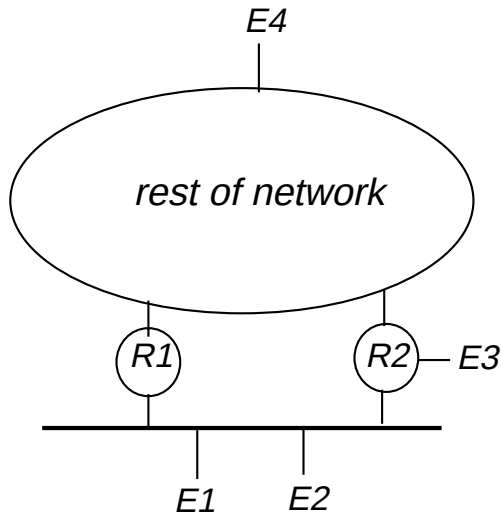
2) ENDNODE ROUTING: (must be simple)

-- NEED TO FIND A ROUTER

-- NEED TO AVOID SUBOPTIMAL ROUTES.

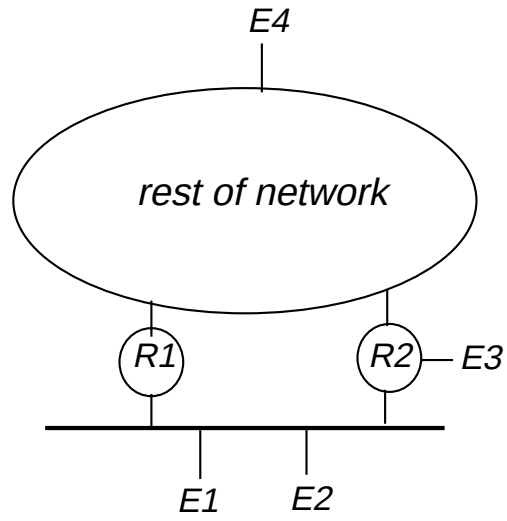
-- NEED TO ROUTE TO LAN DESTINATIONS  
IN ABSENCE OF ROUTER.

## 6 Problems for LAN Endnodes



- Routers need network addresses of endnodes
- Routers need Data Link addresses of endnodes
- Endnodes need Data Link address of at least one router
- E1 to E2 traffic should not have to go through a router
- E1 to E3 traffic should go through R2 eventually.
- If R1, R2 are down, E1 and E2 should be able to communicate.

# OSI Solution



# IP Solution

