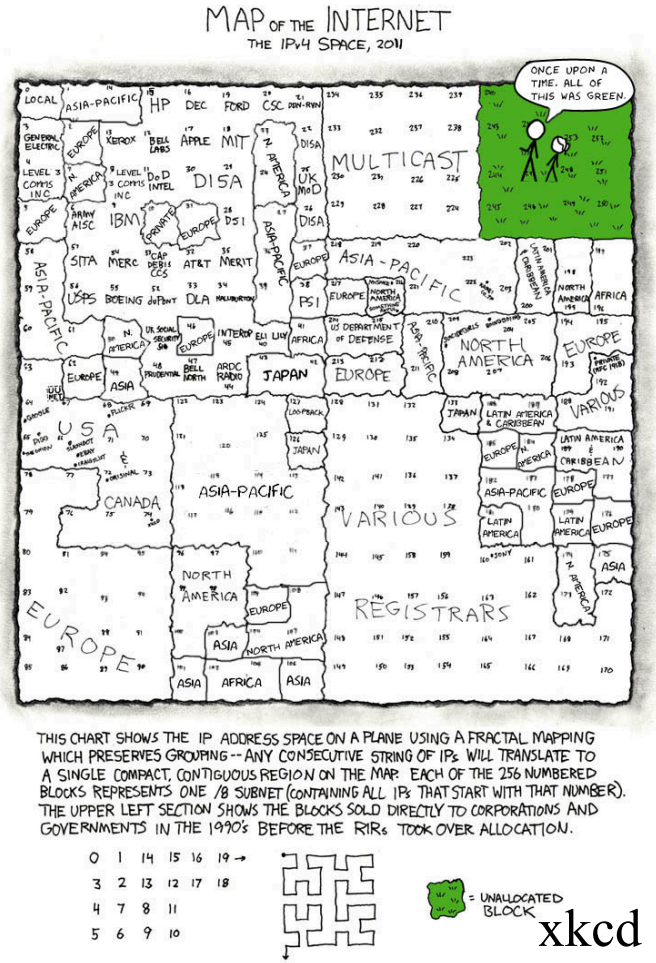
Networking and Internetworking 2

To do …

- Networks at work
- Design principles



IP and addresses

- Challenge of Internet protocol design: naming and addressing scheme and routing – IP
- Transmit datagrams between hosts, via several routers
  - Each router in the Internet implements IP-layer software to provide a routing algorithm
- IP layer must insert a physical net address to the packet – ask address resolution module
- Address resolution, technology dependent
  - For Ethernet, hard-wired into its network interface
  - ARP module in each host has a cache of <IP address,Ethernet address>
  - Cache miss, broadcast; if host is there, responds with Ethernet address

IP and addresses

- Most of the Internet today on IPv4
  - 32b IP written as four decimal numbers separated by '.'
  - Design of the Internet address space

Class A	0	Network ID 7b	Host ID 24b
Class B	10	Network ID: 14b	Host ID: 16b
Class C	110	Network ID: 21b	Host ID: 8b
Class D	1110	Multicast address 28b	
Class E	1111	Unused: 28b	

  - Classes A, B and C to meet the needs of different type of organizations (e.g., A, 2^{24} hosts, reserved for very large ones)

Internet address space

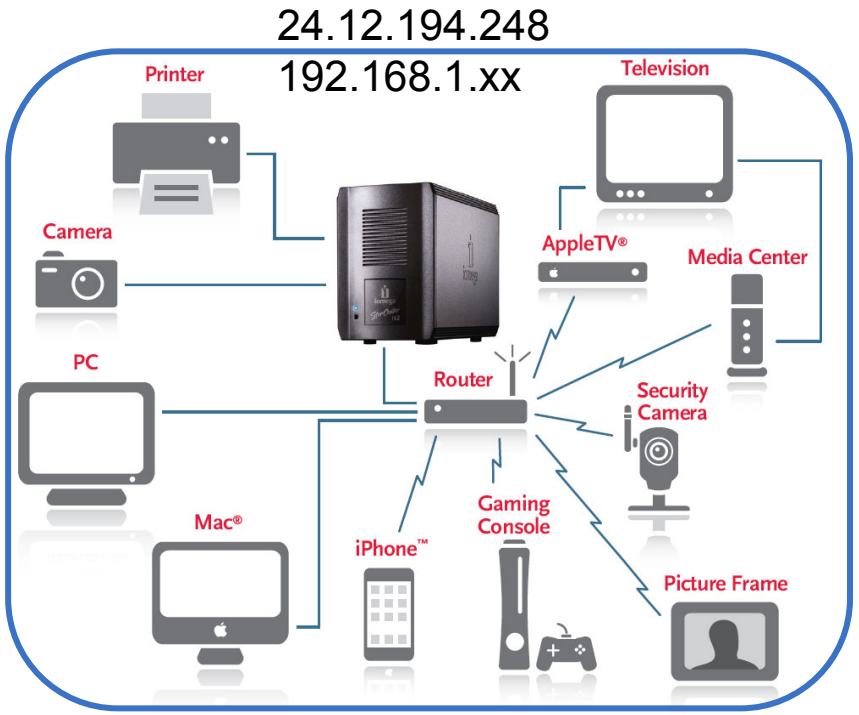
- Allocation scheme turned out not to be very effective
  - *Can't predict how much you will need, so ask for a lot!*
- Since host address includes network id
  - More than one network for a host, more than one IP
  - Moving host to another network, change of IP
- ~1990, a three-way solution
  - Develop of a new IP protocol and addressing scheme: IPv6
  - Modify address allocation scheme – Classless Inter-domain Routing (CIDR)
  - Enable unregistered computers to access the Internet directly: NAT or Network Address Translators

Scaling IPv4 with NAT

- Not all devices need a unique IP address
  - Sharing one “global” IP address at home
  - Router has a “global” IP address from ISP
  - Each machine has a “local” IP address via DHCP
  - NAT enabled router maintains an address translation table

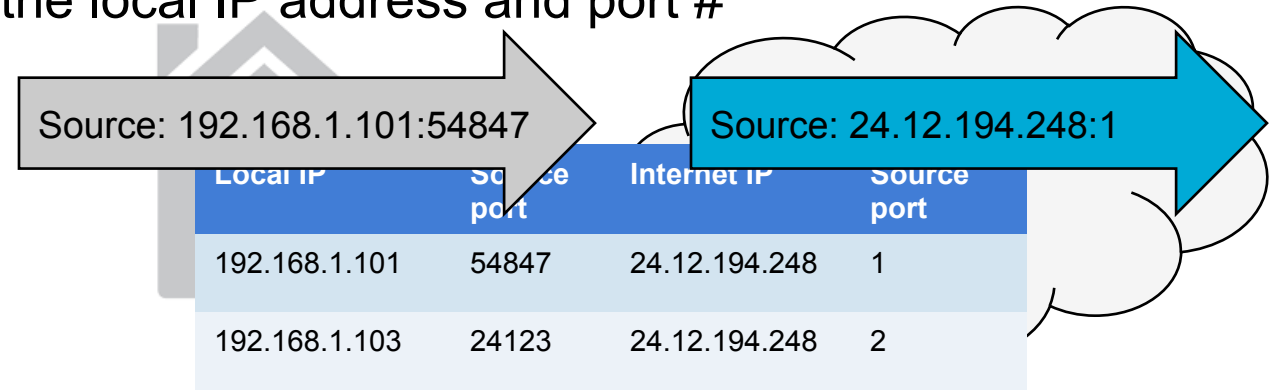
NAT Translation Table

Local IP	Source port	Internet IP	Source port
192.168.1.101	54847	24.12.194.248	1
192.168.1.103	24123	24.12.194.248	2



A simple NAT

- Machine ➔ router
  - Router stores the local IP addr and source port #
  - Table entry indexed by a virtual port #
- Router ➔ outside
  - Put the router IP addr and virtual port # in the packet
- Outside ➔ router
  - Reply to the router IP addr and virtual port #
- Router ➔ machine
  - Use the virtual port # to find table entry
  - Forward to the local IP address and port #



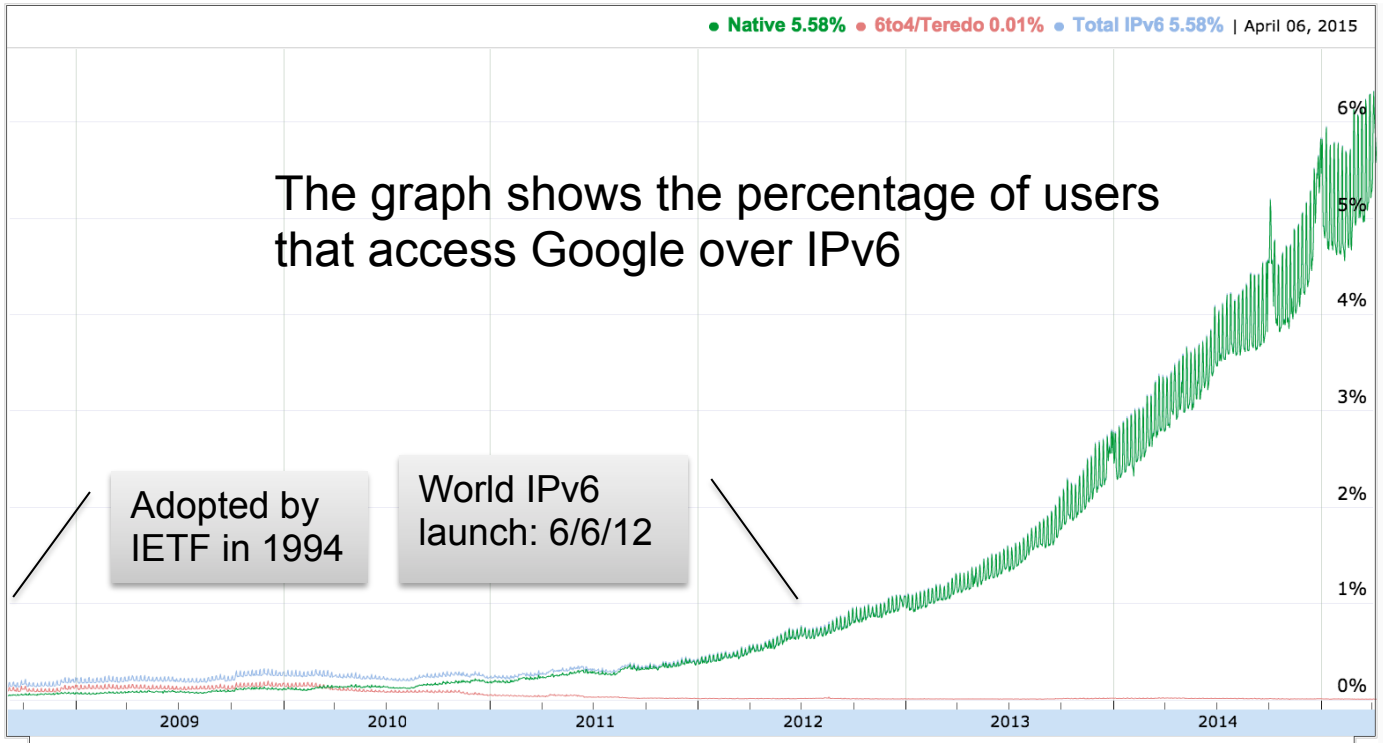
Local IP	Source port	Internet IP	Source port
192.168.1.101	54847	24.12.194.248	1
192.168.1.103	24123	24.12.194.248	2

IPv6 – a more permanent solution

- IP addresses: 128 bits (16 bytes)
  - 3×10^{38} addresses (~7×10^{23} addresses per square meter!)
- Routing speed
  - No data checksum – done at higher levels
  - No packet fragmentation once in transit – determine smallest MTU before transmitting the packet
- Support for real-time and special services with traffic classes and flow levels
- "Next" header field for future evolution
- Security at the IP-level
  - Authentication and encrypted security payload through extension header types

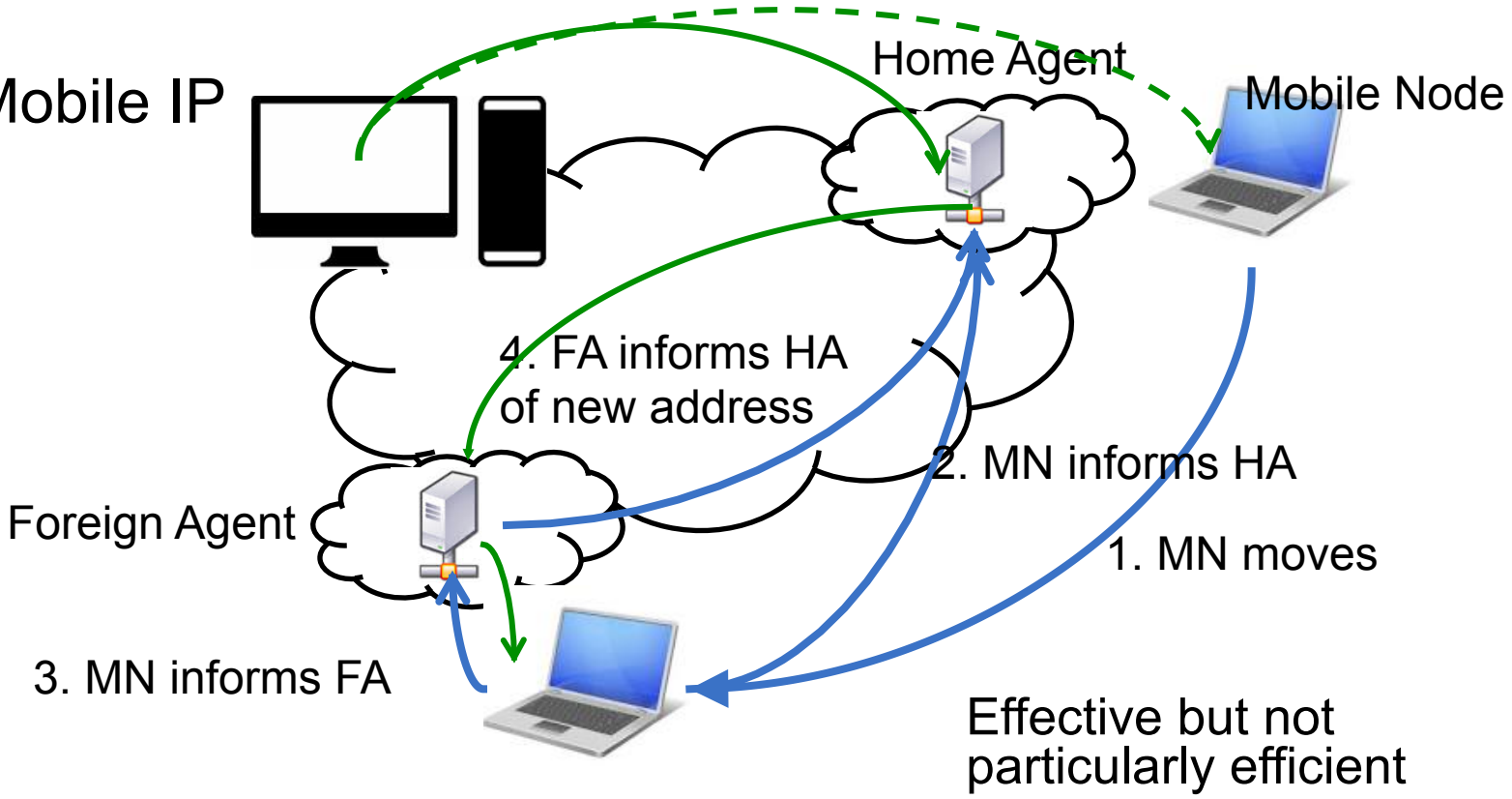
IPv6 – a more permanent solution

- IPv6 adoption
  - Backward compatibility: IPv6 addresses include IPv4 addresses
  - Islands of IPv6 nets, traffic tunnels though other IPv4 nets
- Picking up speed



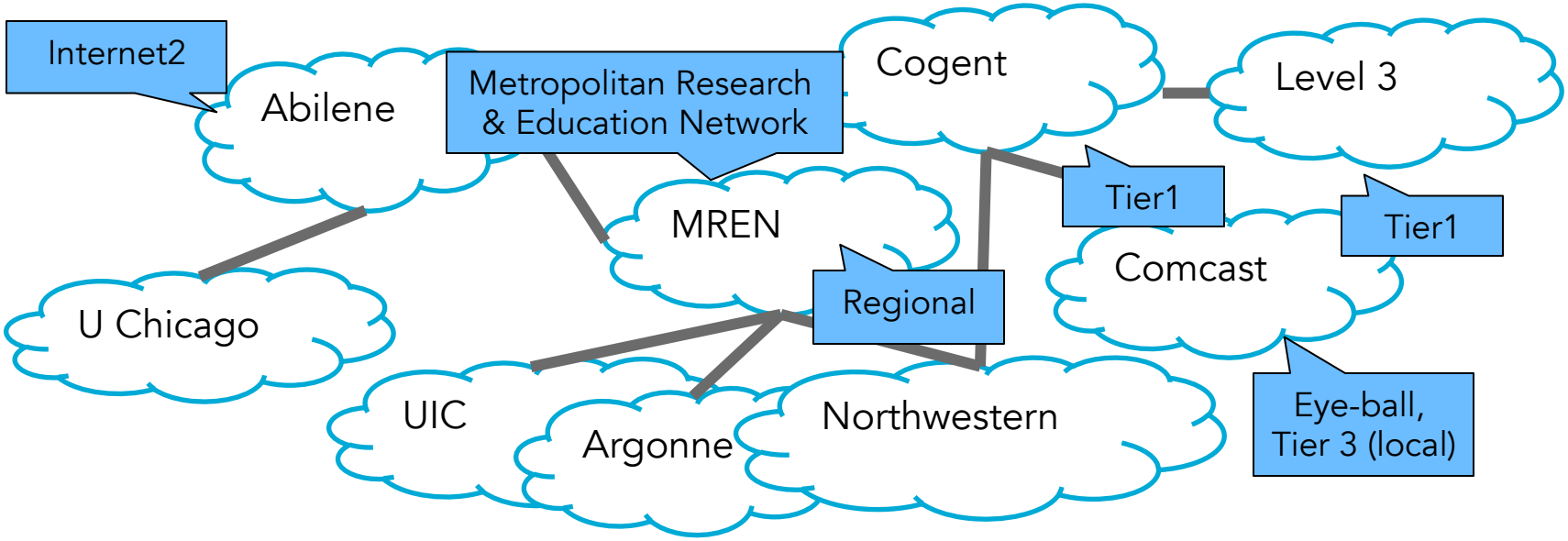
Mobility

- When moving
  - Accessing a service does not require a host to keep a single IP address as it moves – DHCP
  - Continuous access is an issue (e.g. to send you updates)
- Mobile IP



Internet inter-domain system

- ASes of different sizes, functions, business objectives
- ASes exchanged traffic at connection points – peering or exchanged points
  - Connections formed by establishing a link between routers in each AS, called gateway routers
- Not all ASes are equal, same for inter-AS relationships
  - Transit and peering



Internet routing

- Internet routing is hierarchical
  - Two levels intra-AS (intra-domain) and inter-AS
  - Both (1) for scalability (too large, too many routers) and (2) for independence (each AS mange their network independently)
- Intra-AS
  - Link-state routing with OSPF, IS-IS
- Inter-domain routing, a bit more challenging
  - Economic and policy issues

    e.g. “I prefer to send traffic via AS X than Y, but I’ll use Y if it is the only path, and never want to carry traffic between X and Y”

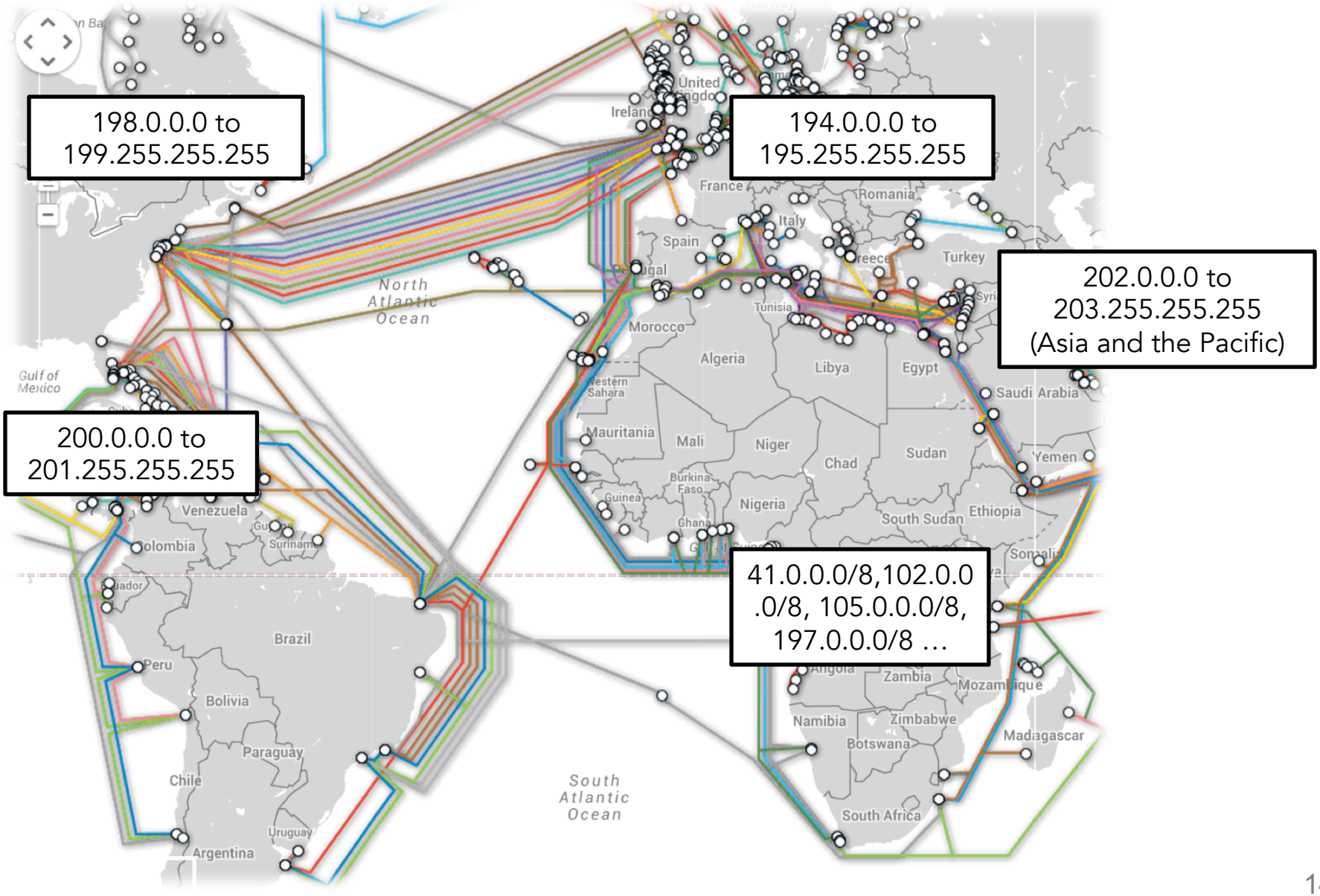
Intra-domain routing with BGP

- Some highlights
- BGP advertises complete paths as an AS list
  - Necessary to enable policy decision and detect loops
- Runs on top of TCP
  - Reliability is covered but need *KEEPALIVE*
- To participate in a BGP session
  - Router opens a connection, sends *OPEN* msg – exchanging tables of all active routers
- BGP *UPDATE*
  - A BGP router will advertise one of several routes to a destination, selected according to its policies
  - It can also send a negative advertisement – withdrawn route

Scale and routing tables

- A full routing table in every routing – too large
  - Can't do it for all routers in today's Internet
  - Two responses: regional locations and default routes
- Regional locations
  - Prior to 1993 nothing could be inferred from an IP about its location; a regional split
    - 194.0.0.0 to 195.255.255.255 Europe
    - 198.0.0.0 to 199.255.255.255 North America
    - 200.0.0.0 to 201.255.255.255 Central and South America
    - 202.0.0.0 to 203.255.255.255 Asia and the pacific
    - …
  - Works because it matches topological regions of the Internet
    - A router outside Europe can have a single entry for Europe IP range, sends all packets to nearest Europe gateway

Submarine cable map (fraction) 2015

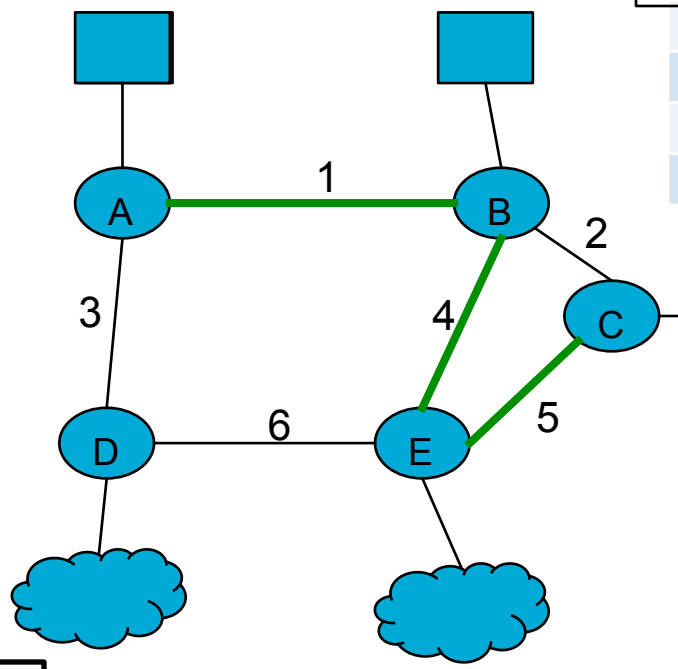


Source: telegeography

And default routes

Routing Table for A

To	Link	Cost
A	Local	0
B	1	1
C	1	2
D	3	1
E	1	2



Routing Table for B

To	Link	Cost
A	1	1
B	Local	0
C	2	1
D	1	2
E	4	1

Packets addressed to A reach via E and B with an extra hop

Routing Table for D

To	Link	Cost
A	3	1
B	3	2

Packets addressed to D reach via default without overhead

Routing Table for E

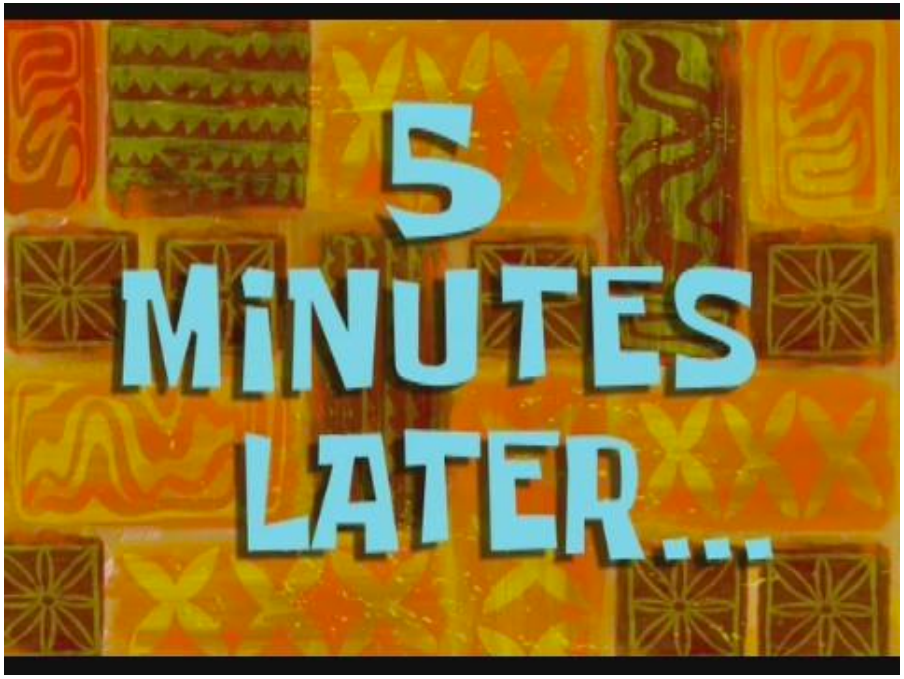
To	Link	Cost
A	4	2
B	4	1
C	5	1
D	6	1
E	Local	0

Routing Table for C

To	Link	Cost
A	2	2
B	2	1
C	Local	0
E	5	1
Default	5	-

Back in 5’

- Internet design and implications



Internet design

- Understanding design history as context to current design extensions
- Fundamental goal
  - Effective multiplexed use of existing interconnected networks
    - Connecting ARPANET with ARPA packet radio network
  - Technique for multiplexing: packet switching, store&forward
  - Alternative – a more integrated unified system
    - Problem is how to add existing network architectures

- *A packet switched communication facility in which a number of distinguishable networks are connected together using packet communication processors called gateways which implement a store and forward packet forwarding algorithm*

Design principles – second level goals

1. Survivability
   Communication must survive loss of networks or gateways
2. Support multiple types of communication services
3. The architecture must accommodate a variety of networks
4. … permit distributed management of its resources
5. … be cost effective
6. … permit host attachment with low level of efforts
7. Allow resource accountability

Not just a checklist! Order matters (think of the order for military or commercial goals)

Survivability

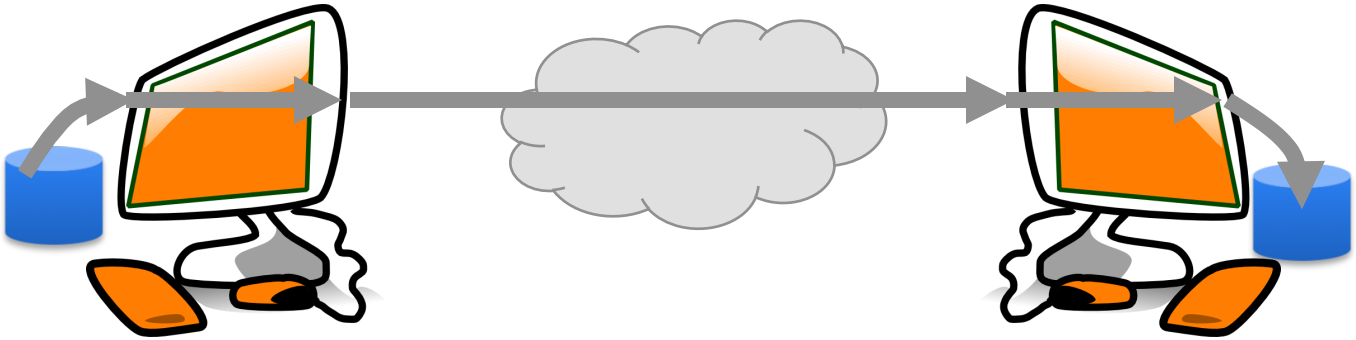
- If two parties are communicating and something causes the Internet to be disrupted and reconfigured
  - Entities should be able to continue without re-establish or reset the high-level state of their conversation
  - i.e., at the top of the transport layer there's only one failure – total partition
- State of ongoing conversation must be protected
  - E.g., # of packets transmitted, # of acks
- Design: keep state at the endpoints
  - Fate-sharing – it's ok to loose the sate associated with an entity if the entity itself is lost
  - Alternative: replication state in the network
    - Can only protect against a certain # of failures (= #replicas)
    - Hard to do well at scale

End-to-end argument

- General argument in system design
- when it is applicable:
  - "The function … can completely and correctly be implemented *only* with the knowledge and help of the application standing at the endpoints of the communication system…"
- the consequence:
  - "… Therefore, providing that questioned function as a feature of the communication system itself is not possible…"
- the exception:
  - "… (Sometimes an incomplete version of the function provided by the communication system may be useful as a performance enhancement.)"

E2e arguments – careful file transfer

- Transfer file from HD on A to HD on B



- What can go wrong?
  - Disk error
  - SW error (OS, FTP, network driver)
  - HW error
  - Communication system
  - System crash

E2e – careful file transfer

- Solution 1: reinforce every step
  - Duplicate copies, time out and retry, redundancy, …
  - Reduce probability of each threat to an acceptable small value
  - Could be hard to make each step full-proof, inefficient, everyone pays …
- Solution 2: e2e check and retry
  - Store file w/ checksum, destination computes checksum after transfer and origin compares
  - If check fails, redo everything … potentially $$$
- Solution 3: Both
  - Point-to-points checks (as IP and TCP), e2e checksums, reduce frequency of problem

E2e – identifying the ends

- Key and maybe not that easy
- Consider Voice over IP service
  - Should the computers be the ends?
  - Or the people? (could you *please repeat that?)*
  - What if you are leaving a message?

The person you are trying to reach is not available right now. Please leave a message after the beep.

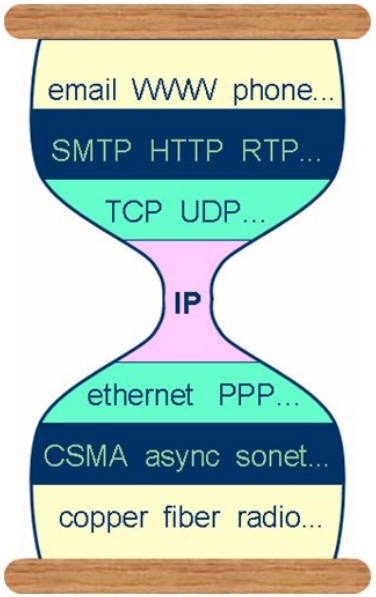
- E2e argument is not an absolute, but a design tool

Types of services

- Support different types of services at transport level
  - ≠ services with ≠ requirements (latency, reliability, …)
- Traditional type – bidirectional reliable (TCP) – initially thought to be enough for everything
  - Not quite, first example XNET – cross-Internet debugger
    - Under stress or failure conditions, reliable communication may prevent communication at all!
  - Another example, real time delivery of digital speech – no reliable, but low jitter
    - The most serious source of delay – mechanism for reliability
- Clearly we needed more than one transport
  - Split TCP/IP, up to this point a single protocol
  - IP providing a building block to build a variety of type of services – datagram (“best effort”)
  - UDP to provide app-level interface to basic datagram service

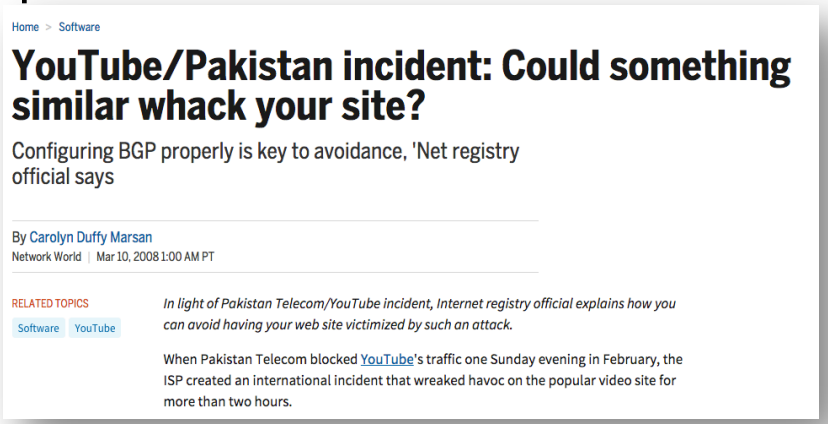
Variety of networks

- Important to be able to incorporate variety of networks
  - Key for this is IP – “narrow waist”
- Make minimal assumptions about the function that the network underneath can provide – transport a packet
- Services explicitly not assumed
  - Reliable or sequenced delivery
  - Network level broadcast
  - Priority ranking
  - Support for multiple types of services
  - …
  - Hide underlying technology from applications
  - Many protocols at app or transport layer, many communication technologies underneath



Other goals

- Lower in importance and so less effectively met
  4. **… permit distributed management of its resources**
  5. … be cost effective
  6. … permit host attachment with low level of efforts
  7. Allow resource accountability
- Kind of; various organization manage their own gateways, for instance
  - But management is very error-prone



Home > Software

YouTube/Pakistan incident: Could something similar whack your site?

Configuring BGP properly is key to avoidance, 'Net registry official says

By Carolyn Duffy Marsan
Network World | Mar 10, 2008 1:00 AM PT

RELATED TOPICS
Software YouTube

In light of Pakistan Telecom/YouTube incident, Internet registry official explains how you can avoid having your web site victimized by such an attack.

When Pakistan Telecom blocked YouTube's traffic one Sunday evening in February, the ISP created an international incident that wreaked havoc on the popular video site for more than two hours.

Other goals

- Lower in importance and so less effectively met
  4. … permit distributed management of its resources
  5. … be cost effective
  6. … permit host attachment with low level of efforts
  7. Allow resource accountability
- Header of Internet packets are long (40B) and if short messages are sent, that's just overhead (remote login packets: 1B of data + 40B of header
- Retransmission end-to-end – simpler network interface code but lower efficiency

Other goals

- Lower in importance and so less effectively met
  4. … permit distributed management of its resources
  5. … be cost effective
  6. **… permit host attachment with low level of efforts**
  7. **Allow resource accountability**
- Cost of attaching a host is higher than in other architectures
  - Need to implement things like ack and retransmission in every host; initially a source of anxiety
  - Poor implementation may hurt not just the host but the network as well
- Initially a goal but not in the current Internet
  - Several efforts focused on that at different points

Summary

- Internet organization, protocols and routing – complex and changing
  - But some changes have become hard to implement (ossification)
  - And we are still trying to understand this evolving entity
  - Ongoing work on understanding the current Internet and designing the future one
- Next – an overview of remote invocation paradigms (request-reply, RPC and RMI)