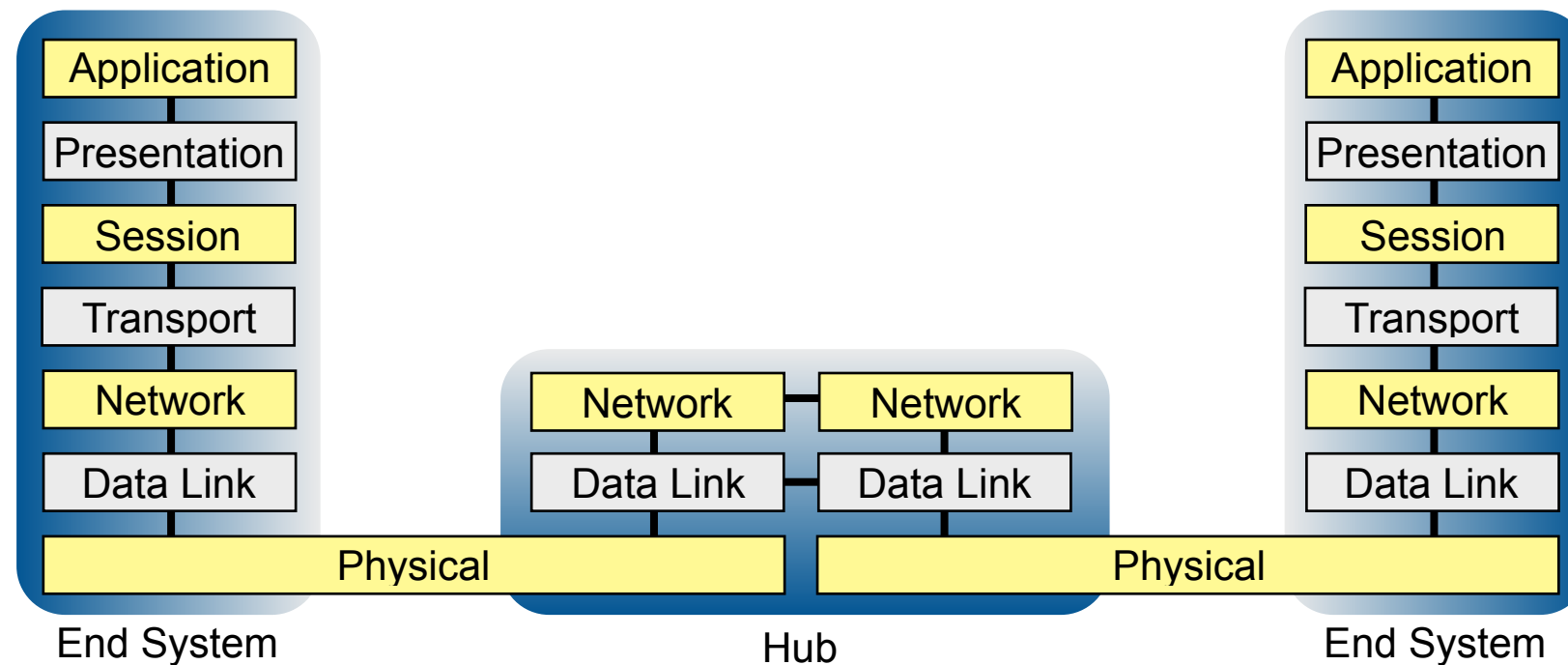


Internetworking

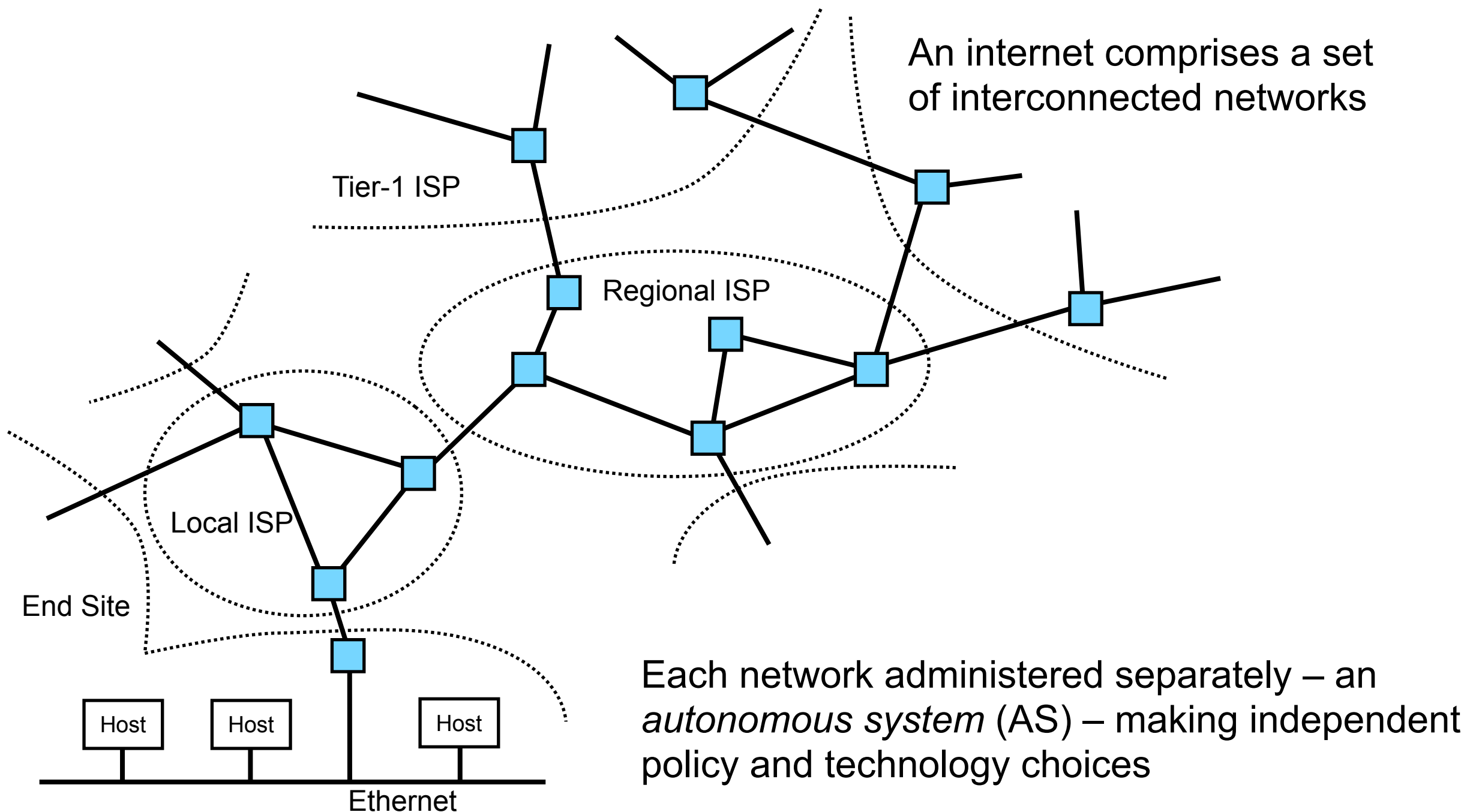
Networked Systems (H) Lecture 4

Role of the Network Layer



- Network layer is first end-to-end layer in the OSI reference model
- Responsible for end-to-end delivery of data:
 - Across multiple link-layer hops and technologies
 - Across multiple *autonomous systems*
 - Building an *Internet*: a set of interconnected networks

Interconnecting Networks



Components of *an* Internet

- A common end-to-end network protocol
 - Provide a single seamless service to transport layer
 - Delivery of data packets/provisioning of circuits
 - Addressing of end systems
- A set of gateway devices (a.k.a. *routers*)
 - Implement the common network protocol
 - Hide differences in link layer technologies
 - Framing, addressing, flow control, error detection and correction
 - Desire to perform the least amount of translation necessary

The Internet

- The globally interconnected networks running the *Internet Protocol* (IP)
 - 1965: Concept of packet switching
 - Paul Baran (RAND), Donald Davies (NPL)
 - 1969: Wide-area packet networks
 - ARPANET (US), CYCLADES (France)
 - 1973: First non-US ARPANET sites
 - UCL
 - 1974: Initial version of the Internet Protocol
 - Vint Cerf and Robert Kahn
 - 1981: Access to ARPANET broadened to non-DARPA-funded sites
 - NSF funds access for universities; production internetworking starts
 - 1983: Network switched to IPv4
 - 1992: Development of IPv6 starts
 - Initial IETF IPng effort led by Allison Mankin and Scott Bradner



Paul Baran



Donald Davis



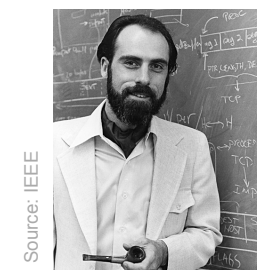
Leonard Kleinrock



Louis Pouzin



Peter Kirstein



Vint Cerf



Robert Kahn

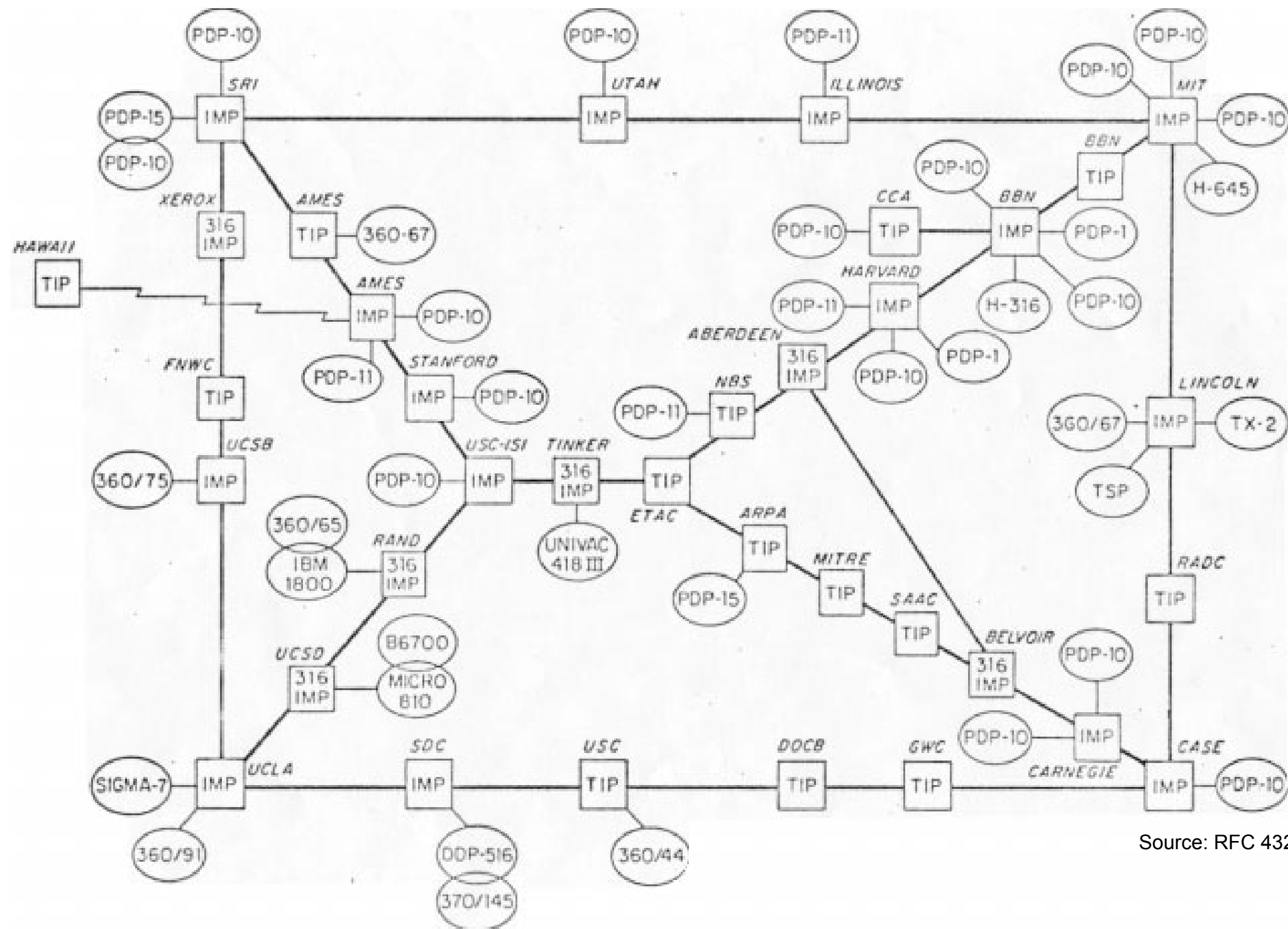


Allison Mankin



Scott Bradner

ARPA Network Map, December 1972



Source: RFC 432

Aside: Should “Internet” be Capitalised?

- Yes, when referring to “the Internet” – the global internetwork we happened to build; since proper nouns are always capitalised in English
- No, when referring to “an internet” – the concept of an internetwork; one of many possible networks, since common nouns are not capitalised in English

INTERNET HISTORIES, 2017
VOL. 1, NO. 3, 203–218
<https://doi.org/10.1080/24701475.2017.1339860>

 **Routledge**
Taylor & Francis Group

ORIGINAL ARTICLE [Check for updates](#)

What is “internet”? The case for the proper noun and why it is important

Morten Bay 

Department of Information Studies, University of California Los Angeles, Los Angeles, CA, USA

ABSTRACT
Academics, style manual editors and others have recently pushed for an elimination of the capitalisation of the word “internet”. This choice may have consequences that reach far beyond language and spelling, as it lends authority to the claim that there could be more than one “internet”, which in turn is based on a historical narrative that is not necessarily accurate. By first exploring the meaning of the word “internet” and subsequently tracing its origins, this article shows how “internet” evolved from an adjective describing a class of networking activities into a proper noun defining the foundation of the current “internet” as early as 1976. It is shown how the use of “internet” as a common noun emerges post-hoc and may have commercial origins rather than historical. The article concludes by showing how both the current, popular, broad definition of “internet”, as well as its historical roots, make the plural use of the term impossible, and why it should only be considered a proper noun, written as Internet.

ARTICLE HISTORY
Received 30 January 2017
Accepted 1 June 2017

KEYWORDS
Internet; history; noun; definition; ARPA; PARC

Introduction

What is more correct: Internet or internet? Whether the word is capitalised or not may seem trivial to the casual reader or the non-native English speaker. However, the importance of this discussion reaches far beyond media style guides and correct spelling. It also has major consequences for how we perceive the history of networks. As I will show in the following, the choice of Internet vs. internet can be seen as a manifestation of a dispute between prominent, but competing historical narratives, one of which I will show to be less credible than the other.

From this point forward, I will use the form “internet” when discussing the term in question. I use “internet” in quotation marks and un-capitalised to be able to discuss the object of study in a somewhat neutral manner, until the exploration in this article is presented in full. The form “internet” should therefore not be seen as representing any lingual choice, but merely serves as a placeholder for the object of discussion until the full argument has been explored.

CONTACT Morten Bay  mortenbay@ucla.edu

© 2017 Informa UK Limited, trading as Taylor & Francis Group

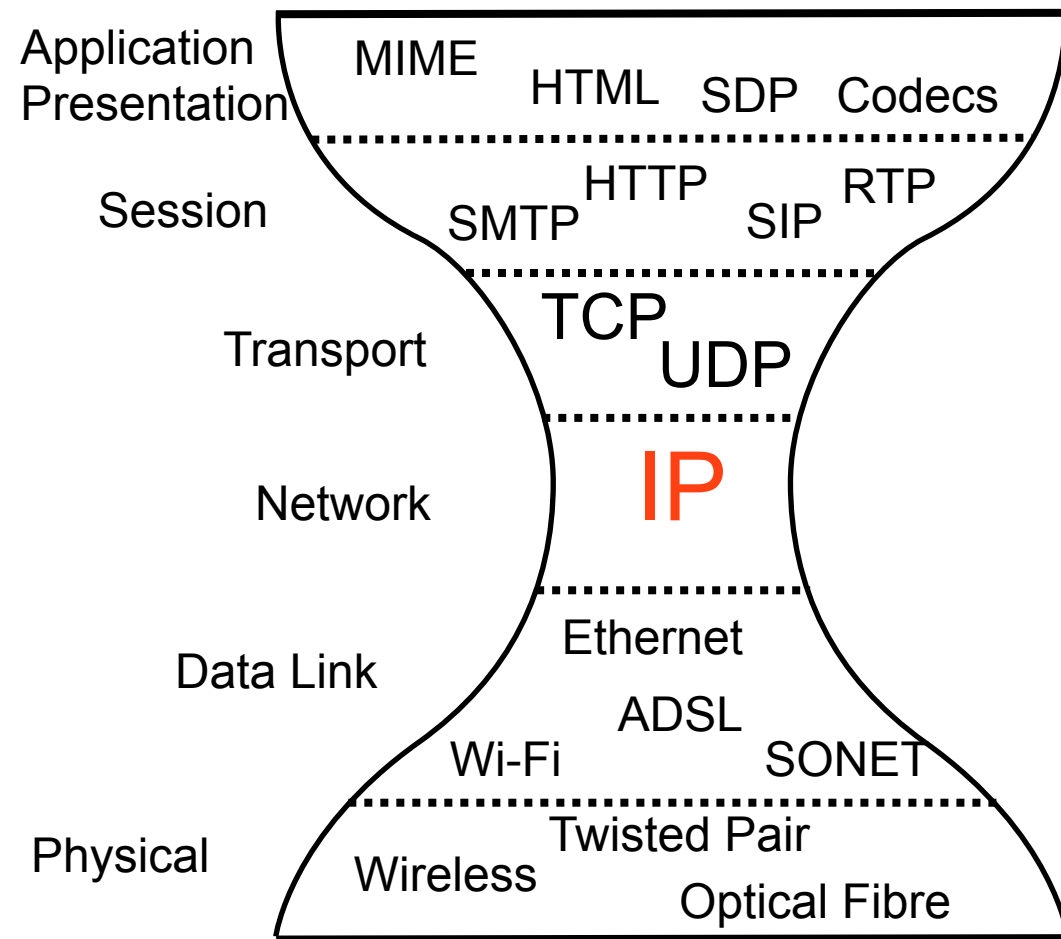
M. Bay. What is “internet”? the case for the proper noun and why it is important. Internet Histories, 1(3):203–218, 2017.<https://doi.org/10.1080/24701475.2017.1339860>

The Internet Protocol (IPv4 and IPv6)

The Internet Protocol

- IP provides an abstraction layer
 - Transport protocols and applications above
 - Assorted data link technologies and physical links below
 - A simple, best effort, connectionless, packet delivery service
 - Addressing, routing, fragmentation and reassembly

Basic Concepts

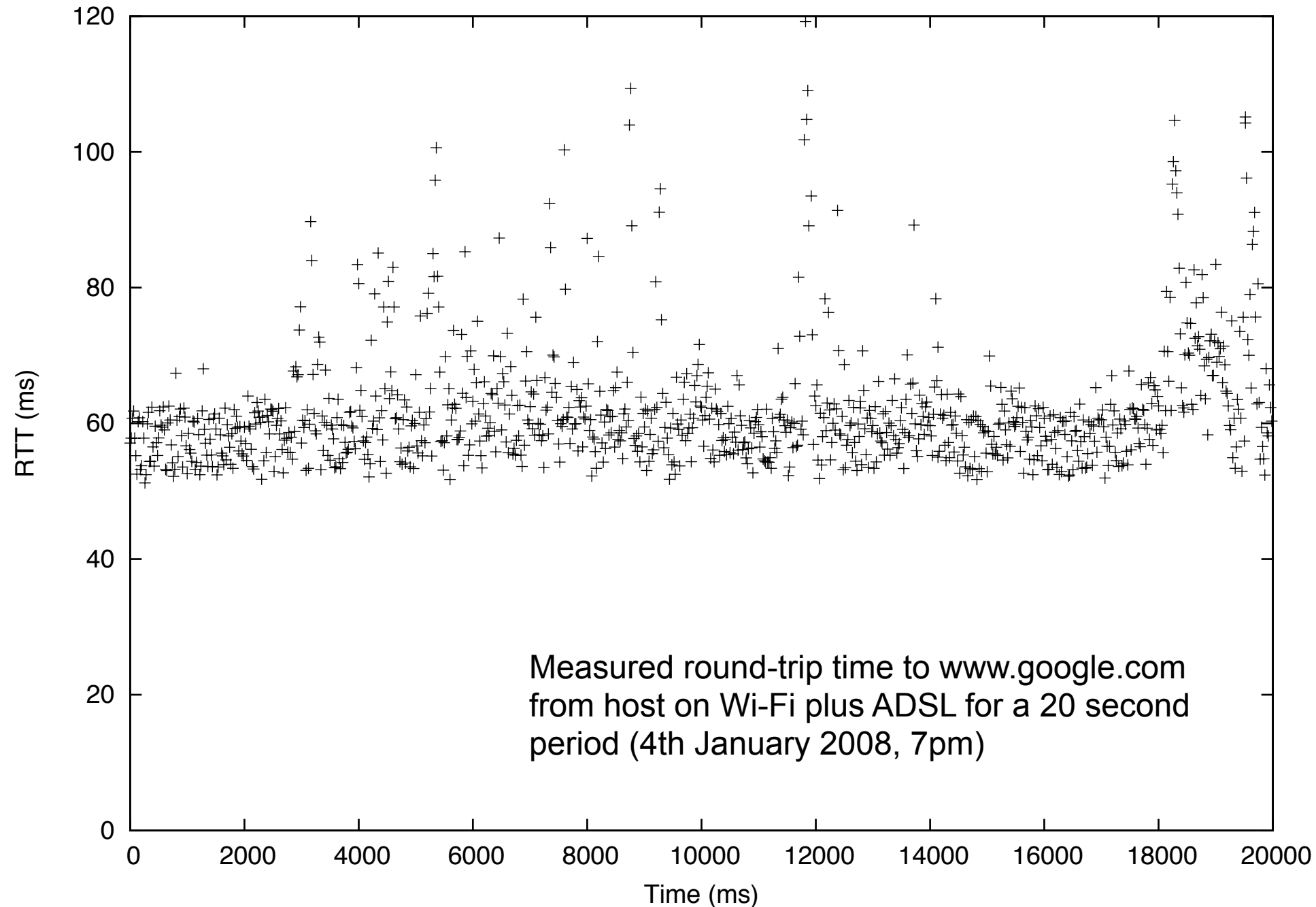


- Global inter-networking protocol
- Hour glass protocol stack
 - Single standard network layer protocol (IP)
 - Packet switched network, best effort service
 - Uniform network and host addressing
 - Uniform end-to-end connectivity (subject to firewall policy)
 - Many transport & application layer protocols
 - Range of link-layer technologies supported

IP Service Model

- Best effort, connectionless, packet delivery
 - Just send – no need to setup a connection first
 - Network makes its *best effort* to deliver packets, but provides no guarantees
 - Time taken to transit the network may vary
 - Packets may be lost, delayed, reordered, duplicated or corrupted
 - The network discards packets it can't deliver
 - Easy to run over any type of link layer
 - Fundamental service: can easily simulate a circuit over packets, but simulating packets over a circuit difficult

Best Effort Packet Delivery



Versions of the Internet Protocol

- Two versions of IP in use:
 - IPv4 – the current production Internet
 - IPv6 – the next generation Internet
- IPv5 was assigned to the Internet Stream Protocol
 - An experimental multimedia streaming protocol developed between 1979 and 1995 [<http://www.ietf.org/rfc/rfc1819.txt>], but no longer used

IPv4 Packet Format

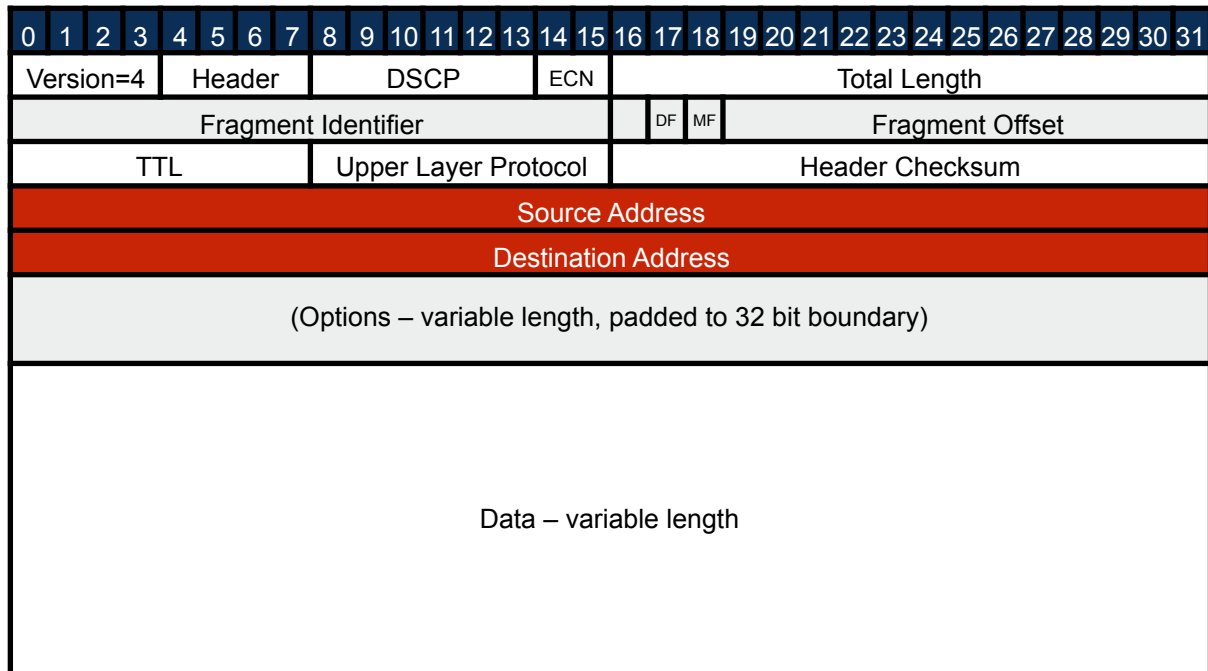
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version=4				Header				DSCP				ECN		Total Length																	
Fragment Identifier																	DF	MF	Fragment Offset												
TTL								Upper Layer Protocol								Header Checksum															
Source Address																															
Destination Address																															
(Options – variable length, padded to 32 bit boundary)																															
Data – variable length																															

IPv6 Packet Format

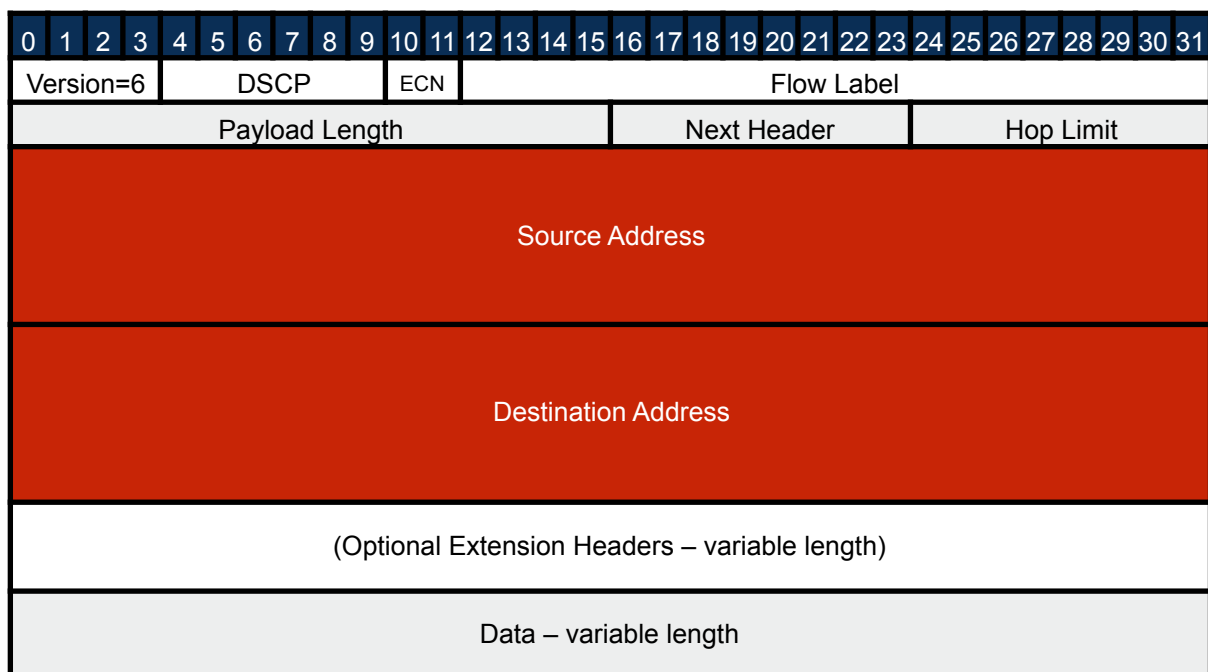
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version=6				DSCP						ECN		Flow Label																			
Payload Length																Next Header								Hop Limit							
Source Address																															
Destination Address																															
(Optional Extension Headers – variable length)																															
Data – variable length																															

Compared to IPv4: simpler header format, larger addresses, removes support for fragmentation, adds flow label

Addressing



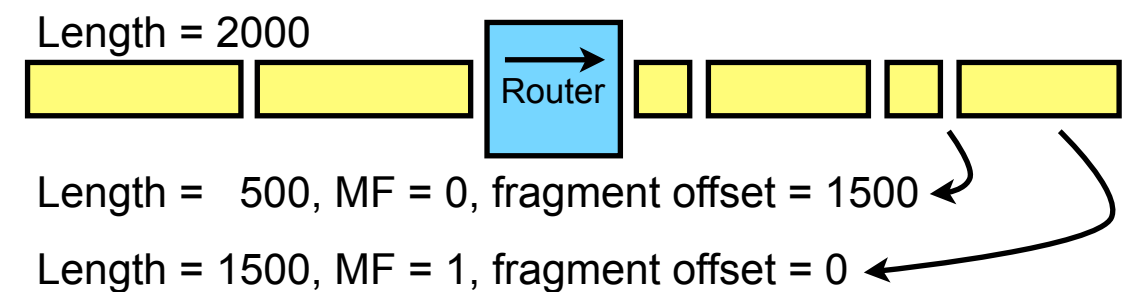
- Every network interface on every host is intended to have a unique address
 - Hosts may change address over time to give illusion of privacy
 - Addressable \neq reachable: firewalls exist in both IPv4 and IPv6
- IPv4 addresses are 32 bits
 - Example: 130.209.247.112
 - Significant problems due to lack of IPv4 addresses \rightarrow details later
- IPv6 addresses are 128 bits
 - Example: 2001:4860:4860::8844



Fragmentation

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
Version=4				Header				DSCP				ECN		Total Length																				
Fragment Identifier																	DF	MF	Fragment Offset															
TTL								Upper Layer Protocol								Header Checksum																		
Source Address																																		
Destination Address																																		
(Options – variable length, padded to 32 bit boundary)																																		
Data – variable length																																		

- Link layer has a maximum packet size (MTU)
- IPv4 will routers fragment packets that are larger than the MTU
 - MF bit is set if more fragments follow: reconstruct using fragment offset and fragment identifier



- DF bit is set → routers shouldn't fragment, must discard large packets
- IPv6 doesn't support fragmentation
 - Hard to implement for high rate links
 - End-to-end principle

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version=6				DSCP						ECN			Flow Label																		
Payload Length																Next Header								Hop Limit							
Source Address																															
Destination Address																															
(Optional Extension Headers – variable length)																															
Data – variable length																															

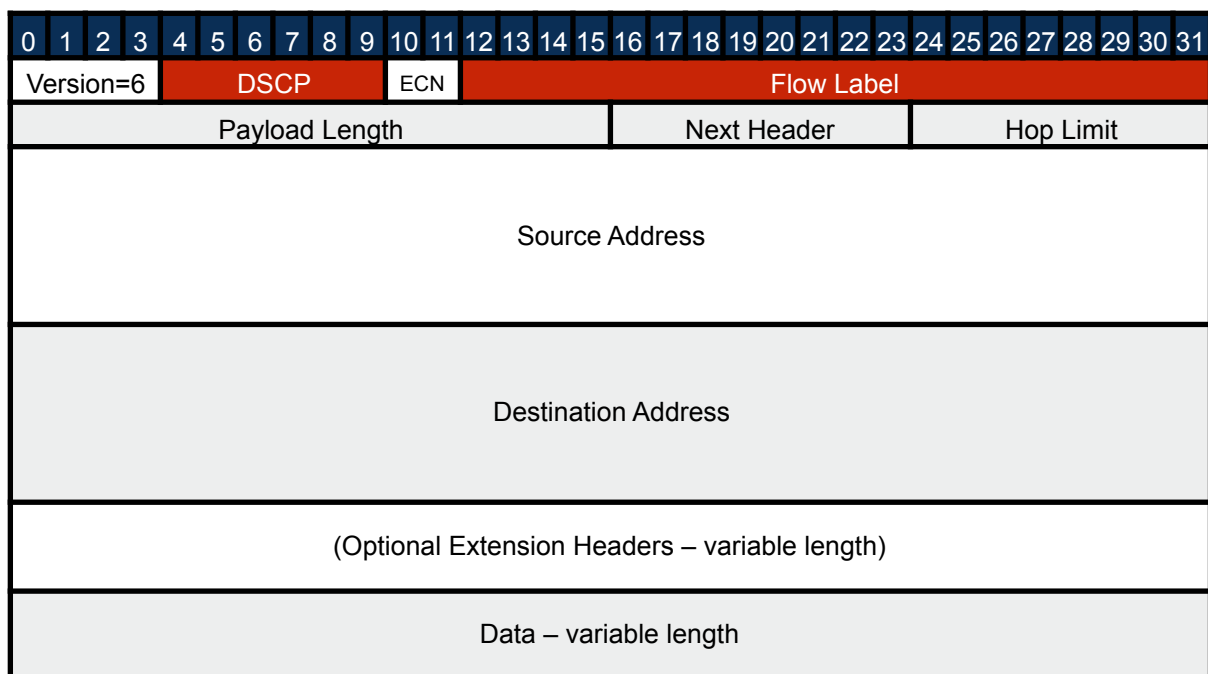
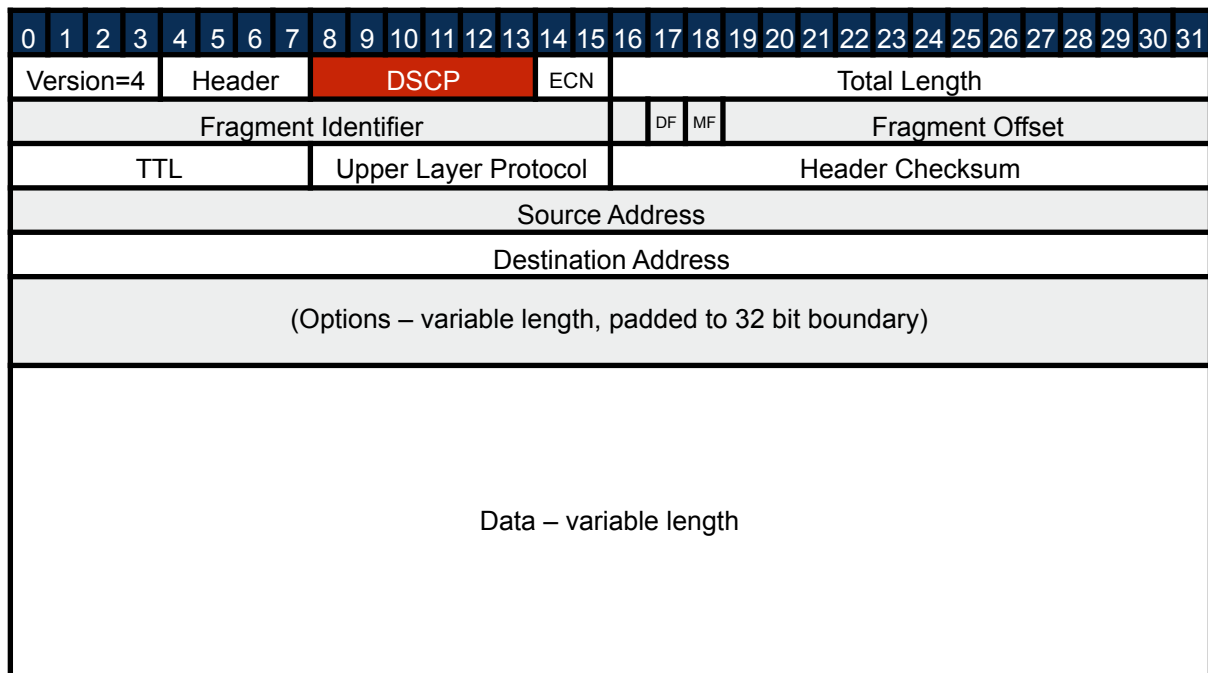
Loop Protection

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31			
Version=4				Header				DSCP						ECN		Total Length																		
Fragment Identifier																	DF		MF		Fragment Offset													
TTL								Upper Layer Protocol								Header Checksum																		
Source Address																																		
Destination Address																																		
(Options – variable length, padded to 32 bit boundary)																																		
Data – variable length																																		

- Packets include a forwarding limit:
 - Set to a non-zero value when the packet is sent (typically 64 or 128)
 - Each router that forwards the packet reduces this value by 1
 - If zero is reached, packet is discarded
- Stops packets circling forever if a network problem causes a loop
 - Assumption: network diameter is smaller than initial value of forwarding limit

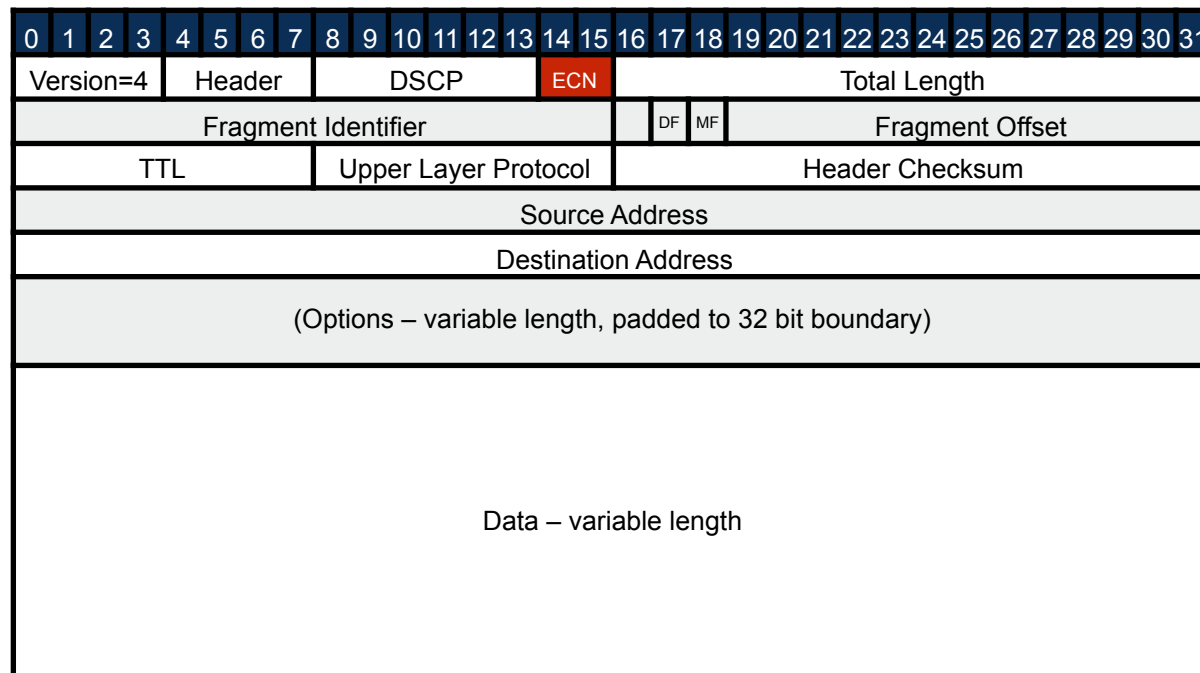
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version=6				DSCP						ECN			Flow Label																		
Payload Length																Next Header								Hop Limit							
Source Address																															
Destination Address																															
(Optional Extension Headers – variable length)																															
Data – variable length																															

Differentiated Services

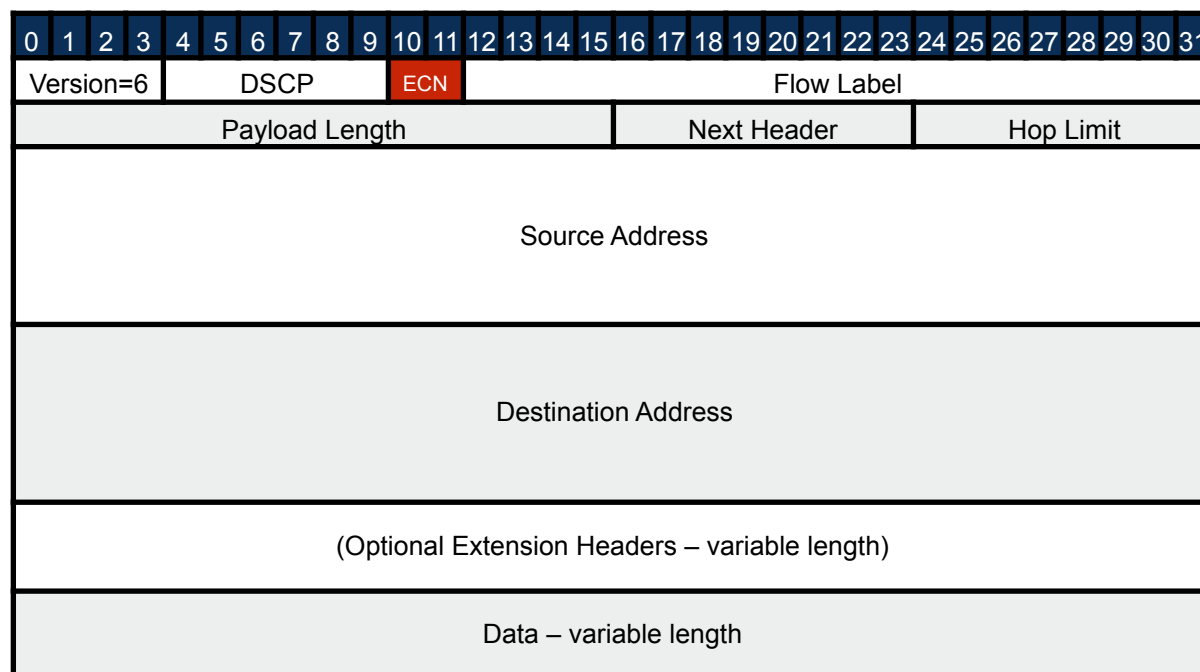


- End systems can request special service from the network
 - Telephony or gaming might prefer low latency over high bandwidth
 - Emergency traffic could be prioritised
 - Background software updates might ask for low priority
- Signalled by differentiated service code point (DSCP) field in header
- Provides a hint to the network, not a guarantee
 - Often stripped at network boundaries
 - Difficult economic and network neutrality issues – who is allowed to set the DSCP and what are they charged for doing so?
- IPv6 provides a flow label to group related traffic flows together

Explicit Congestion Notification



- Routers typically respond to network congestion by dropping packets
 - A “best effort” packet delivery service
 - Transport protocols detect the loss, and can request a retransmission if necessary
- Explicit congestion notification gives routers a way to signal congestion is approaching
 - If ECN=00 explicit congestion notification is disabled
 - If a sending host sets ECN=10 or ECN=01, routers monitor link usage, and can change the field to ECN=11 indicating congestion is imminent
 - A host receiving ECN=11 needs to reduce it’s sending rate – or the congested router will start dropping packets



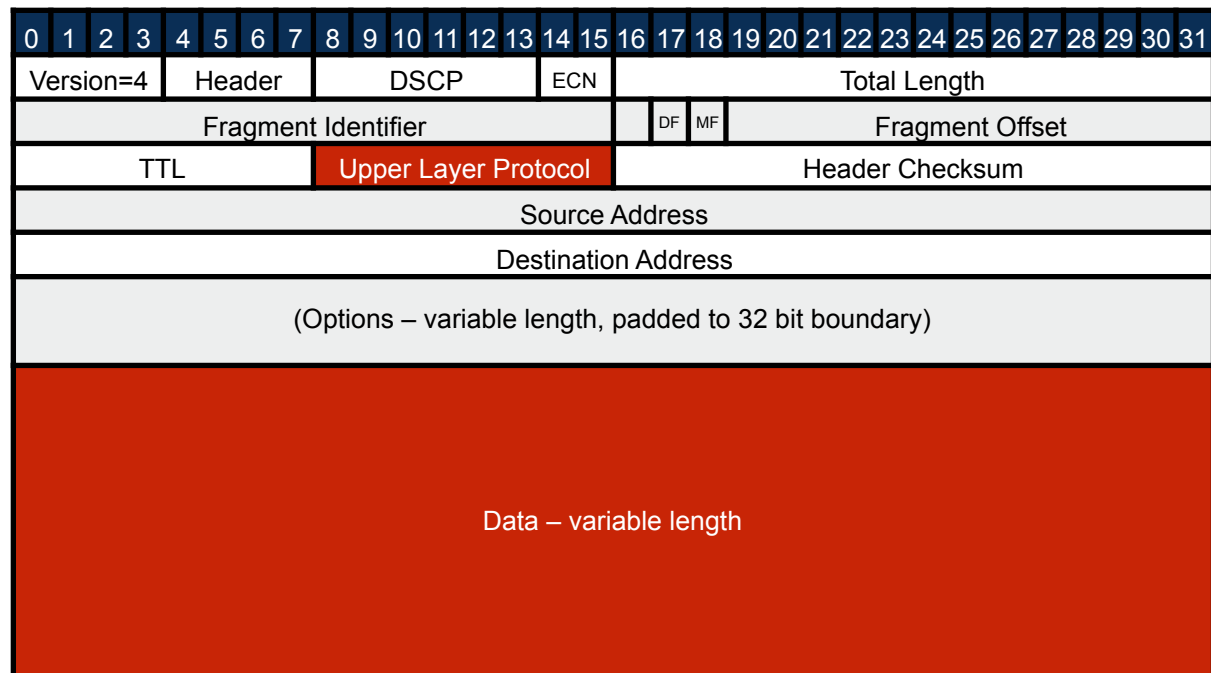
Header Checksum

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31		
Version=4				Header				DSCP				ECN		Total Length																			
Fragment Identifier																		DF	MF	Fragment Offset													
TTL								Upper Layer Protocol								Header Checksum																	
Source Address																																	
Destination Address																																	
(Options – variable length, padded to 32 bit boundary)																																	
Data – variable length																																	

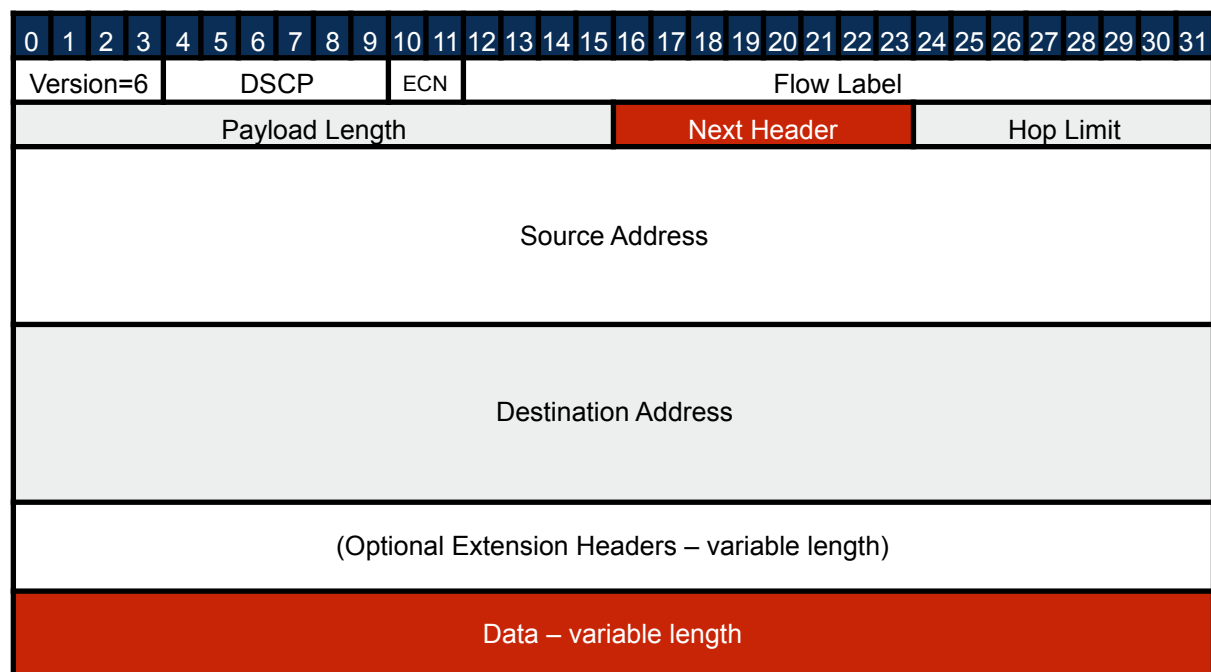
- IPv4 header contain a checksum to detect transmission errors
 - Conceptually similar to link-layer checksum, although uses a different algorithm
 - Protects the IP header only, not the payload data protected (must be protected by upper layer protocol, if needed)
- IPv6 does not contain checksum – assumes the data is protected by a link layer checksum

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Version=6				DSCP						ECN			Flow Label																		
Payload Length																Next Header								Hop Limit							
Source Address																															
Destination Address																															
(Optional Extension Headers – variable length)																															
Data – variable length																															

Transport Layer Protocol Identifier



- Network layer packets include the transport layer data as payload
- Must identify what transport layer protocol is used, to pass the data to the correct upper-layer protocol
 - TCP = 6
 - UDP = 17
 - DCCP = 33
 - ICMP = 1
- Protocols managed by the IANA: <http://www.iana.org/assignments/protocol-numbers/>



IPv4 or IPv6?

- IPv4 has reached end-of-life: insufficient addresses
- IPv6 intended as long term replacement for IPv4
 - Primary goal: increase the size of the address space, to allow more hosts on the network
 - Also simplifies the protocol, makes high-speed implementations easier
- Not yet clear if IPv6 will be widely deployed
 - But, straight-forward to build applications that work with both IPv4 and IPv6
 - DNS query using `getaddrinfo()` will return IPv6 address if it exists, else IPv4 address; all other socket calls use the returned value
 - Write new code to support both IPv6 and IPv4

Internet Protocol Addresses

Addressing

- How to name hosts in a network?
 - Is the address an identity or a location?
 - Does it name the host, or the location at which it attaches to the network
 - How should addresses be allocated?
 - Hierarchical or flat?
 - What is the address format?
 - Human or machine readable?
 - Textual or binary? Structured or unstructured?
 - Fixed or variable length? How large?

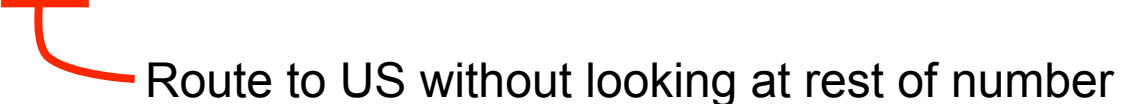
Identity and Location

- Addresses can denote host identity
 - Give hosts a consistent address, irrespective of where or when they attach to the network
 - Simple upper-layer protocols
 - Transport layer and applications unaware of multi-homing or mobility
 - Puts complexity in network layer
 - Network must determine location of host before it can route data
 - Often requires in-network database to map host identity to routable address
 - E.g., mobile phone numbers

Identity and Location

- Alternatively, an address can indicate the *location* at which a host attaches to the network
 - Address structure matches the network structure
 - Network can directly route data given an address
 - E.g., geographic phone numbers: +44 141 330 4256
 - Simplifies network layer, by pushing complexity to the higher layers
 - Multi-homing and mobility must be handled by transport layer or applications – transport layer connections break when host moves

Address Allocation

- Are addresses allocated hierarchically?
 - Allows routing on aggregate addresses
 - E.g., phone call to +1 703 243 9422
Route to US without looking at rest of number
 - Forces address structure to match network topology
 - Requires rigid control of allocations
- Or is there a flat namespace?
 - Flexible allocations, no aggregation → not scalable

Address Formats

- Textual or binary? Fixed or variable length?
 - Fixed length binary easier (faster) for machines to process
 - Variable length textual easier for humans to read
 - Which are you optimising for?

IP Addresses

- IP addresses have the following characteristics:
 - They specify location of a network interface
 - They are allocated hierarchically
 - They are fixed length binary values
 - IPv4: 32 bits
 - IPv6: 128 bits
- Domain names are a separate *application level* namespace

IP Addresses

- Both IPv4 and IPv6 addresses encode location
 - Addresses are split into a *network part* and a *host part*
 - A *netmask* describes the number of bits in the network part
 - The network itself has the address with the host part equal to zero
 - The broadcast address for a network has all bits of host part equal to one(allows messages to be sent to all hosts on a network)
 - A host with several network interfaces will have one IP addresses per interface
 - E.g., laptop with an Ethernet interface and a Wi-Fi interface will have two IP addresses

IPv4 Addresses

- 32 bit binary addresses

IP address: 130.209.247.112 = 10000010 11010001 11110111 01110000

Netmask: 255.255.240.0 = 11111111 11111111 11110000 00000000



20 bits → network = 130.209.240.0/20

Broadcast address:

130.209.255.255 = 10000010 11010001 11111111 11111111

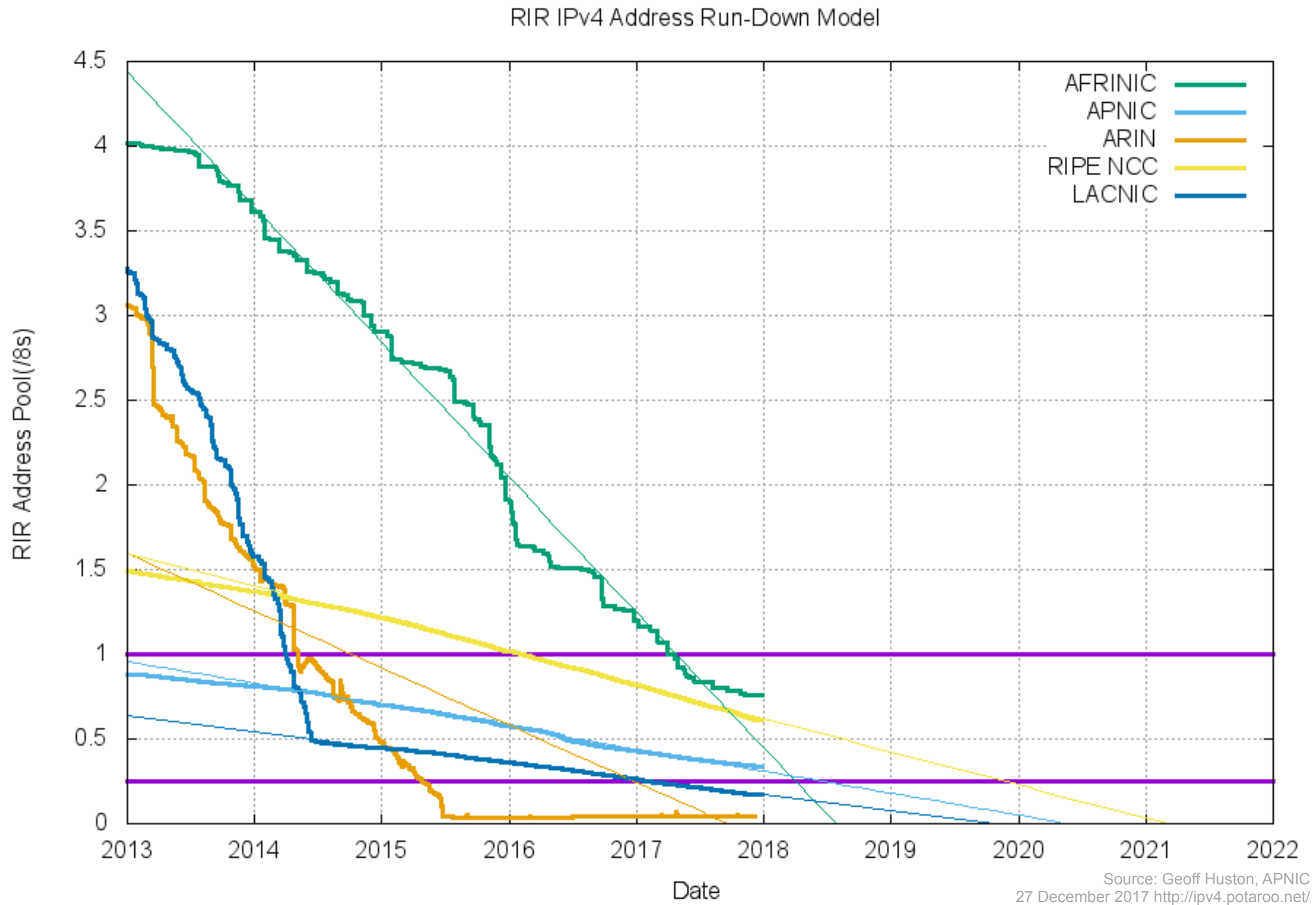
Aside: Classes of IP address

- IP addresses used to be allocated so the netmask was a multiple of 8 bits
 - Class A → a /8 network (~16 million addresses)
 - Class B → a /16 network (65536 addresses)
 - Class C → a /24 network (256 addresses)
 - Inflexible, and wasted addresses
- Old terminology still used sometimes...*
- Arbitrary length netmask allowed since 1993:
 - The Glasgow SoCS network is a /20

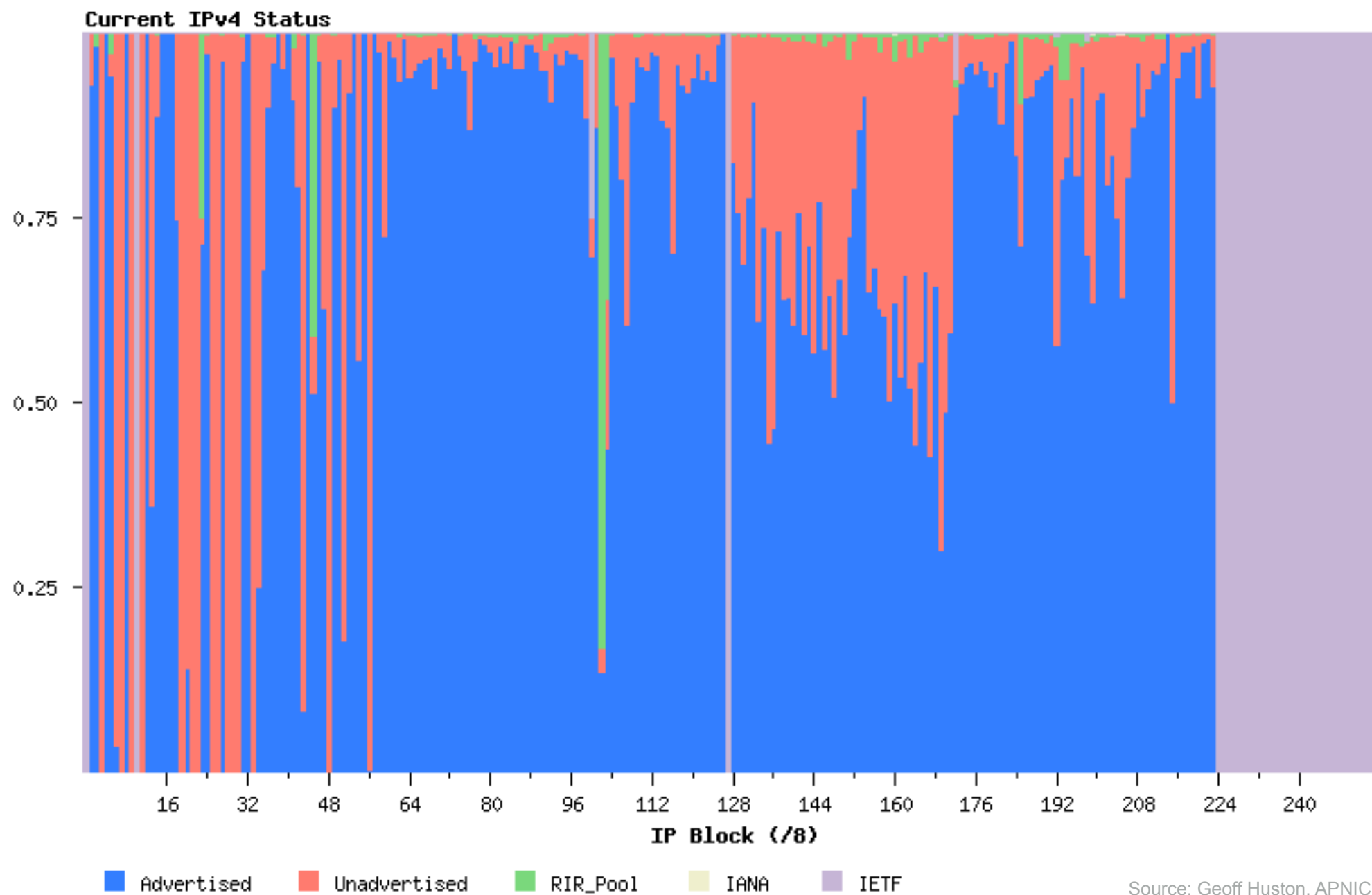
IP Address Management

- IPv4 has $2^{32} = 4,294,967,296$ addresses
 - IANA administers the pool of unallocated addresses
 - Historically would assign addresses directly to ISPs, large enterprises, etc.
 - Now, addresses assigned to regional Internet registries (RIRs) as needed:
 - AfriNIC (Africa), APNIC (Asia-Pacific), ARIN (North America), LACNIC (Latin America and Caribbean), and RIPE (Europe, Middle East, Central Asia)
 - Allocations made one /8 ($2^{24} = 16,777,216$ addresses) at a time
 - RIRs allocate addresses to ISPs and large enterprises within their region; ISPs allocate to their customers
- IANA has allocated all available addresses to RIRs
 - Last allocation made on 3 February 2011

IPv4 Address Space Exhaustion



IPv4 Address Space Utilisation



Source: Geoff Huston, APNIC
27 December 2017 <http://ipv4.potaroo.net/>

The IPv4 Address Space is Fully Used

- In practical terms, we have run out of IPv4 address space



IPv6

- IPv6 provides 128 bit addresses – if deployed it will solve address shortage for a *long* time
 - $2^{128} = 340,282,366,920,938,463,463,374,607,431,768,211,456$ addresses
- Approximately 665,570,793,348,866,943,898,599 addresses per square metre of the Earth's surface

IPv6 Addresses

- 128 bit binary addresses, written as 8 “:” separated 16 bit hexadecimal fields

2a00:1098:0000:0086:1000:0000:0000:0010

- Usually written in a shortened form: [RFC 5952]
 - Leading zeros in each 16 bit field are suppressed
 - A run of more than one consecutive 16 bit field that is all zero is omitted and replaced with a “::” (if there is more than one such run, the longest is replaced; if there are several runs of equal length, the first is replaced)
 - The “::” must not be used to replace a single 16 bit field

2a00:1098:0:86:1000::10

IPv6 Addresses

- Local identifier part of IPv6 address is 64 bits:

2001:0db8:85a3:08d3:1319:8a2e:0370:7334

← Local identifier part →

- Can be derived from Ethernet/Wi-Fi MAC address:

48 bit IEEE MAC: 0014:5104:25ea

Expand to 64 bits: 0014:51ff:fe04:25ea

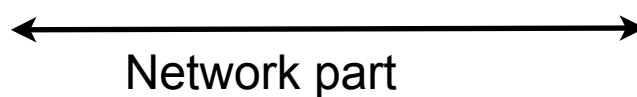
Invert bit 6: 0214:51ff:fe04:25ea

- Or randomly chosen, with bit 6 set to zero, to give illusion of privacy

IPv6 Addresses

- Routers advertise network part, hosts auto-configure address:

2001:0db8:85a3:08d3:1319:8a2e:0370:7334

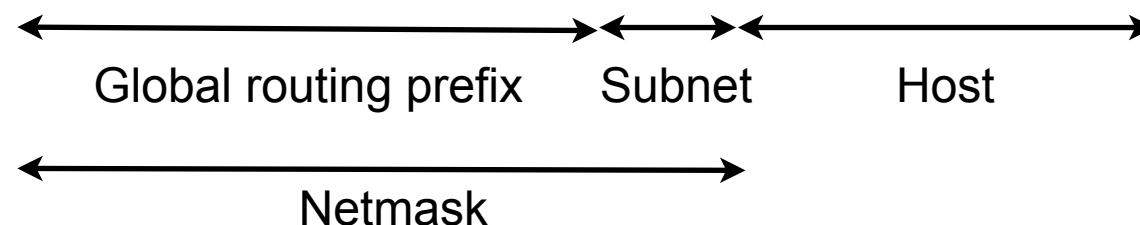


Network part

- Network part is split into global routing prefix (up to 48 bits) and a subnet identifier:

Formalises the distinction present in IPv4:

130.209.247.112 = 10000010 11010001 11110111 01110000

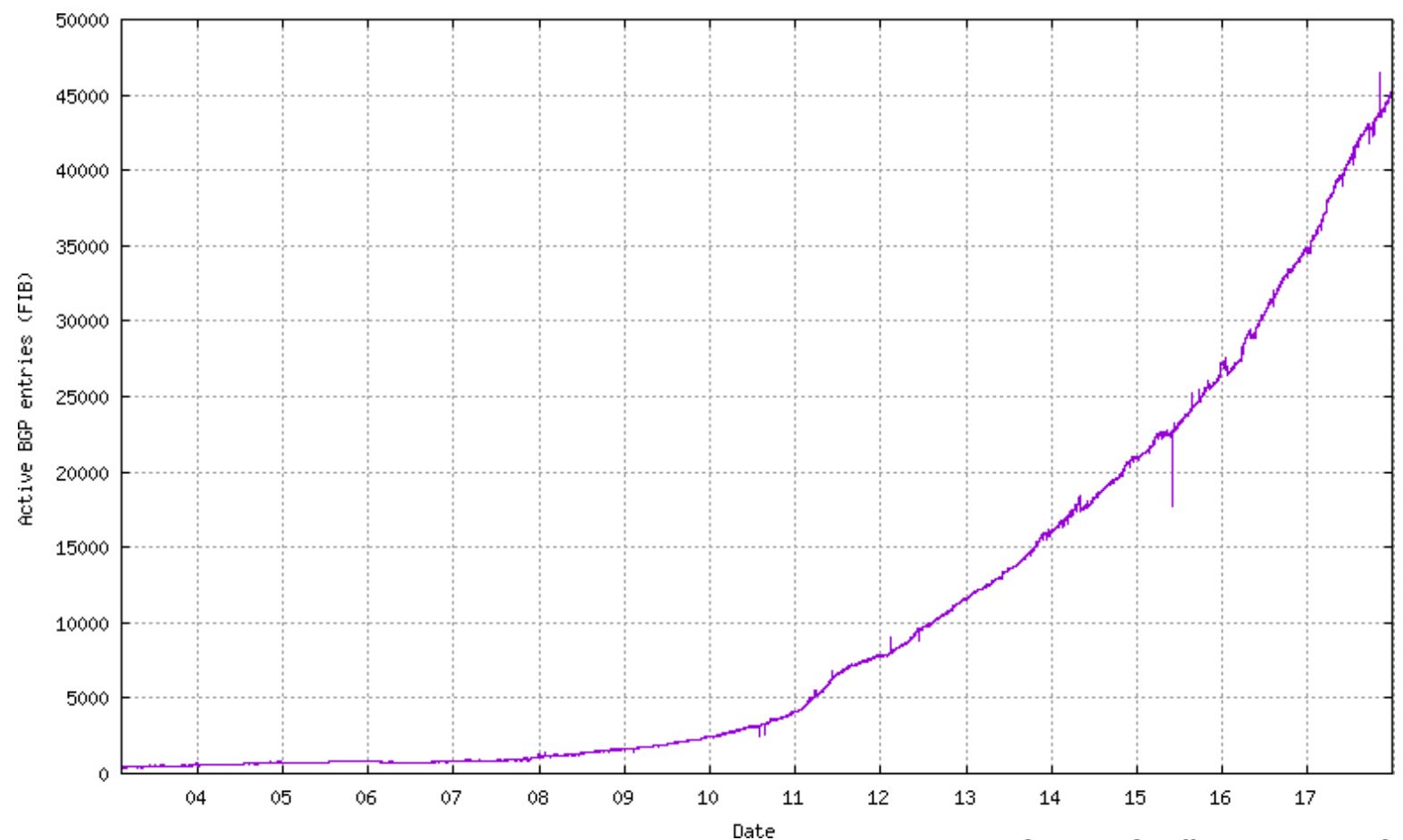


IPv6 Deployment Issues

- IPv6 requires changes to *every* single host, router, firewall, and application...
 - Significant deployment challenge!
 - Host changes done: MacOS X, Windows, Linux, FreeBSD, Symbian, iOS, Android, etc.
 - Backbone routers generally support IPv6, home routers and firewalls are starting to be updated
 - Many applications have been updated

NAT vs. IPv6

- NAT widely deployed for IPv4
 - Initially seems simple: no host changes; web browsing and email still work
 - But... hugely complicated for peer-to-peer applications → lecture 16
 - Very difficult to debug problems, or deploy new classes of application
- IPv6 starting to see large-scale use:



Source: Geoff Huston, APNIC
27 December 2017 <http://bgp.potaroo.net/v6/as2.0/>

Summary

- Internetworking
 - History of the Internet
 - The Internet Protocol: IPv4 and IPv6
- Addressing concepts
 - Identifiers vs. locators
 - Fixed- vs. variable-length
 - Flat vs. hierarchical
 - Textual vs. binary
- Internet addressing:
 - IPv4 and address exhaustion
 - IPv6

