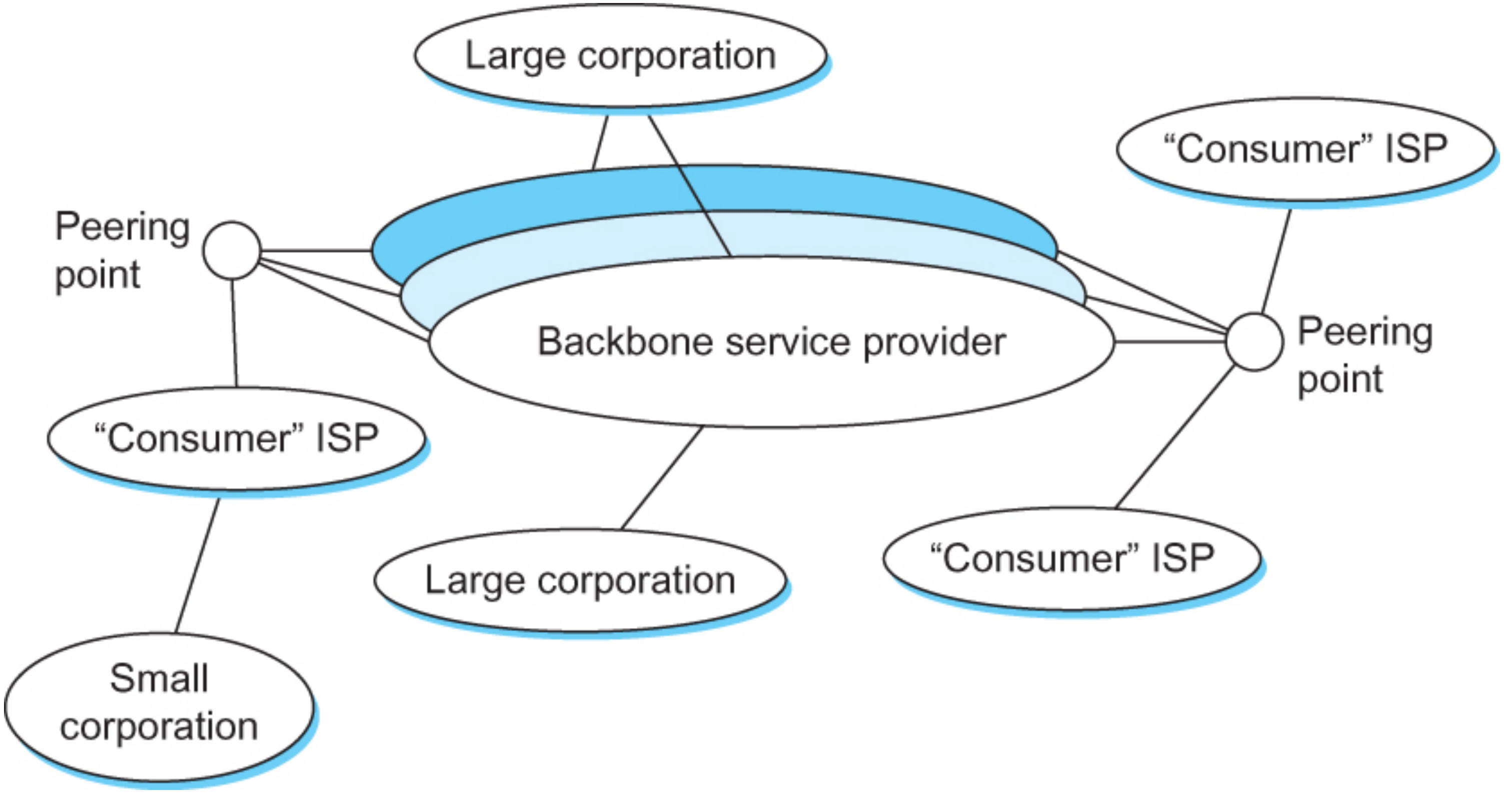
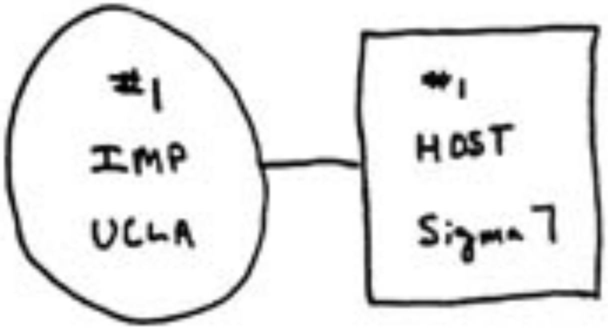
Scaling up routing



Overview

- Last time:
  - Intradomain routing
    - Distance-vector (RIP, EIGRP)
    - Link-state (OSPF)
- This time:
  - Scaling to a global network
  - Interdomain routing
    - Path-vector

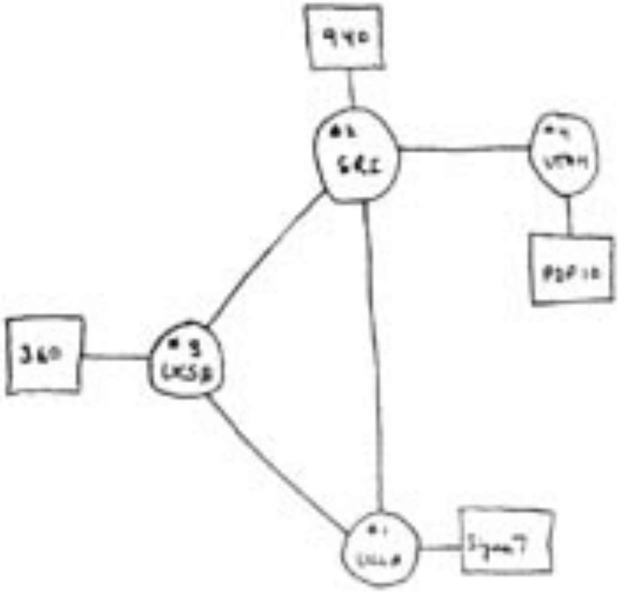
The Internet: 1969



THE ARPA NETWORK

SEPT 1969

1 NODE

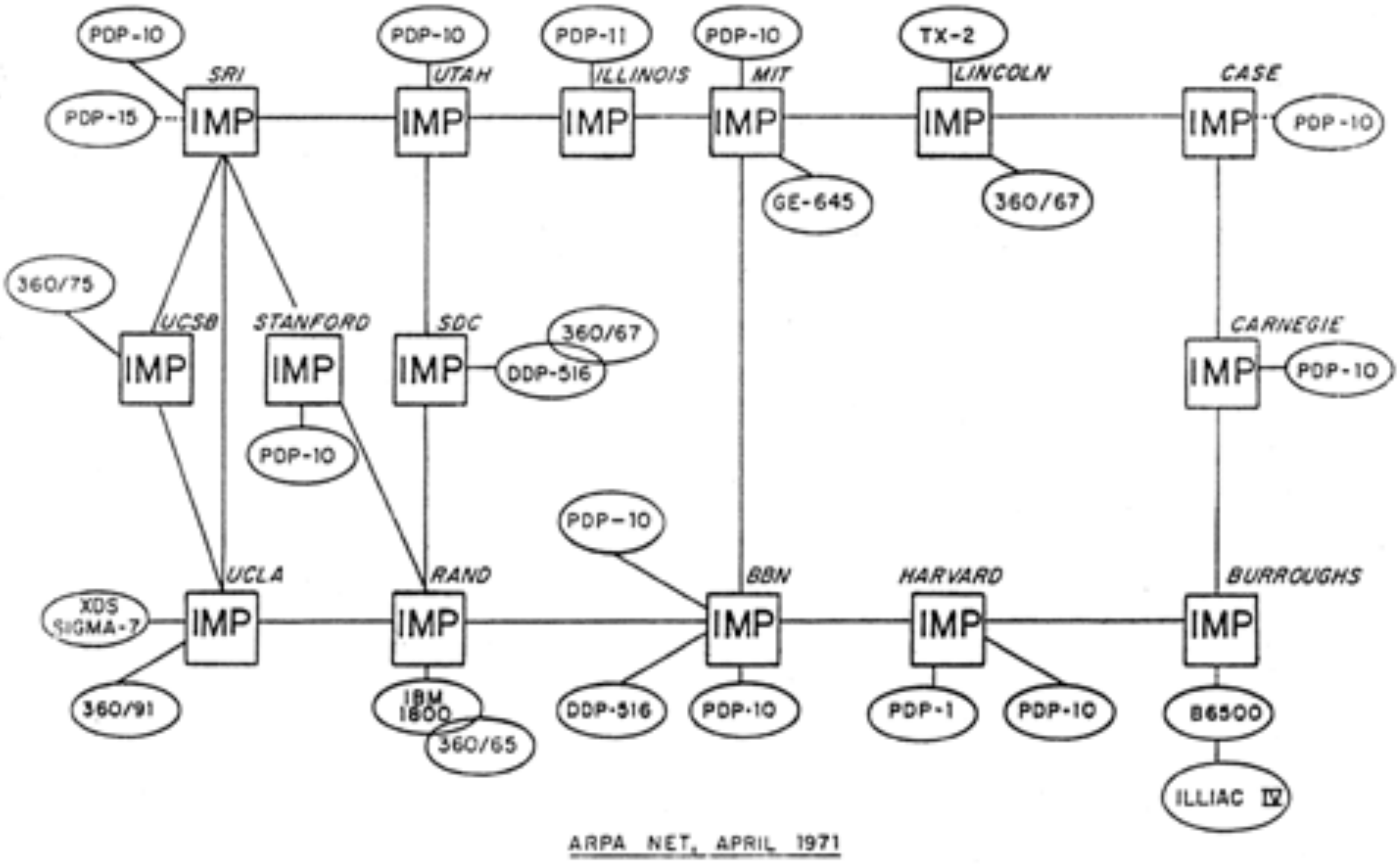


THE ARPA NETWORK

DEC 1969

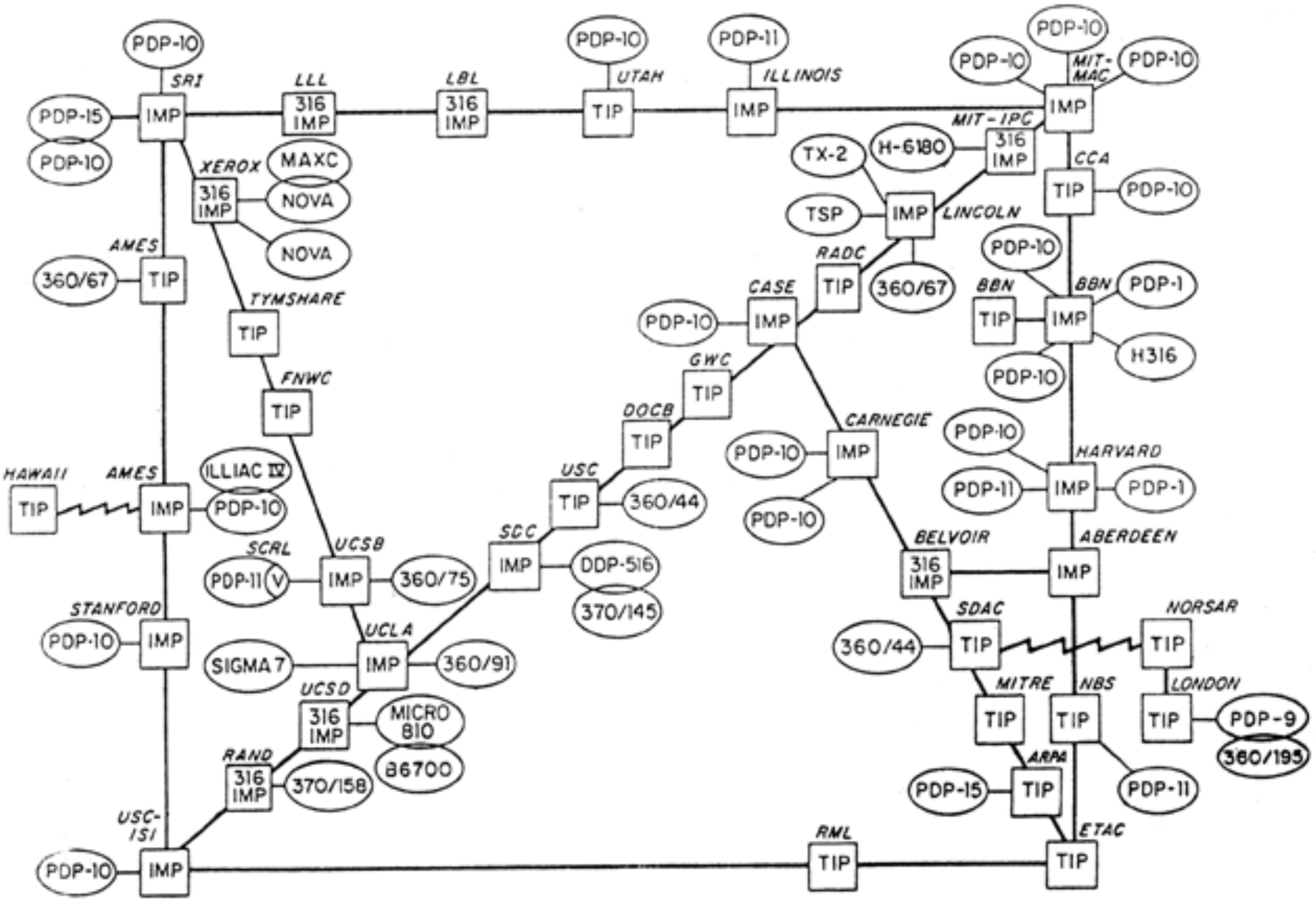
4 NODES

The Internet: 1971

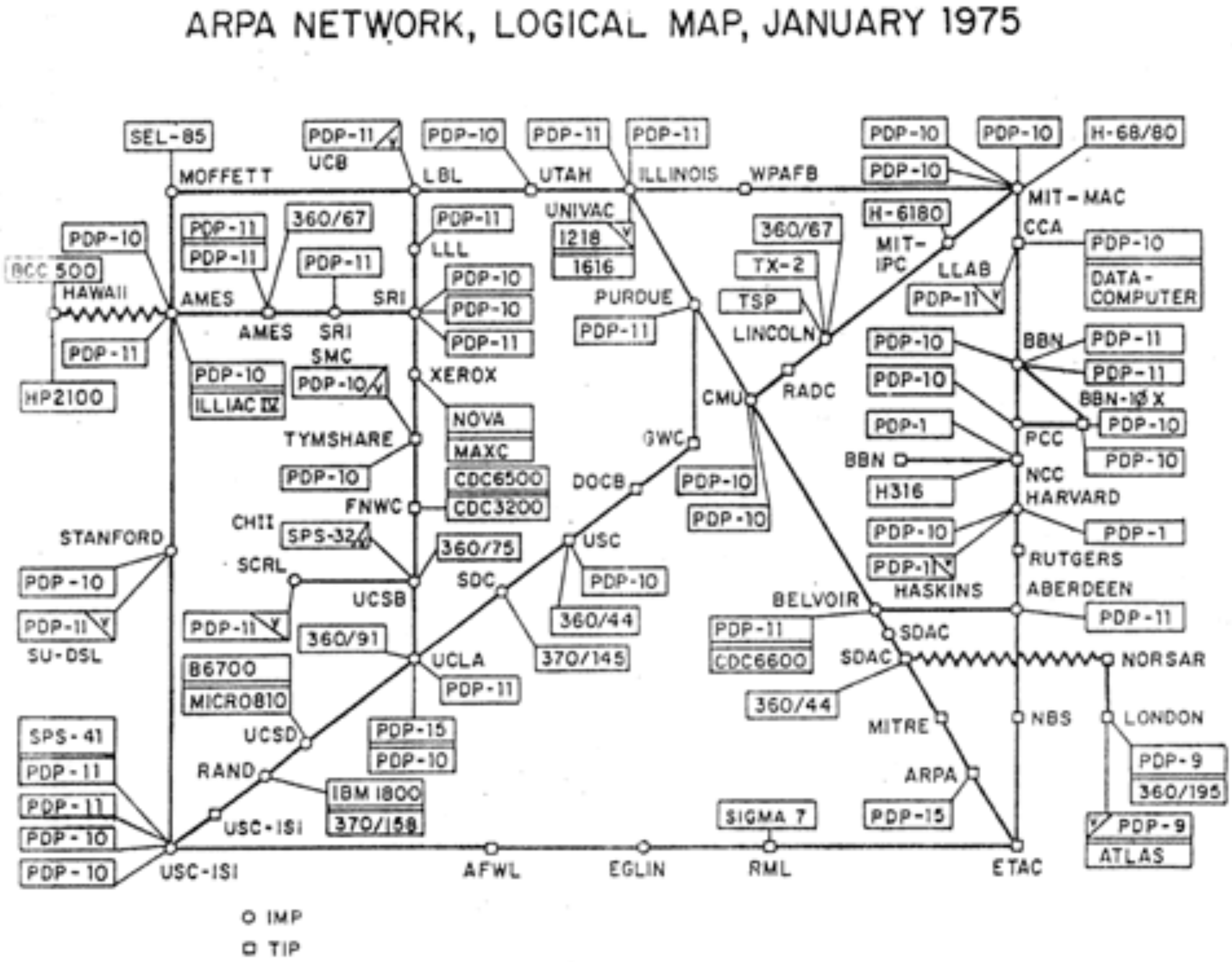


The Internet: 1973

ARPA NETWORK, LOGICAL MAP, SEPTEMBER 1973

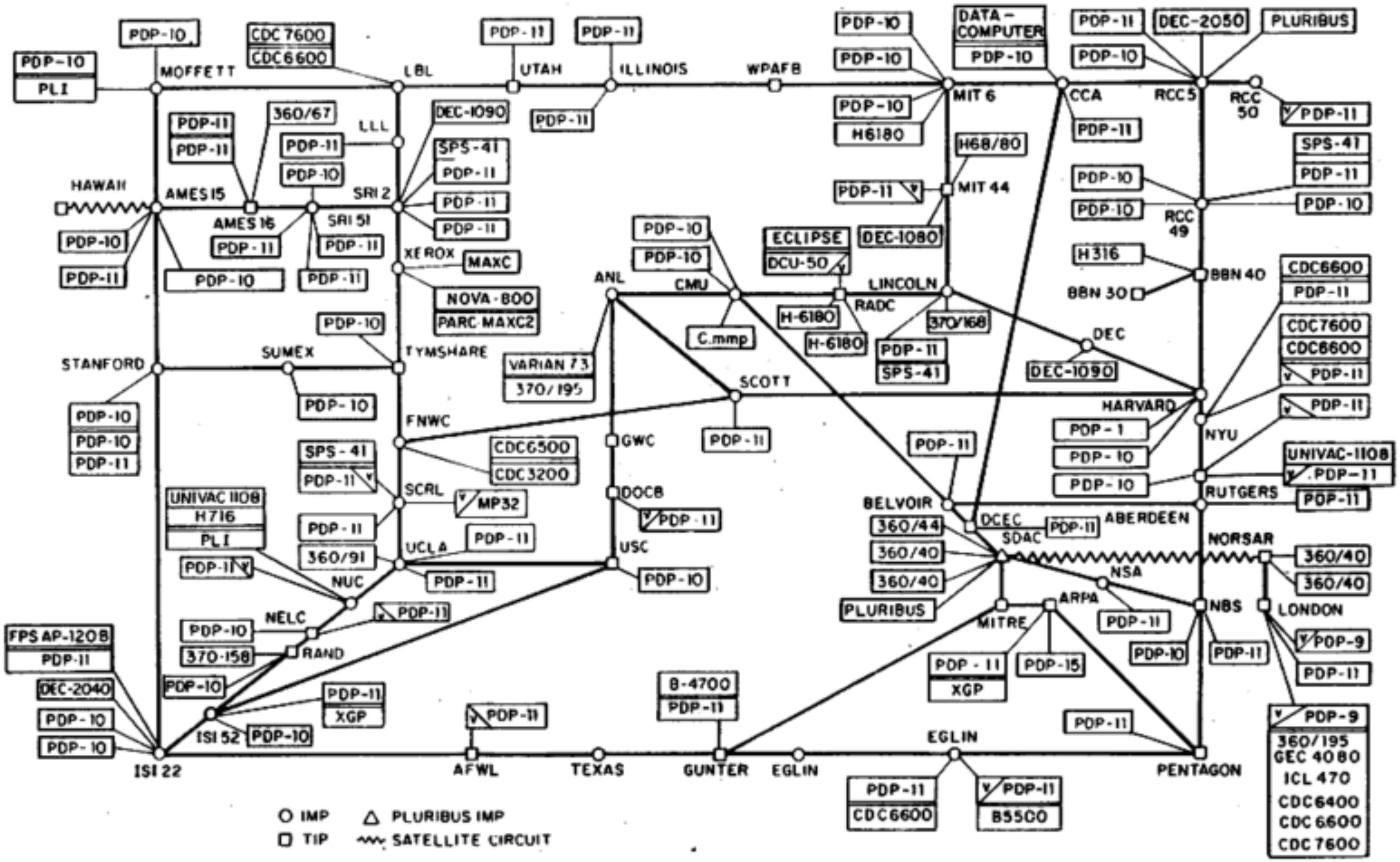


The Internet: 1975



The Internet: 1977

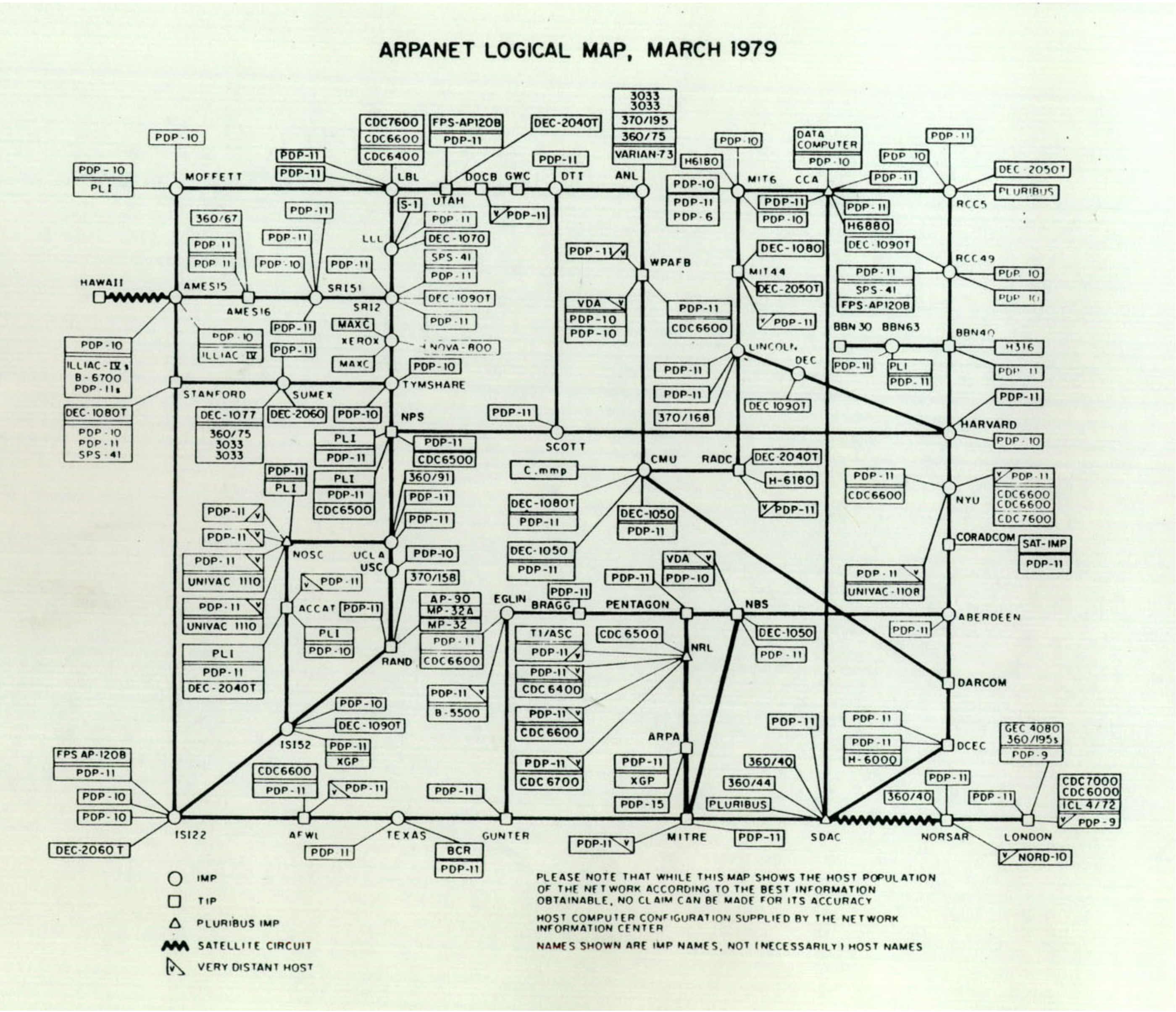
ARPANET LOGICAL MAP, MARCH 1977



(PLEASE NOTE THAT WHILE THIS MAP SHOWS THE HOST POPULATION OF THE NETWORK ACCORDING TO THE BEST INFORMATION OBTAINABLE, NO CLAIM CAN BE MADE FOR ITS ACCURACY)

NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

The Internet: 1979


ARPANET LOGICAL MAP, MARCH 1979

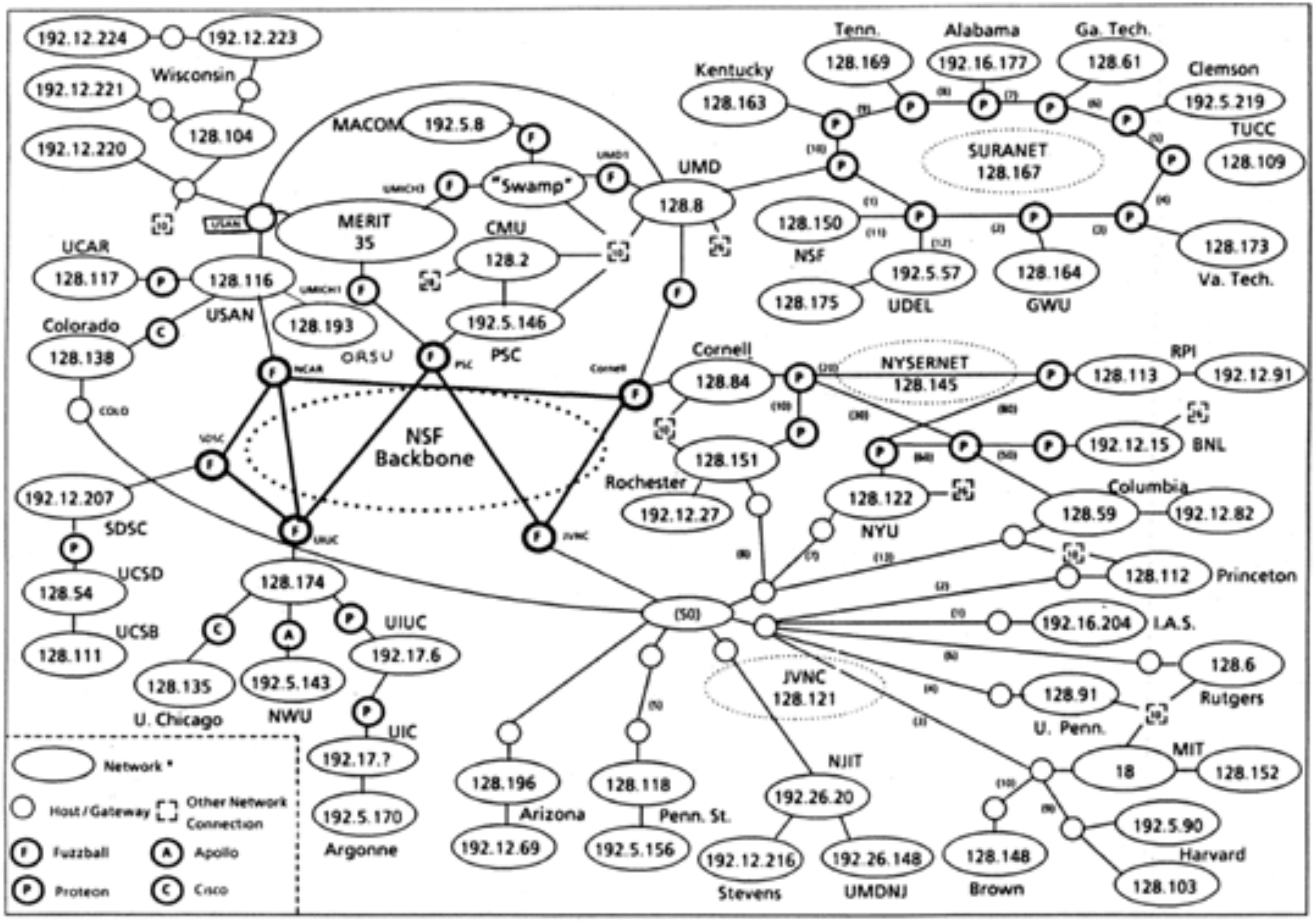
PLEASE NOTE THAT WHILE THIS MAP SHOWS THE HOST POPULATION OF THE NETWORK ACCORDING TO THE BEST INFORMATION OBTAINABLE, NO CLAIM CAN BE MADE FOR ITS ACCURACY

HOST COMPUTER CONFIGURATION SUPPLIED BY THE NETWORK INFORMATION CENTER

NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

The Internet: 1987

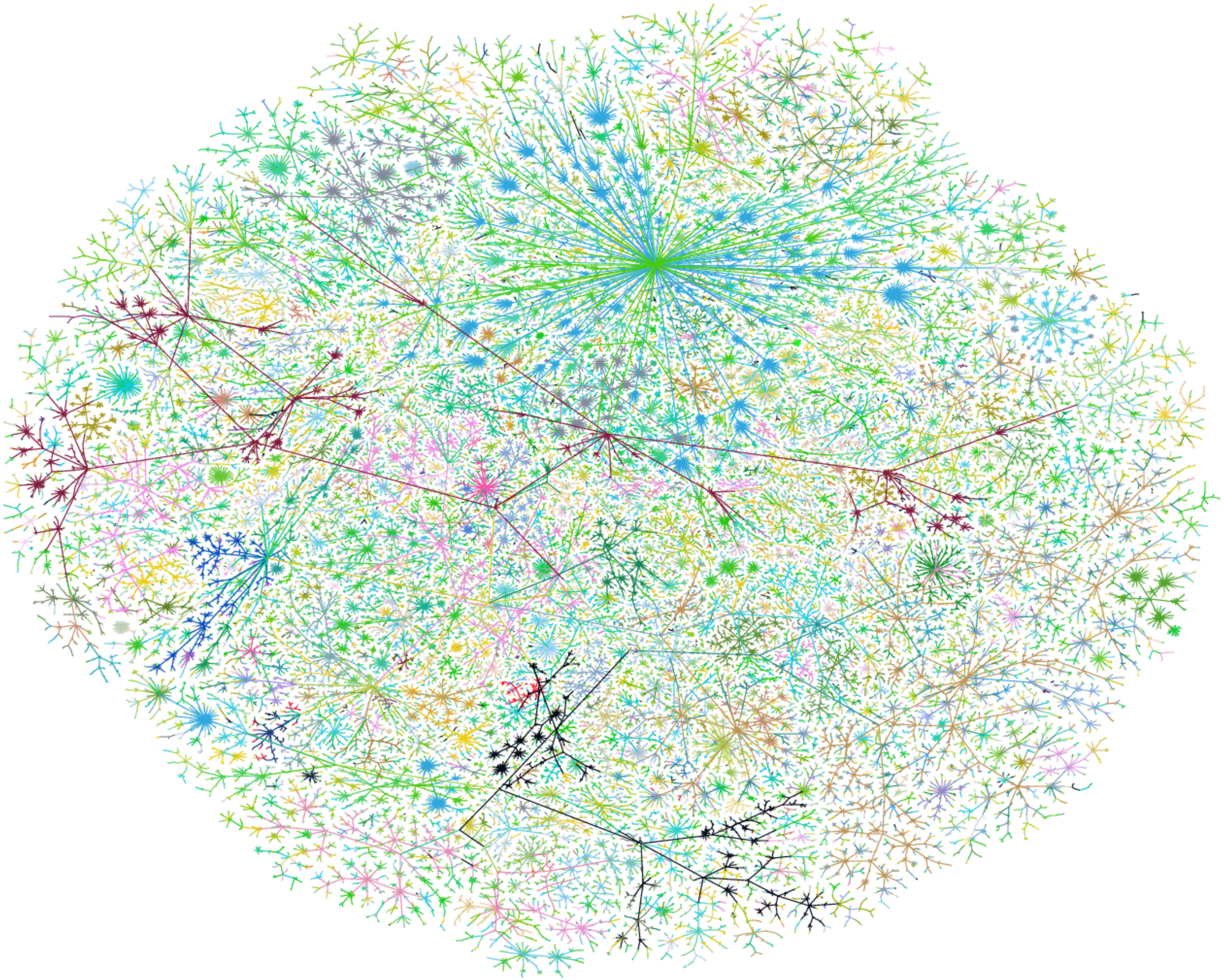
NSFNet Physical Connectivity -- April 87



* For some networks internal structure (e.g. subnets) is suppressed.

ISI 4/17/87

The Internet: 1999

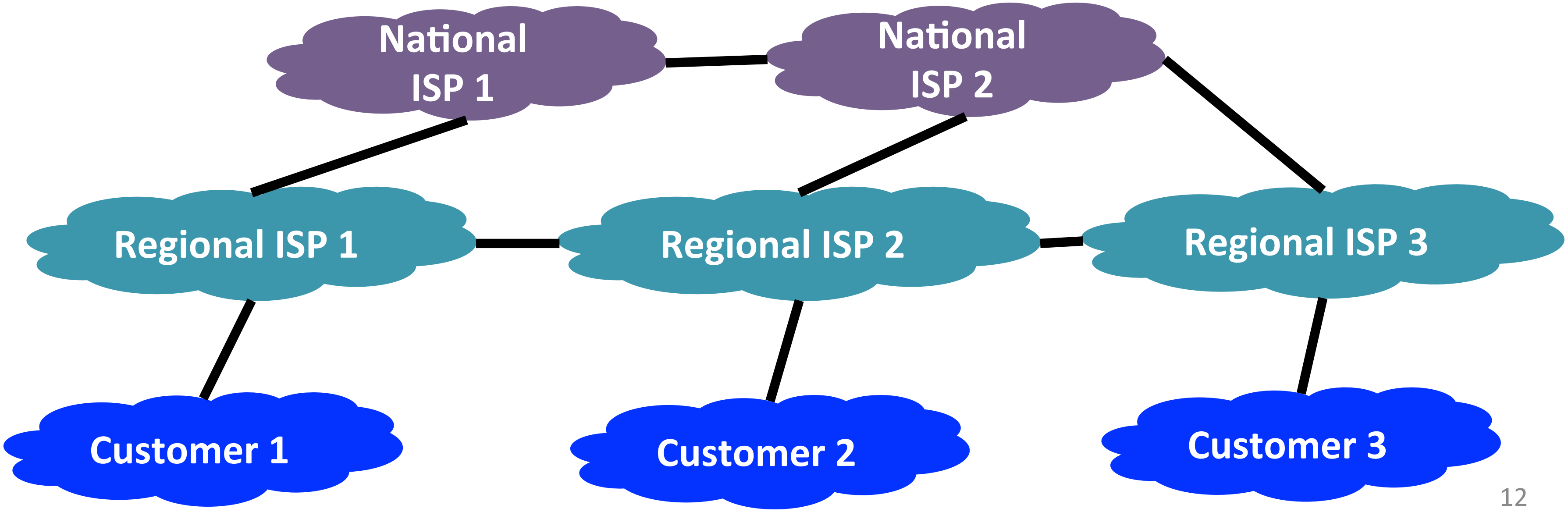


Distance-vector vs. Link-state

Distance-vector	Link-state
Knowledge of neighbors' distance to destinations	Knowledge of every router's links (entire network graph)
Router has O(# neighbors * # nodes)	Router has O(# edges)
Messages only between neighbors	Messages between all nodes
Trust a peer's routing computation	Trust a peer's info Do routing yourself
Bellman-Ford algorithm	Dijkstra's algorithm
Enhanced Interior Gateway Routing Protocol (EIGRP) Proprietary Cisco protocol	Open Shortest Path First (OSPF) Open protocol standard
Advantages: Less info has to be stored Lower computation overhead	**Advantages:** Fast to react to changes

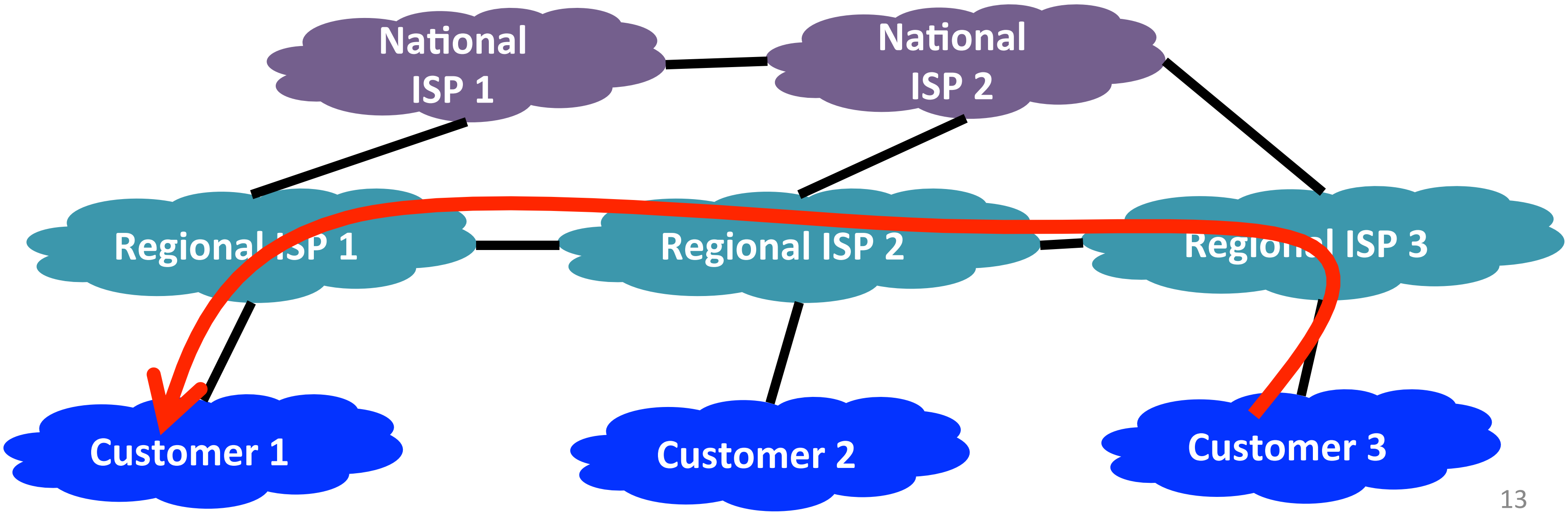
Shortest path routing

- Problems with always taking shortest path:
  - All traffic must travel on shortest path
  - All nodes must do same link cost calculation
  - Not possible to enforce various business rules



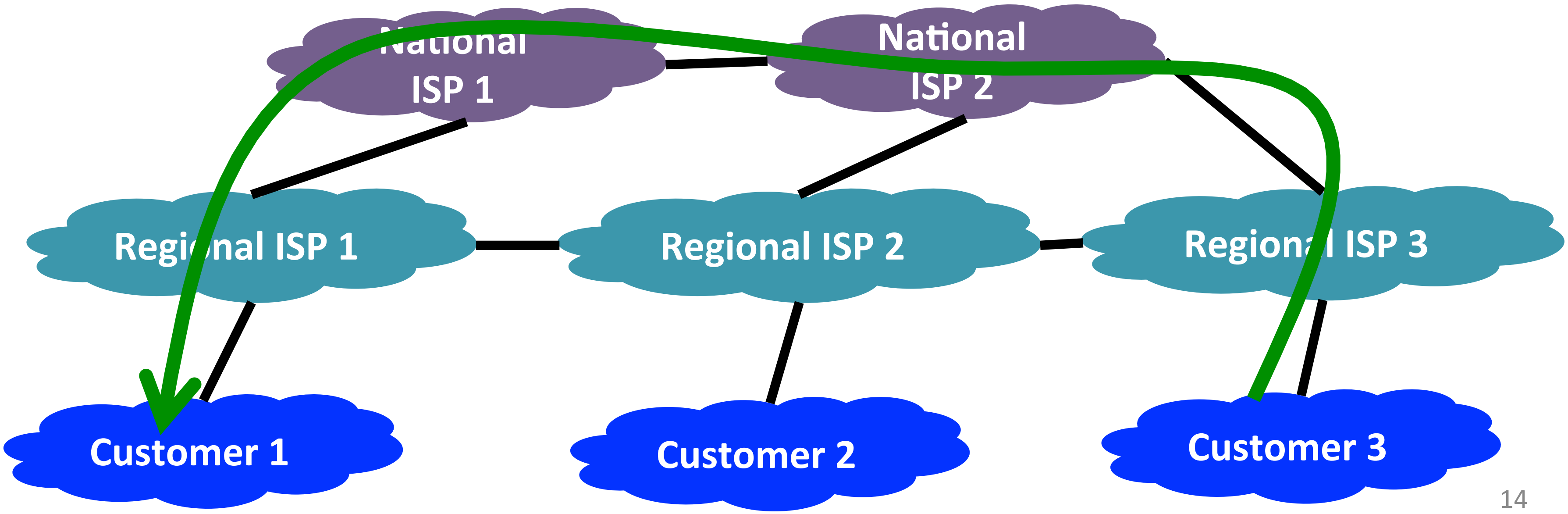
Shortest path routing

- Example: customer 3 talking to customer 1
  - Shortest path transits Regional ISP 2
  - Regional ISP 2 isn't being paid by either customer



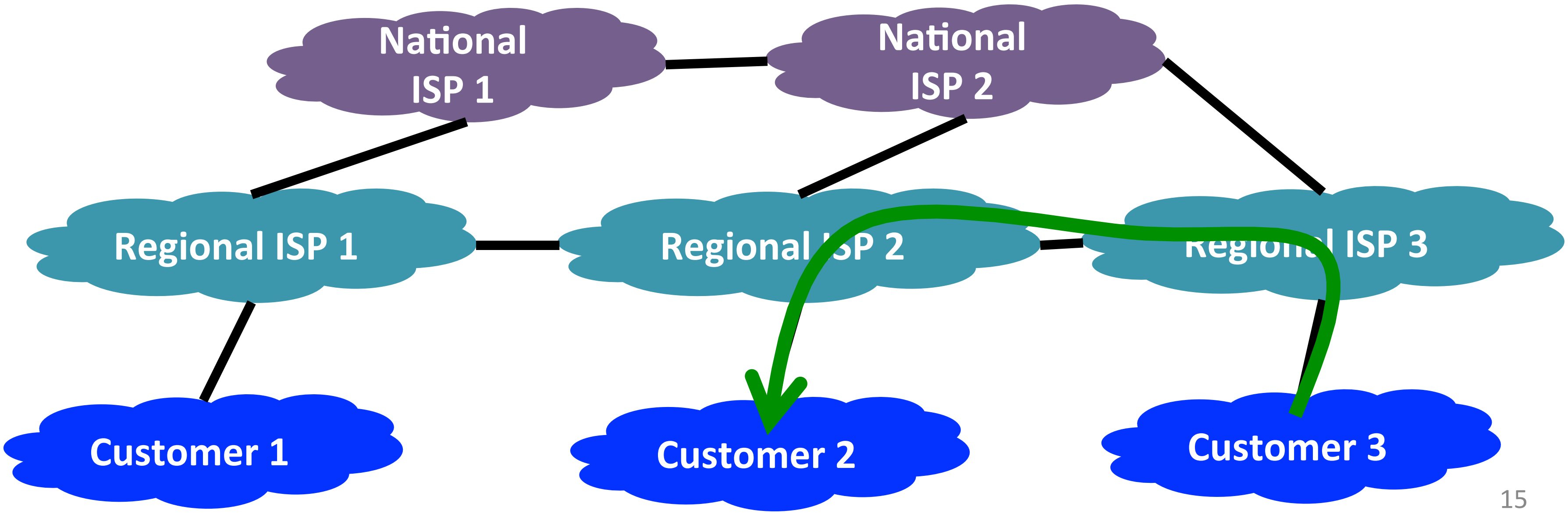
Shortest path routing

- Example: customer 3 talking to customer 1
  - Goes through National ISP 1 & 2
  - Regional 3 is paying National ISP 2
  - Regional 1 is paying National ISP 1



Shortest path routing

- Example: customer 3 talking to customer 2
  - Regional 2 and 3 are peered
  - Avoid going through National ISP 2 since then both regionals would incur expense



Other routing issues

- Policies may be political, security, or economic:
  - Some examples (Tanenbuam):
    - Don't carry commercial traffic on educational network
    - Never send Pentagon traffic through Iraq
    - Use TeliaSonera instead of Verizon because it is cheaper
    - Don't use AT&T in Australia because performance is poor
    - Traffic starting or ending at Apple should not transit Google

Link-state, disadvantages

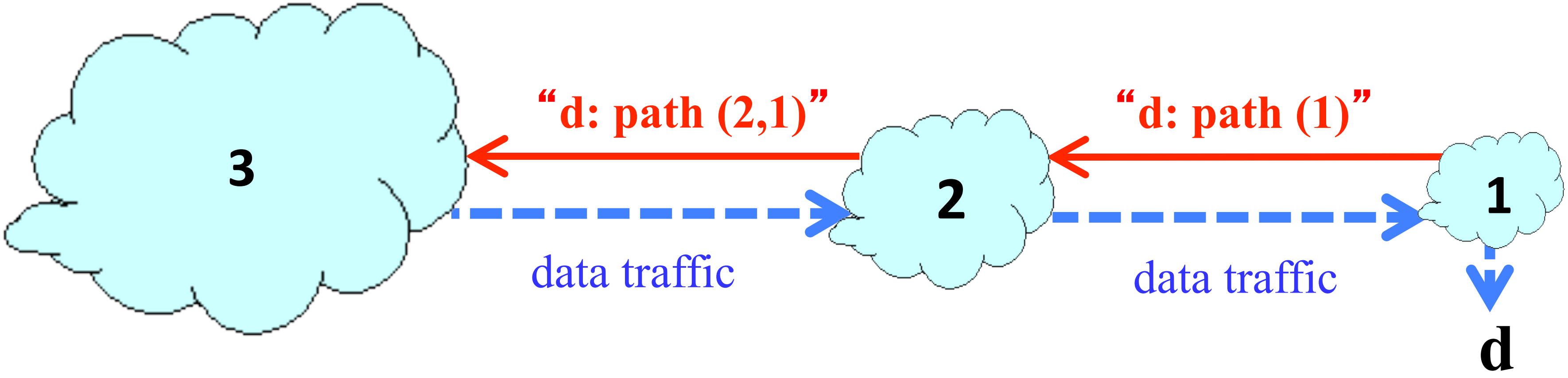
- Floods topology information
  - High bandwidth and storage requirements
  - Nodes divulge potentially sensitive information
- Entire path computed locally
  - High processing overhead for large network
- Distance calculation hides information
  - Everyone has to have a shared notion of link cost
- Typically used within one organization
  - Autonomous System (AS)
    - e.g. university, company, ISP
  - Popular link-state protocols: OSPF, IS-IS

Distance-vector

- Disadvantages:
  - Count to infinity, "bad news travels slow"
  - Slow to converge
  - Hides information that you might need in an interdomain setting
- Advantages:
  - Summarizes details of network topology
    - Trades optimality for scalability
  - Each node only needs to know about next hop

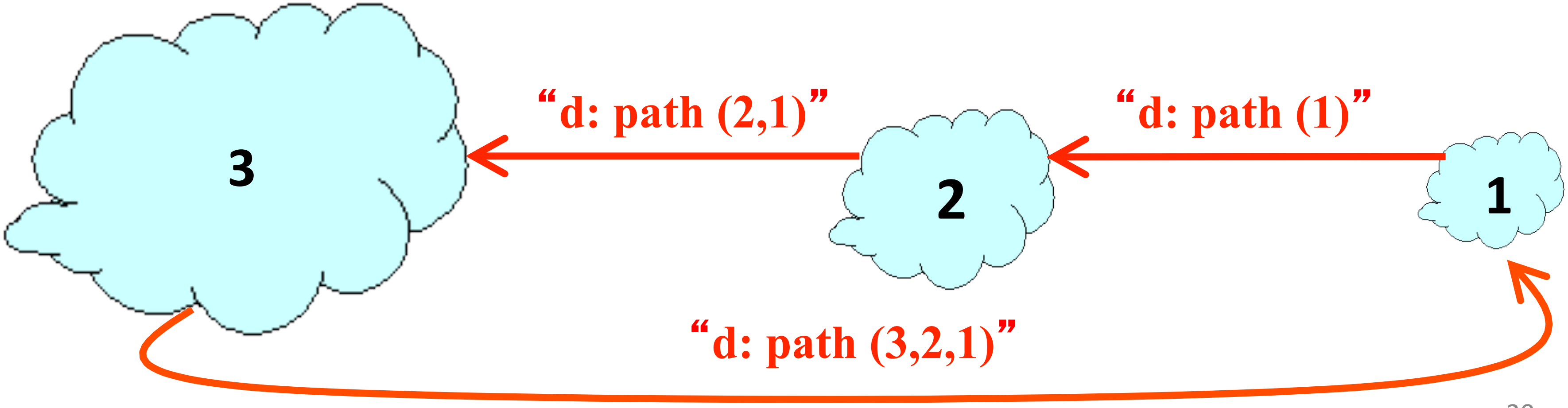
Path-vector routing

- Extension of distance-vector
  - Support flexible routing policies
  - Avoid count-to-infinity problem
- Key idea: advertise the entire path
  - Distance vector: send *distance metric* per destination d
  - Path vector: send the *entire path* per destination d



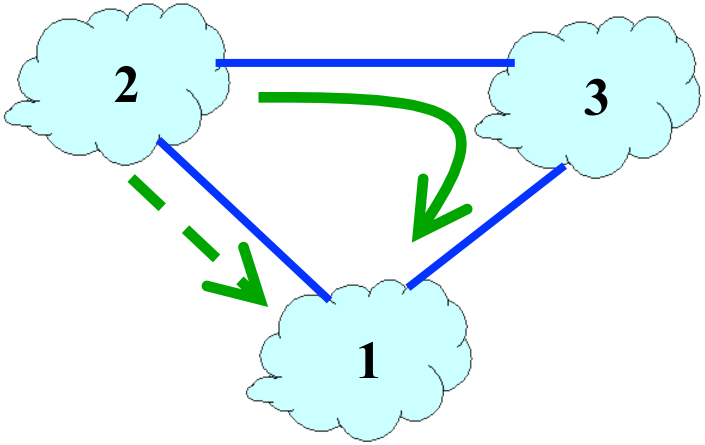
Detecting loops

- Path-vector can easily detect loop
  - – Look for your own node ID in the path
  - – e.g. node 1 sees itself in path "3, 2, 1"
- Node can discard paths with loops
  - – e.g. node 1 drops advertisement

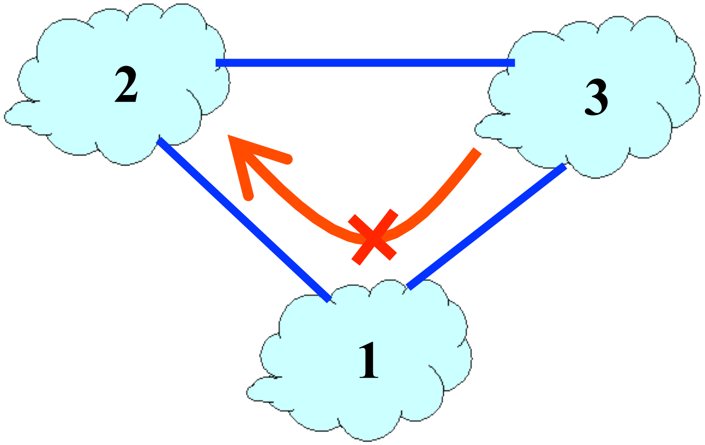


Flexible routing policies

- Each node can apply local policies:
  - Path selection: Which path to use?
  - Path export: Which path to advertise?



Node 2 may prefer the path "2, 3, 1" over the path "2, 1". Perhaps it is cheaper.

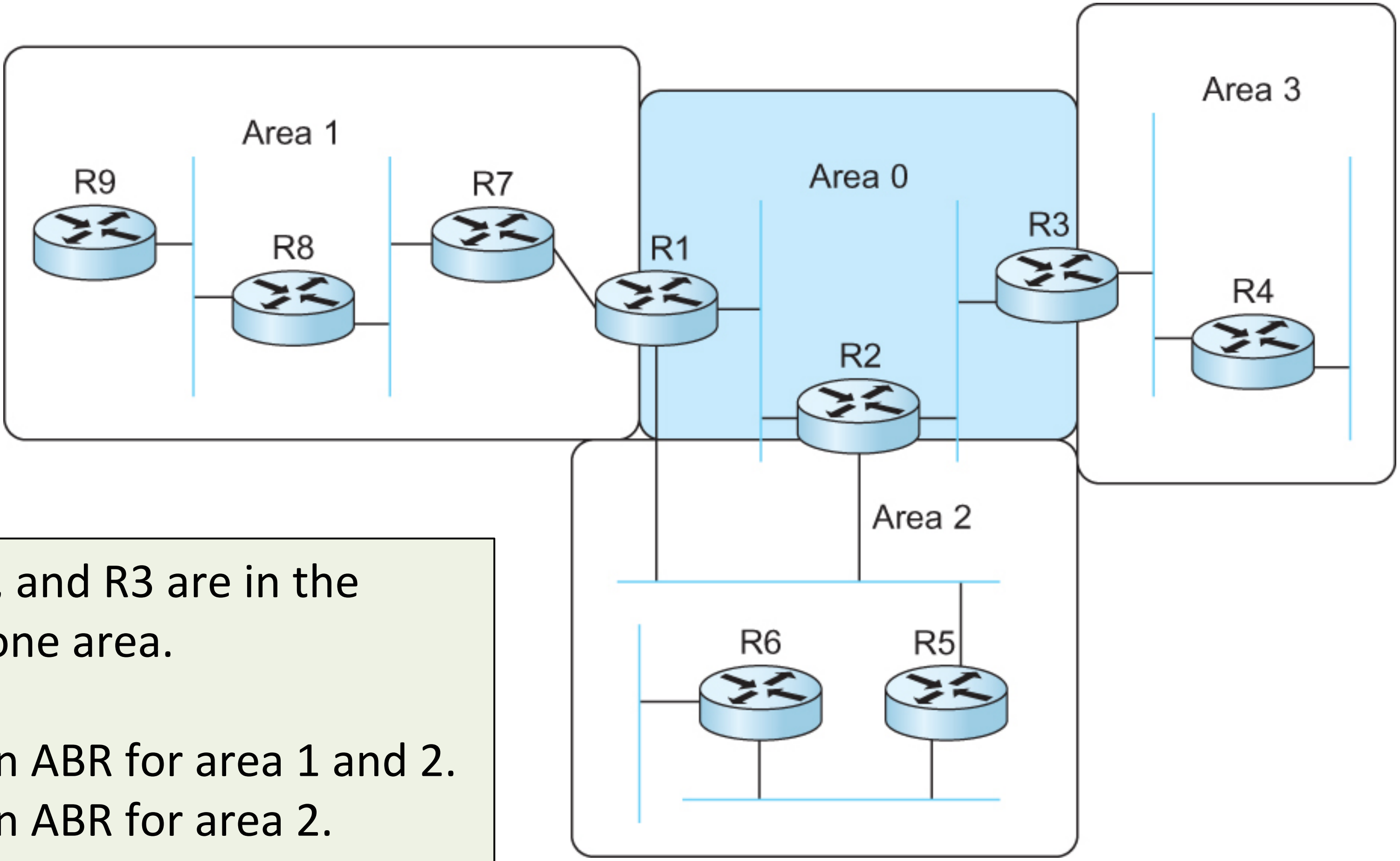


Node 1 may not export the path "1, 2". Perhaps node 1 reserves the 1->2 link for special traffic.

Scaling up

- How to scale a single company's network?
  - Add a level of hierarchy
    - Within a single organization (aka autonomous system)
  - Routing areas
    - Most routers in a single area
      - Routers only send information within their area
      - Detailed topology for only their area
      - Traffic going outside of area, send to backbone
    - Area 0 = backbone
      - Some routers in both backbone and other area(s)
      - Area Border Router (ABR)

Routing areas



R1, R2, and R3 are in the backbone area.

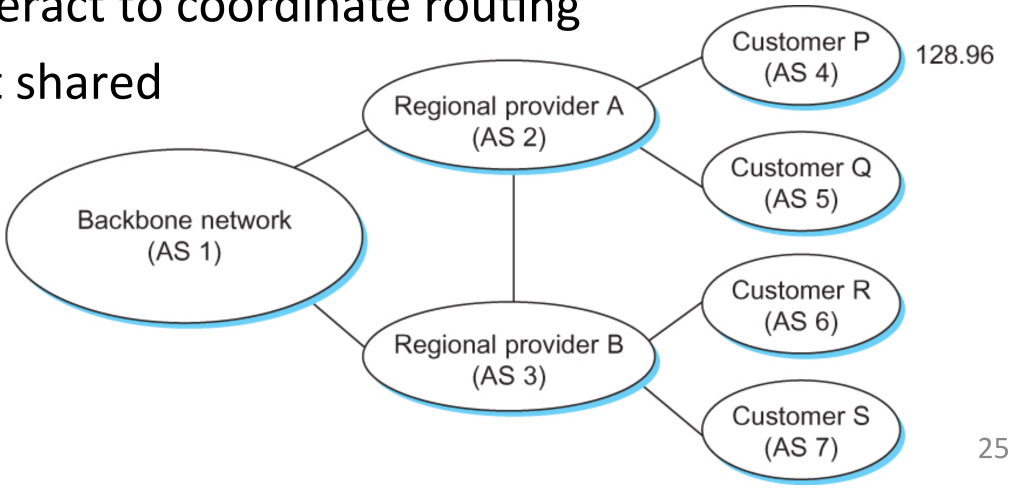
R1 is an ABR for area 1 and 2.
R2 is an ABR for area 2.
R3 is an ABR for area 3.

Routing areas

- Contains information flow
  - Link-state advertisements contained in your area
  - Reduces cost of flooding and route calculation
- Summarizes information
  - ABRs advertise cost of networks in their areas as if directly connected to the ABR
- Non-optimal routing
  - Going via backbone may not be the fastest route

Scaling up and up

- How to scale to the global Internet?
  - Add another level of hierarchy!
  - Routing amongst Autonomous Systems (ASes)
    - Distinct regions of admin control
    - Routers/links managed by a single institution
    - ASes can use policy-based routing
  - Interaction between ASes
    - Neighboring ASes interact to coordinate routing
    - Internal topology not shared

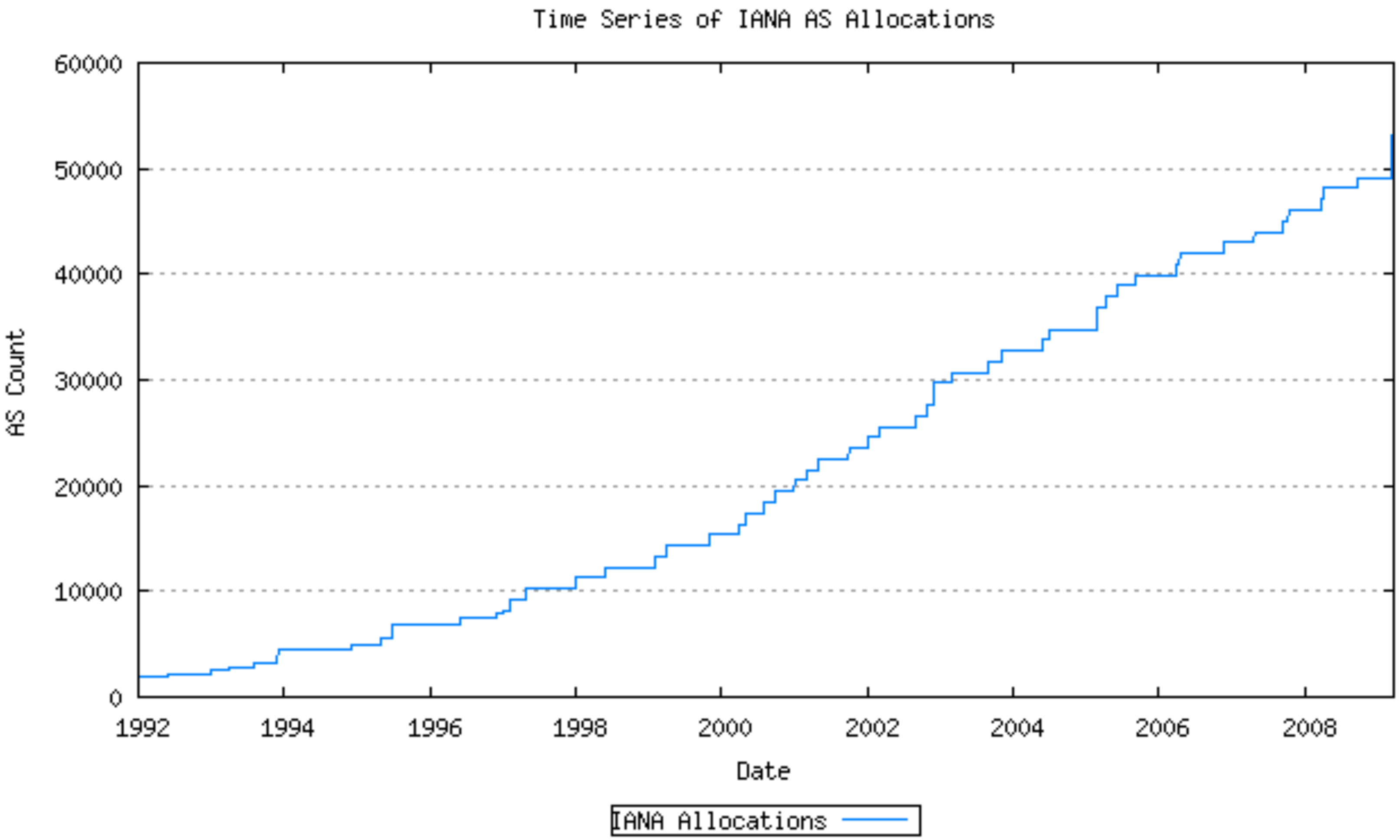


Autonomous System Numbers

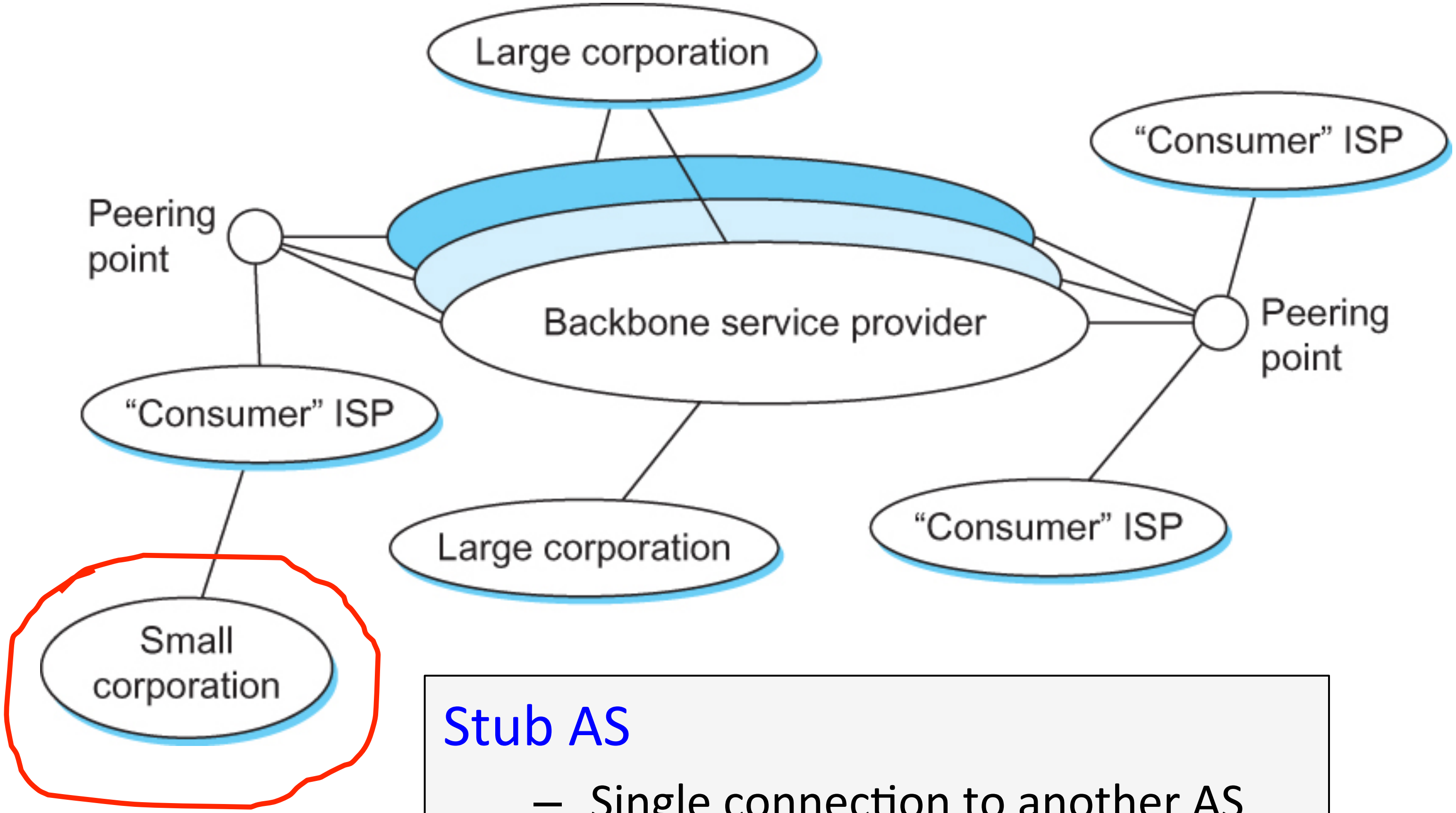
- Each AS assigned a unique number
  - Before 2007: AS Numbers 16-bit
  - After 2007: IANA began allocating 32-bit AS numbers
  - Currently over 50,000 allocated

- Level 3: 1
- MIT: 3
- Harvard: 11
- Yale: 29
- Princeton: 88
- AT&T: 7018, 6341, 5074, …
- UUNET: 701, 702, 284, 12199, …
- Sprint: 1239, 1240, 6211, 6242, …
- …

Autonomous System Numbers



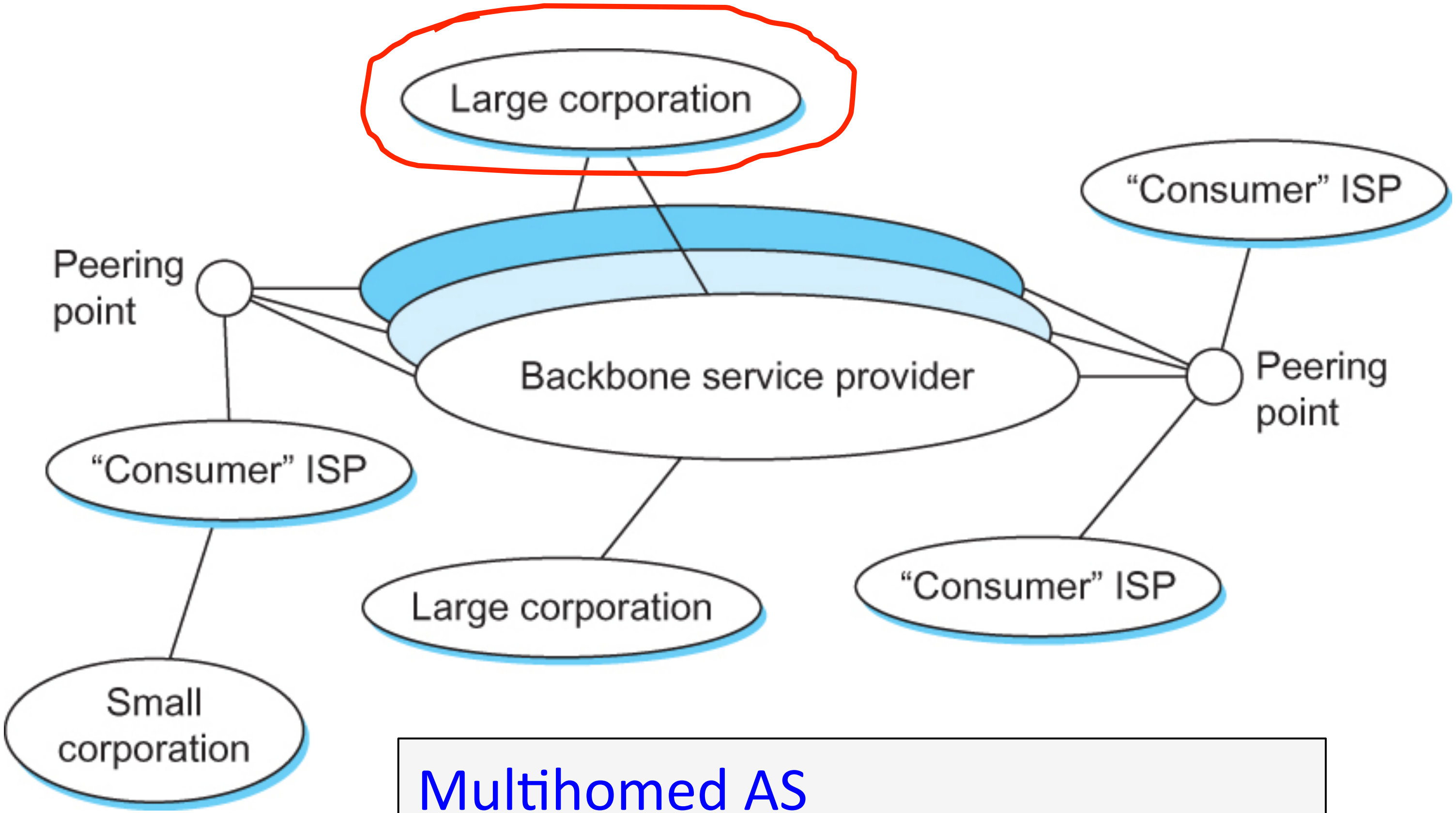
AS stub



Stub AS

- Single connection to another AS
- AS only carries local traffic
- e.g. Small corporation, university

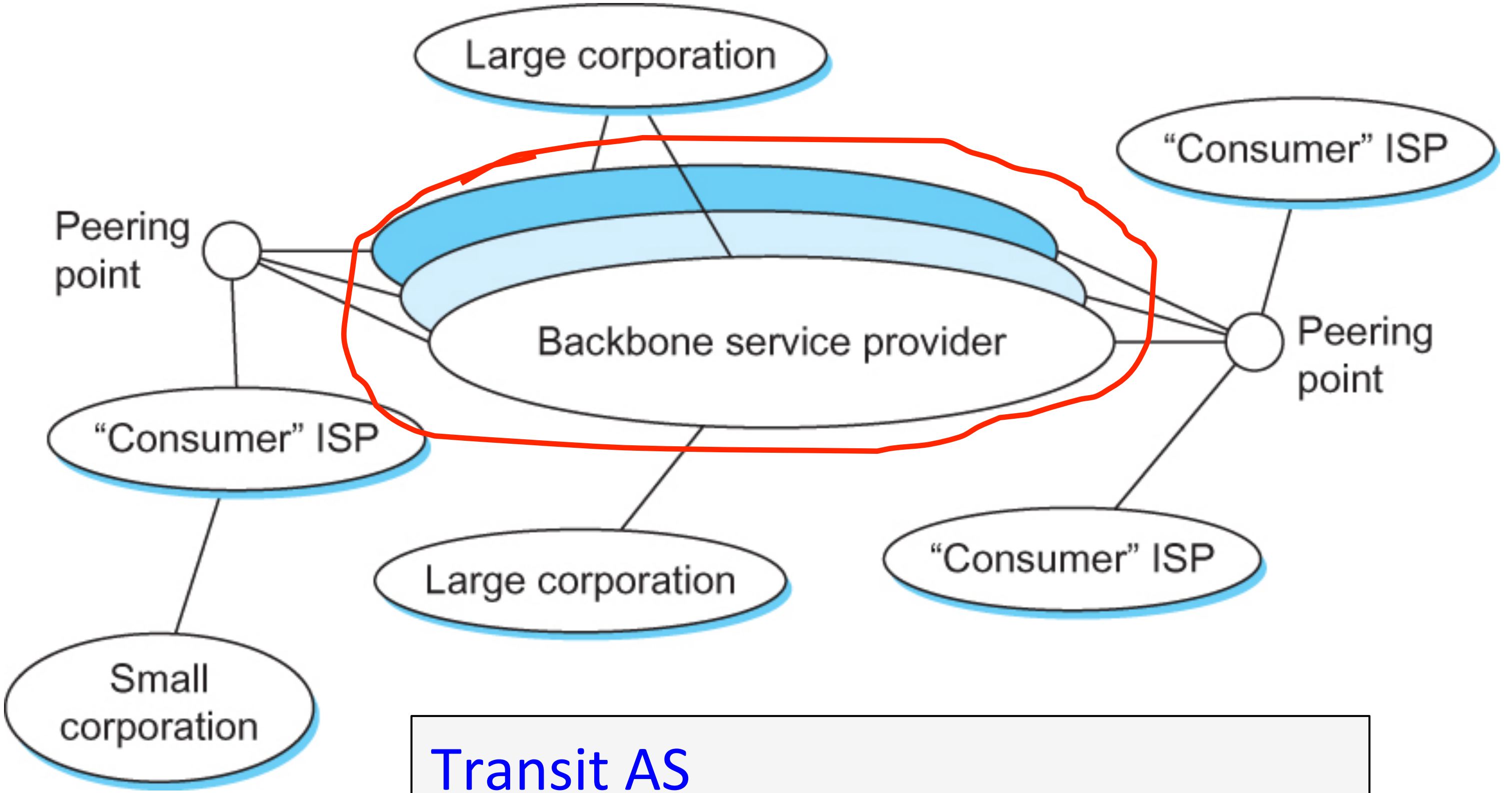
AS multihomed



Multihomed AS

- Connected to multiple ASes
- Refuses to carry transit traffic
- Improves reliability

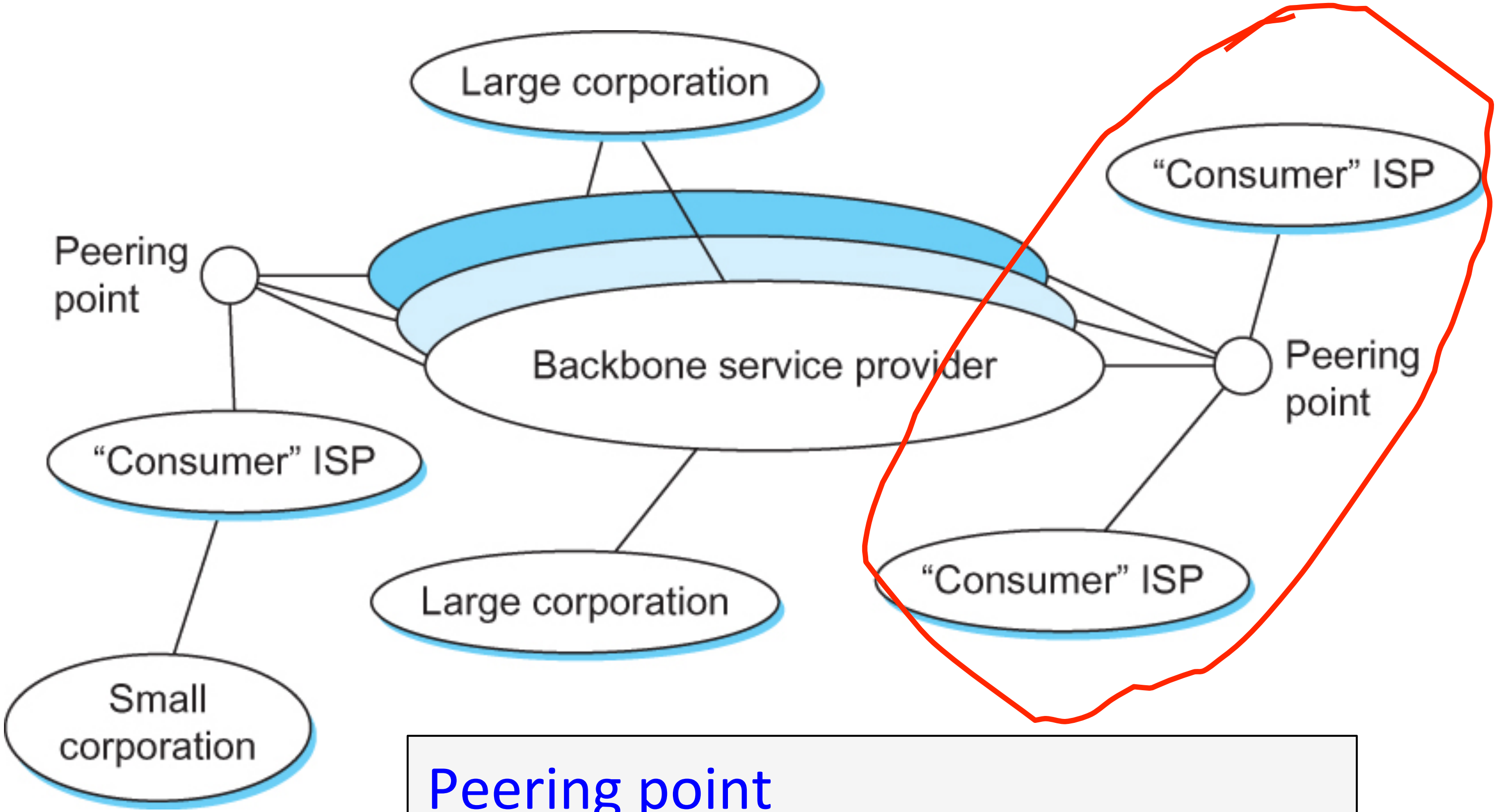
AS transit



Transit AS

- Connected to multiple ASes
- Designed to carry transit and local traffic

Peering point

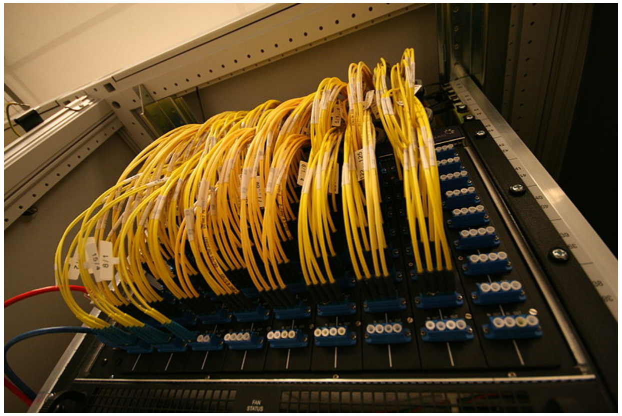


Peering point

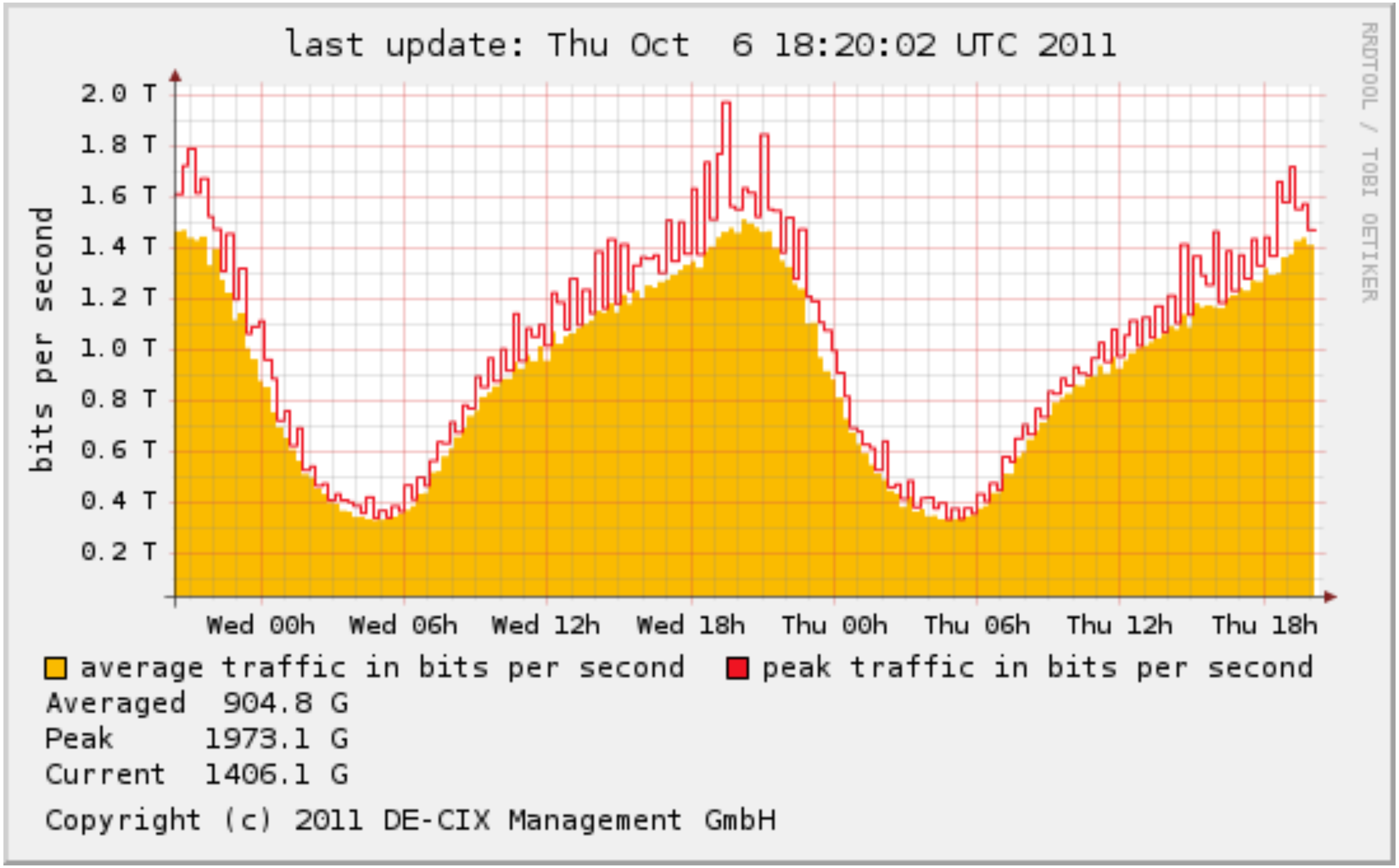
- Allows ASes to connect directly, bypassing a transit AS.

Peering point

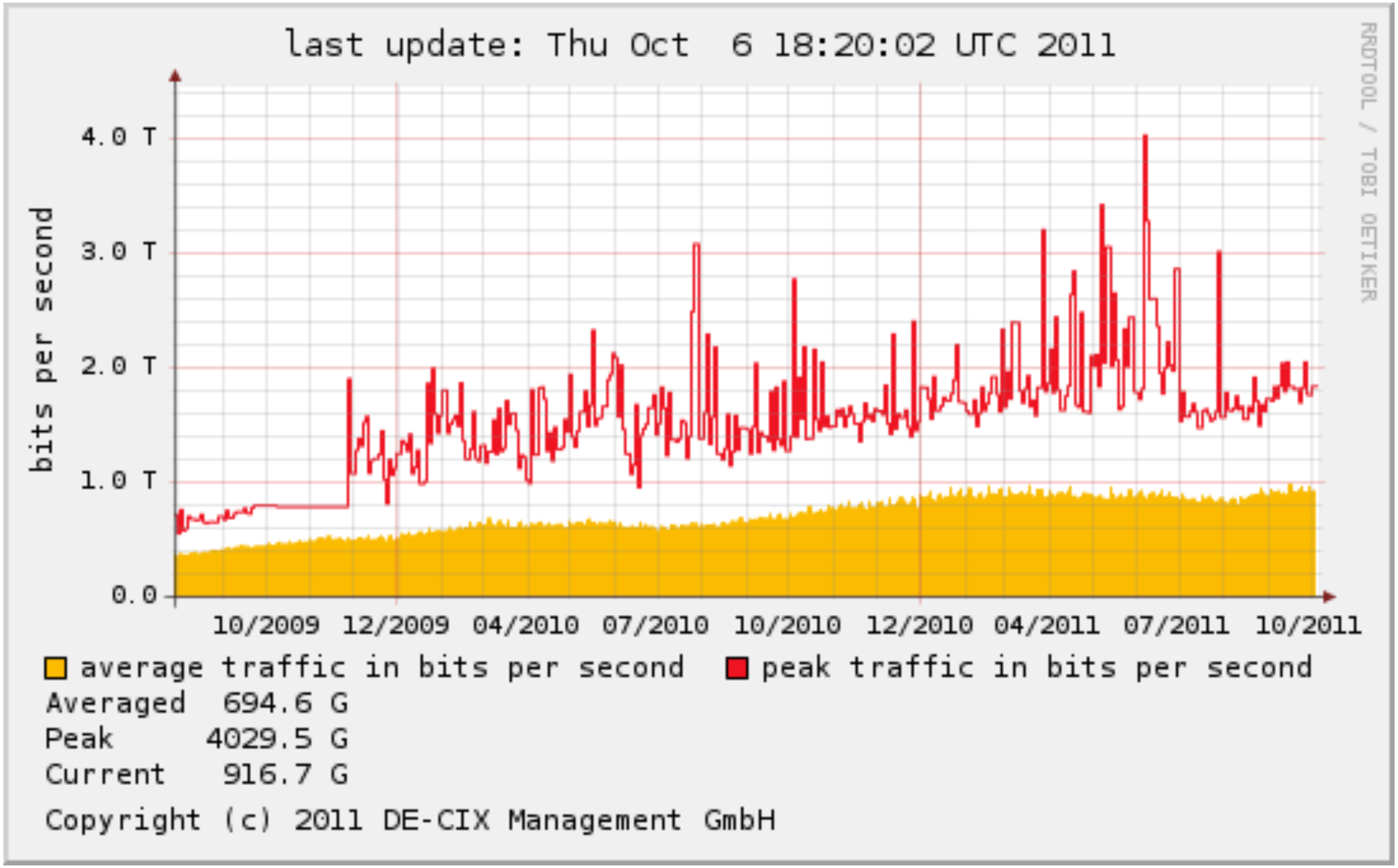
- Peering point
  - Many networks come together in one location
  - Exchange traffic
    - reduce cost
    - improve performance
    - improve reliability
  - e.g. DE-CIX
    - One of the world's largest peering points
    - 400 ISPs from 45+ countries



DE-CIX daily graph



DE-CIX yearly graph

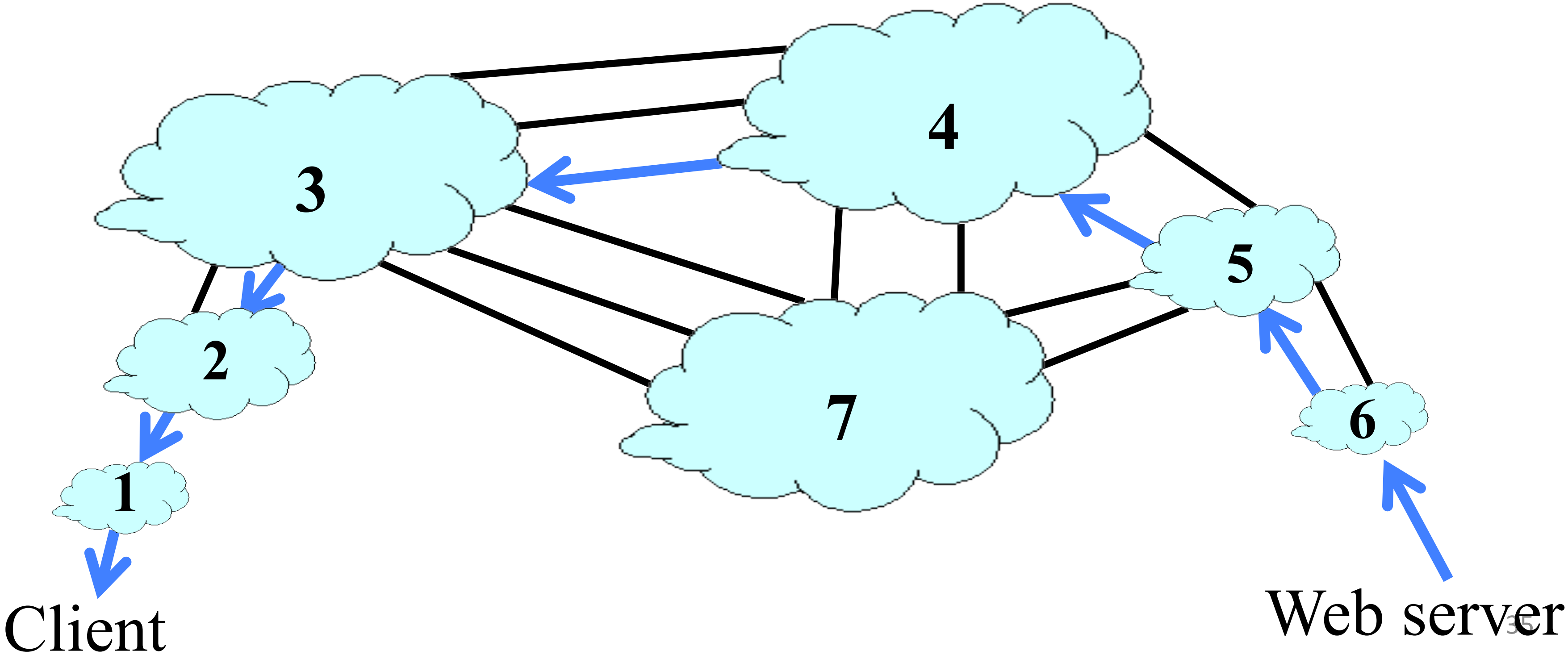


🟨 average traffic in bits per second 🟥 peak traffic in bits per second

```
Averaged   694.6 G
Peak      4029.5 G
Current    916.7 G
```

Interdomain routing

- AS-level topology
  - Destinations are IP prefixes (e.g. 12.0.0.0/8)
  - Nodes are Autonomous Systems (ASes)
  - Edges are links and business relationships



Interdomain routing challenges

- Scale:
  - IP prefixes: 200,000+
  - ASes: 20K+ visible, 50k+ allocated
  - Routers: millions
- Privacy:
  - ASes don't want other to known topology
  - ASes don't want business relationships exposed
- Policy:
  - No internet-wide notion of link cost metric
  - Need control over where you send traffic, who you send traffic through, etc.

Border Gateway Protocol

- Interdomain routing protocol for the Internet
  - Prefix-based path-vector protocol
  - Policy-based routing using AS paths
  - Evolved over the past 18 years

- **1989 : BGP-1 [RFC 1105], replacement for EGP**
- **1990 : BGP-2 [RFC 1163]**
- **1991 : BGP-3 [RFC 1267]**
- **1995 : BGP-4 [RFC 1771], support for CIDR**
- **2006 : BGP-4 [RFC 4271], update**

Summary

- Dealing with scale of Internet
  - Separate into autonomous systems (ASes)
  - Within an AS:
    - Use an intradomain routing protocol (OSPF)
    - Route optimally
  - Between ASes:
    - Use an interdomain routing protocol that routes between ASes
    - Path-vector routing allows scaling and implementation of policy compliant paths
- Next: details of BGP, IPv6, NAT