

Internet and Open Source

A story of freedom

Riccardo Lancellotti

SECloud research group

University of Modena and Reggio Emilia

25/10/2025

Introduction

Technology is not just technology

GUADEC 2025



Some relevant quotes

Georges Basile Stavrakas Neto

If we drop our code and we rewrite it from scratch, would it still be GNOME? Yes because we would be the same people that wrote it in the first time. GNOME is the output of a community not just a piece of software



Federico Mena Quintero

GNOME is about people, not about software. Developing GNOME meant making friends, For example, friends working to the Linux Kernel that helped us creating the iNotify API for the file manager



Internet is complex

Send a Message (set of bits) from one host to another. Looks easy...

Where is the problem?

Political Map of the World, January 2015



- Heterogeneity
 - What hw/sw features do the computers have?
 - How are the computers interconnected?
 - How do you manage the transit of messages through intermediate nodes?
 - What services can the user use?
 - ...
- Things can go wrong
 - Electrical interference (bit level errors)
 - Congestion (message level errors/delay)
 - Failures of links and intermediate nodes
 - Sender/destination node software issues
 - Out-of-order message delivery
 - ...

What to do when the complexity is very high?

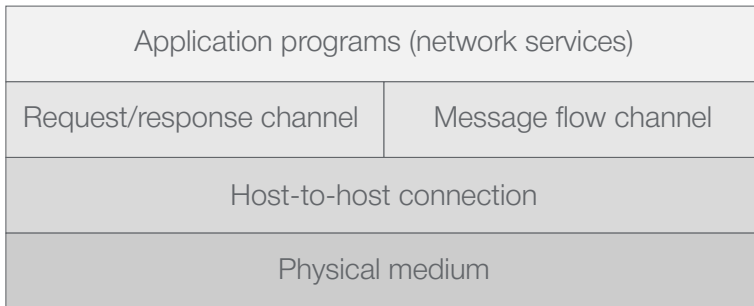
- **Divide** the problem into sub-problems
- **Solve** the sub-problems
- **Connect** the partial solutions

From an IT point of view

- **Layering** is the typical “IT way” to deal with the complexity of a problem
- Use **abstractions** to mask complexity
- Abstraction naturally leads to layering

First example: HW – OS – applications

In network architectures there can be different abstractions and also different alternatives for each level



The history of Internet can be divided into 3 main phases

- Initial vision
- TCP/IP foundation
- Services (and tuning for new application scenarios)

Let's meet a few guys from the past...

Initial vision

Minimal Bio:

- Ph.D. From MIT in 1963
- From 1963 @UCLA (still working there)
- Laurea Honoris Causa from University of Bologna in 2005

Paradigm-shifting contribution:

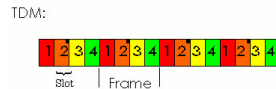
- Ph.D. work “Information Flow in Large Communication Nets” started in 1961



Two ways to transfer data

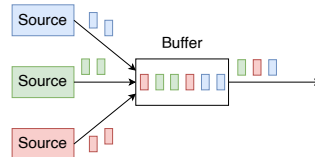
Circuit switching

- A dedicated virtual circuit for each communication
- Examples:
 - Frequency Division Multiplexing (FDM)
 - Time Division Multiplexing (TDM)
- The idea behind the telephone system



Packet switching

- The data is divided into “parts” and sent across the network
- The idea behind the Internet



Packet switching is a revolutionary idea

- Kelinrock develops mathematical tools (**Queuing Theory**) for studying performance in packet-switching networks
- He demonstrates that packet switching outperforms traditional communication models
- Throughout the 1960s (and many thereafter...), telecommunications “experts”, supporters of circuit-switching communications, declared “It will never work”

From “The history of the Internet written by those who created it”, 1997

Minimal Bio:

- Joseph Carl Robnett Licklider
- Associate professor in the '50 @MIT
- 1962: first director of the Information Processing Techniques Office (IPTO) of the Advanced Research Projects Agency (ARPA) of the Department of Defense

Relevant publications

- Paper “Man Computer symbiosis” (1960)
- Memorandum For Members and Affiliates of the Intergalactic Computer Network (1963)
- Book “Libraries of the Future” (1965)



From “Man Computer symbiosis”

- *[Online] thinking centers [that] incorporate the functions of present-day libraries*
- *A network of such centers, connected by wide-band communications lines*

From Memorandum For Members and Affiliates of the IGCN

- *Consider the situation in which several different centers are netted together, each center being highly individualistic and having its own special language and its own special way of doing things. Is it not desirable, or even necessary for all the centers to agree upon some language or, at least, upon some conventions*
- *It seems to me to be interesting and important, nevertheless, to develop a capability for integrated network operation*

TCP/IP foundation

Bob Kahn

- Ph.D. from Princeton
- CEO of Corporation for National Research Initiatives

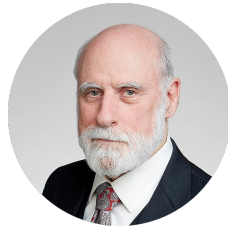


Vint Cerf

- Master degree from Stanford, Ph.D. from UCLA
- Currently Chief Internet Evangelist @Google

Main contribution

- Main architects of TCP/IP stack
- IP and TCP protocols
- 2004 Turing Award



Protocol stack comparison

ISO/OSI (*De Jure standard*)

Application
Presentation
Session
Transport
Network
Data link
Physical

TCP/IP (*De Facto standard*)

Application
...
Transport
Network
Host to Network

TCP/IP

- Availability of a good open source stack implementation in the mid-1980s on BSD Unix
- Availability of a good set of APIs (BSD socket API) to develop network applications: not perfect, but working

ISO/OSI

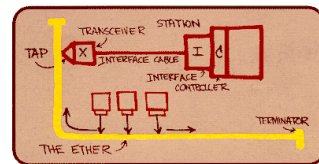
- The ISO/OSI committee defined the stack specifications
- Working implementations of the ISO/OSI specifications lagged far behind those already available TCP/IP

Minimal Bio:

- Robert Metcalfe
- Ph.D. from Harvard on packet switching technologies
- Developed Ethernet while @Xerox PARC
- Founder of 3com
- Grace Murray Hopper + Turing Awards
- Eating his article on Internet collapse @WWW6 conference (1997)



- Today dominant position on the market
- The original design included only one shared channel to connect the nodes
- Based on Radio data transmission (University of Hawaii)
- Strong points
 - Simple and Cheap
 - Easy to extend and adapt
 - works well for bursty traffic
- More complex protocols (e.g., Token Ring) no longer supported



Samuel J. Leffler:

- Work on BSD/FreeBSD Unix
- Various positions @ Lucasfilm, Pixar, SGI, VMWare

William N. Joy:

- Co-founder of Sun Microsystems (Solaris)
- Re-wrote from scratch TCP/IP Stack for BSD unix
- Grace Murray Hopper Award

Robert S. Fabry:

- Professor @ Berkeley

Main joint contributions:

- **Socket API** for TCP/IP in BSD
- BSD Unix filesystem



- TCP/IP implemented in kernel
- Applications run in User Space
- Need for interface to make TCP/IP usable by applications
- Design principles
 - Support generic IPC
 - Leverage unix file abstraction
 - Easily extended to other protocols (including IPv6)
- Adopted by most OS to support TCP/IP
 - Reference to prof. Rizzo in Windows credits



Why Unix?

- Unix born as successor of MULTICS project
- Developed initially by AT&T
- People working at Unix
 - Brian Kernighan and Dennis Ritchie (C language)
 - Ken Thompson
- AT&T has has telephony monopoly
- No economic activities allowed outside telephony



Why Berkeley?

- AT&T cannot sell Unix
 - Can use and maintain it internally
 - Can give it away for free
- Many universities are interested in Unix
- Great OS to study and modify
- Ken Thompson spent sabbatical year @Berkeley



Services (and tuning)

Jon Postel

- Master degree @UCLA
- RFC editor
- Email (With D. Crocker)
- DNS
- IANA
- “Be conservative in what you do, be liberal in what you accept from others”
- V. Cerf: I Remember IANA (RFC 2468)



- Naming is a critical function of every distributed Systems
- IP addresses are not human-friendly (Who can remember phone numbers?)
- Original approach was to keep a list of name-IP conversions and periodically synchronize every Internet host
- Not scalable, error prone

Furthermore, the namespace was flat. Only one machine could be named FRODO. Furthermore, names were relatively short (24 characters maximum), so users had to become increasingly creative about hostnames as the network grew. In 1982, the Internet community set out to replace HOSTS.TXT with a distributed database.

[Craig Partridge, BBN]

Minimal Bio:

- Physics M.Sc. from Oxford
- Work @CERN (1984 and 1989)
- Founder of W3C @MIT
- Turing Award

Main contribution:

- World Wide Web

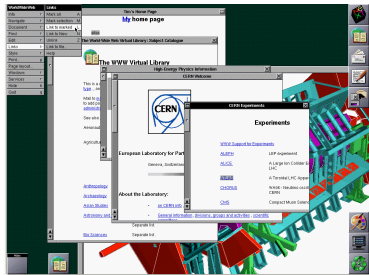


WWW is based on existing ideas:

- DNS for identifying hosts
- Status codes similar to SMTP (e-mail)
- TCP/IP networking
- Idea of hypertext (Nelson & Engelbart, 1965)
- Markup languages (SGML in particular)

Reason for WWW success:

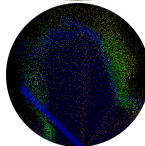
- Working prototype
- Idea of URL
- Easy to extend and adapt to new scenarios
- Multimedia (thanks to NeXT toolkit)



Research group @Google

- Neal Cardwell
- Yuchung Cheng
- C. Stephen Gunn
- Van Jacobson (many other contributions)

Working on BBR TCP congestion control



Two tasks of TCP (thanks to Vint Cerf)

- **Congestion control**: the sending host must decrease the packet transmission rate when the network is congested
- **Flow control**: the sending host must not overload the receiving host

Mechanism used

- Define how many data can be **in-flight**
- Sent but not yet acknowledged
- Use sliding Windows

Loss-based

- Congestion is detected when a packet is **lost** (routers queues are full)
- Congestion **must occur** before acting + Problem with Long Fat Networks
- Traditional examples: Tahoe, Reno (both obsolete)
- LFN-aware examples: BIC, Cubic (Current default in most OS)

Time-Based

- Congestion is detected before occurrence (monitor RTT)
- Too sensitive to other TCP flows (**friendliness** problem)
- Example: Vegas

Model Based (only BBR)

- Creates performance model of transmission channel
- Insensitive to loss and to interference
- Developed by Google for YouTube servers

Concluding remarks

- Internet is not the work of single person
- **Community** of people
 - Most developer **knew** each other (several academic ties)
 - Remember Licklider memo!
 - Academia, industries and political bodies working **together**
- Focus on **open standards**
 - Guaranteeing access to everyone
 - (Almost) no bias in basic design
 - Solving today problem without compromising future development

A look at the future

- The critical point of **net neutrality**
- The growing weight of **corporate entities**
- **Hyperscalers**
 - Google owns 8.5% of world submarine cables
 - 70% of network expenses related to hyperscalers
- **Gender bias** (ok for the '60, **not ok** for the 2025)
 - 8% woman in IETF (2023)
 - ~25% woman in computer science (now)
 - ~35% woman in computer science (1980)
 - No Grace Hopper/Dorothy Vaughan for Internet (yet?)
- **Geographic bias**
 - No hyperscalers from Europe/Africa/LA

