计算机网络与Internet

徐恪
清华大学计算机系

提纲

- Internet简介
- 协议分层和OSI参考模型
- End-to-End Arguments
- Internet设计原则

提纲

- ❐ *Internet简介*
- ❐ 协议分层和OSI参考模型
- ❐ End-to-End Arguments
- ❐ Internet设计原则

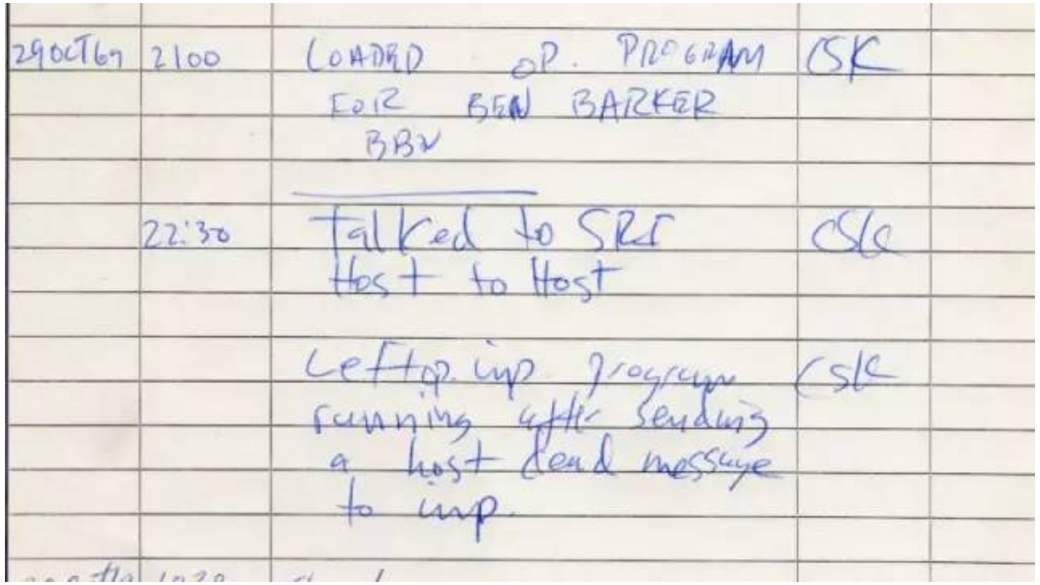
The Internet

- Global scale, general purpose, heterogeneous-technologies, public, computer network
- Internet Protocol
  - Open standard: Internet Engineering Task Force (IETF) as standard body http://www.ietf.org
  - Technical basis for other types of networks
    - Intranet: enterprise IP network
- Developed by the research community

1969-2019


1969-2019
50 Years of the Internet 互联网50年


APOLLO
50
NEXT GIANT LEAP


29OCT69 2100 LOADED OP. PROGRAM CSK
FOR BEN BARKER
BBN
22:30 Talked to SRI CSK
Host to Host
Left op. imp program CSK
running after sending
a host dead message
to imp.

Internet History

1961-1972: Early packet-switching principles

- 1961: Kleinrock - queueing theory shows effectiveness of packet-switching
- 1964: Baran - packet-switching in military nets
- 1967: ARPAnet conceived by Advanced Research Projects Agency –Larry Roberts
- 1969: first ARPAnet node operational

- 1972:
  - ARPAnet demonstrated publicly
  - NCP (Network Control Protocol) first host-host protocol
  - first e-mail program
  - ARPAnet has 15 nodes

Internet History

1972-1980: Internetworking, new and proprietary nets

- 1970: ALOHAnet satellite network in Hawaii
- 1973: Metcalfe's PhD thesis proposes Ethernet
- 1974: Cerf and Kahn - architecture for interconnecting networks
- late 70's: proprietary architectures: DECnet, SNA, XNA
- late 70's: switching fixed length packets (ATM precursor)
- 1979: ARPAnet has 200 nodes

Cerf and Kahn's internetworking principles:

- minimalism, autonomy
  - no internal changes required to interconnect networks
- best effort service model
- stateless routers
- decentralized control

define today's Internet architecture

Internet History

1980-1990: new protocols, a proliferation of networks

- 1982: SMTP e-mail protocol defined
- 1983: deployment of TCP/IP
- 1983: DNS defined for name-to-IP-address translation
- 1985: FTP protocol defined
- 1988: TCP congestion control

- new national networks: Csnet, BITnet, NSFnet, Minitel
- 100,000 hosts connected to confederation of networks

Internet History

1990, 2000's: commercialization, the Web, new apps

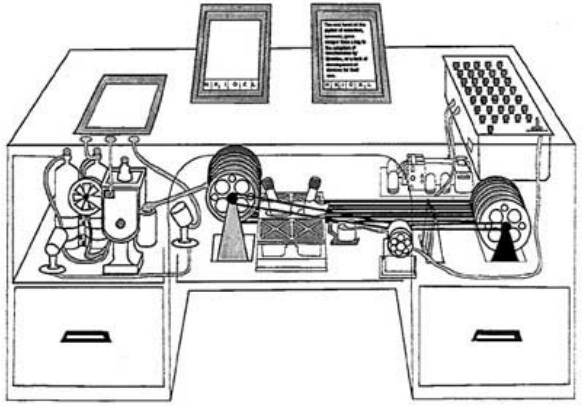
- Early 1990's: ARPAnet decommissioned
- 1991: NSF lifts restrictions on commercial use of NSFnet (decommissioned, 1995)
- early 1990s: Web
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, later Netscape
  - late 1990's: commercialization of the Web

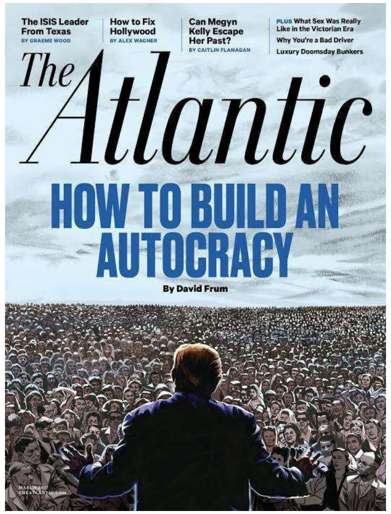
Late 1990's – 2000's:

- more killer apps: instant messaging, peer2peer file sharing (e.g., BT, Napster)
- network security to forefront
- est. 50 million host, 100 million+ users
- backbone links running at Gbps

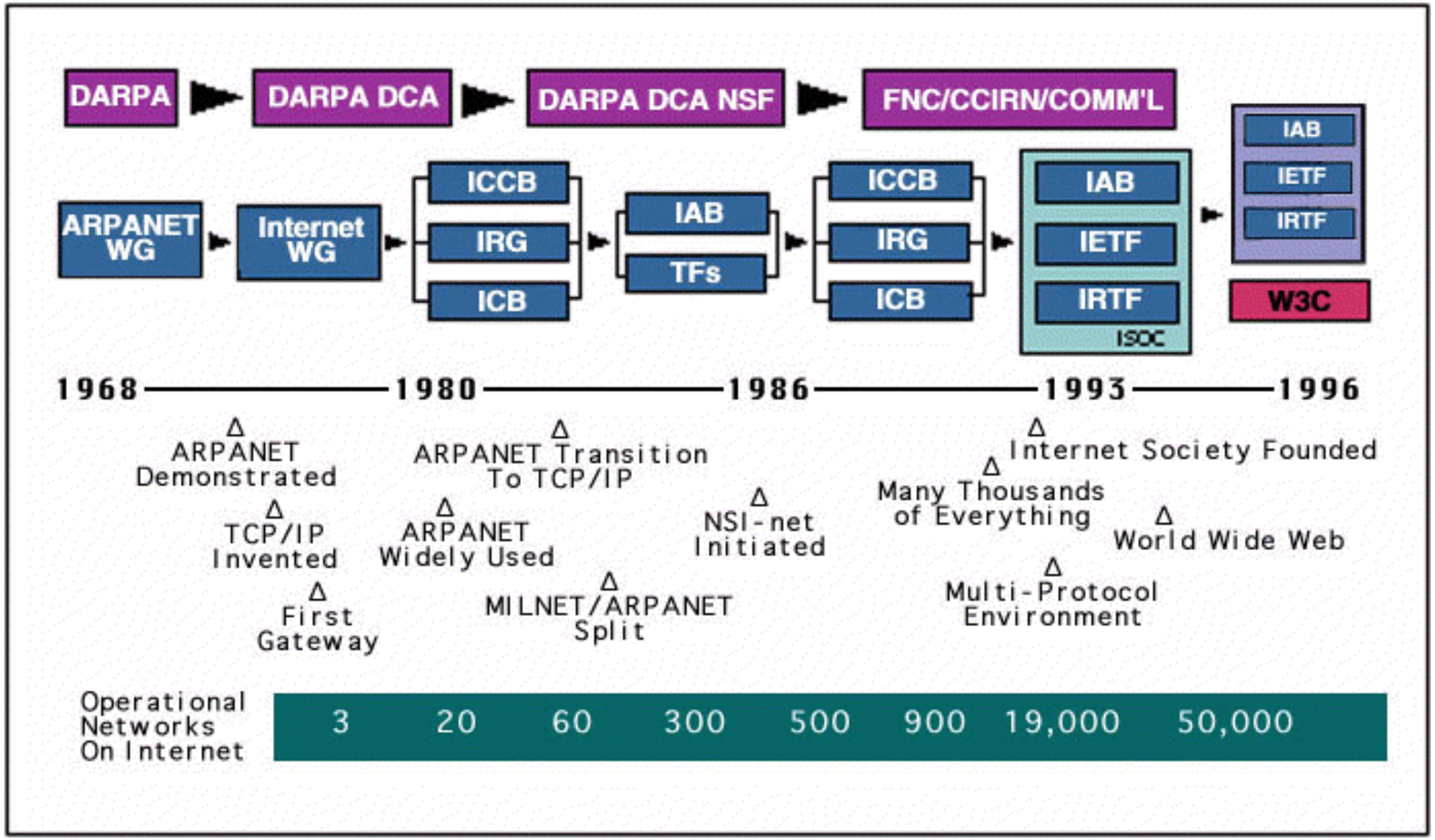
Vannevar Bush and the Memex

- Vannevar Bush
  - 曼哈顿计划的发起者
  - 发起建立NSF
- Memex(Memory-Extender)
  - 1945年，Vannevar Bush在《大西洋月刊》上发表了一篇文章《As We May Think》，提出一种信息机器的构想
  - 机器内部用微缩胶卷存储信息，也就是自动翻拍，可以不断添加新的信息；桌面上有阅读屏，用来放大阅读微缩胶卷；还有许多个按钮，每一个按钮代表一个主题，按一下，相应的微缩胶卷就会显示
  - *读者可以建立指向某些微缩胶卷片段的链接，并依照自己的喜好形成新的线性顺序*，甚至加上自己的补充或评论。这些可以成为共享，他人只要键入建立链接的作者的索引代码，就可以追溯到这些关联





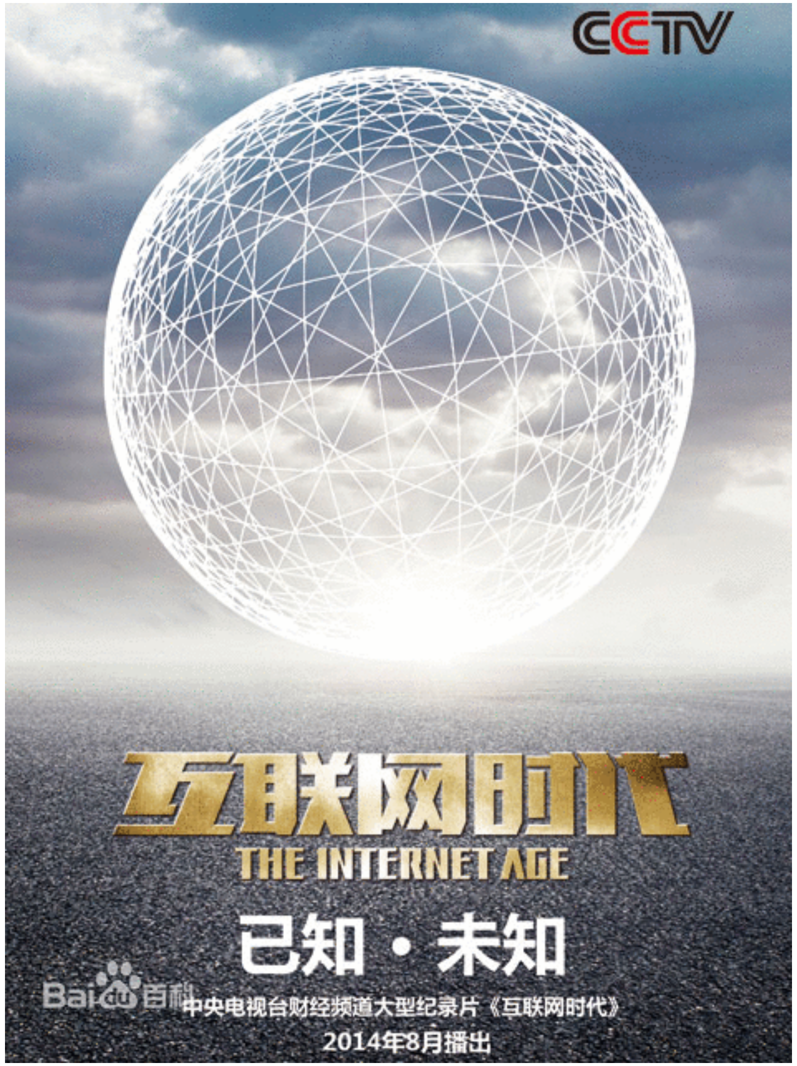
Time Line of the Internet



• *Source: Internet Society*

互联网时代

- 中央电视台
- 2014年上映
- 共10集：
  - 时代、浪潮
  - 能量、再构
  - 崛起、迁徙
  - 控制、忧虑
  - 世界、眺望

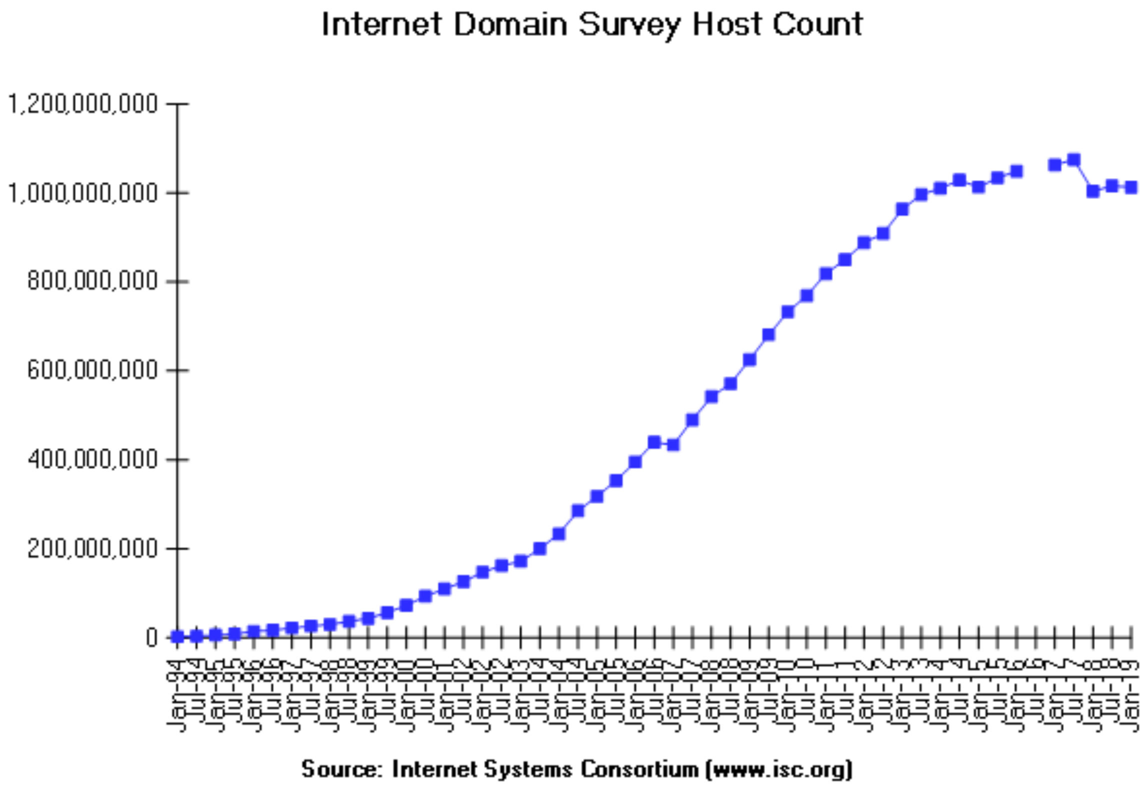


https://tv.sohu.com/s2014/hlwsd/

Growth of the Internet

- Number of Hosts on the Internet:

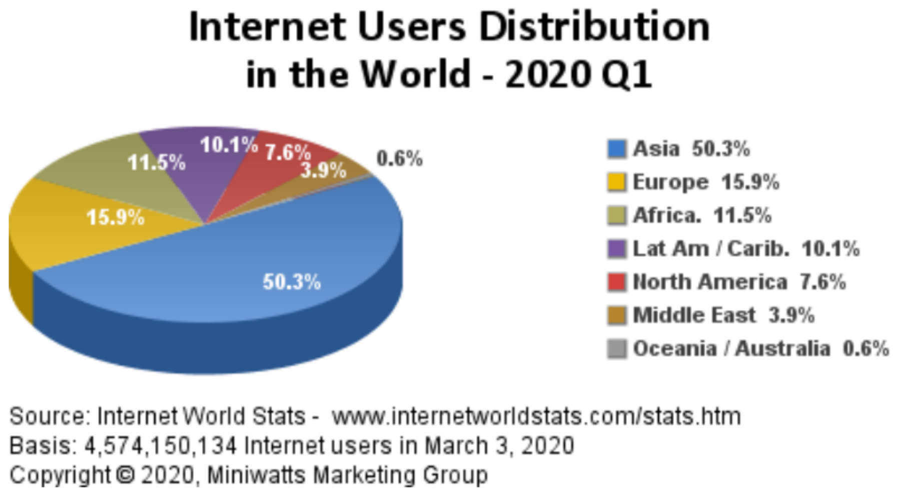
Aug. 1981	213
Oct. 1984	1,024
Dec. 1987	28,174
Oct. 1990	313,000
Oct. 1993	2,056,000
Apr. 1995	5,706,000
Jan. 1997	16,146,000
Jan. 1999	56,218,000
Jan. 2001	109,374,000
Jan. 2003	171,638,297
Jan. 2006	394,991,609
July 2007	489,774,269
Jan. 2009	625,226,456
July 2009	681,064,561
July 2010	768,913,036
July 2011	849,869,781
Jan. 2012	888,239,420
July 2012	908,585,739
July 2013	996,230,757
July 2014	1,028,544,414
Jan. 2018	1,003,604,363

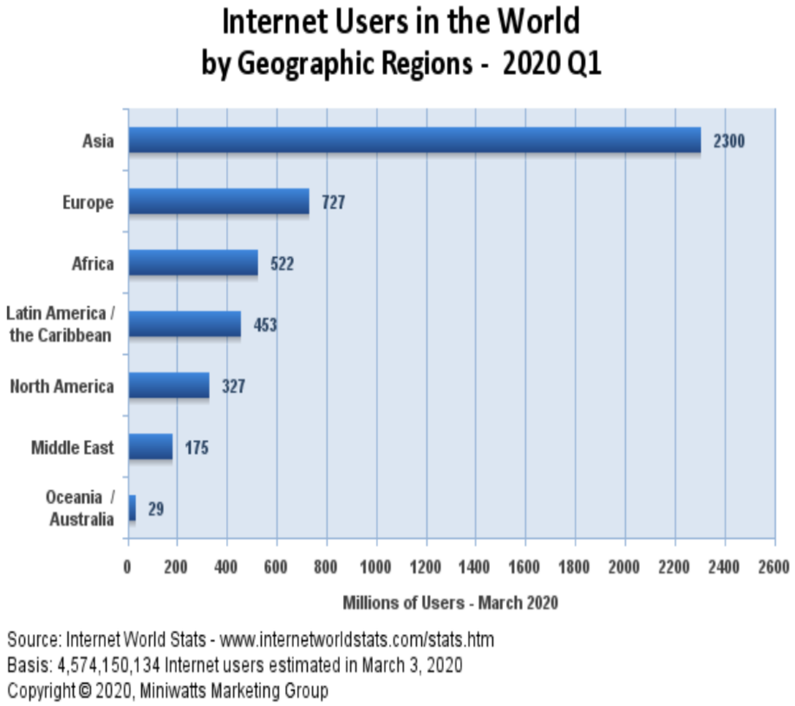


Data available at:
https://www.isc.org/survey/

Growth of the Internet

- In 2020, Total Internet user reach 4.57 billion, Internet penetration rate reach 59.0%.





中国互联网发展历程

探索

- 1986-1994
- X.25 + E-mail

起步

- 1994-1997
- 全功能接入Internet
- 四大骨干网建设

发展

- 1997-2002
- 互联网开始普及
- 网络应用快速增长

繁荣

- 2002-
- WWW, P2P
- Social Networks
- Mobile, Cloud, IoT……

探索：跨越长城，走向世界

- 1987年9月14日（网民节），钱天白教授在北京成功发出中国第一封电子邮件："Across the Great Wall we can reach every corner in the world"，揭开了中国人使用互联网的序幕

E-mail（北京）

↓ *300bps*

公用分组网 ITAPAC（意大利）

↓

卡尔斯鲁厄大学（德国）

Who is Who in the Internet ?

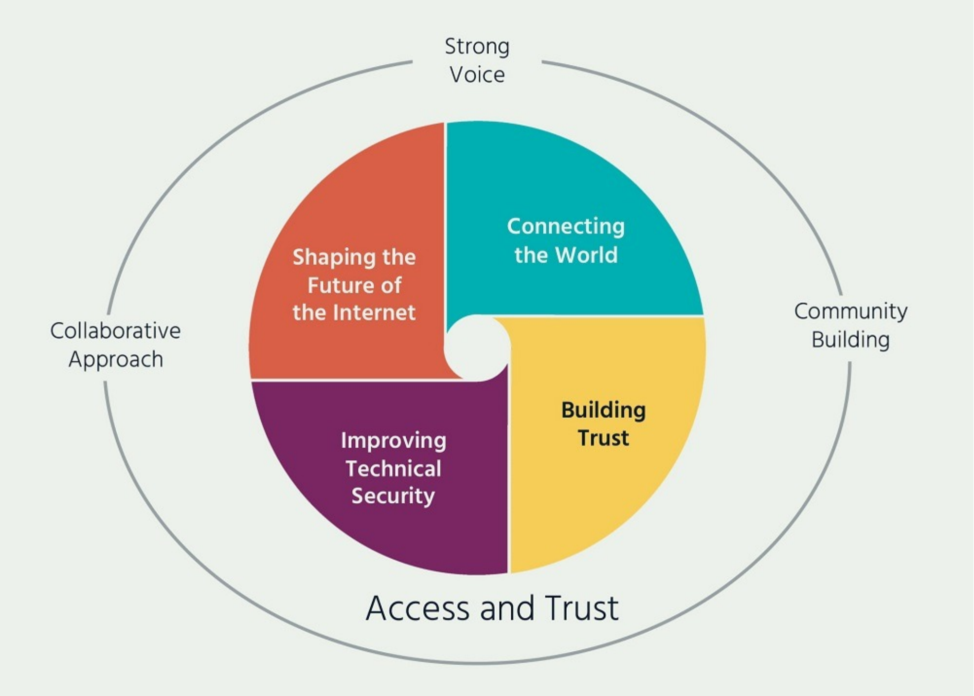
- Internet Society (ISOC)
- Internet Engineering Task Force (IETF)
- Internet Research Task Force (IRTF)
- Internet Architecture Board (IAB)
- Internet Corporation for Assigned Names and Numbers (ICANN)

Who is Who in the Internet ?

- Internet Society (ISOC)

  https://www.internetsociety.org/

  - ISOC is a professional membership society with 97 organization and over 73,000 individual members in over 180 countries
  - It provides leadership in addressing issues of the Internet, and is the organization home for the groups responsible for Internet infrastructure standards, including IETF and IAB



Internet Way of Networking 18 September 2020

Internet Society: U.S. Administration ban of TikTok and WeChat is a direct attack on the Internet

The U.S. Administration's move to ban TikTok and WeChat for U.S. app stores is a direct attack on the Internet. It is an extreme measure that fundamentally undermines the foundation of the Internet. It's especially a threat to the principles of openness and accessibility as well as its decentralized management. **The Internet has no center. This type of top-down intervention is worrisome**

Who is Who in the Internet ?

❑ Internet Hall of Fame

- **Pioneers:** design and development of the Internet with exceptional achievements
- **Innovators:** made outstanding technological, commercial, regulatory or policy advances and helped to expand the Internet's positive impact on the lives of others
- **Global Connectors:** who have made major contributions to the growth, connectivity, and use of the Internet

Who is Who in the Internet ?

- Internet Engineering Task Force (IETF)
  - The IETF is the protocol engineering and development arm of the Internet
  - Subdivided into many working groups, which specify Request For Comments or RFCs

Who is Who in the Internet ?

- IRTF (Internet Research Task Force)
  - The Internet Research Task Force is composed of a number of focused, long-term and small Research Groups
- Internet Architecture Board (IAB)
  - The IAB is responsible for defining the overall architecture of the Internet, providing guidance and broad direction to the IETF

Who is Who in the Internet ?

- The Internet Engineering Steering Group (IESG)
  - The IESG is responsible for technical management of IETF activities and the Internet standards process
  - Composed of the Area Directors of the IETF working groups

Who is Who in the Internet ?

- IETF and IESG Chair
  - Alissa Cooper, Cisco
- Applications and Real-Time Area (art)
  - Barry Leiba, Futurewei Technologie
  - Murray Kucherawy, Facebook
- Internet Area (int)
  - Erik Kline, Google
  - Éric Vyncke, Cisco
- Operations and Management Area (ops)
  - Warren Kumari, Google
  - Robert Wilton, Cisco
- Routing Area (rtg)
  - Deborah Brungard, AT&T
  - Alvaro Retana, Futurewei Technologies
  - Martin Vigoureux, Nokia
- Security Area (sec)
  - Roman Danyliw, CERT/SEI
  - Benjamin Kaduk, Akamai Technologies
- Transport Area (tsv)
  - Martin Duke, F5 Networks
  - Magnus Westerlund, Ericsson

Internet Standardization Process

- All standards of the Internet are published as RFC (Request for Comments). But not all RFCs are Internet Standards
  - available: http://www.ietf.org
- A typical (but not only) way of standardization is:
  - BOF (Birds of a feather)
  - Internet Drafts
  - RFC
  - Proposed Standard
  - Draft Standard (requires 2 working implementation)
  - Internet Standard (declared by IAB)

Internet Standardization Process

- David Clark, MIT, 1992:

We reject: kings, presidents, and voting. We believe in: rough consensus and running code.

Who is Who in the Internet ?

- Internet Corporation for Assigned Names and Numbers (ICANN)
  - To reach another person on the Internet you have to type an address into your computer -- a **name** or a **number**
  - That address must be unique so computers know where to find each other
  - ICANN coordinates these unique identifiers across the world. Without that coordination, we wouldn't have one global Internet

Services Provided by the Internet

- Shared access to computing resources
  - Telnet (1970's)
- Shared access to data/files
  - FTP, NFS, AFS (1980's)
- Communication medium over which people interact
  - Email (1980's), on-line chat rooms (1990's)
  - Instant messaging, IP Telephony (2000's)

Services Provided by the Internet

- A medium for information dissemination
  - USENET (1980's)
  - WWW (1990's)
    - Replacing newspaper, magazine?
  - Audio, video (2000's)
    - Replacing radio, CD, TV…

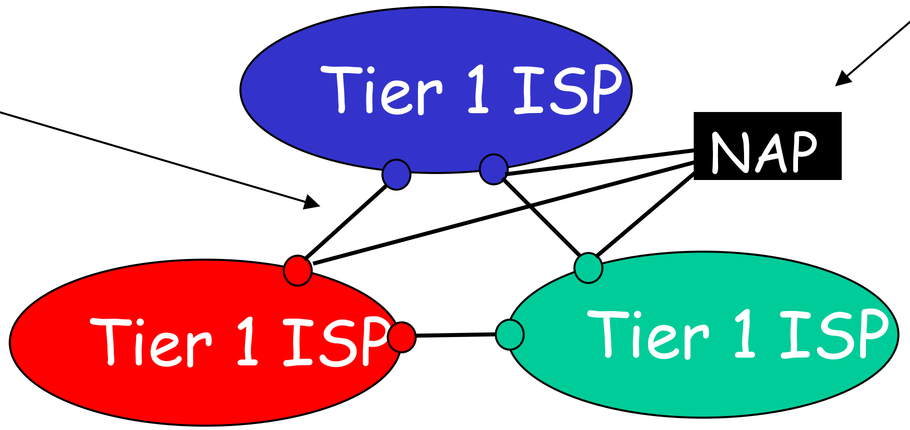
年印刷量分档	家数	2019年印刷量		2018年印刷量（亿对开印张）
		亿对开印张	较上年±%	
7亿印张以上	19	198.60	-12.96	228.17
4亿-7亿印张	18	99.30	-9.96	110.28
2亿-4亿印张	28	76.57	-11.25	86.28
1亿-2亿印张	29	42.06	-12.16	47.88
1亿印张以下	54	26.66	-9.81	29.56
合计	148	443.19	-11.75	[illegible]

cankaoxiaoxi.com

Internet structure: network of networks

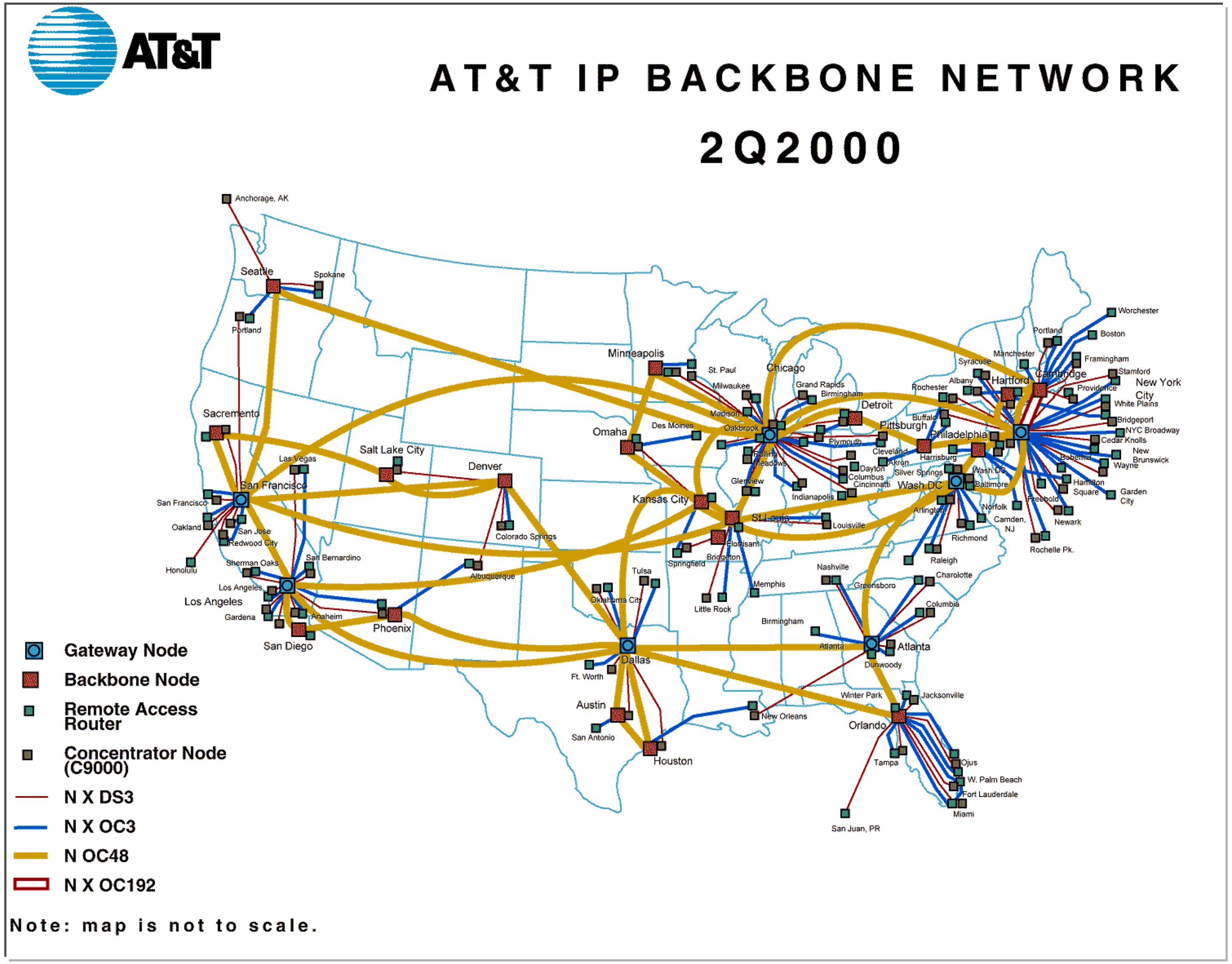
- roughly hierarchical
- at center: "tier-1" ISPs (e.g., UUNet, BBN/Genuity, Sprint, AT&T), national/international coverage
  - treat each other as equals

Tier-1 providers interconnect (peer) privately



Tier-1 providers also interconnect at public network access points (NAPs)

Tier-1 ISP: e.g., AT&T



Tier-1 ISPs

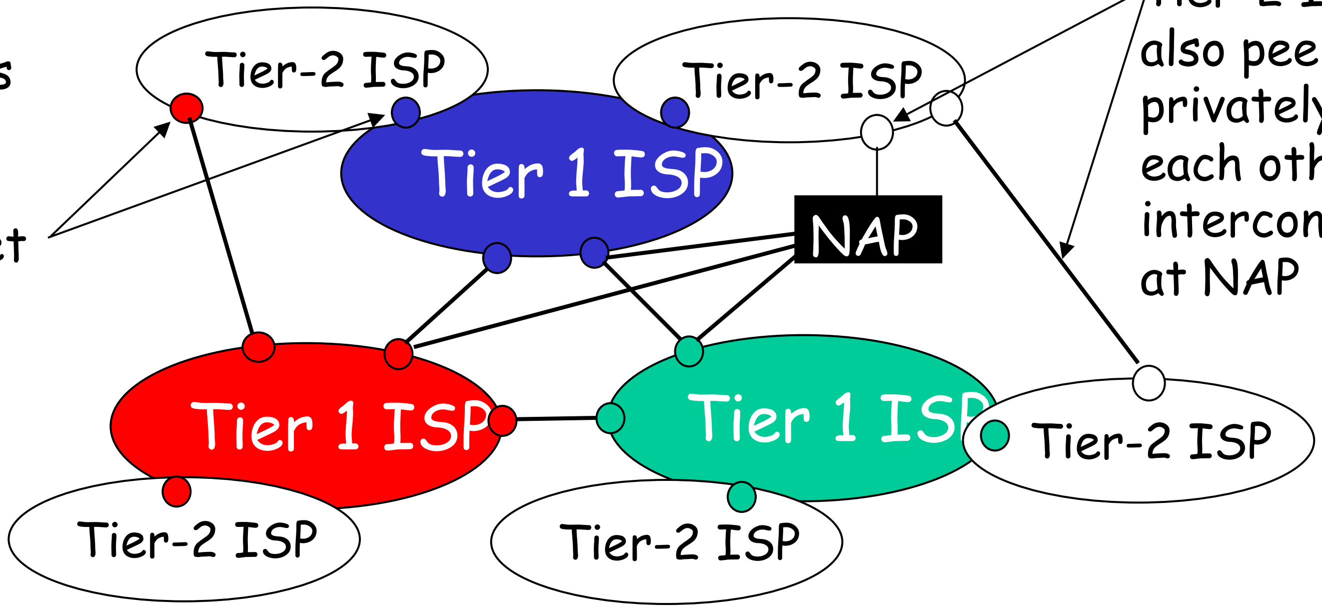
Name	AS number	September, 2007 degree[2][3]	Peering policy
Qwest	209	828	North America; International
Verizon Business	701	1452	Verizon UUNET Peering policy 701, 702, 703
Sprint	1239	880	
TeliaSonera International Carrier	1299		
NTT Communications	2914		
Tinet	3257		
Level 3 Communications (L3)	3356		
Global Crossing (GBLX)	3549	499	Global Crossing Peering policy (2003)
Savvis	3561		
AT&T	7018	1382	AT&T Peering policy
Tata Communications	6453		Peering Policy

http://en.wikipedia.org/wiki/Tier_1_network

Internet structure: network of networks

- "Tier-2" ISPs: smaller (often regional) ISPs
  - Connect to one or more tier-1 ISPs, possibly other tier-2 ISPs

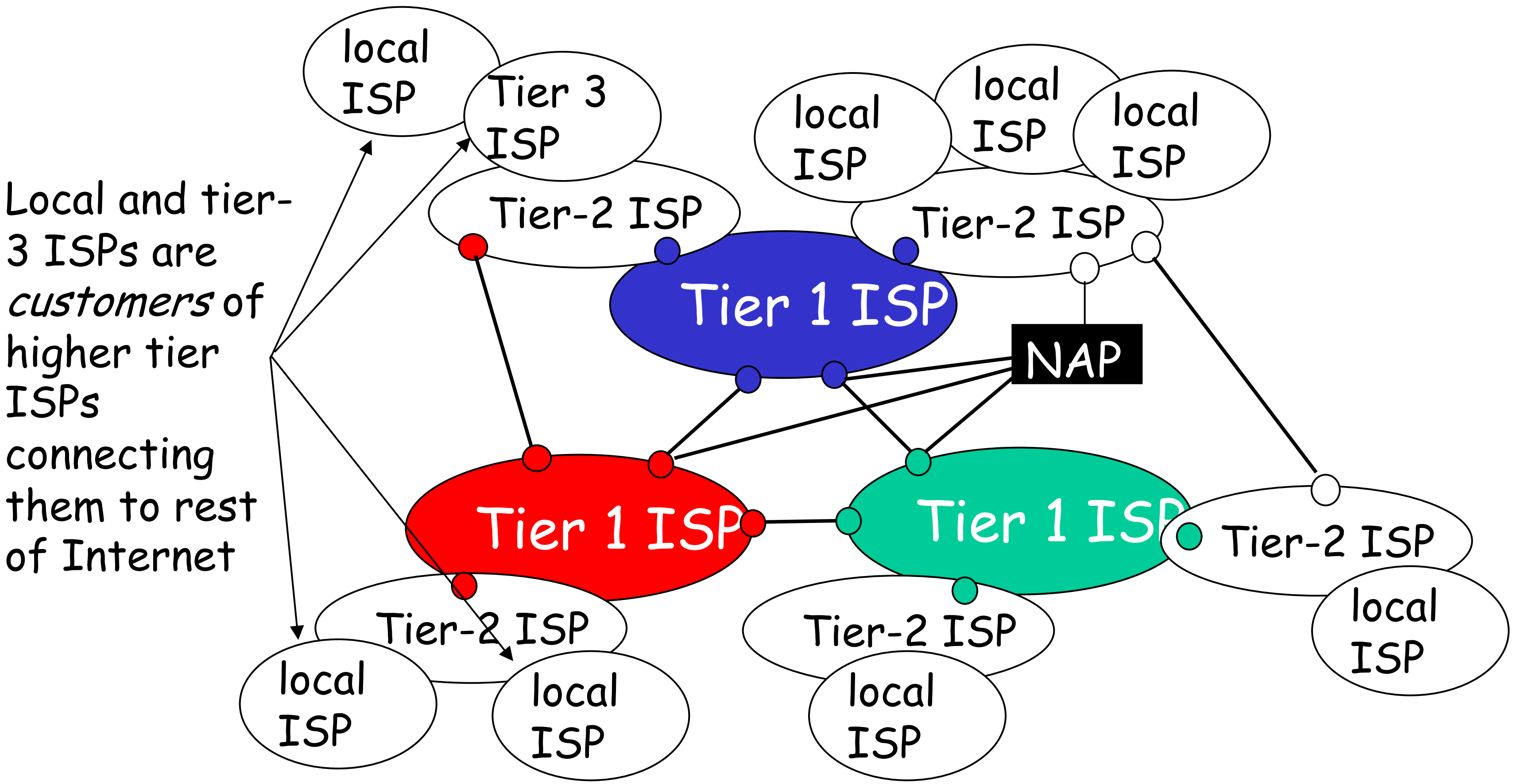
Tier-2 ISP pays tier-1 ISP for connectivity to rest of Internet
- tier-2 ISP is *customer* of tier-1 provider



Tier-2 ISPs also peer privately with each other, interconnect at NAP

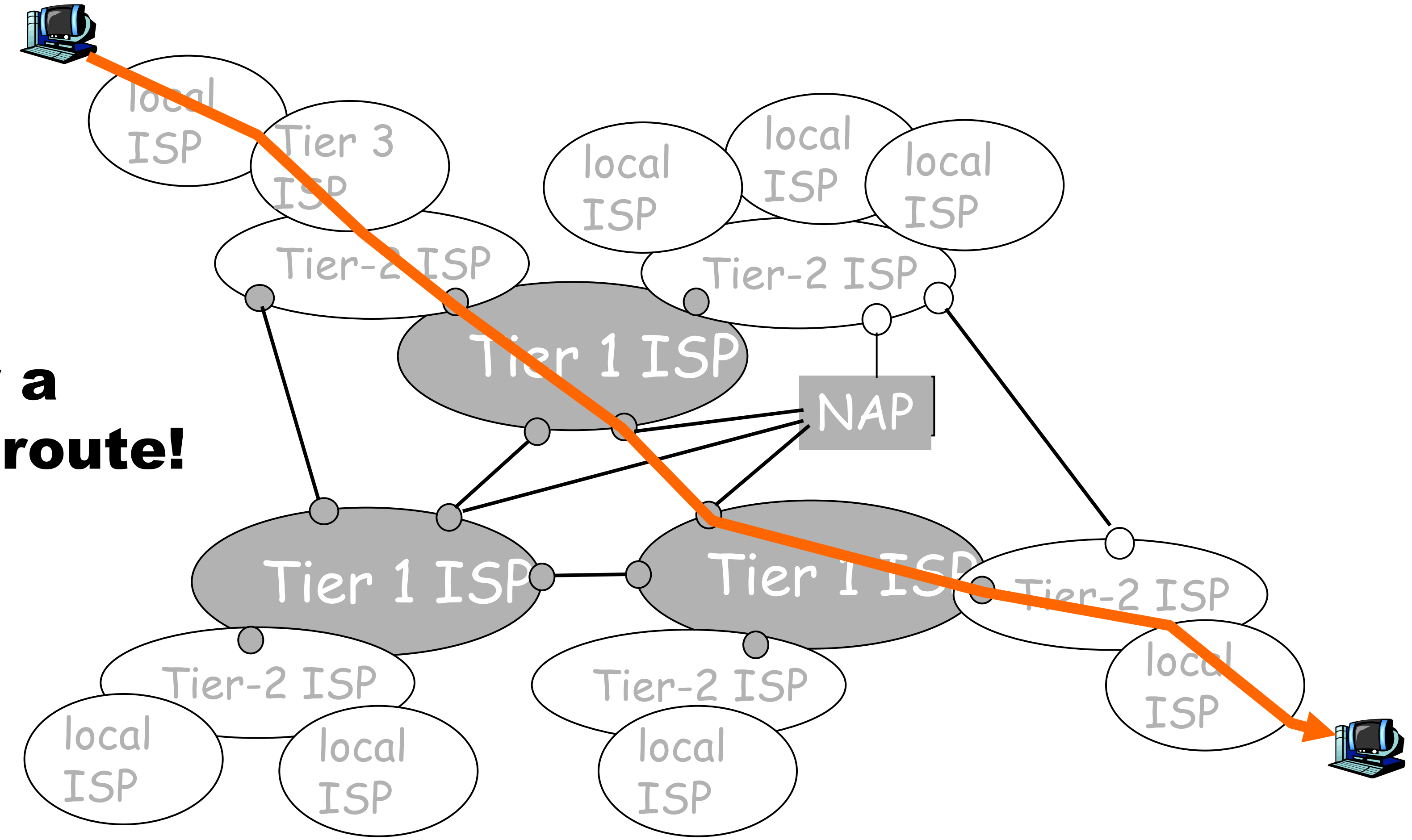
Internet structure: network of networks

- "Tier-3" ISPs and local ISPs
  - last hop ("access") network (closest to end systems)

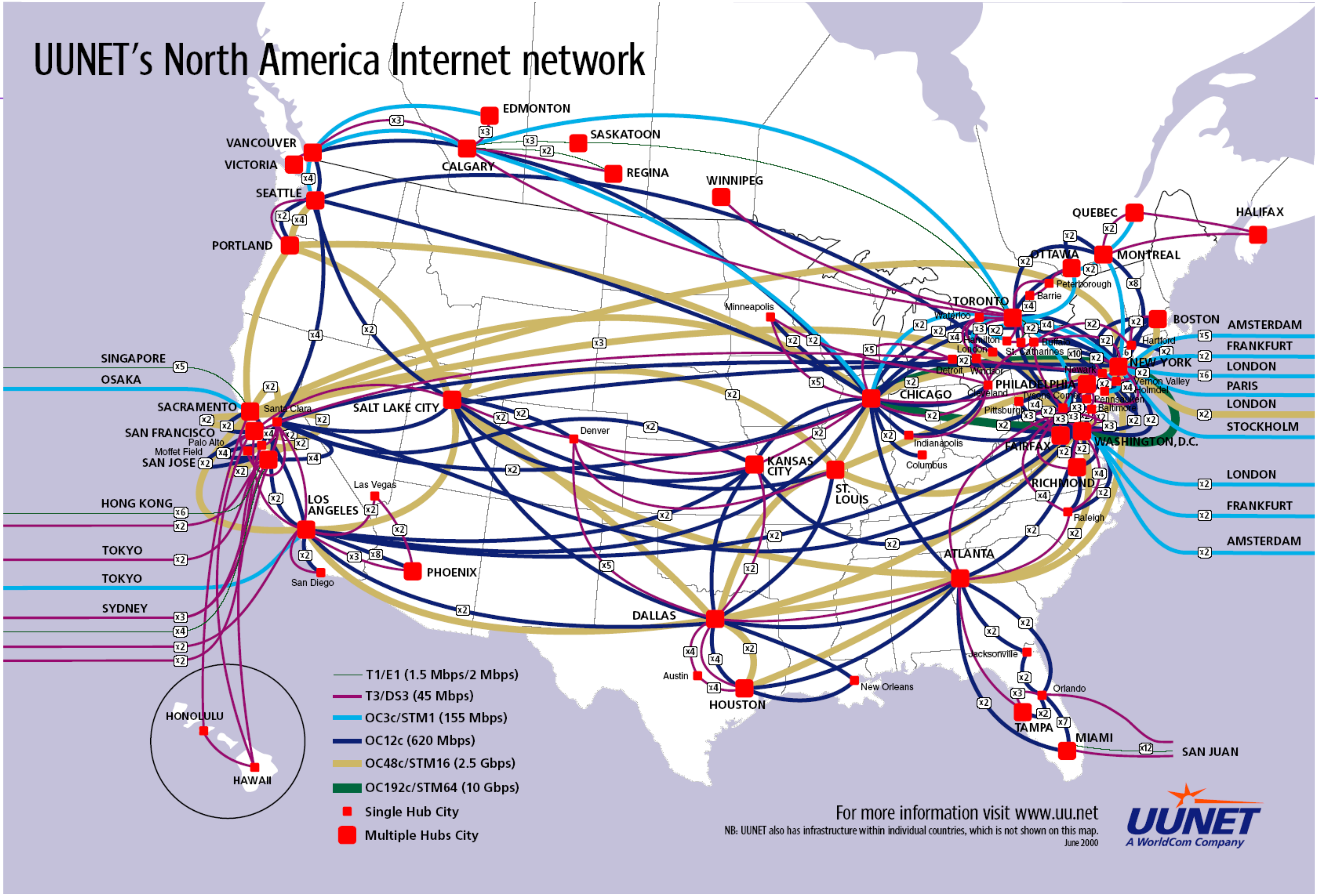


Internet structure: network of networks

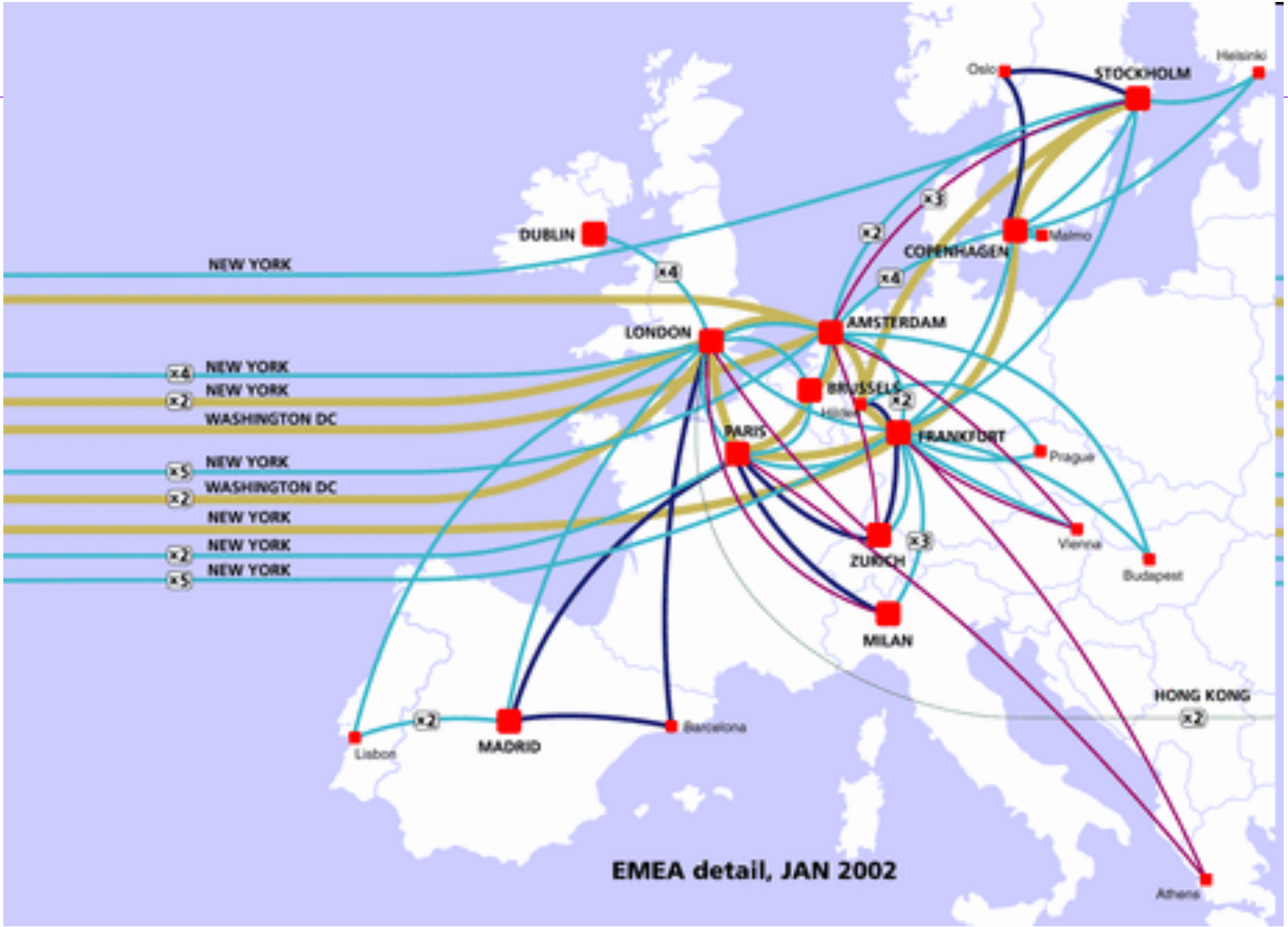
- a packet passes through many networks!



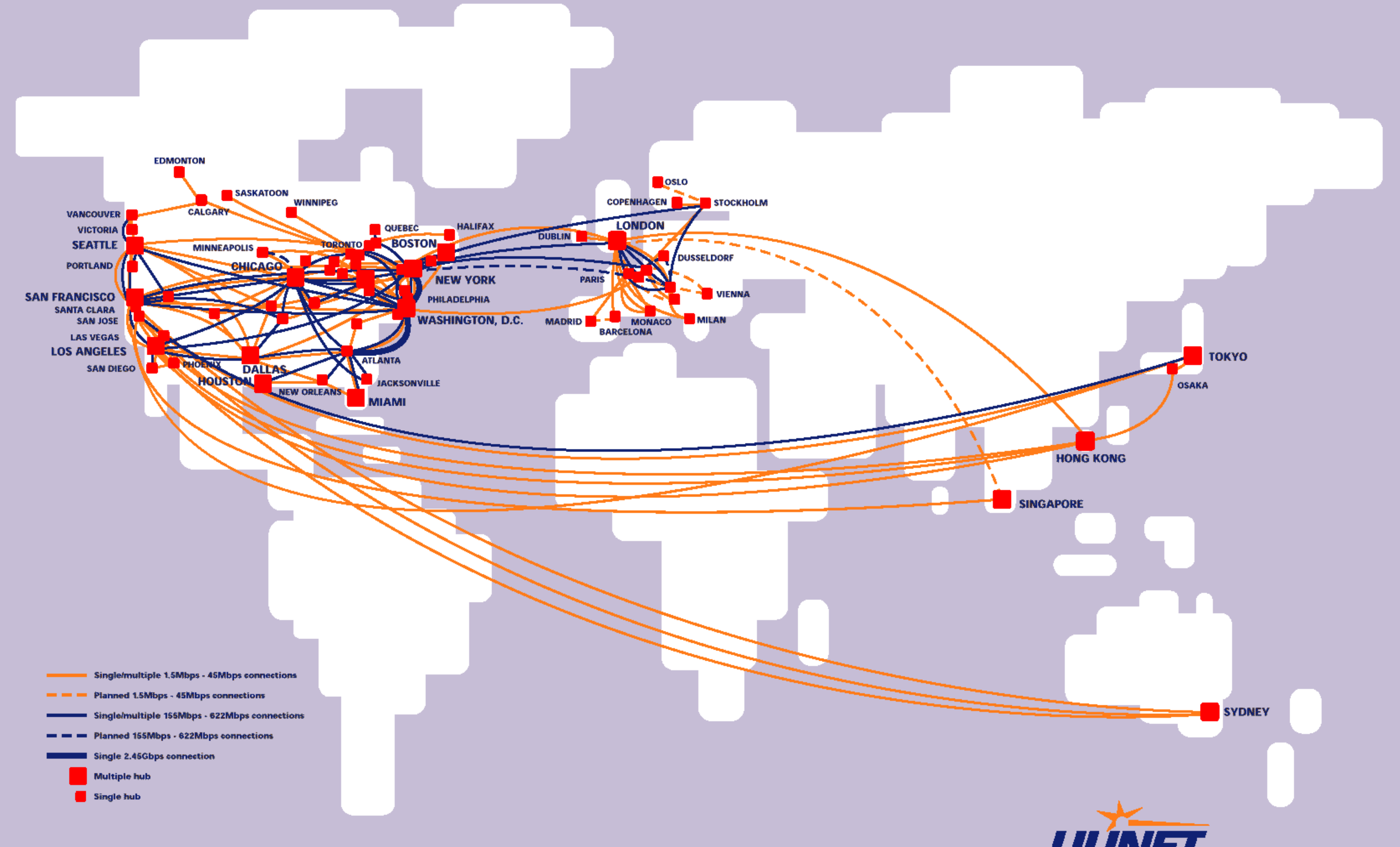
Try a traceroute!



UUNET's North America Internet network
EDMONTON
SASKATOON
VANCOUVER
VICTORIA
CALGARY
REGINA
WINNIPEG
SEATTLE
PORTLAND
QUEBEC
HALIFAX
OTTAWA
MONTREAL
TORONTO
BOSTON
AMSTERDAM
FRANKFURT
LONDON
PARIS
LONDON
STOCKHOLM
NEW YORK
PHILADELPHIA
WASHINGTON, D.C.
FAIRFAX
RICHMOND
LONDON
FRANKFURT
AMSTERDAM
SINGAPORE
OSAKA
SACRAMENTO
SAN FRANCISCO
SAN JOSE
SALT LAKE CITY
CHICAGO
KANSAS CITY
ST. LOUIS
HONG KONG
TOKYO
TOKYO
SYDNEY
LOS ANGELES
PHOENIX
DALLAS
ATLANTA
HOUSTON
TAMPA
MIAMI
SAN JUAN
HONOLULU
HAWAII
Denver
Las Vegas
San Diego
Minneapolis
Indianapolis
Columbus
Austin
New Orleans
Jacksonville
Orlando
Raleigh
T1/E1 (1.5 Mbps/2 Mbps)
T3/DS3 (45 Mbps)
OC3c/STM1 (155 Mbps)
OC12c (620 Mbps)
OC48c/STM16 (2.5 Gbps)
OC192c/STM64 (10 Gbps)
Single Hub City
Multiple Hubs City
For more information visit www.uu.net
NB: UUNET also has infrastructure within individual countries, which is not shown on this map.
June 2000
UUNET
A WorldCom Company


NEW YORK
DUBLIN
×4
NEW YORK
NEW YORK
WASHINGTON DC
NEW YORK
WASHINGTON DC
NEW YORK
NEW YORK
NEW YORK
LONDON
AMSTERDAM
BRUSSELS
PARIS
FRANKFURT
ZURICH
MILAN
COPENHAGEN
Malmo
Oslo
STOCKHOLM
Helsinki
Prague
Vienna
Budapest
MADRID
Lisbon
Barcelona
HONG KONG
Athens
EMEA detail, JAN 2002

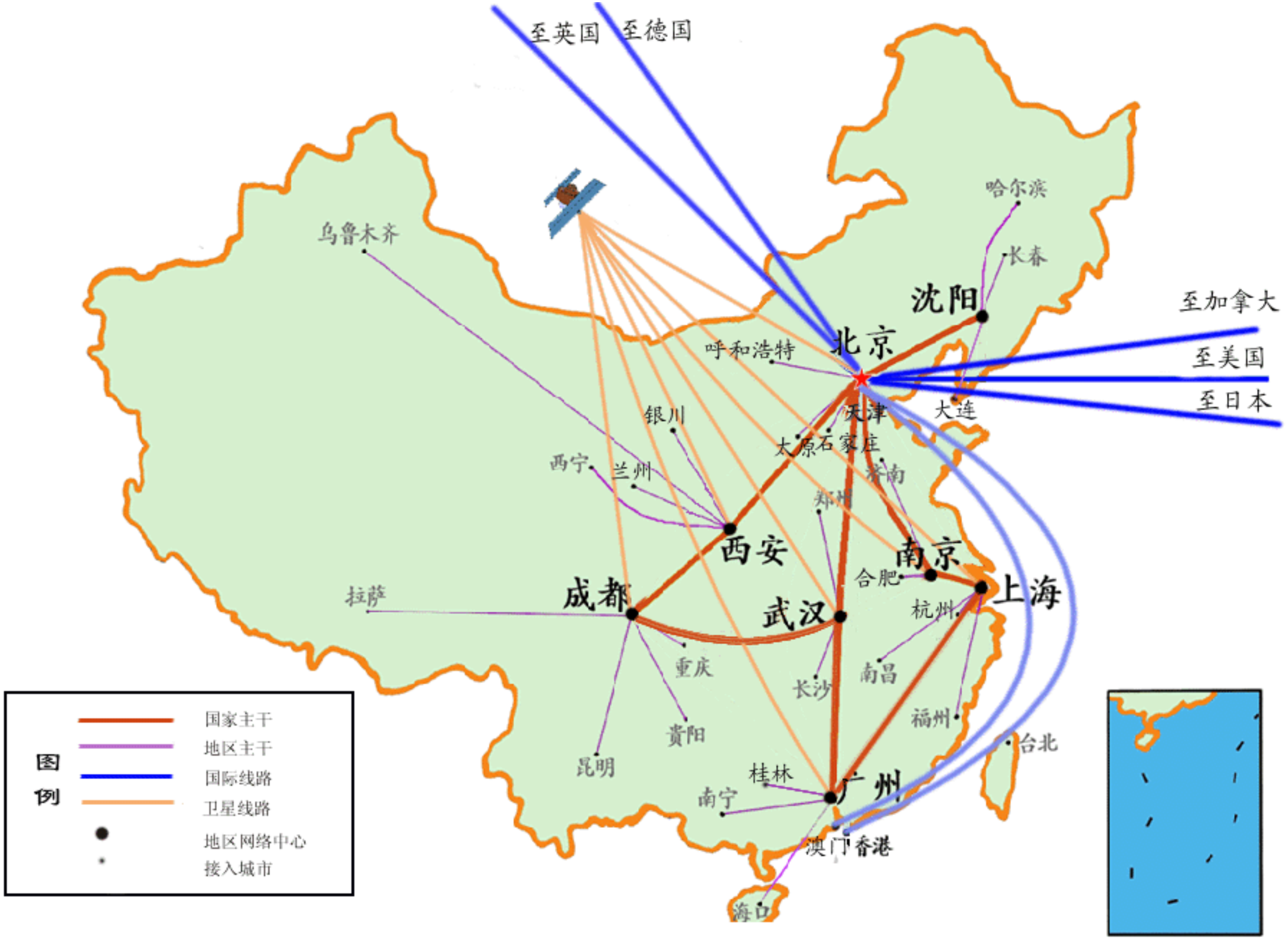
UUNET's Global Internet Backbone



For more information visit www.uu.net

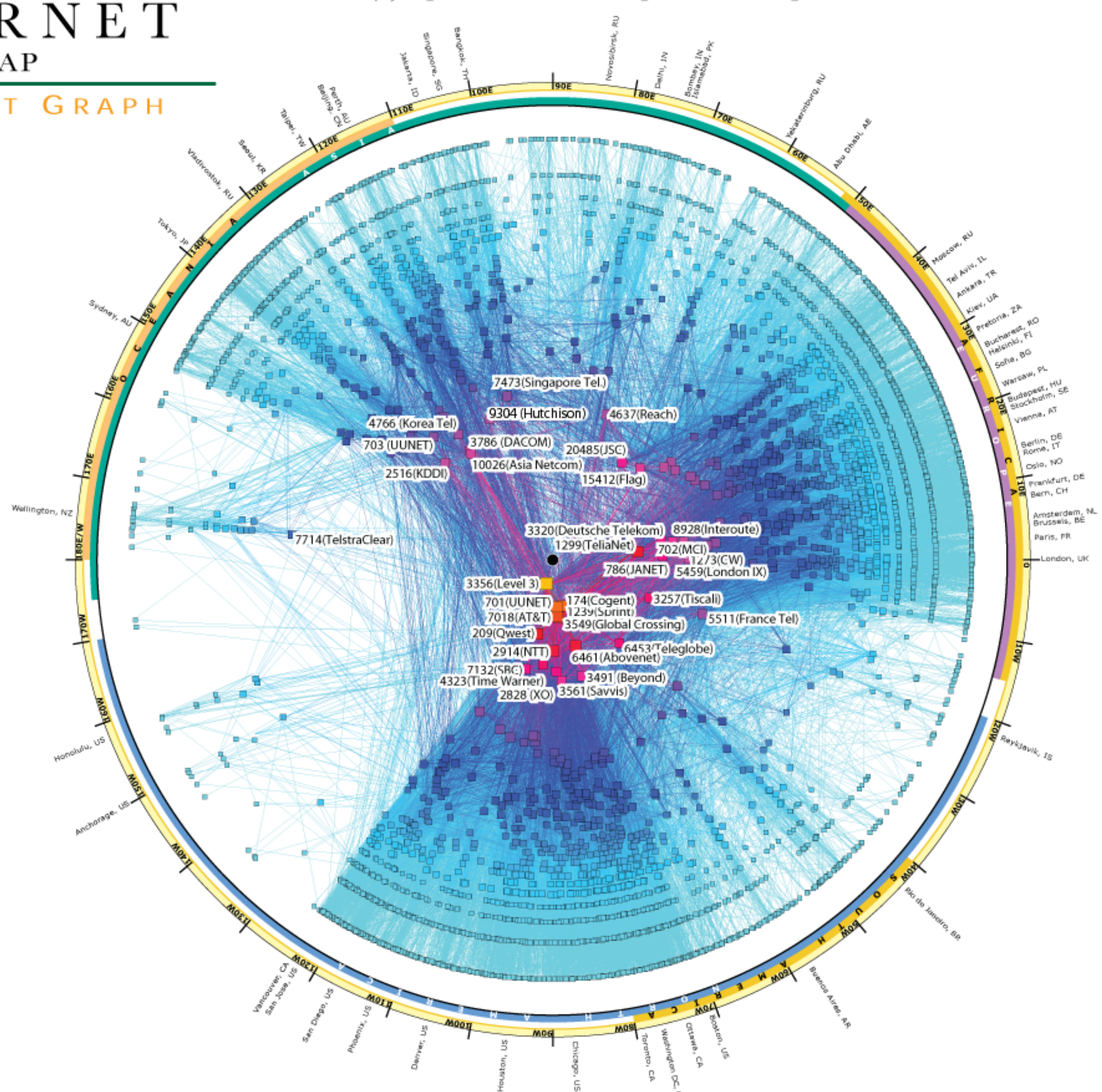
NB: With the exception of North America, Spain & Germany, all major 'in-country' links have been excluded from this map. 'In-state' links within the USA have also been excluded.

May/June 1999 2nd edition



中国教育和科研计算机网 CERNET 主干网

IPv4 INTERNET
TOPOLOGY MAP
AS-level INTERNET GRAPH



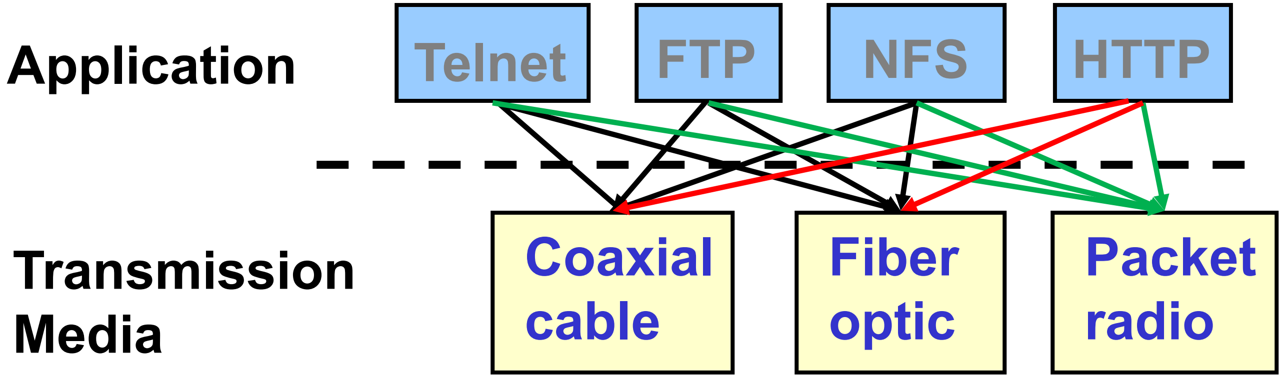
提纲

- ❑ Internet简介
- ❑ *协议分层和OSI参考模型*
- ❑ End-to-End Arguments
- ❑ Internet设计原则

问题是什么？

- Many different network styles and technologies
  - circuit-switched vs packet-switched, etc.
  - wireless vs wired vs optical, etc.
- Many different applications
  - ftp, email, web, P2P, etc.
- How do we organize this mess?

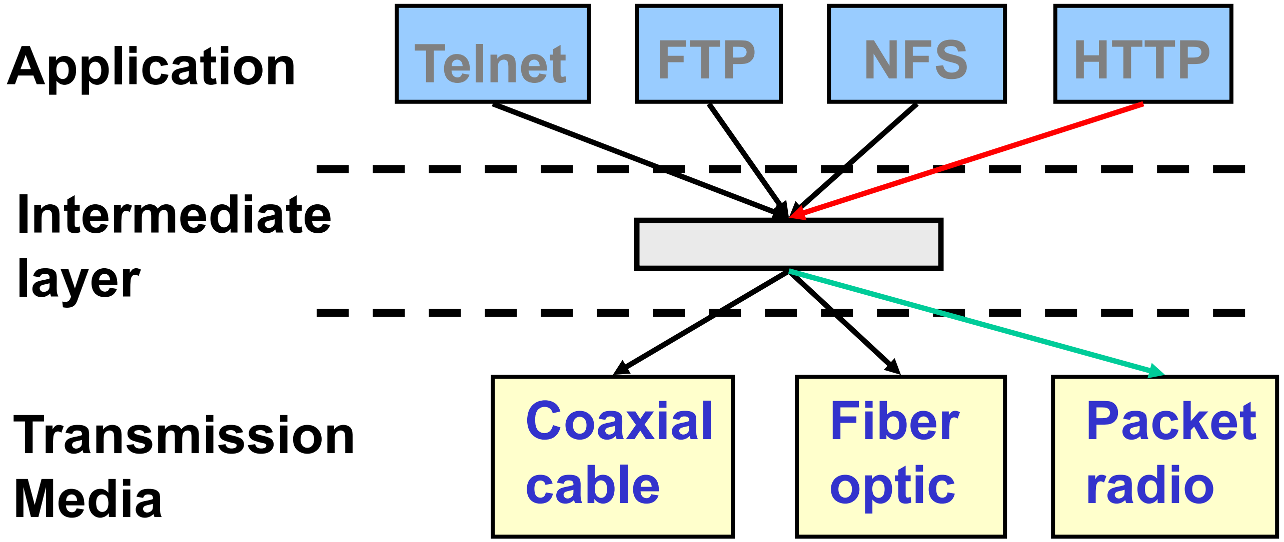
为什么要分层



- Do we re-implement every application for every technology?
- Obviously not, but how does the Internet architecture avoid this?

为什么要分层

- Solution: introduce an intermediate layer that provides a <u>unique</u> abstraction for various network technologies



体系结构

- Architecture is not the implementation itself
- Architecture is how to "organize" implementations
  - what interfaces are supported
  - where functionality is implemented
- Architecture is the modular design of the network

软件模块化

Break system into modules:

- Well-defined interfaces gives flexibility
    - can change implementation of modules
    - can extend functionality of system by adding new modules
- Interfaces hide information
    - allows for flexibility
    - but can hurt performance

微软垄断案

- 比尔·盖茨2002年4月22日在出庭作证时指出，如果将微软的ＩＥ网络浏览器剥离，**视窗的其它功能将被降级，视窗操作系统将停止运行**
- 2003年5月，微软和美国在线达成和解协议，愿意支付网景7.5亿美元

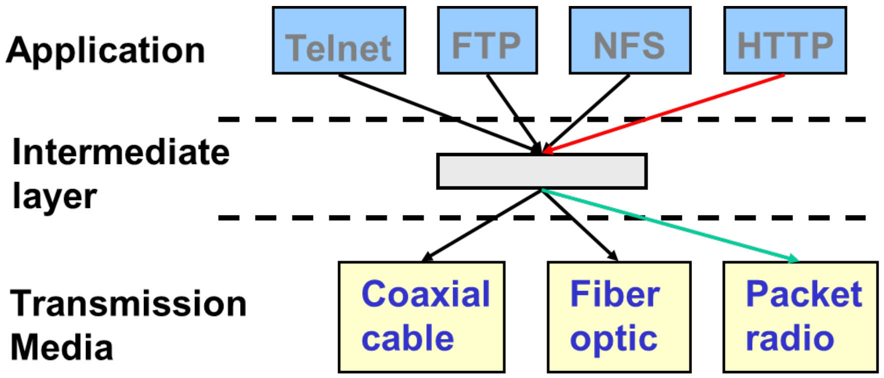
网络模块化

Like software modularity, but with a twist:

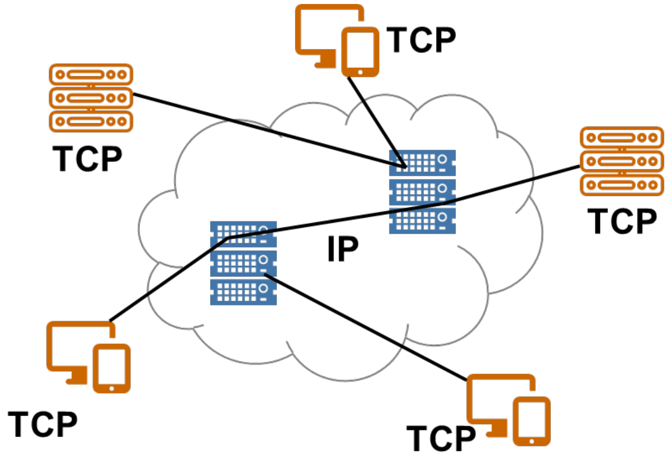
- Implementation distributed across routers and hosts
- Must decide both:
  - how to break system into modules
  - where modules are implemented
- Lecture will address these questions in turn

体系结构的两个方面

- Layering
  - how to break network functionality into modules
- The End-to-End Argument
  - where to implement functionality



Layering



End-to-End

分层

- Layering is a particular form of modularization
- The system is broken into a vertical hierarchy of logically distinct entities (layers)
- The service provided by one layer is based solely on the service provided by layer below

分层

- Advantages
  - Modularity – protocols easier to manage and maintain
  - Abstract functionality –lower layer can be changed <u>without</u> affecting the upper layer
  - Reuse – upper layer can reuse the functionality provided by lower layer
- Disadvantages
  - Information hiding – inefficient implementations

分层

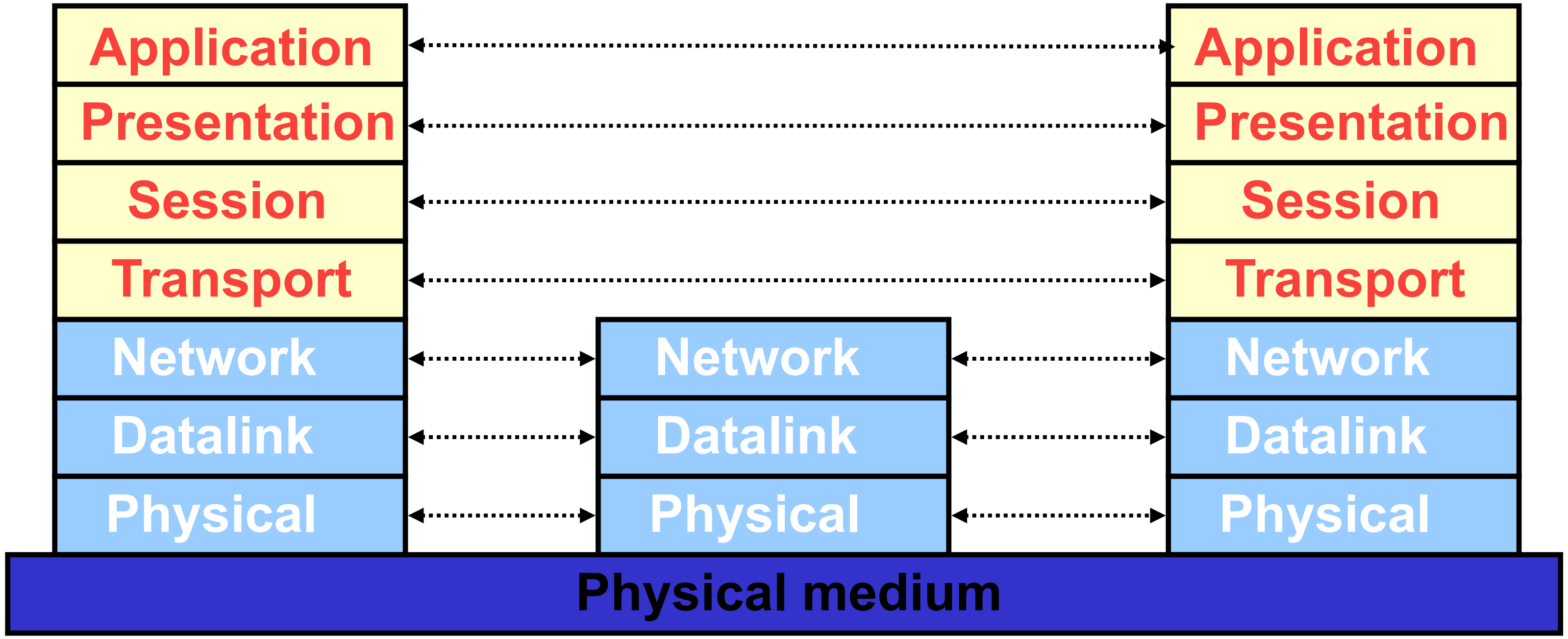
- Layer N software on the destination computer must receive exactly the message sent by layer N software on the sending computer
- Mathematically, if the sender applies a transformation T, the receiver must apply the inverse T^{-1}
- Recall the concept of "protocol"

ISO OSI Reference Model

- ISO – International Standard Organization
- OSI – Open System Interconnection
- Started to 1978; first standard 1979
  - ARPANET started in 1969; TCP/IP protocols ready by 1974
- Goal: a general open standard
  - Allow vendors to enter the market by using their own implementation and protocols

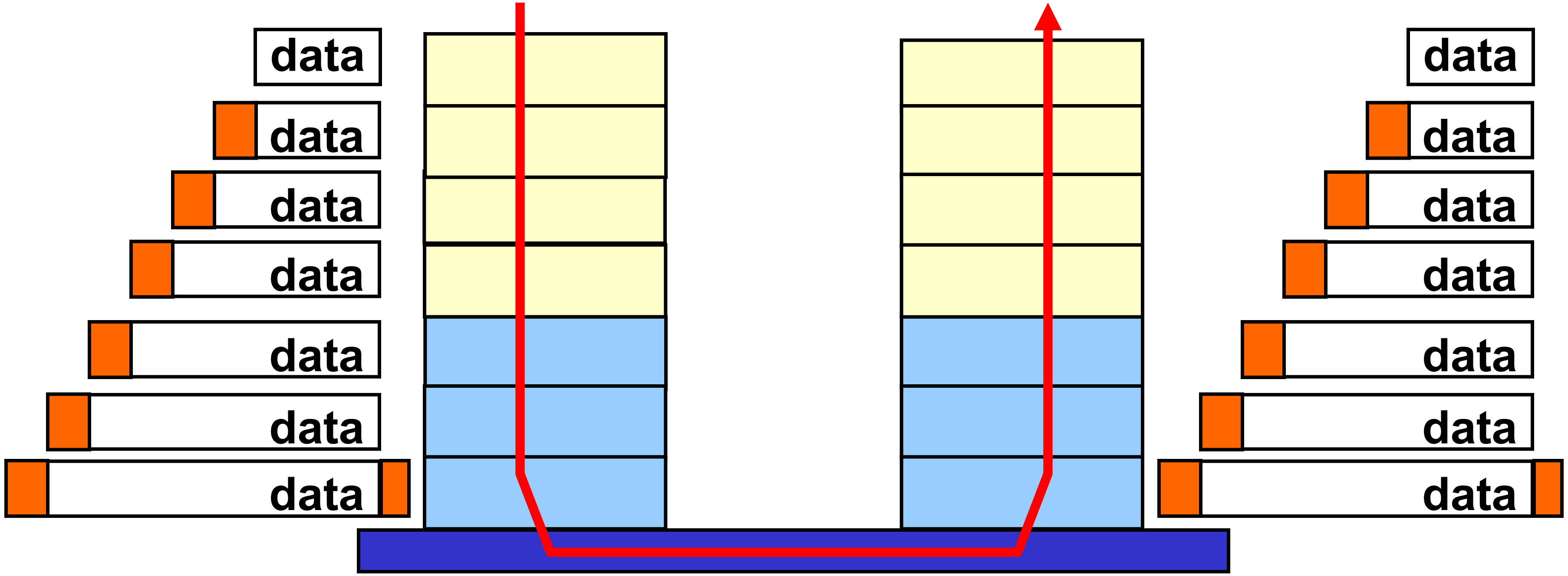
ISO OSI Reference Model

- Seven layers
  - Lower three layers are peer-to-peer
  - Next four layers are end-to-end



Data Transmission

- A layer can use only the service provided by the layer immediate below it
- Each layer may change and add a header to data packet



OSI Model Concepts

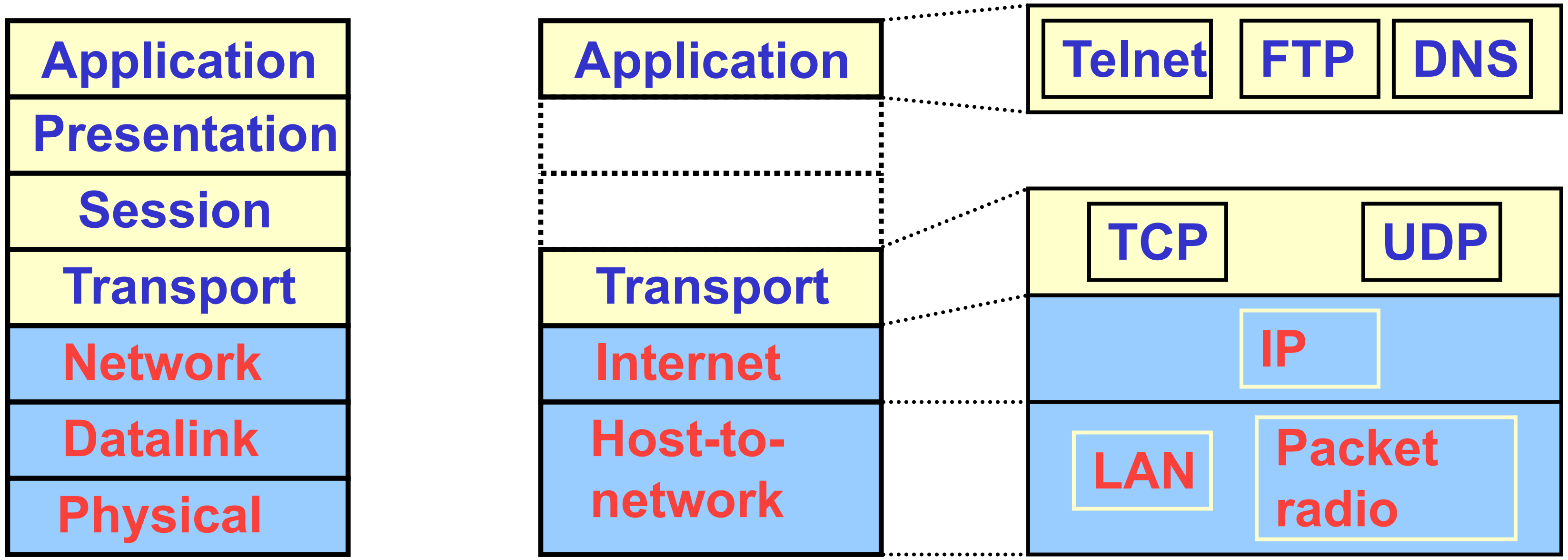
- Service – says what a layer does
- Interface – says how to access the service
- Protocol – says how is the service implemented
  - A set of rules and formats that govern the communication between two peers
  - protocol does not govern the implementation on a single machine, but how the layer is implemented between machines

The Origins of OSI

- http://williamstallings.com/Extras/OSI.html
- Much of the work on the design of OSI was actually done by a group at Honeywell Information Systems
  - Charlie Bachman as the principal technical member
- The group studied some of the existing solutions, including IBM's system network architecture (SNA), the work on protocols being done for ARPANET, the result of this effort was the development by 1977 of a seven-layer architecture known internally as the distributed systems architecture (DSA)
- In 1977, ISO formed a subcommittee on Open Systems Interconnection (Technical Committee 97, Subcommittee 16)
- This model was chosen as the only proposal to be submitted to the ISO subcommittee
- A consensus was reached at that meeting that this layered architecture would satisfy most requirements of Open Systems Interconnection, and had the capacity of being expanded later to meet new requirements. A provisional version of the model was published in March of 1978
- The next version, with some minor refinements, was published in June of 1979 and eventually standardized
- The resulting OSI model is essentially the same as the DSA model developed in 1977

OSI vs. TCP/IP

- OSI: conceptually define: service, interface, protocol
- Internet: provide a successful implementation



OSI模型的问题

- Andrew S. Tanenbaum 在 “Computer Networks”第三版中评价OSI:
  - Bad timing (too late)
  - Bad technology (both the model and the protocol are flawed)
  - Bad implementations (huge, unwieldy, and slow)
  - Bad politics (government and organizations bureaucrats)
- Internet 标准化名言 (David Clark of MIT):
  - “We reject kings, presidents, and voting；we believe in rough consensus and running code”

关键设计决策

- How do you divide functionality across the layers?

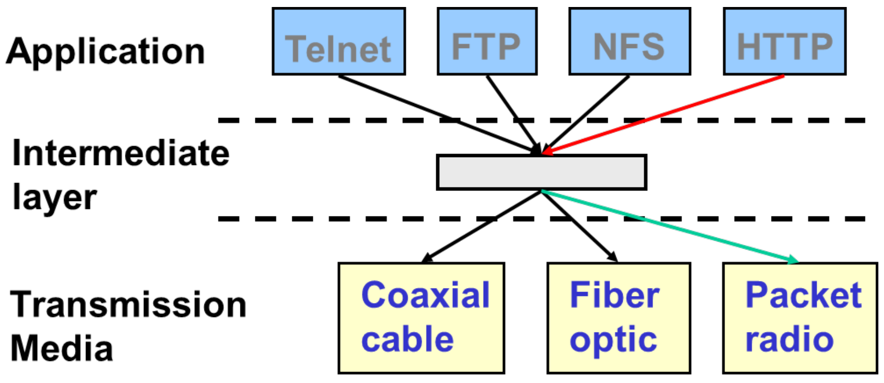
提纲

- Internet简介
- 协议分层和OSI参考模型
- *End-to-End Arguments*
- Internet设计原则

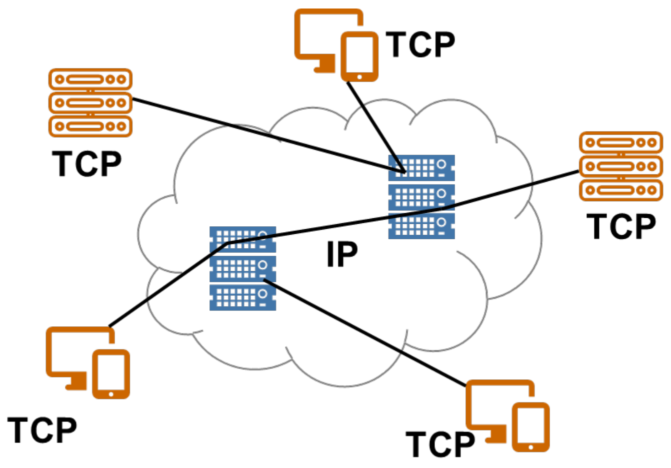
体系结构的两个方面

- Layering
  - how to break network functionality into modules

- The End-to-End Argument
  - where to implement functionality



Layering



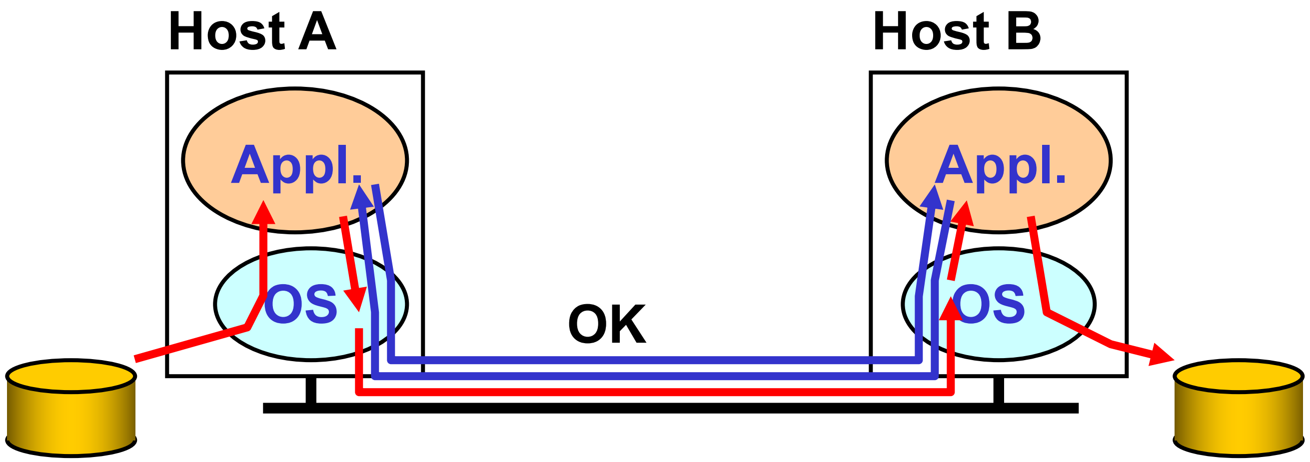
End-to-End

边缘论

- Think <u>twice</u> before implementing a functionality that you believe that is useful to an application at a lower layer
- If the application can implement a functionality correctly, implement it at a lower layer only as a <u>performance enhancement</u>
- **“核心简单，边缘复杂”**

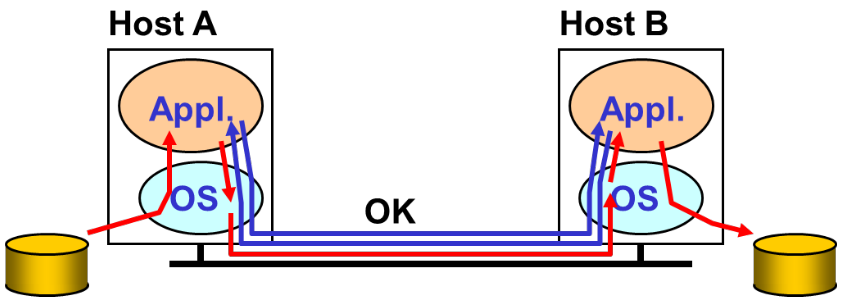
David D. Clark

Example: Reliable File Transfer



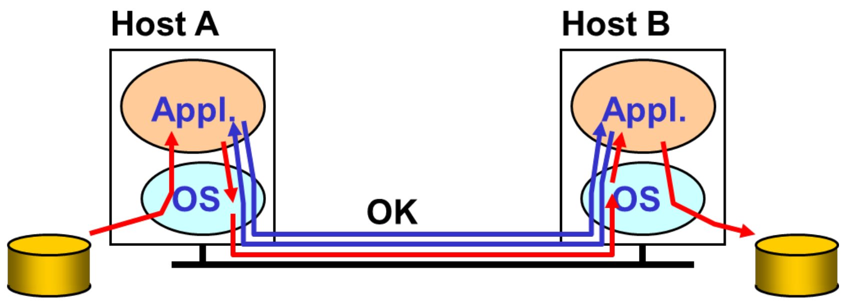
- Solution 1: make each step reliable, and then concatenate them
- Solution 2: end-to-end check and retry

Discussion



- Solution 1 is *not* complete
  - What happens if any network element misbehaves?
  - The receiver has to do the check anyway!
- Solution 2 is complete
  - Full functionality can be entirely implemented at application layer with *no* need for reliability from lower layers
- Is there any need to implement reliability at lower layers?

Discussion



Q: Is there any reason to implement reliability at lower layers?

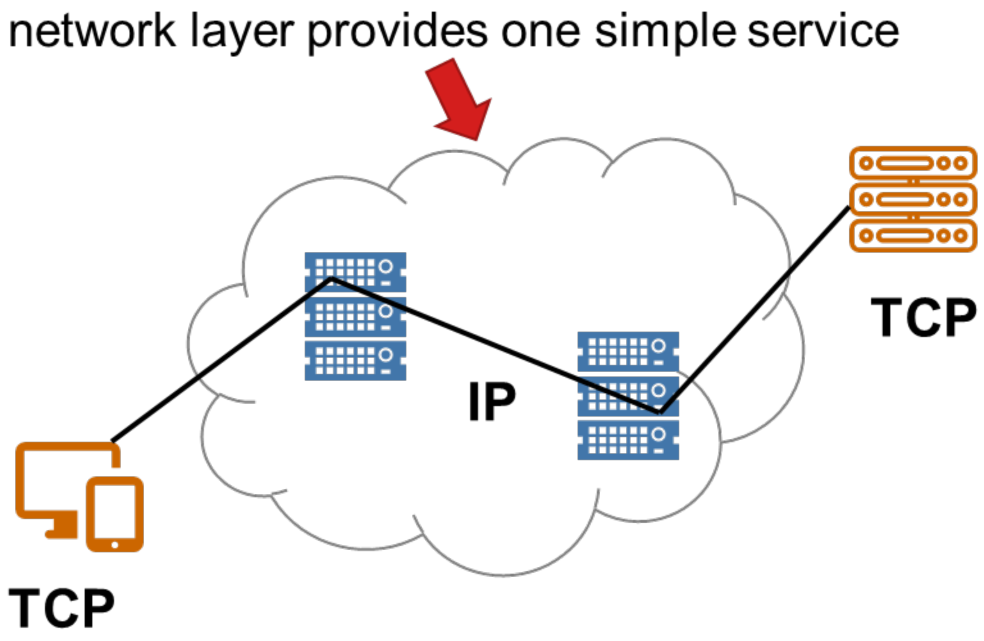
A: YES. “easier” (and more efficient) to check and recovery from errors at each intermediate hop

- e.g: faster response to errors, localized retransmissions

Trade-offs

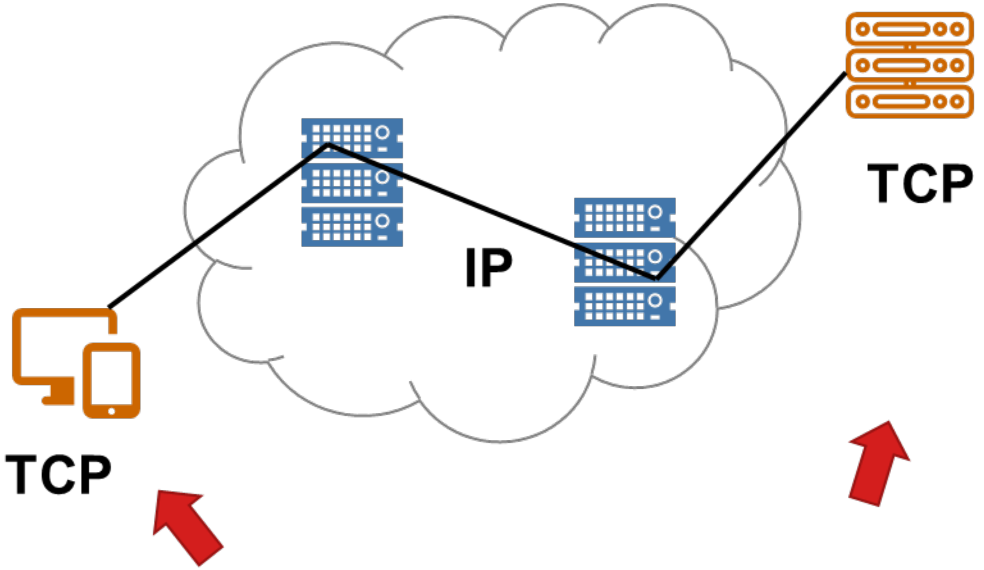
- application has more information about the data and semantics of required service (e.g., can check only at the end of each data unit)
- lower layer has more information about constraints in data transmission (e.g., packet size, error rate)
- *Note:* these trade-offs are a direct result of layering!

Internet & End-to-End Argument



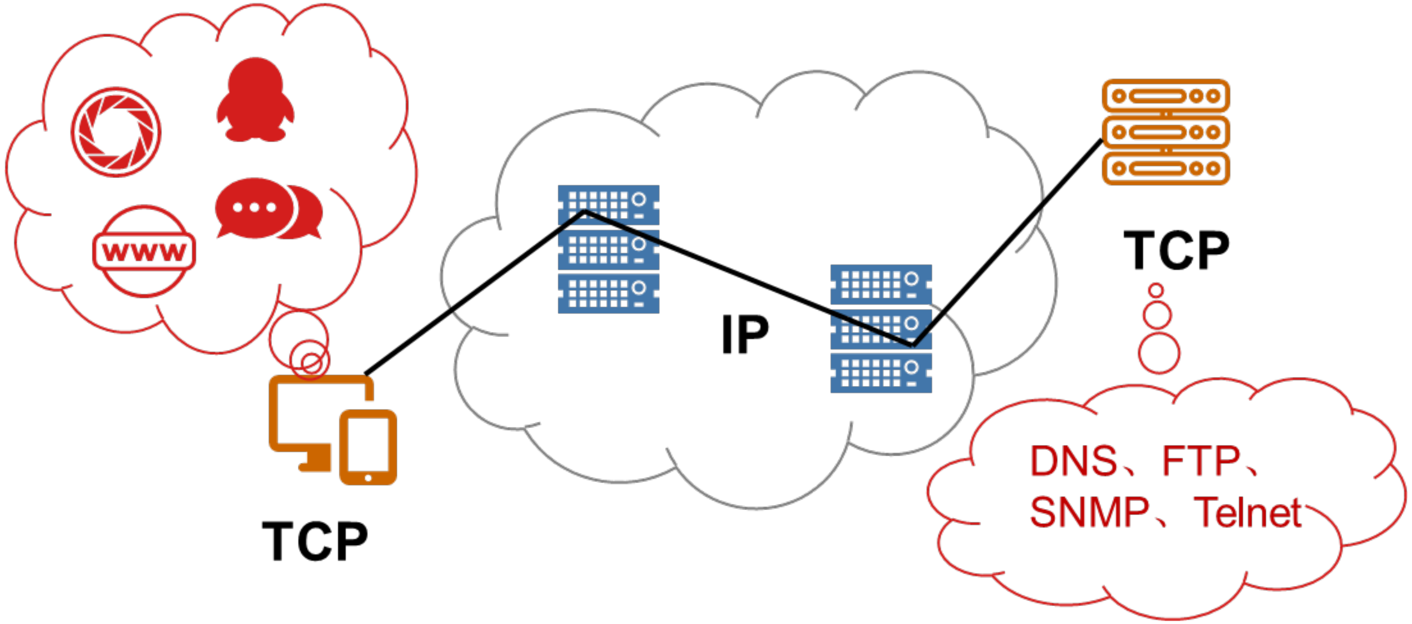
- network layer provides one simple service:
  - best effort datagram (packet) delivery

Internet & End-to-End Argument



- transport layer at network edge (TCP) provides end-to-end error control

Internet & End-to-End Argument



- all other functionality …
  - all application layer functionality
  - network services: DNS

implemented at *application* level

Internet & End-to-End Argument

Discussion: congestion control, flow control: why at transport, rather than link or application layers?

- congestion control needed for many application
- many applications "don't care" about congestion control – it's the network's concern
- consistency across applications- you *have* to use it if you use TCP (social contract – everybody does)
- why do it at the application level
  - Flow control – application knows how/when it wants to consume data
  - Congestion control – application can do tcp-friendly

Internet & End-to-End Argument

Why not at the link layer

- 1: not every application needs/want it
- 2: lots of state at each router (each connection needs to buffer, need back pressure) – it's hard

E2E Argument: Interpretations

- One interpretation:
  - A function can only be completely and correctly implemented with the knowledge and help of the applications *standing at the communication endpoints*
- Another: (more precise…)
  - a system (or subsystem level) should consider only functions that can be *completely and correctly* implemented within it

E2E Argument: Interpretations

- Alternative interpretation: (also correct …)
  - Think twice before implementing a functionality that you believe that is useful to an application at a lower layer
  - If the application can implement a functionality correctly, implement it a lower layer *only* as a performance enhancement

End-to-End Argument: Critical Issues

- end-to-end principle emphasizes:
  - *function placement*
  - *correctness, completeness*
  - *overall system costs*
- Philosophy: if application can do it, don't do it at a lower layer -- application best knows what it needs
  - add functionality in lower layers iff
  - (1) used by and improves performances of many applications
  - (2) does not hurt other applications
- allows *cost-performance* tradeoff

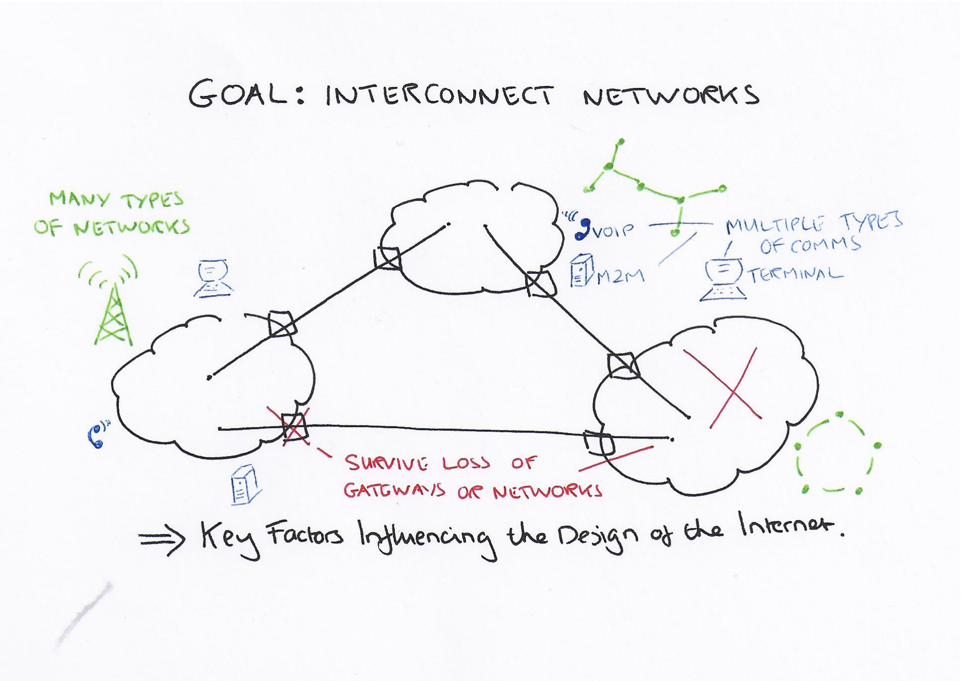
提纲

- Internet简介
- 协议分层和OSI参考模型
- End-to-End Arguments
- ***Internet设计原则***

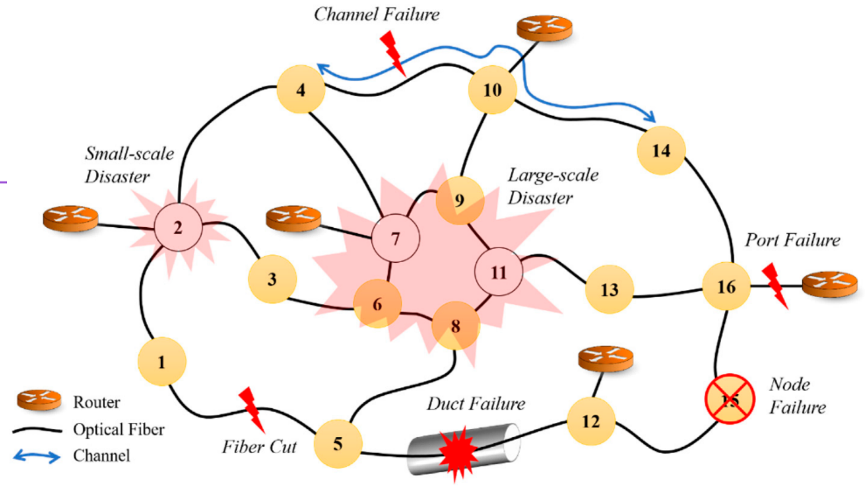
Internet Design Philosophy (Clark'88)

In order of importance:

0. Connect existing networks
   - initially ARPANET, ARPA packet radio, packet satellite network
1. Survivability
   - ensure communication service even with network and router failures
2. Support multiple types of services
3. Must accommodate a variety of networks
4. Allow distributed management
5. Allow host attachment with a low level of effort
6. Be cost effective
7. Allow resource accountability

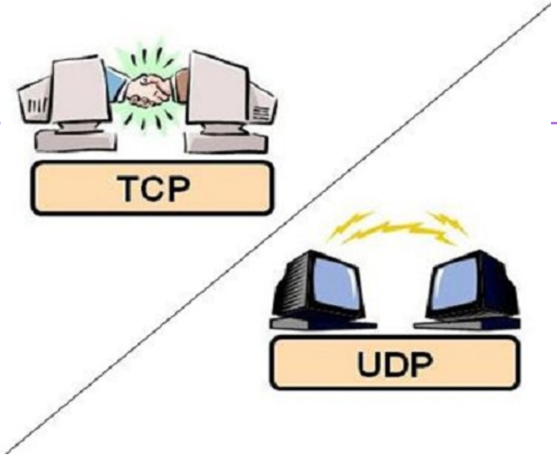


1. Survivability



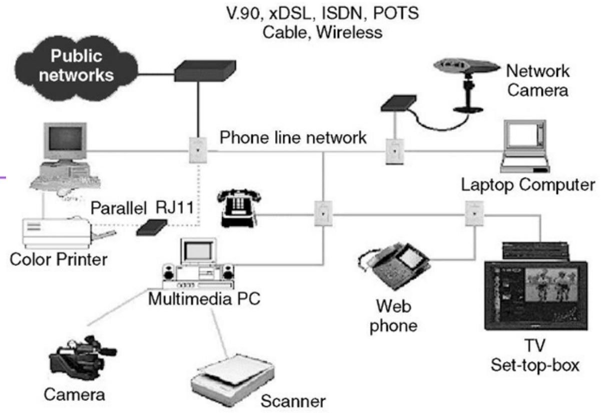
- Continue to operate even in the presence of network failures (e.g., link and router failures)
  - as long as network is not partitioned, two endpoints should be able to communicate
  - any other failure (excepting network partition) should be transparent to endpoints
- Decision: maintain e-e transport state only at end-points
  - eliminate the problem of handling state inconsistency and performing state restoration when router fails
- Internet: stateless network-layer architecture
  - No notion of a session/call at network layer
- Grade: A-
  - routing algorithm failover path is non-optimal, non-traffic sensitive (Note: ISPs worry about this)

2. Types of Services



- add UDP to TCP to better support other apps
  - e.g., "real-time" applications
- arguably main reason for separating TCP, IP
- datagram abstraction: lower common denominator on which other services can be built
  - service differentiation was considered (remember ToS?), but this has never happened on the large scale (Why?)
- Grade: AB
  - Need something (reliability) between TCP and UDP? Why not just build on top of UDP
  - Need time sensitivity for multimedia application
  - Need a quality of service notion: give me throughput X or give me a busy signal (this is what ATM is/was advocating)

3. Variety of Networks



- Very successful (why?)
  - because the minimalist service; it requires from underlying network only to deliver a packet with a "reasonable" probability of success
- …does not require:
  - reliability
  - in-order delivery
- The mantra: IP over everything
  - Then: ARPANET, X.25, DARPA satellite network..
  - Now: ATM, SONET, WDM…
- Grade: A
  - because it runs over everything

Other Goals

- Allow distributed management
  - Administrative autonomy: IP interconnects networks
    - each network can be managed by a different organization
    - different organizations need to interact only at the boundaries
    - … but this model complicates routing
  - Grade: B
  - Why: some stuff managed centrally: DNS, IP address allocation (but it's not that much)
  - Today's distributed management makes it easy for misconfigurations or malicious users to corrupt infrastrcuture (e.g., AT&T routing black hole)

Other Goals

- Cost effective
  - sources of inefficiency
    - header overhead
    - retransmissions
    - routing
  - …but "optimal" performance never been top priority
  - Grade: AB (500 million people can't be wrong!)

Other Goals

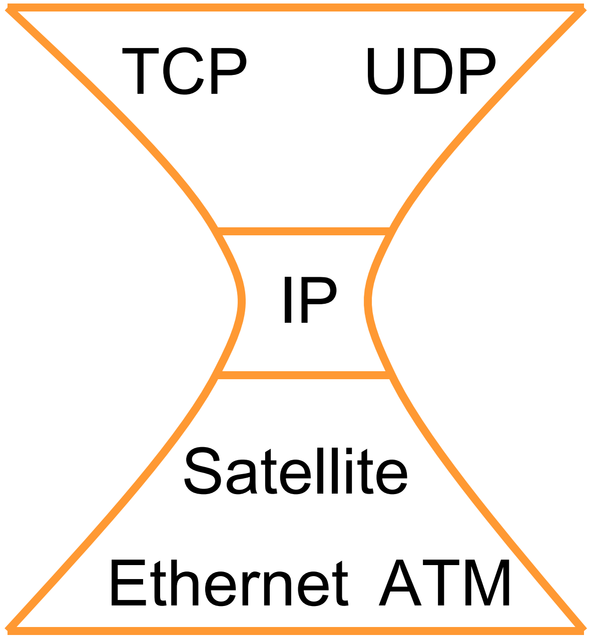
- Low cost of attaching a new host
  - not a strong point → higher than other architecture because the intelligence is in hosts (e.g., telephone vs. computer)
  - bad implementations or malicious users can produce considerably harm
  - Grade: B
    - Leverages low cost of end-system hardware (Ethernet NICs ~ $20), DHCP makes self configuration easy
    - Very hard to debug problems
- Accountability
  - Grade: F

What About the Future

- Datagram not the best abstraction for:
  - resource management, accountability, QoS
- new abstraction: flow (see IPv6)
  - but no one knows what a flow *is*
- routers require to maintain per-flow state
- state management: recovering lost state is hard
- here (1988) we see the first proposal of "soft state"!
  - soft-state: end-hosts responsible to maintain the state

Summary: Internet Architecture

- packet-switched datagram network
- IP is the glue (network layer overlay)
- IP hourglass architecture
  - all hosts and routers run IP
- stateless architecture
  - no per flow state inside network

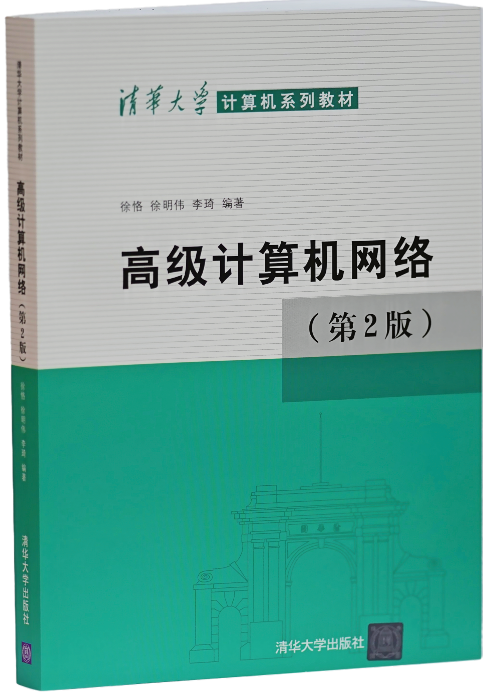


IP hourglass

Summary: Minimalist Approach

- Dumb network
  - IP provide minimal functionalities to support connectivity
  - addressing, forwarding, routing
- Smart end system
  - transport layer or application performs more sophisticated functionalities
  - flow control, error control, congestion control
- Advantages
  - accommodate heterogeneous technologies (Ethernet, modem, satellite, wireless)
  - support diverse applications (telnet, ftp, Web, X windows)
  - decentralized network administration

公众号与参考书



- ❑ 徐恪 徐明伟 等 "高级计算机网络"，清华大学出版社
- ❑ 徐恪 任丰原 等 "计算机网络体系结构-设计、建模、分析与优化"，清华大学出版社