

CSCI1680

Network Layer: IP & Forwarding

Administrivia

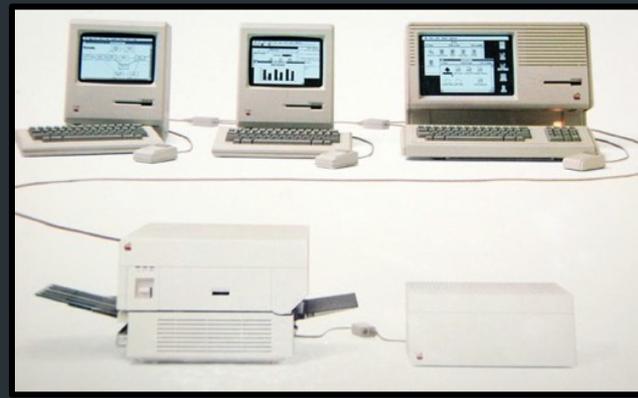
- Snowcast: now due Thursday (2/12)
 - Look for an announcement with updates to tester, tips for debugging
- IP project: out Thursday
 - Team form: out TODAY, due THURSDAY 2/12
 - Gearup: THURSDAY 2/12 5-7pm CIT 165
- HW1: out later today, due next Thurs
 - Some practice for IP!

Today

Start of network layer

- Network layer: Internet Protocol (IP) (v4)
- Mechanics of IP forwarding
- Intro to IP project

Recap: the link layer



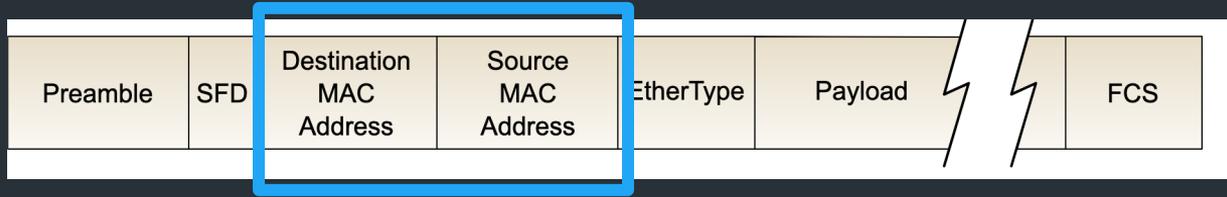
Recap: the link layer

Goal: How to connect hosts on a "small" network

- Hosts connect to network via interfaces
- Every interface has a link-layer address
 - Ethernet/Wifi: MAC address (`0c:45:22:c1:be:03`)

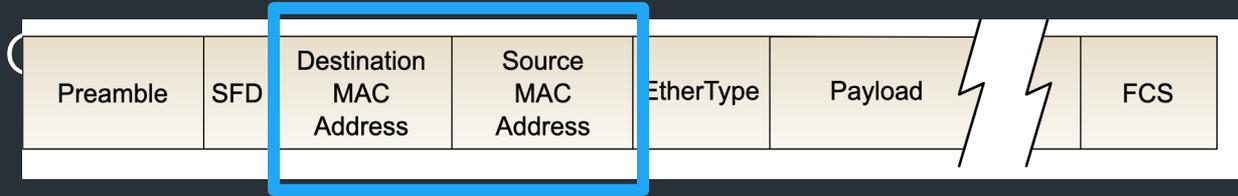


Ethernet Addresses (mac addresses)



00:1c:43:00:3d:09 (Samsung adapter)

Ethernet Addresses (mac addresses)



Globally unique, 48-bit address per interface

00:1c:43:00:3d:09 (Samsung adapter)

First 24 bits: Registered to manufacturers

=> Other protocols have adopted this address format (eg. Wifi, Bluetooth, ...)

=> Nowadays, we call them "mac addresses" or "hardware addresses"

Recap: the link layer

Goal: How to connect hosts on a “small” network

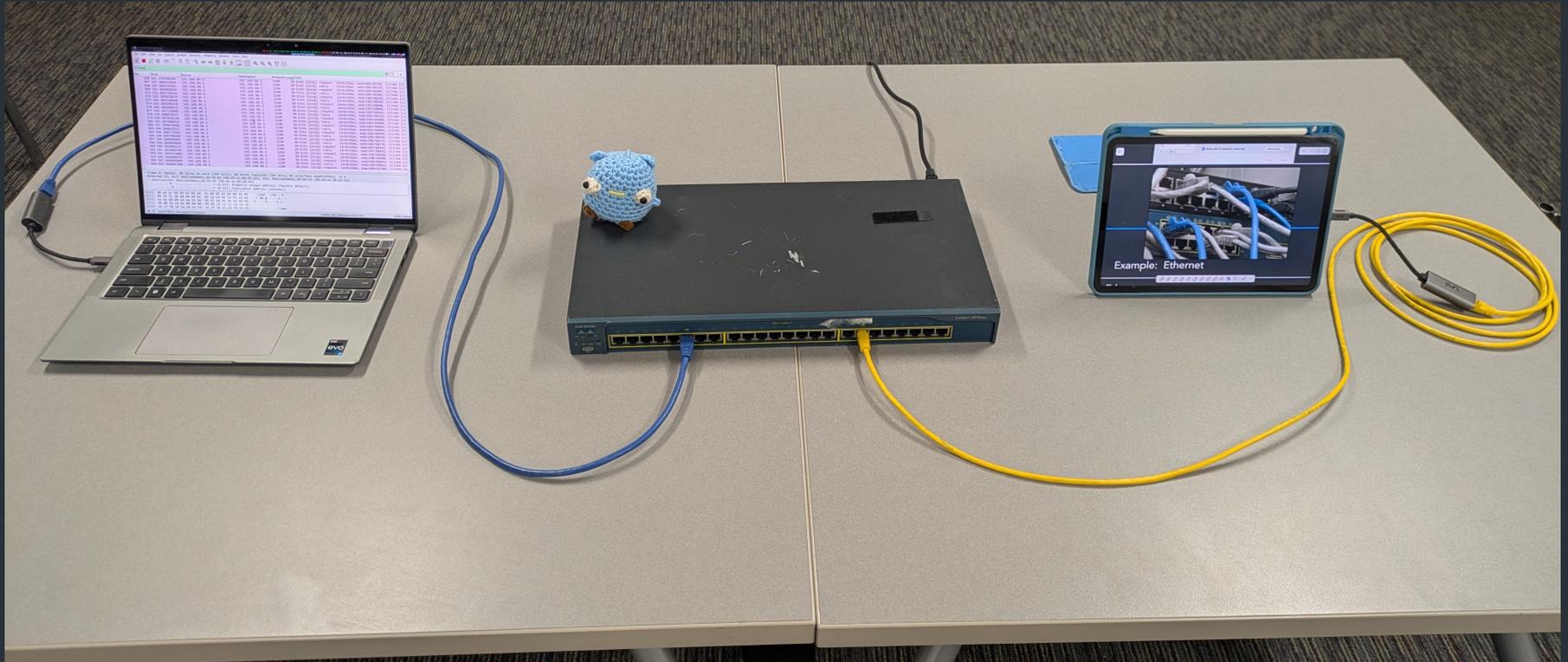
- Hosts connect to network via interfaces
- Every interface has a link-layer address
 - Ethernet/Wifi: MAC address (`0c:45:22:c1:be:03`)

Mental model for the link layer

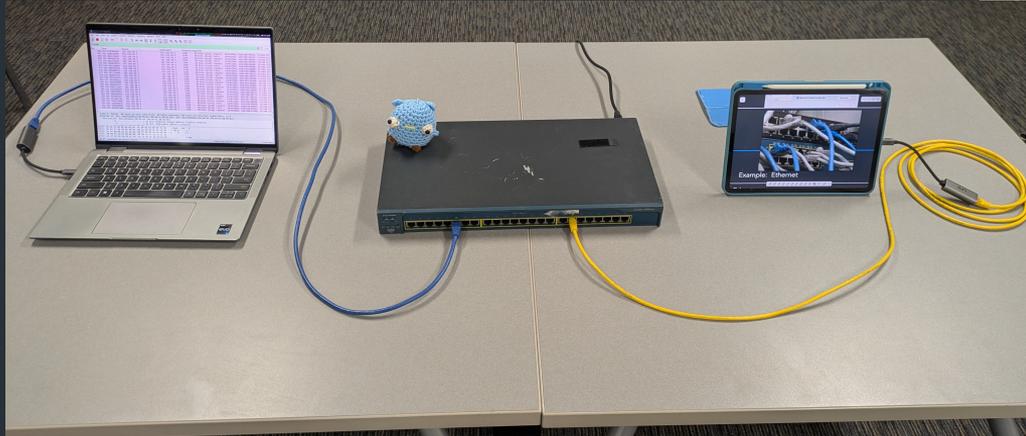
- Every host connected to every other host (at least logically)
- Given a link-layer address, know how to reach host **on your network**

=> Devices: Switches, Wifi APs: forward packets between nodes on same network

Switch: network device that forwards frames (packets) between *ports*



Switch: network device that forwards frames (packets) between *ports*



Example: Ethernet switch MAC table

```
R6#sh mac-address-table
EHWIC: 0
Destination Address Address Type VLAN Destination Port
-----
5c45.27e0.8383      Dynamic      1 GigabitEthernet0/1/3
7641.7b63.584a      Dynamic      20 GigabitEthernet0/1/3
5c45.27e0.8381      Dynamic      10 GigabitEthernet0/1/3
0000.5e00.0101      Dynamic      10 GigabitEthernet0/0/1
ca3f.aee3.e3e6      Dynamic      20 GigabitEthernet0/1/3
644b.f012.7f75      Dynamic      20 GigabitEthernet0/1/3
f018.9815.8eb8      Dynamic      20 GigabitEthernet0/1/3
ecb5.fa13.4677      Dynamic      20 GigabitEthernet0/0/2
a0a4.c5c2.4165      Dynamic      20 GigabitEthernet0/0/1
4c71.0c92.4f10      Dynamic      10 GigabitEthernet0/1/3
12d3.acae.bbc0      Dynamic      20 GigabitEthernet0/0/1
04d4.c448.9cf7      Dynamic      20 GigabitEthernet0/1/3
```

Example: Ethernet switch MAC table

```
R6#sh mac-address-table
EHWIC: 0
Destination Address Address Type VLAN Destination Port
-----
5c45.27e0.8383      Dynamic      1 GigabitEthernet0/1/3
7641.7b63.584a      Dynamic      20 GigabitEthernet0/1/3
5c45.27e0.8381      Dynamic      10 GigabitEthernet0/1/3
0000.5e00.0101      Dynamic      10 GigabitEthernet0/0/1
ca3f.aee3.e3e6      Dynamic      20 GigabitEthernet0/1/3
644b.f012.7f75      Dynamic      20 GigabitEthernet0/1/3
f018.9815.8eb8      Dynamic      20 GigabitEthernet0/1/3
ecb5.fa13.4677      Dynamic      20 GigabitEthernet0/0/2
a0a4.c5c2.4165      Dynamic      20 GigabitEthernet0/0/1
4c71.0c92.4f10      Dynamic      10 GigabitEthernet0/1/3
12d3.acae.bbc0      Dynamic      20 GigabitEthernet0/0/1
04d4.c448.9cf7      Dynamic      20 GigabitEthernet0/1/3
```

Does it scale? 🤔

Why doesn't it scale?

Enter: IP

Application

Service: user-facing application.
Application-defined messages

Transport

Service: multiplexing applications
Reliable byte stream to other node (TCP),
Unreliable datagram (UDP)

Network

Service: move packets to any other node in the network
Internet Protocol (IP)

Link

Service: move frames to other node across link.
May add reliability, medium access control

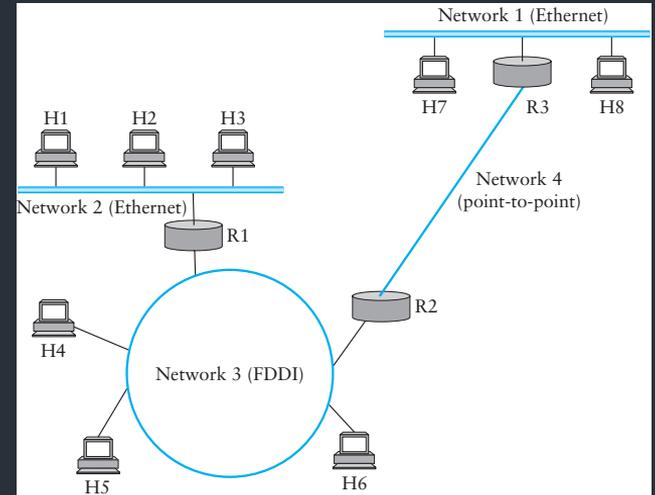
Physical

Service: move bits to other node across link

Internet Protocol (IP) Goals

How to connect *everyone*?

- Glue lower-level networks together
- A network of networks!
- Router: device that forwards packets between *networks*

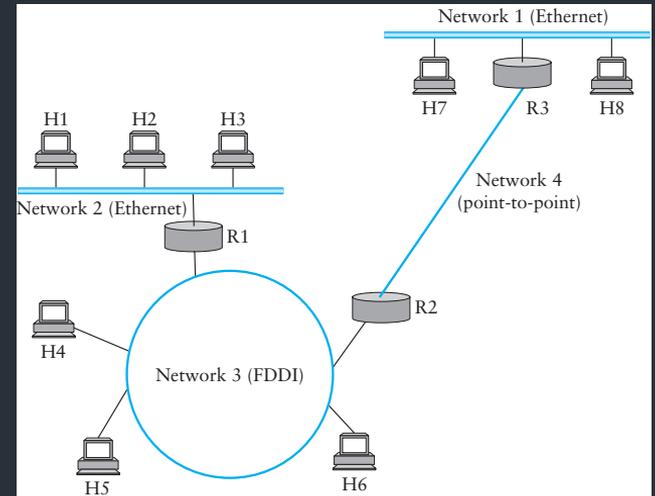


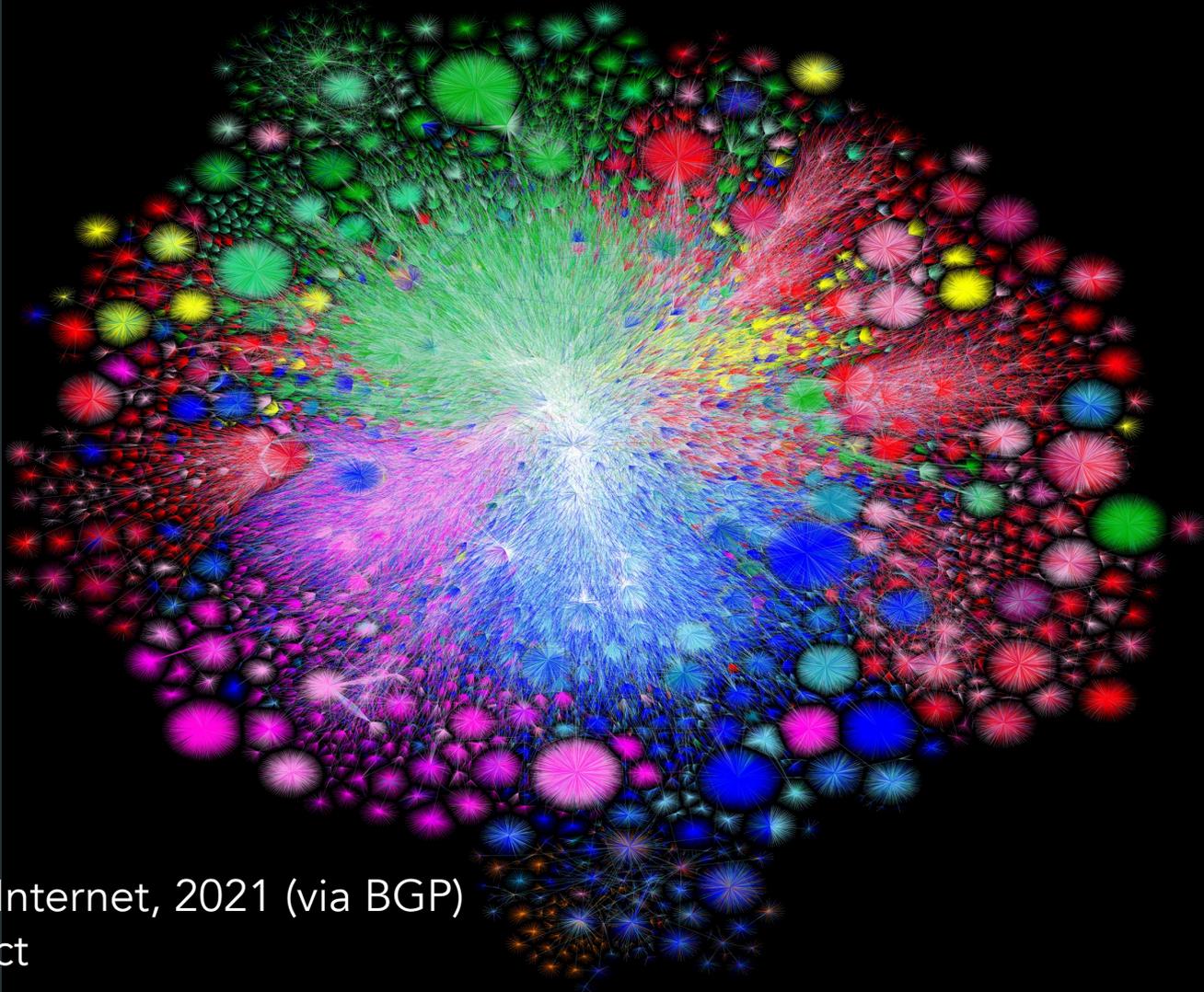
Internet Protocol (IP) Goals

How to connect *everyone*?

- Glue lower-level networks together
- A network of networks!
- Router: device that forwards packets between *networks*

=> Doesn't that sound like switching?



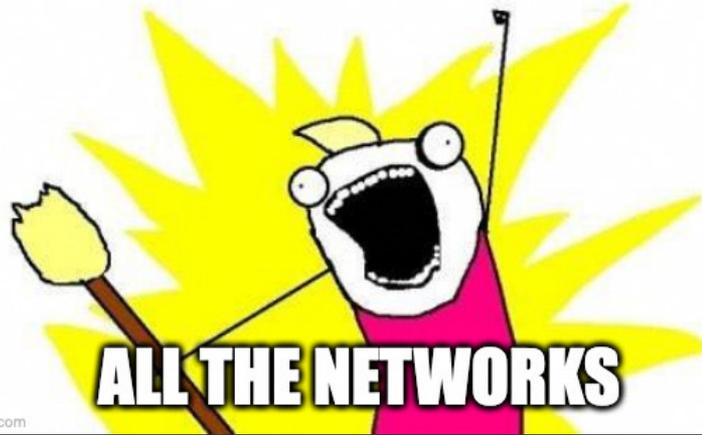


Color Chart

North America (ARIN)	Blue
Europe (RIPE)	Green
Asia Pacific (APNIC)	Red
Latin America (LANIC)	Purple
Africa (AFRINIC)	Yellow
Backbone	White
US Military	Brown

Map of the Internet, 2021 (via BGP)
OPTE project

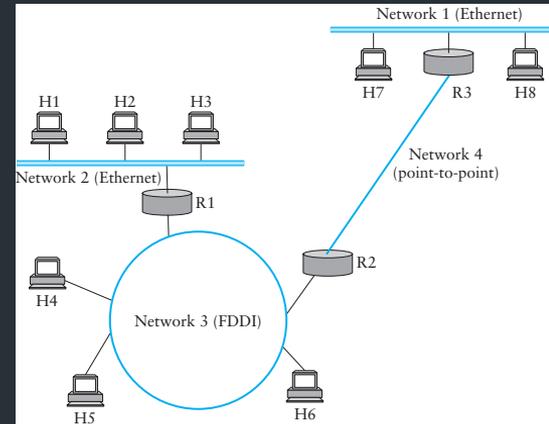
CONNECT



imgflip.com

New Challenges

- Networks are heterogeneous (eg. Wifi vs. Ethernet)
 - Different frame formats
 - Some are more reliable than others
 - Different packet sizes/bandwidths
- Scaling: link-layer strategies don't work!



What came before the Internet?

The (landline) telephone network

(Plain Old Telephone Service (POTS))



A Host

The (landline) telephone network

(Plain Old Telephone Service (POTS))



A Host





A large telephone exchange, 1943

Example: long distance telephone call

*Dramatization from an episode of
radio program "Dragnet", 1949*

Early telephone systems

- Circuit switching: set up whole path for call beforehand

Does it scale?

Early Internet goals

Early Internet goals

- ⇒ Build system that can connect different *networks*
- ⇒ Operate over long distances
- ⇒ Managed by different entities

Early Internet goals

- ⇒ Build system that can connect different *networks*
- ⇒ Operate over long distances
- ⇒ Managed by different entities

Need: *devices* and *protocols* to make this work

Design questions

- How to deal with heterogeneous networks?
- How to find hosts?
- Should messages be reliable or unreliable?
- What to do when a device joins/leaves?
- ...

A Bit of History

Early Packet switched networks: Arpanet's IMPs

- Late 1960's => RFC 1 (1969)
- Reliable network with many features we know today



A Bit of History

Early Packet switched networks: Arpanet's IMPs

- Late 1960's => RFC 1, 1969!
- Reliable network with many features we know today

Initial version: Network Control Program (NCP)

- Assumed IMPs were reliable



A Bit of History

Early Packet switched networks: Arpanet's IMPs

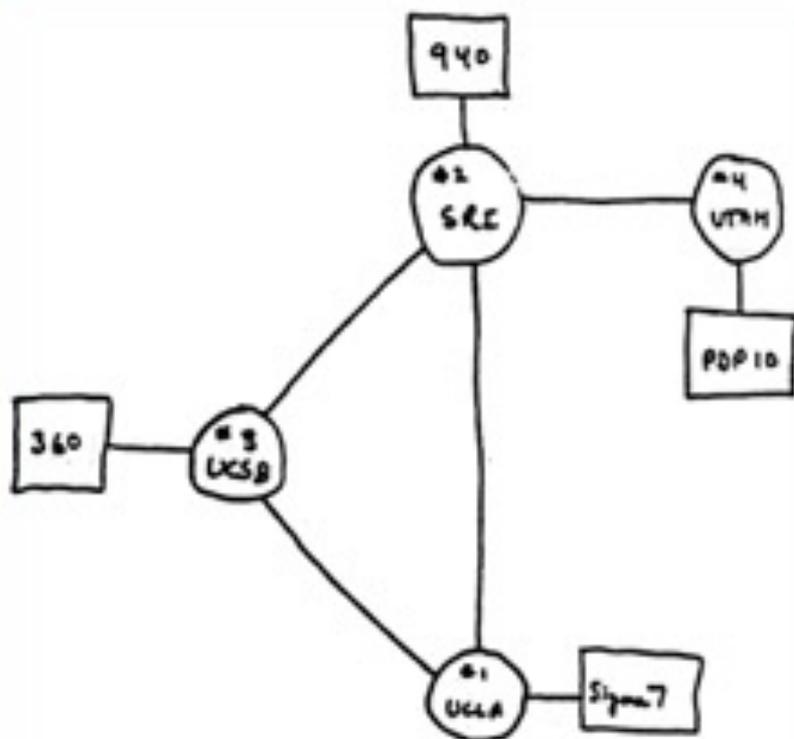
- Late 1960's => RFC 1, 1969!
- Reliable network with many features we know today

Initial version: Network Control Program (NCP)

- Assumed IMPs were reliable

What about when network isn't reliable?





THE ARPA NETWORK
DEC 1969

How to make such a protocol?

- How to deal with heterogeneous networks?
- How to find hosts?
- Should messages be reliable or unreliable?
- What to do when a device joins/leaves?
- ...

Big concerns

- ⇒ Not every application needs all features
- ⇒ Can't assume much functionality from (heterogeneous link layer)

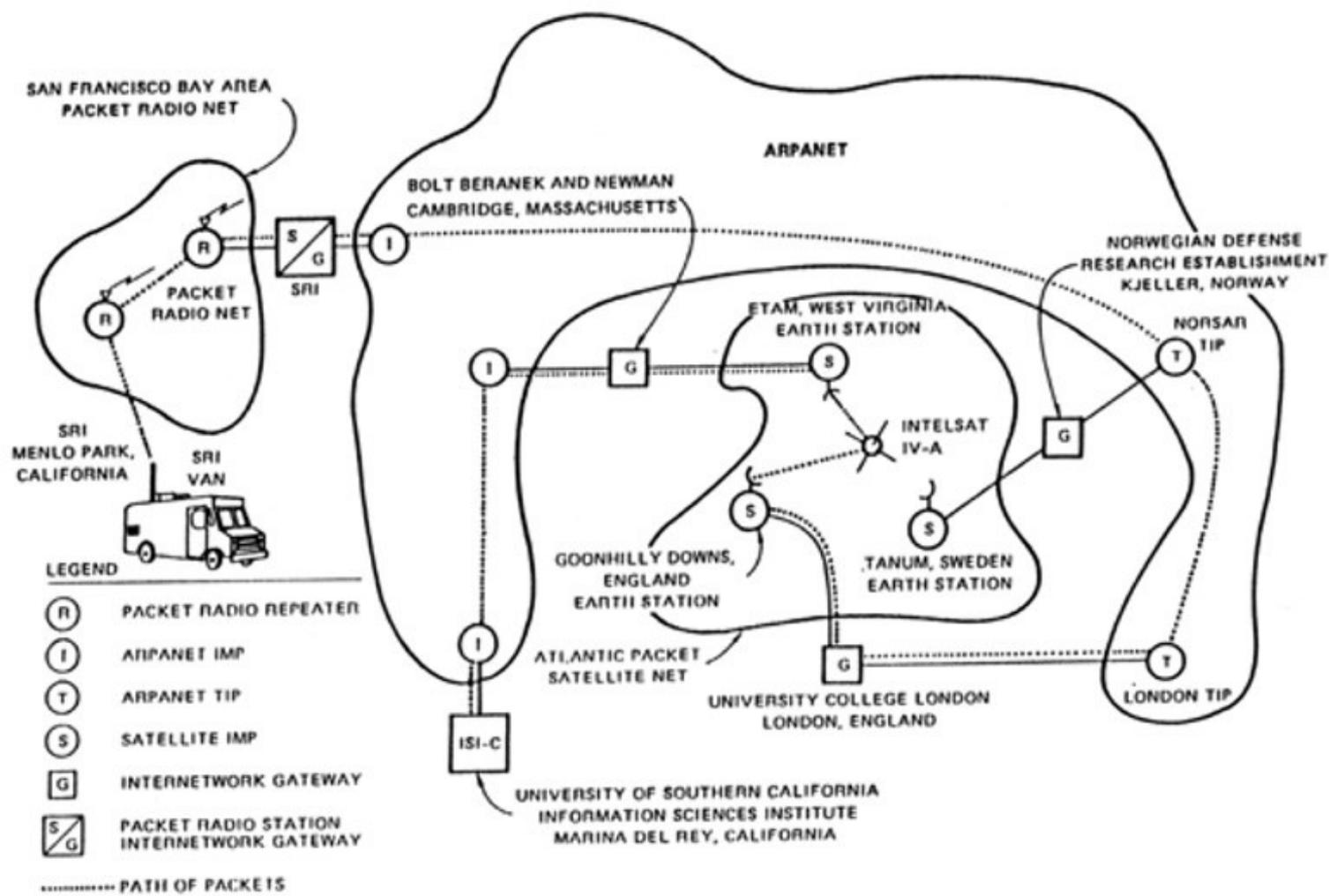
1974: TCP/IP Introduced

- Vint Cerf, Robert Kahn build protocol to replace NCP
- Initial design: single protocol providing a reliable pipe

1974: TCP/IP Introduced

- Vint Cerf, Robert Kahn build protocol to replace NCP
- Initial design: single protocol providing a reliable pipe

Eventually, separated into different protocols we know today



IP's Decisions

- Connectionless, packet-switched network
- "Best-effort" service

How to reach hosts?

IP's Decisions

- Connectionless, packet-switched network
 - => Routers are "simple" => no connection state
- "Best-effort" service: other layers add reliability if you need it
 - => Packets might be dropped, reordered, delayed, ...

How to reach hosts?

- Common message format: IP header
- Every host identified by an IP address

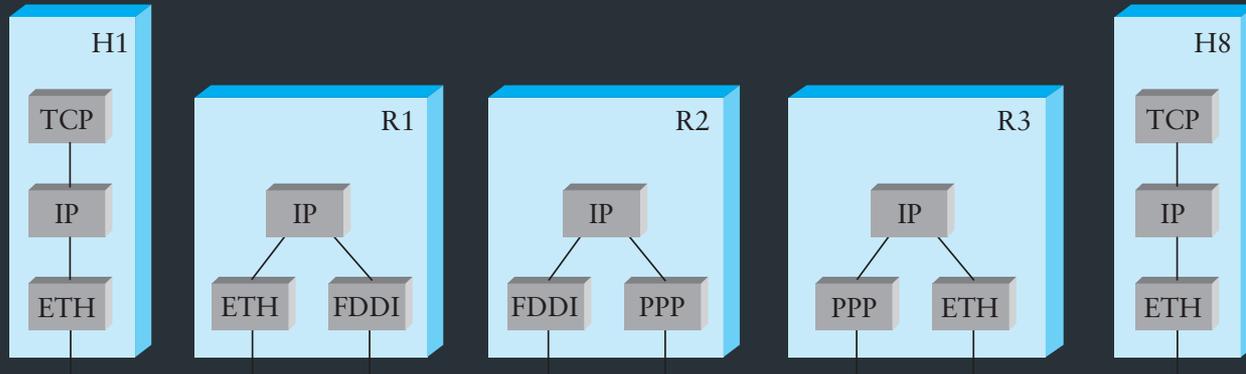
An excellent read

David D. Clark, "The design Philosophy of the DARPA Internet Protocols", 1988

Primary goal: multiplexed utilization of existing interconnected networks

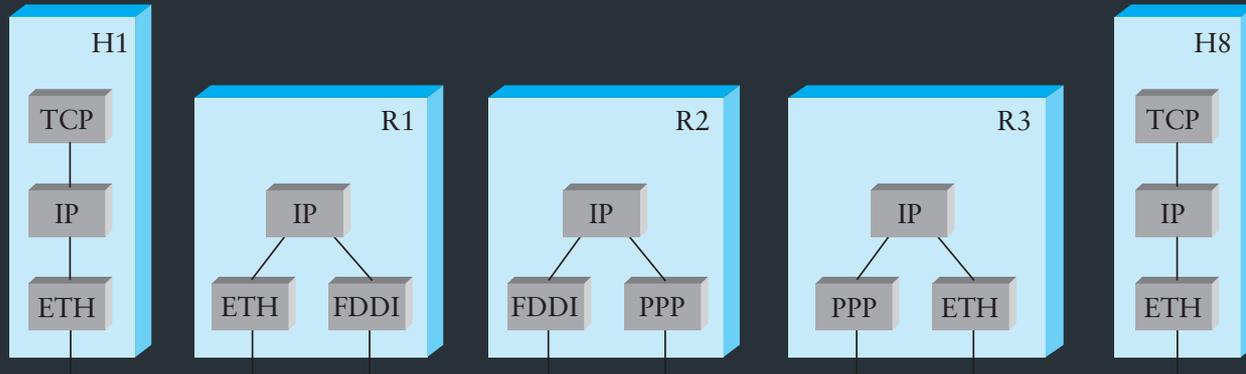
- Other goals:
 - Communication continues despite loss of networks or gateways
 - Support a variety of communication services
 - Accommodate a variety of networks
 - Permit distributed management of its resources
 - Be cost effective
 - Low effort for host attachment
 - Resources must be accountable

The Internet Protocol (IP): Runs on all hosts and routers



The Internet Protocol (IP): Runs on all hosts and routers

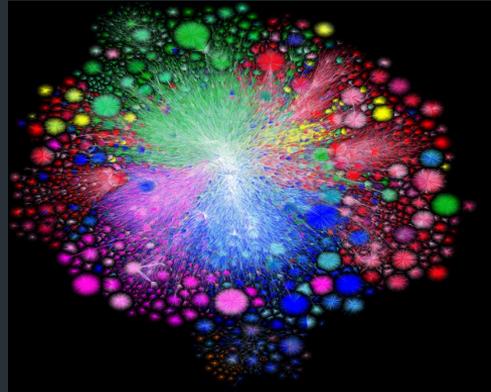
- Addressing: how we name nodes in an IP network
- Provides forwarding: how routers move packets based on the destination address
- (later) Routing: how routers build forwarding rules



IP Addressing

What's an IP address

- Unique number to identify “all” hosts on the Internet
- A number with **structure** => the number tells the network where the host is



Example: phone numbers



+ 1 401 863 1000



Analogy: back to phones

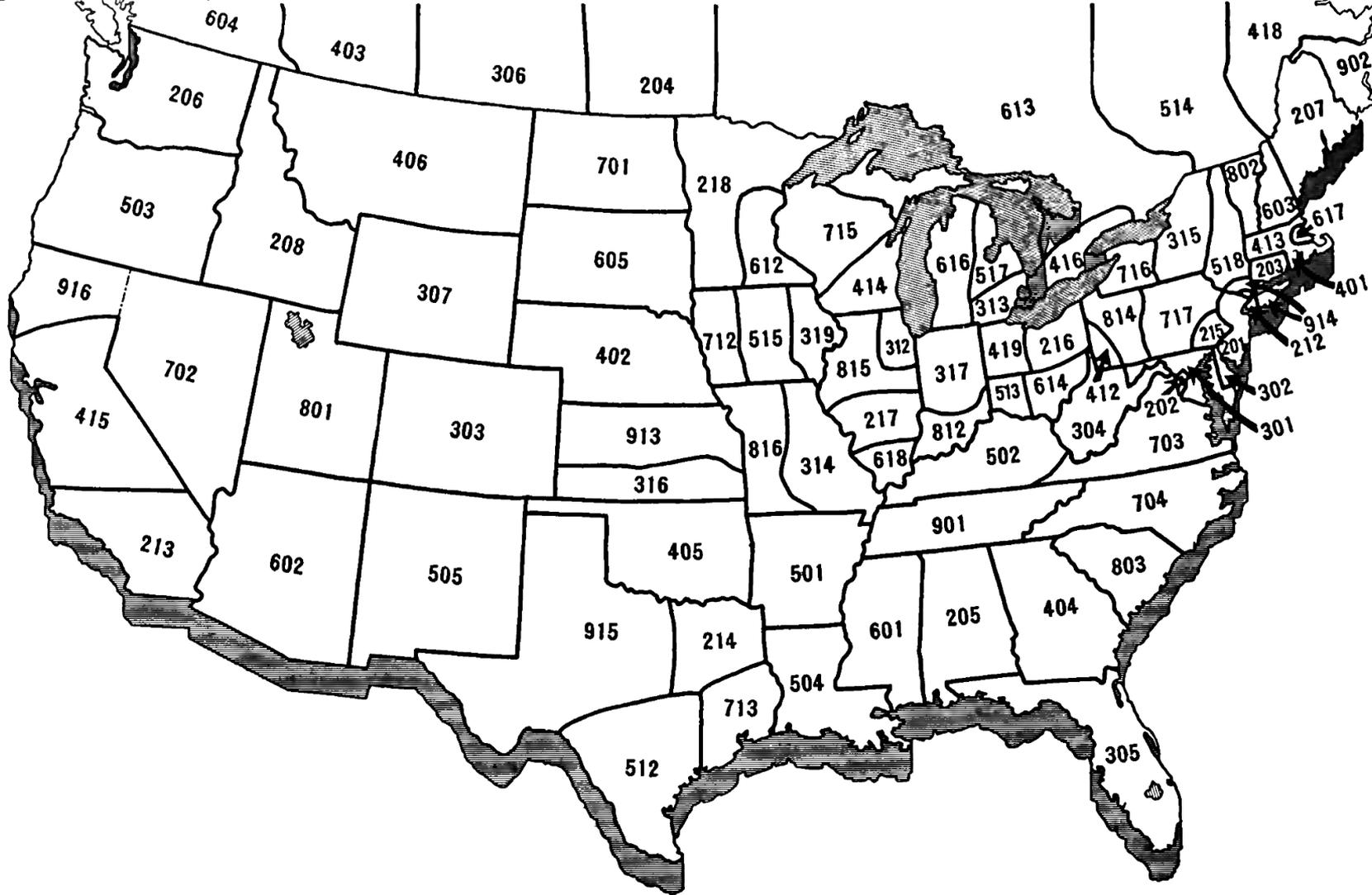
Telephone numbers have a structure to them

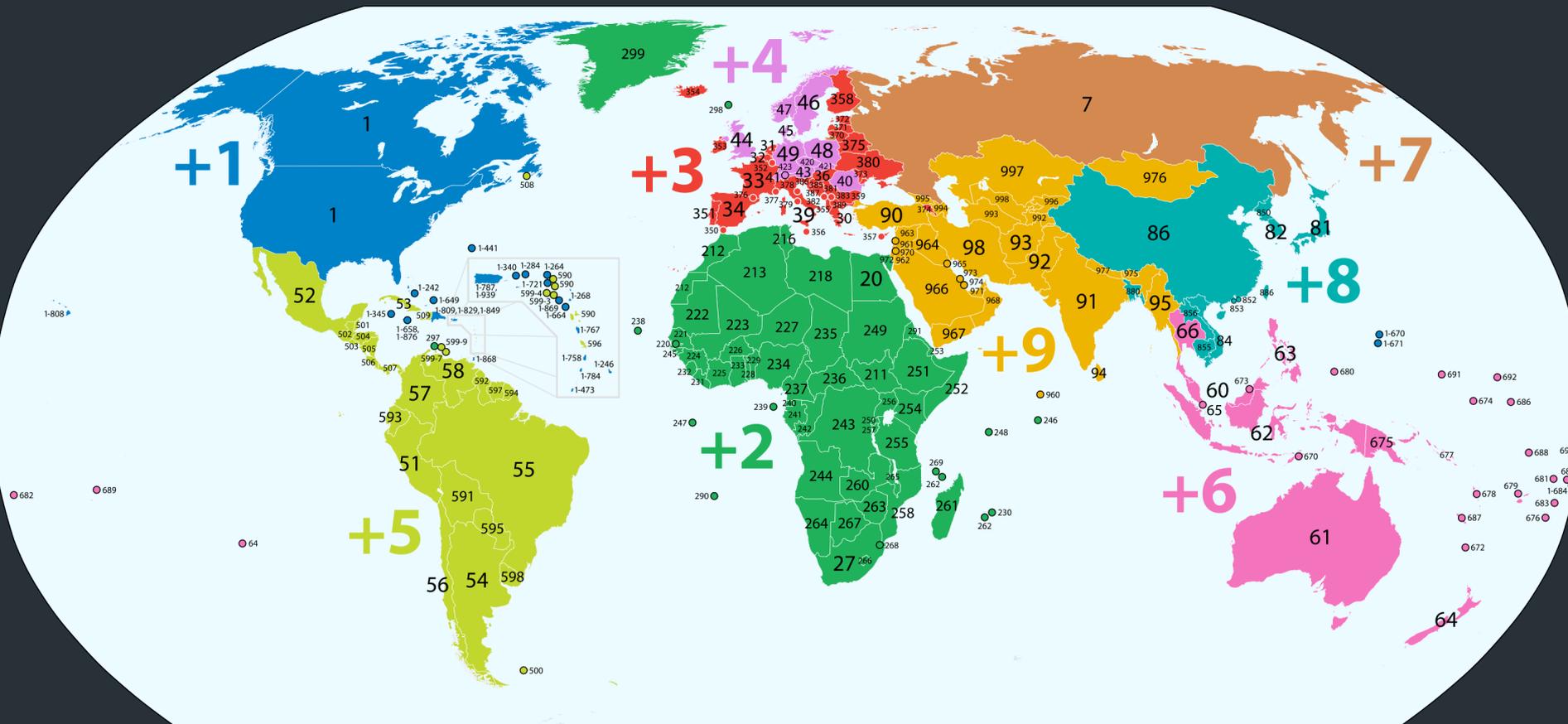
+ 1 401 863 1000

+1 212 555 4253



Part of the number tells where you are!
(or at least it did before cell phones)





... and not all numbers are the same length!

IP Addressing



128.148.16.7

IP Version 4: Each address is a 32-bit number:

128.148.16.7

10000000 10010100 00010000 00000111

Notation

- Write each byte ("octet") as a decimal number
- This is called "dotted decimal" or "dotted quad" notation

IP Addressing



128.148.16.7

IP Version 4: Each address is a 32-bit number:

128.148.16.7

10000000 10010100 00010000 00000111

Notation

- Write each byte ("octet") as a decimal number
- This is called "dotted decimal" or "dotted quad" notation

32 bits => 2^{32} possible addresses...
problem?

Conceptually: an IP address has two parts

128.148.16.7

10000000 10010100 00010000 00000111

Conceptually: an IP address has two parts

=> Network part: identifies this network to the Internet

=> Host part: identifies hosts on that network

128.148.16.7

10000000 10010100 00010000 00000111

Size of host part vs. network part can vary (more on this later)

IP Addressing

Brown owns the range:

128.148.xxx.xxx

10000000 10010100 xxxxxxxxxx xxxxxxxxxx

Network part

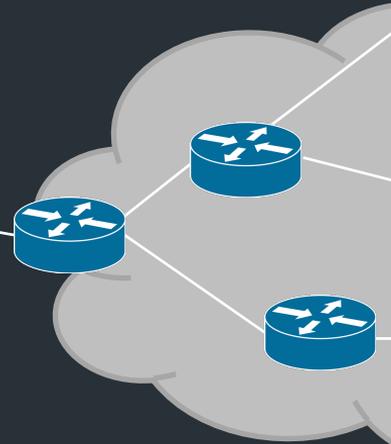
Identifies Brown (to the Internet)

Host part

Denotes individual hosts
within the Brown Network

Assigning numbers

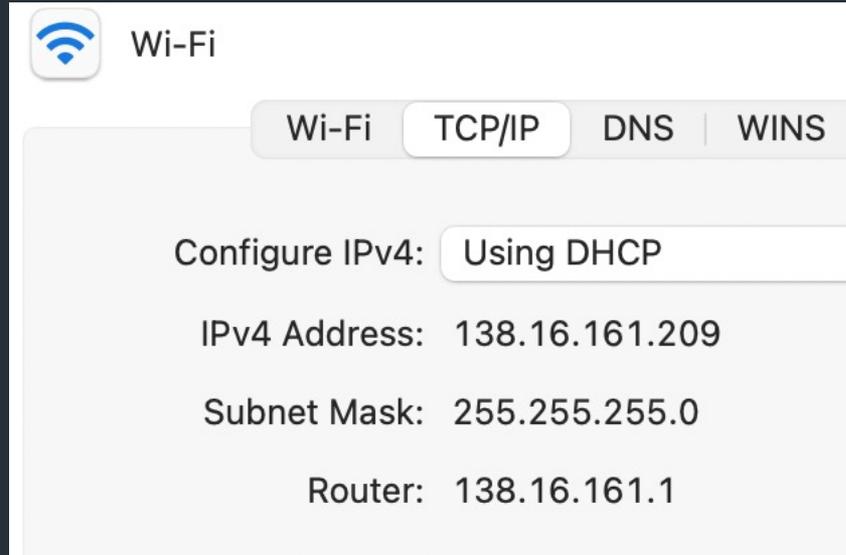
- Networks are allocated ranges of IPs by global authority (ICANN)
 - Further subdivided by regions, ISPs, ...



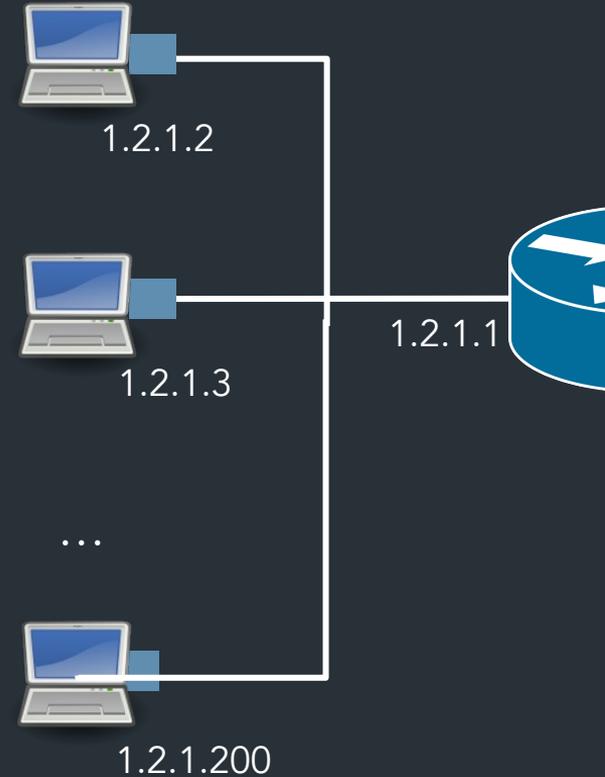
- Some IPs have special uses (eg. 127.0.0.1) 128.148.16.7

eg. Brown owns 128.148.xxx.xxx, 138.16.xxx.xxx

A typical configuration

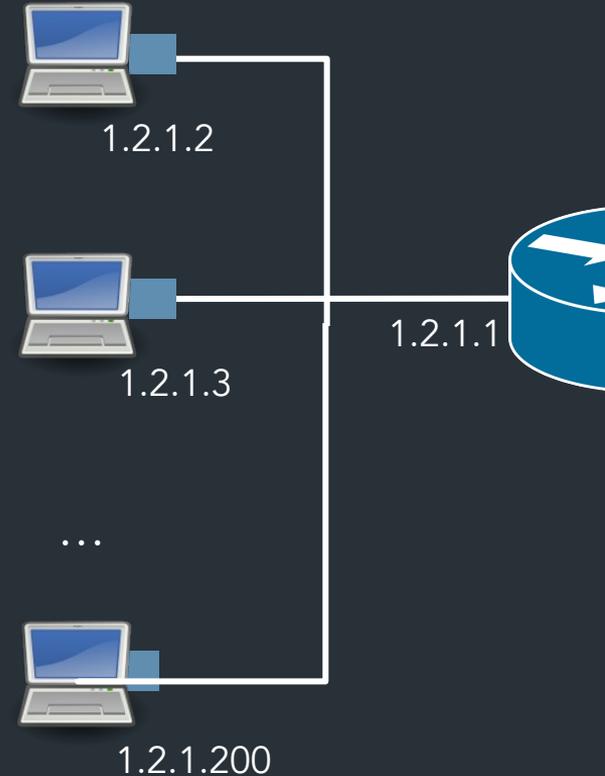


What do we do with this number?



What do we do with this number?

- Link layer: know how to communicate with devices on “your network”
- Routers: know about multiple networks => Use address to decide how to forward packets between them



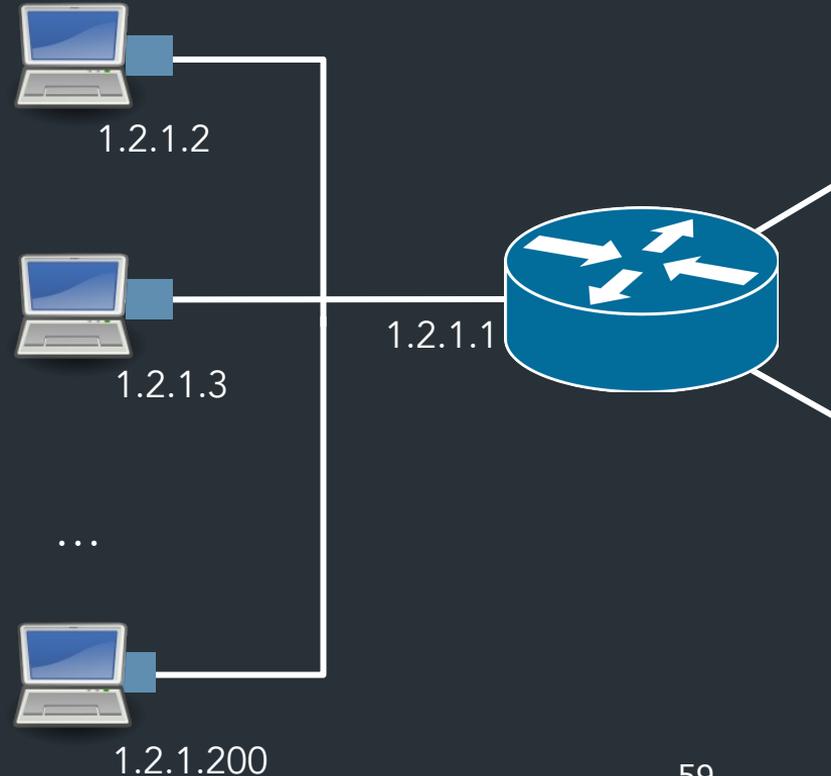
How IP forwarding works

Assume:

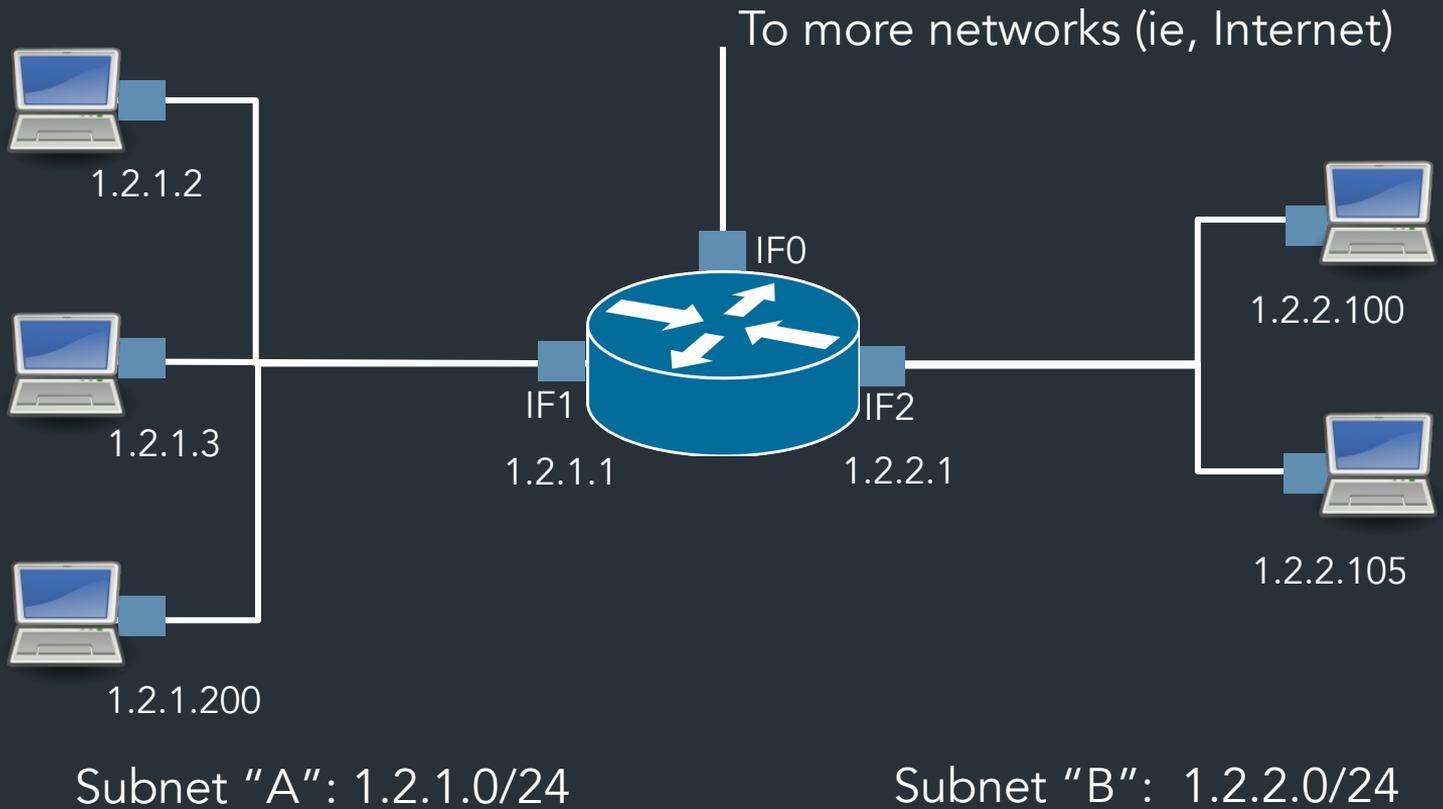
- Communicating on same network is easy—this is the link-layer's job!
- Can map IP addresses to MAC addresses (more on this later)

How to reach an address *outside* this network?

Send packets to a router, which forwards IP packets to other networks

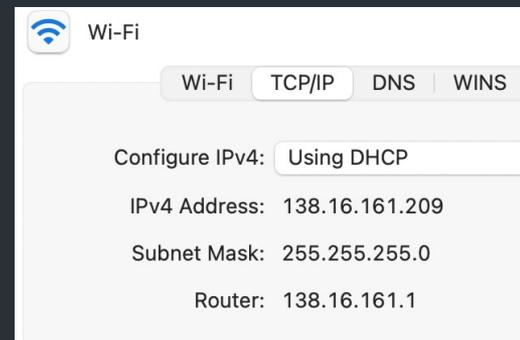


*How do we move
packets between networks?*



What does it mean to be on the same network?

All systems with an IP address have a configuration like this



138.16.161.209

Addr:

138.16.161.209

10001010 00010000 10100001 11010001

Mask:

255.255.255.0

&

11111111 11111111 11111111 00000000

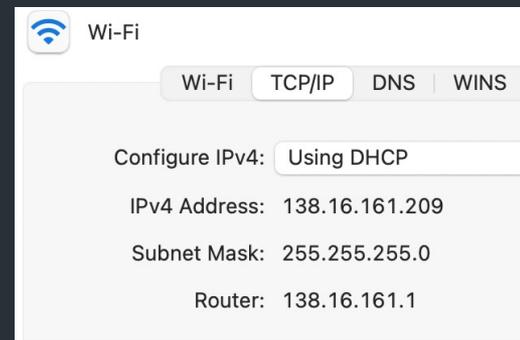
10001010 00010000 10100001 00000000

24 bits

138.16.161.0

Can also write as 138.16.161.209/24
"Prefix notation" or "CIDR notation"

All systems with an IP address have a configuration like this



138.16.161.209

Addr:

138.16.161.209

10001010 00010000 10100001 11010001

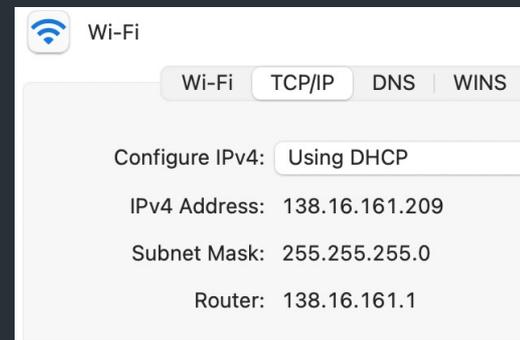
Mask:

255.255.255.0

11111111 11111111 11111111 00000000

=> Bitmask used to "filter out" which part is for hosts on the same network

All systems with an IP address have a configuration like this



138.16.161.209

Addr:

138.16.161.209

10001010 00010000 10100001 11010001

Mask:

255.255.255.0

&

11111111 11111111 11111111 00000000

10001010 00010000 10100001 00000000

24 bits

138.16.161.0

Can also write as 138.16.161.209/24
"Prefix notation" or "CIDR notation"

138.16.161.0/24 10001010 00010000 10100001 xxxxxxxx

138.16.161.204 10001010 00010000 10100001 10100001

1.2.3.4 00000001 00000010 00000011 00000100

⇒ The mask can be any size 0-32
Not just checking the first three digits!

*How do we move
packets between networks?*

A forwarding table (my laptop)

```
deemer@ceres ~ % ip route default via 10.3.128.1 dev wlp2s0
10.3.128.0/18 dev wlp2s0 proto dhcp scope link src 10.3.135.44 metric 3003
172.18.0.0/16 dev docker0 proto kernel scope link src 172.18.0.1
192.168.1.0/24 dev enp0s31f6 proto kernel scope link src 192.168.1.1
```

A routing table

```
R6#sh ip ro
Gateway of last resort is 108.34.215.1 to network 0.0.0.0

S*   0.0.0.0/0 [1/0] via 108.34.215.1
     10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks
C     10.1.0.0/24 is directly connected, wlan-ap0
L     10.1.0.2/32 is directly connected, wlan-ap0
O IA  10.1.44.1/32 [110/1001] via 10.20.30.33, 3w4d, Tunnel0
C     10.1.48.0/24 is directly connected, Loopback0
L     10.1.48.1/32 is directly connected, Loopback0
C     10.20.30.32/31 is directly connected, Tunnel0
L     10.20.30.32/32 is directly connected, Tunnel0
     108.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C     108.34.215.0/24 is directly connected, GigabitEthernet0/0
L     108.34.215.208/32 is directly connected, GigabitEthernet0/0
     172.16.0.0/16 is variably subnetted, 2 subnets, 2 masks
C     172.16.98.0/24 is directly connected, Vlan98
L     172.16.98.1/32 is directly connected, Vlan98
     172.17.0.0/16 is variably subnetted, 6 subnets, 3 masks
O IA  172.17.44.0/24 [110/1001] via 10.20.30.33, 3w4d, Tunnel0
C     172.17.48.0/24 is directly connected, Vlan20
L     172.17.48.1/32 is directly connected, Vlan20
C     172.17.49.0/25 is directly connected, Vlan50
```

A routing table

```
R6#sh ip ro
```

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, * - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP  
+ - replicated route, % - next hop override
```

```
Gateway of last resort is 108.34.215.1 to network 0.0.0.0
```

```
S* 0.0.0.0/0 [1/0] via 108.34.215.1  
10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks  
C 10.1.0.0/24 is directly connected, wlan-ap0  
L 10.1.0.2/32 is directly connected, wlan-ap0  
O IA 10.1.44.1/32 [110/1001] via 10.20.30.33, 3w4d, Tunnel0  
C 10.1.48.0/24 is directly connected, Loopback0  
L 10.1.48.1/32 is directly connected, Loopback0  
C 10.20.30.32/31 is directly connected, Tunnel0  
L 10.20.30.32/32 is directly connected, Tunnel0
```

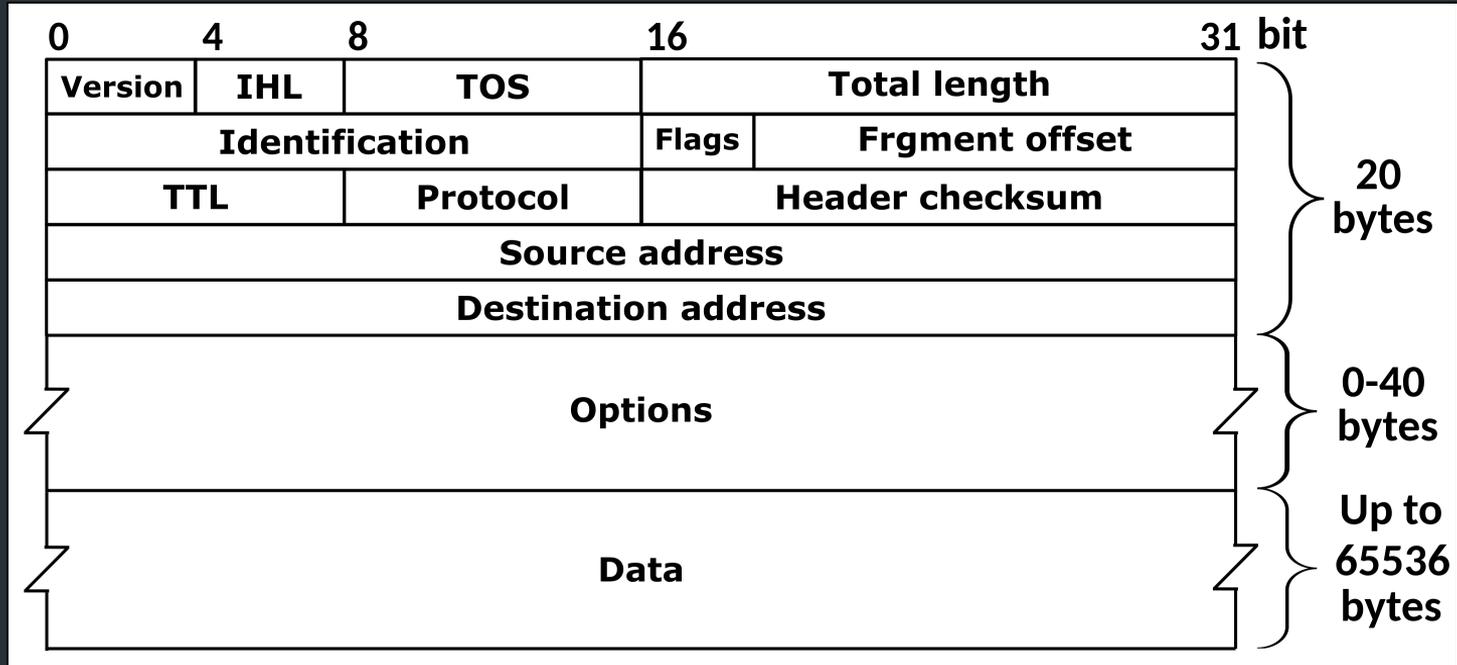
A large table

```
rviews@route-server.ip.att.net>show route table inet.0 active-path
```

```
inet.0: 866991 destinations, 13870153 routes (866991 active, 0 holddown, 0 hidden)  
+ = Active Route, - = Last Active, * = Both
```

```
0.0.0.0/0          *[Static/5] 5w0d 19:43:09  
                  > to 12.0.1.1 via em0.0  
1.0.0.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 13335 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.4.0/22        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.4.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.5.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.6.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0
```

The IPv4 Header



Defined by RFC 791
RFC (Request for Comment): defines network standard

Traceroute example

```
[deemer@Warsprite ~]$ traceroute -q 1 google.com
traceroute to google.com (142.251.40.174), 30 hops max, 60 byte packets
 1  router1-nac.linode.com (207.99.1.13)  0.621 ms
 2  if-0-1-0-0-0.gw1.cjj1.us.linode.com (173.255.239.26)  0.499 ms
 3  72.14.222.136 (72.14.222.136)  0.949 ms
 4  72.14.222.136 (72.14.222.136)  0.919 ms
 5  108.170.248.65 (108.170.248.65)  1.842 ms
 6  lga25s81-in-f14.1e100.net (142.251.40.174)  1.812 ms
```

Traceroute example

```
[deemer@Warsprite ~]$ traceroute -q 1 amazon.co.uk
traceroute to amazon.co.uk (178.236.7.220), 30 hops max, 60 byte packets
 1  router2-nac.linode.com (207.99.1.14)  0.577 ms
 2  if-11-1-0-1-0.gw2.cjj1.us.linode.com (173.255.239.16)  0.461 ms
 3  ix-et-2-0-2-0.tcore3.njy-newark.as6453.net (66.198.70.104)  1.025 ms
 4  be3294.ccr41.jfk02.atlas.cogentco.com (154.54.47.217)  2.938 ms
 5  be2317.ccr41.lon13.atlas.cogentco.com (154.54.30.186)  69.725 ms
 6  be2350.rcr21.b023101-0.lon13.atlas.cogentco.com (130.117.51.138)  69.947 ms
 7  a100-row.demarc.cogentco.com (149.11.173.122)  71.639 ms
 8  150.222.15.28 (150.222.15.28)  78.217 ms
 9  150.222.15.21 (150.222.15.21)  84.383 ms
10  *
11  150.222.15.4 (150.222.15.4)  74.529 ms
    . . .
30  178.236.14.162 (178.236.14.162)  83.659 ms
```

A routing table

```
R6#sh ip ro
```

```
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP  
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area  
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2  
E1 - OSPF external type 1, E2 - OSPF external type 2  
i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2  
ia - IS-IS inter area, * - candidate default, U - per-user static route  
o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP  
+ - replicated route, % - next hop override
```

```
Gateway of last resort is 108.34.215.1 to network 0.0.0.0
```

```
S* 0.0.0.0/0 [1/0] via 108.34.215.1  
10.0.0.0/8 is variably subnetted, 7 subnets, 3 masks  
C 10.1.0.0/24 is directly connected, wlan-ap0  
L 10.1.0.2/32 is directly connected, wlan-ap0  
O IA 10.1.44.1/32 [110/1001] via 10.20.30.33, 3w4d, Tunnel0  
C 10.1.48.0/24 is directly connected, Loopback0  
L 10.1.48.1/32 is directly connected, Loopback0  
C 10.20.30.32/31 is directly connected, Tunnel0  
L 10.20.30.32/32 is directly connected, Tunnel0
```

A large table

```
rviews@route-server.ip.att.net>show route table inet.0 active-path
```

```
inet.0: 866991 destinations, 13870153 routes (866991 active, 0 holddown, 0 hidden)  
+ = Active Route, - = Last Active, * = Both
```

```
0.0.0.0/0          *[Static/5] 5w0d 19:43:09  
                  > to 12.0.1.1 via em0.0  
1.0.0.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 13335 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.4.0/22        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.4.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.5.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0  
1.0.6.0/24        *[BGP/170] 1d 10:24:47, localpref 100, from 12.122.83.238  
                  AS path: 7018 3356 4826 38803 I, validation-state: valid  
                  > to 12.0.1.1 via em0.0
```

