Il software libero: genesi, stato e prospettive

Alberto Cammozzo
alberto (at) cammozzo.com

31 maggio 2018

Seminario per Sistemi Operativi – Corso di laurea in Informatica
Università di Padova

I - genesi

tre storie:

1- GNU & FSF

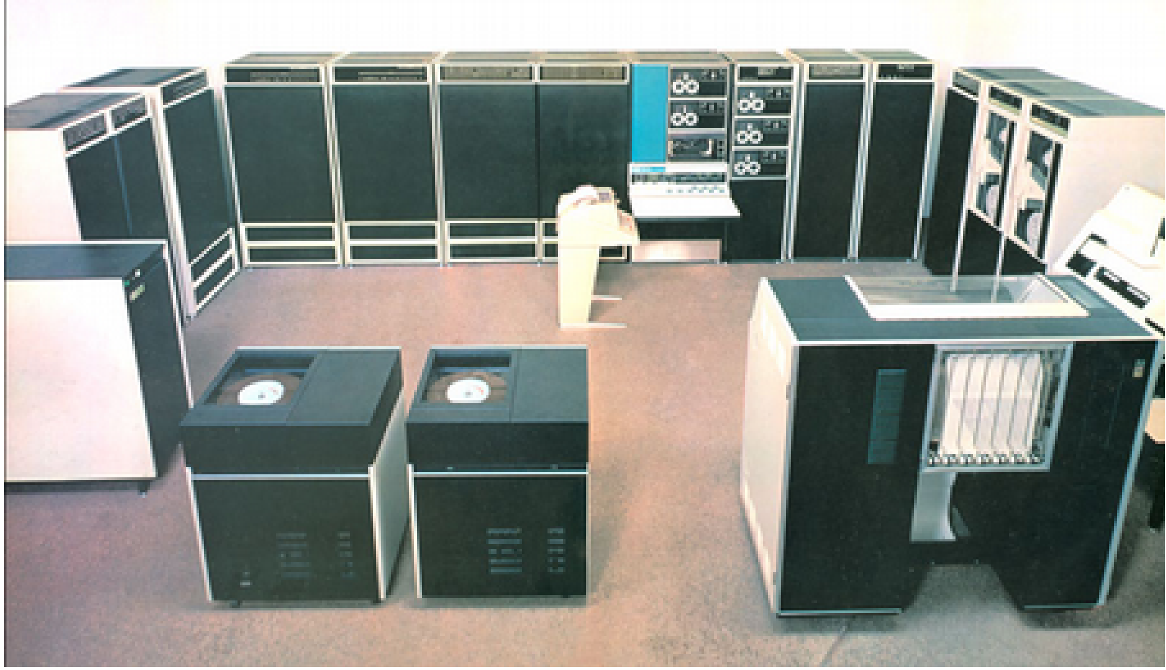
2- Linux

3- Open Source

1963 – Mainframe DEC PDP-6

Sui primi computer il software era distribuito in sorgente, e condiviso in una *comunità* di utenti/sviluppatori.

Con la diffusione dei minicomputer e dei PC (1981):
si apre un mercato del software venduto a parte (UNIX, DOS, CP/M,...), solo in forma eseguibile per evitare concorrenza,
a degli *utenti isolati*.



Richard Stallman, MIT AI lab, USA,

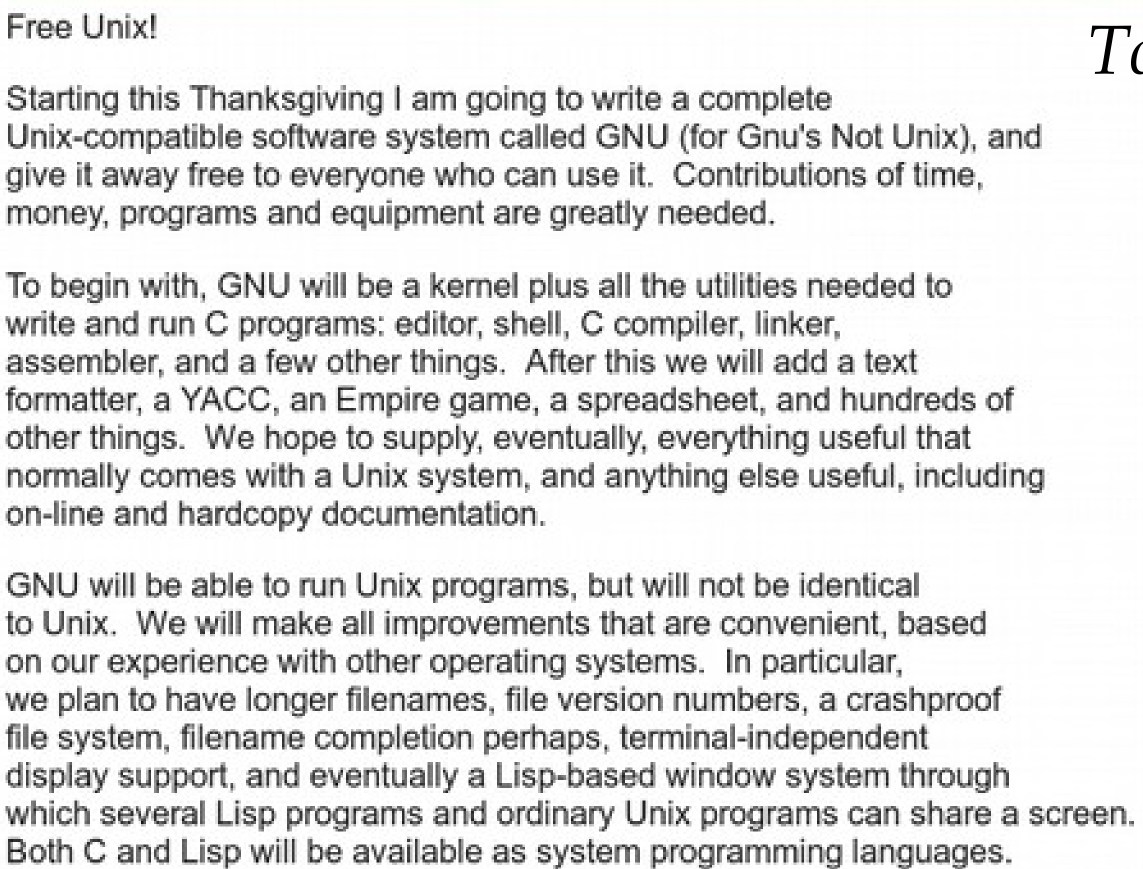
«When the AI Lab bought a new PDP-10 in 1982, its administrators decided to use Digital's nonfree timesharing system instead of ITS»

«The modern computers of the era, such as the VAX or the 68020, had their own operating systems, but none of them were free software: you had to sign a nondisclosure agreement even to get an executable copy»

"ci fu impedito di fare cose utili"= aggressione alla libertà

1983 *GNU project:* come Unix, ma completamente libero

27/09/83



Free Unix!

Starting this Thanksgiving I am going to write a complete Unix-compatible software system called GNU (for Gnu's Not Unix), and give it away free to everyone who can use it. Contributions of time, money, programs and equipment are greatly needed.

To begin with, GNU will be a kernel plus all the utilities needed to write and run C programs: editor, shell, C compiler, linker, assembler, and a few other things. After this we will add a text formatter, a YACC, an Empire game, a spreadsheet, and hundreds of other things. We hope to supply, eventually, everything useful that normally comes with a Unix system, and anything else useful, including on-line and hardcopy documentation.

GNU will be able to run Unix programs, but will not be identical to Unix. We will make all improvements that are convenient, based on our experience with other operating systems. In particular, we plan to have longer filenames, file version numbers, a crashproof file system, filename completion perhaps, terminal-independent display support, and eventually a Lisp-based window system through which several Lisp programs and ordinary Unix programs can share a screen. Both C and Lisp will be available as system programming languages.

Tools: editor (Emacs), compilatore (gcc), linker (ld), make, etc..

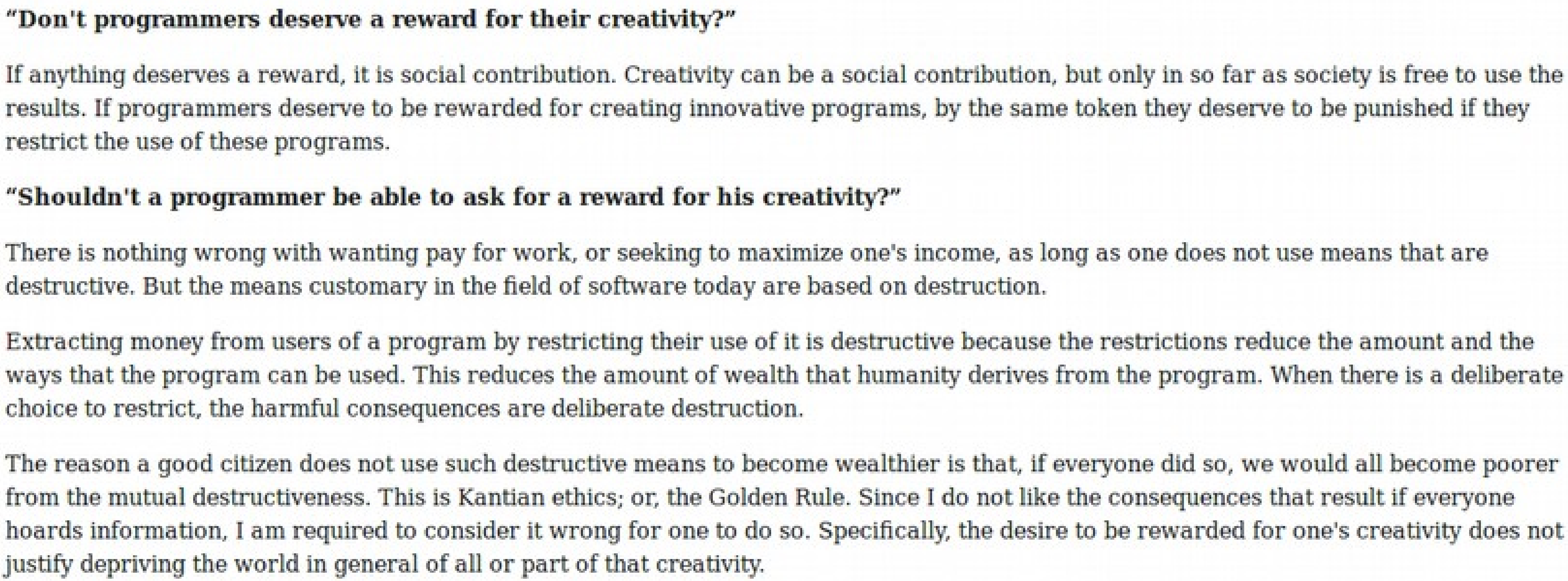
“Make the world a better place”

1985 Free Software Foundation (FSF)

GNU General Public License (GPL)

GNU Manifesto

https://www.gnu.org/gnu/manifesto.html



“Don't programmers deserve a reward for their creativity?”

If anything deserves a reward, it is social contribution. Creativity can be a social contribution, but only in so far as society is free to use the results. If programmers deserve to be rewarded for creating innovative programs, by the same token they deserve to be punished if they restrict the use of these programs.

“Shouldn't a programmer be able to ask for a reward for his creativity?”

There is nothing wrong with wanting pay for work, or seeking to maximize one's income, as long as one does not use means that are destructive. But the means customary in the field of software today are based on destruction.

Extracting money from users of a program by restricting their use of it is destructive because the restrictions reduce the amount and the ways that the program can be used. This reduces the amount of wealth that humanity derives from the program. When there is a deliberate choice to restrict, the harmful consequences are deliberate destruction.

The reason a good citizen does not use such destructive means to become wealthier is that, if everyone did so, we would all become poorer from the mutual destructiveness. This is Kantian ethics; or, the Golden Rule. Since I do not like the consequences that result if everyone hoards information, I am required to consider it wrong for one to do so. Specifically, the desire to be rewarded for one's creativity does not justify depriving the world in general of all or part of that creativity.

GNU e Le 4 libertà

0 Libertà di eseguire il programma, per qualsiasi scopo.

1 Libertà di studiare come funziona il programma, e adattarlo *alle proprie necessità*.

2 Libertà di ridistribuire le copie *in modo da aiutare il prossimo*.

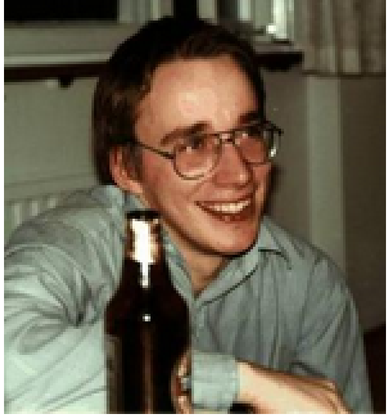
3 Libertà di migliorare il programma, e distribuirne pubblicamente i miglioramenti, *in modo tale che tutta la comunità ne tragga beneficio*.

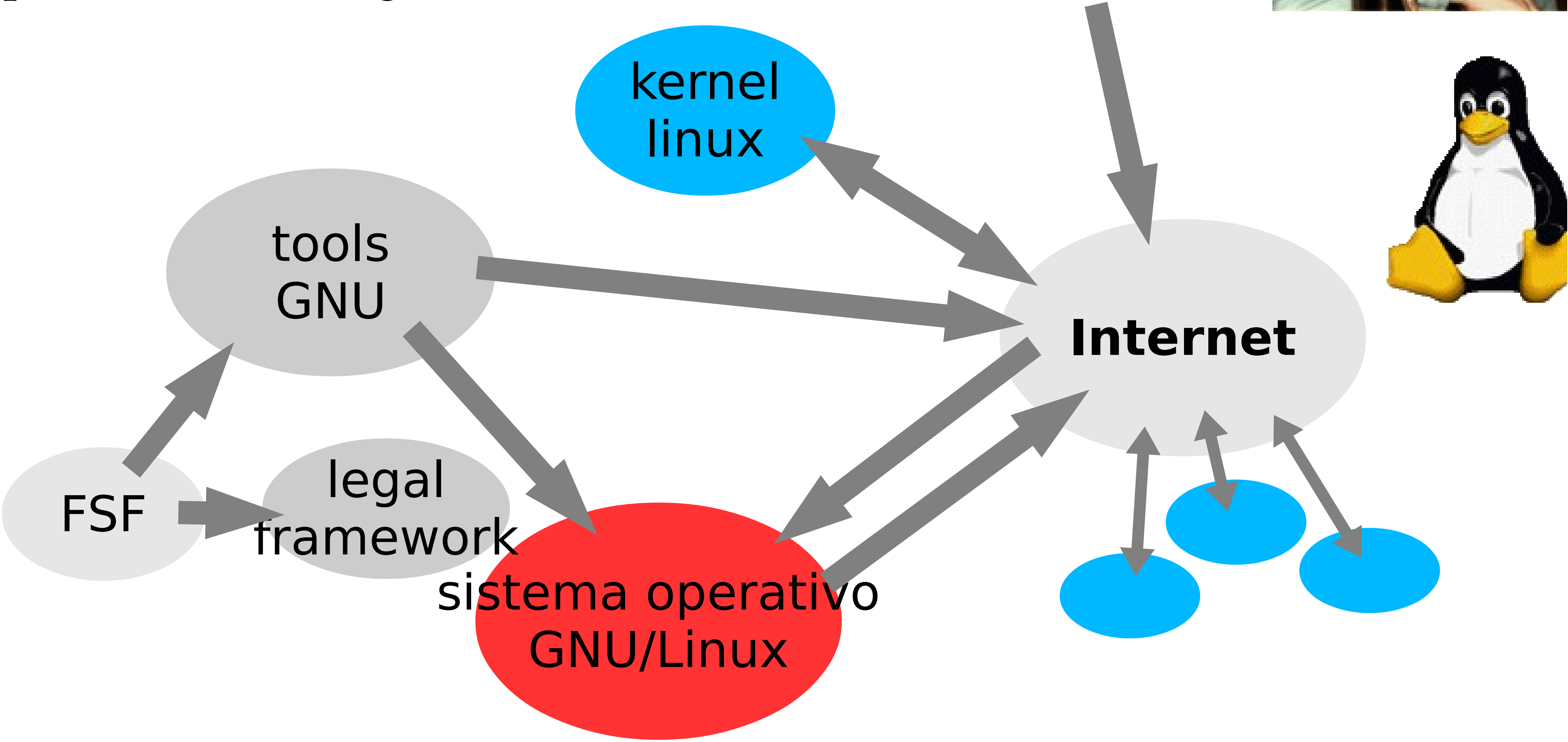
Richard M.Stallman, Cambridge MA, 1985

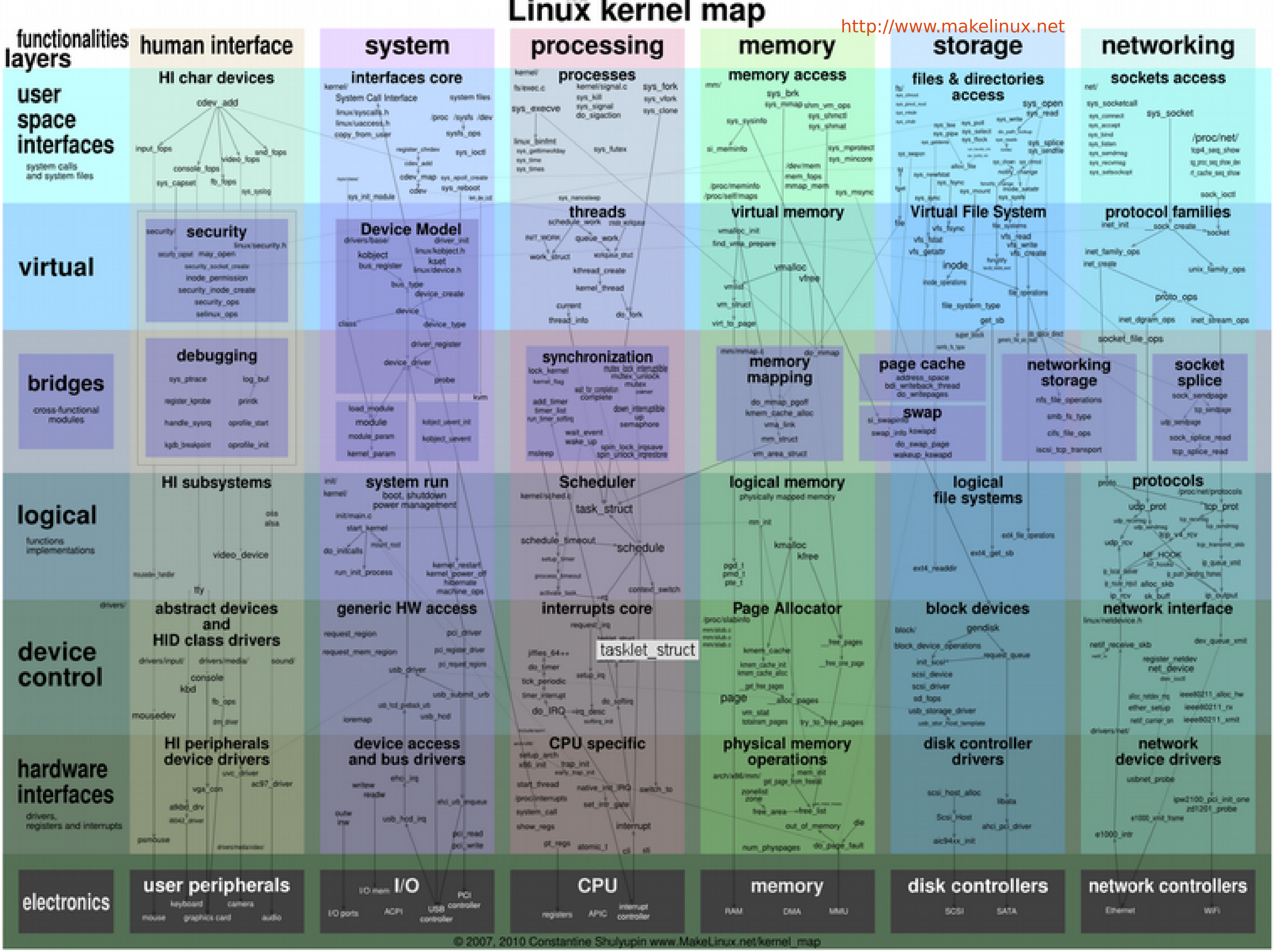
1990 *Kernel,* il nucleo: GNU Hurd: è molto avanzato e più difficile del previsto: blocca il progetto

Linus Torvalds, Helsinki University, Finlandia, 25 Ago 1991

"Hello everybody out there using minix - I'm doing a (free) operating system (just a hobby, won't be big and professional like gnu) for 386(486) AT clones..."







Constantine Shulyupin

1989 Cygnus Software, Michael Tyman: prima impresa di assistenza su free software

Il successo di Linux e degli altri programmi liberi attira le imprese.

1993 RedHat

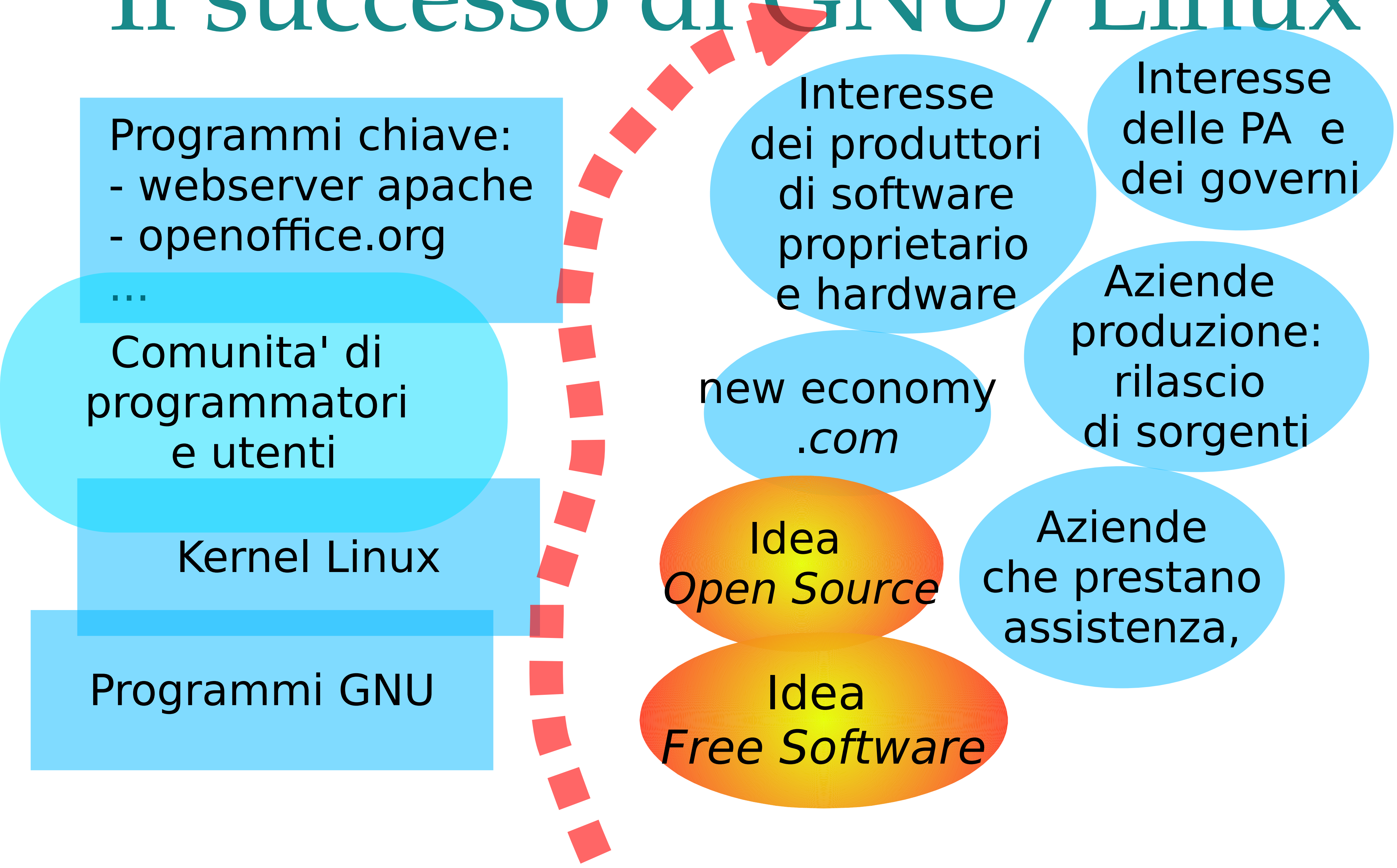
1994 SUSE Linux 1.0

Free suona *gratis* : tabù per il business
→ *“Open Source”* è il *maketing term per “free software”.*

1998 Netscape, sotto la pressione della concorrenza Microsoft, decide di rilasciare il sorgente del browser Mozilla. → Open Source Initiative (Raymond, Perens).

2000 Sun rilascia OpenOffice.org, 2006 Sun rilascia Java, 2007 Google rilascia Android, 2010 Sun comprata da Oracle, 2011 fork LibreOffice

Il successo di GNU/Linux



Programmi chiave:
- webserver apache
- openoffice.org

...

Comunita' di programmatori e utenti

Kernel Linux

Programmi GNU

Interesse dei produttori di software proprietario e hardware

Interesse delle PA e dei governi

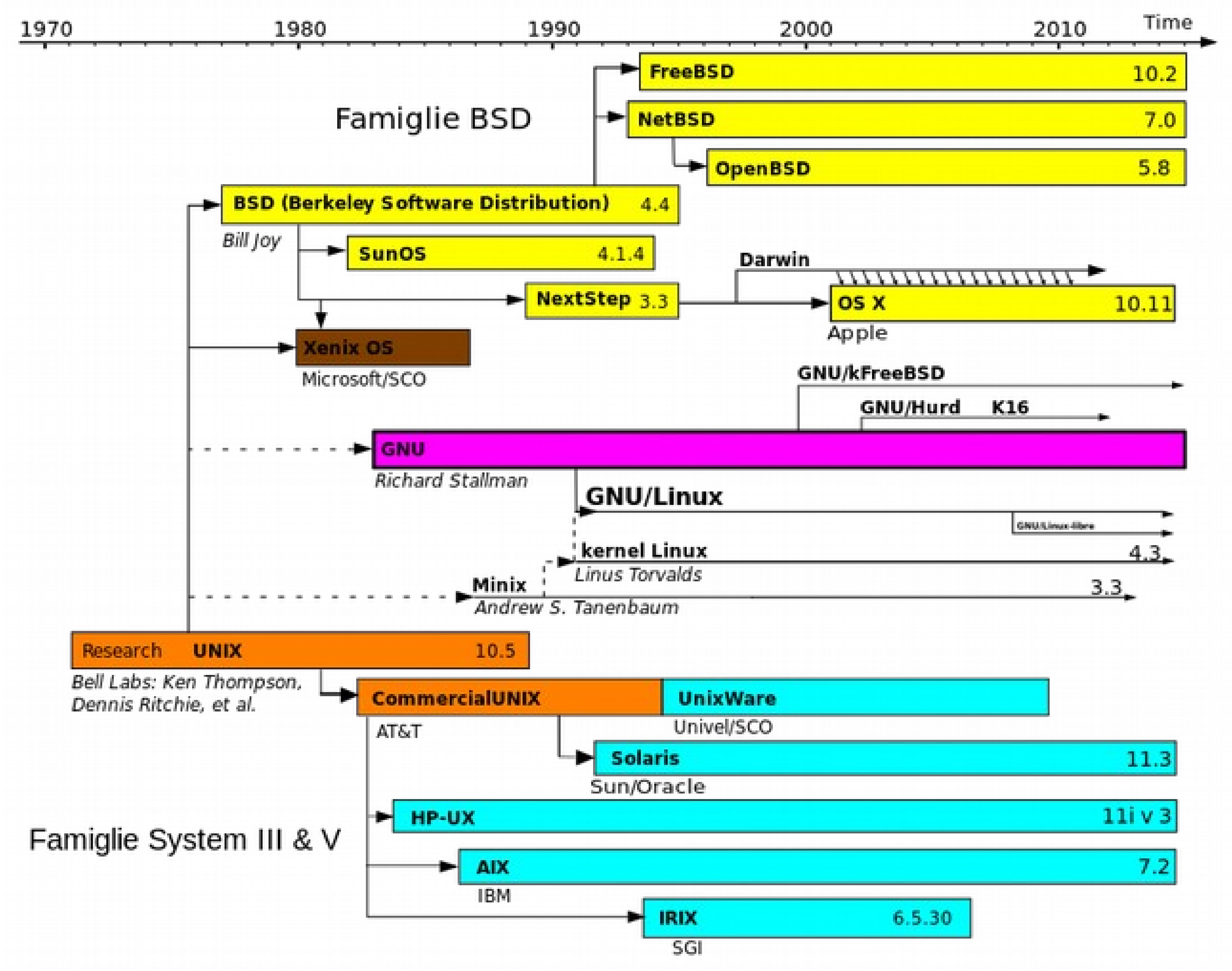
new economy *.com*

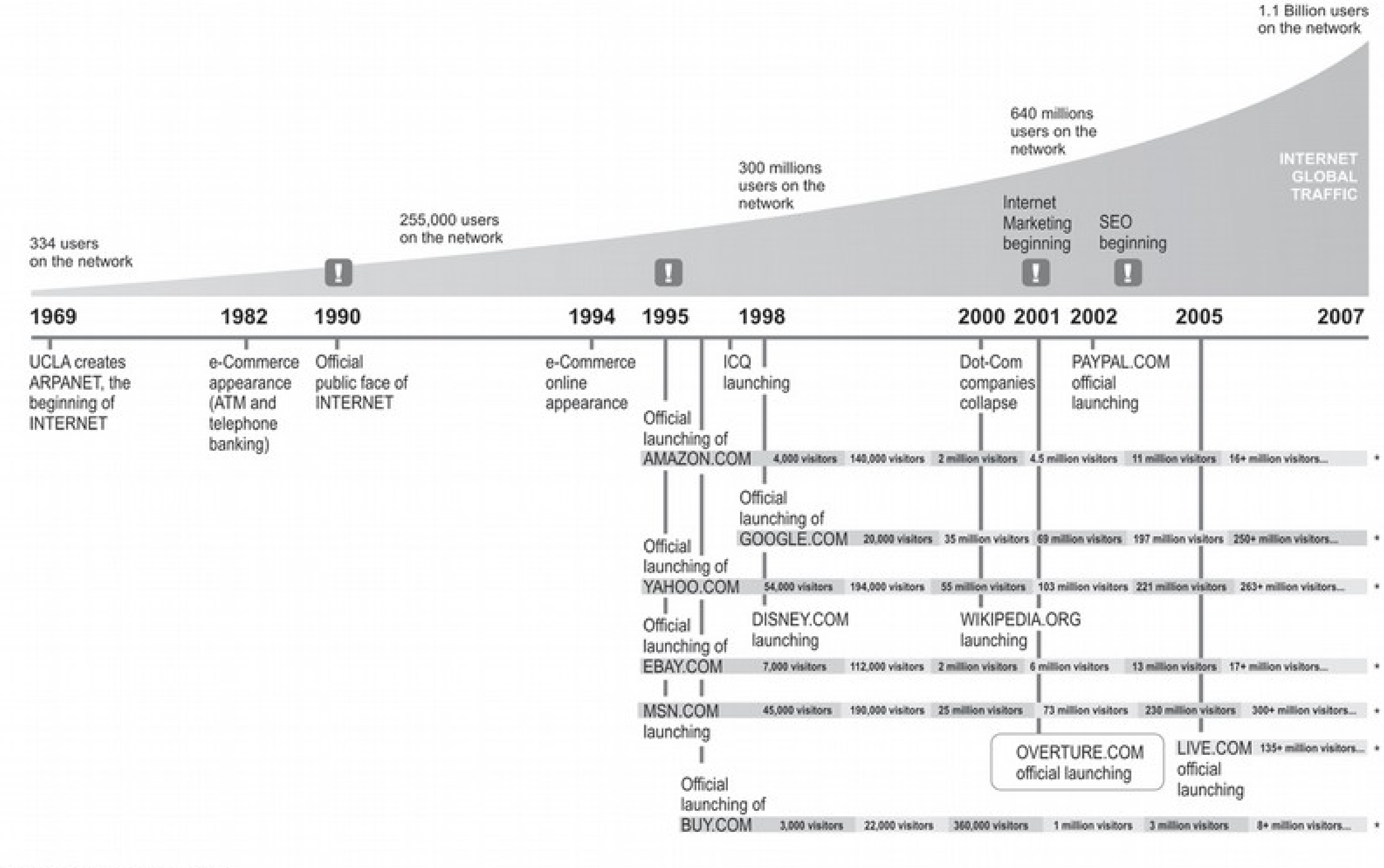
Aziende produzione: rilascio di sorgenti

Idea *Open Source*

Aziende che prestano assistenza,

Idea *Free Software*


1970
1980
1990
2000
2010
Time
Famiglie BSD
FreeBSD 10.2
NetBSD 7.0
OpenBSD 5.8
BSD (Berkeley Software Distribution) 4.4
Bill Joy
SunOS 4.1.4
Darwin
NextStep 3.3
OS X 10.11
Apple
Xenix OS
Microsoft/SCO
GNU/kFreeBSD
GNU/Hurd K16
GNU
Richard Stallman
GNU/Linux
GNU/Linux-libre
kernel Linux 4.3
Linus Torvalds
Minix 3.3
Andrew S. Tanenbaum
Research UNIX 10.5
Bell Labs: Ken Thompson, Dennis Ritchie, et al.
CommercialUNIX
AT&T
UnixWare
Univel/SCO
Solaris 11.3
Sun/Oracle
HP-UX 11i v 3
AIX 7.2
IBM
IRIX 6.5.30
SGI
Famiglie System III & V



* User traffic calculation per day

http://www.whatgoddoes.com/wp-content/uploads/2012/09/internet_timeline.jpg

II - stato

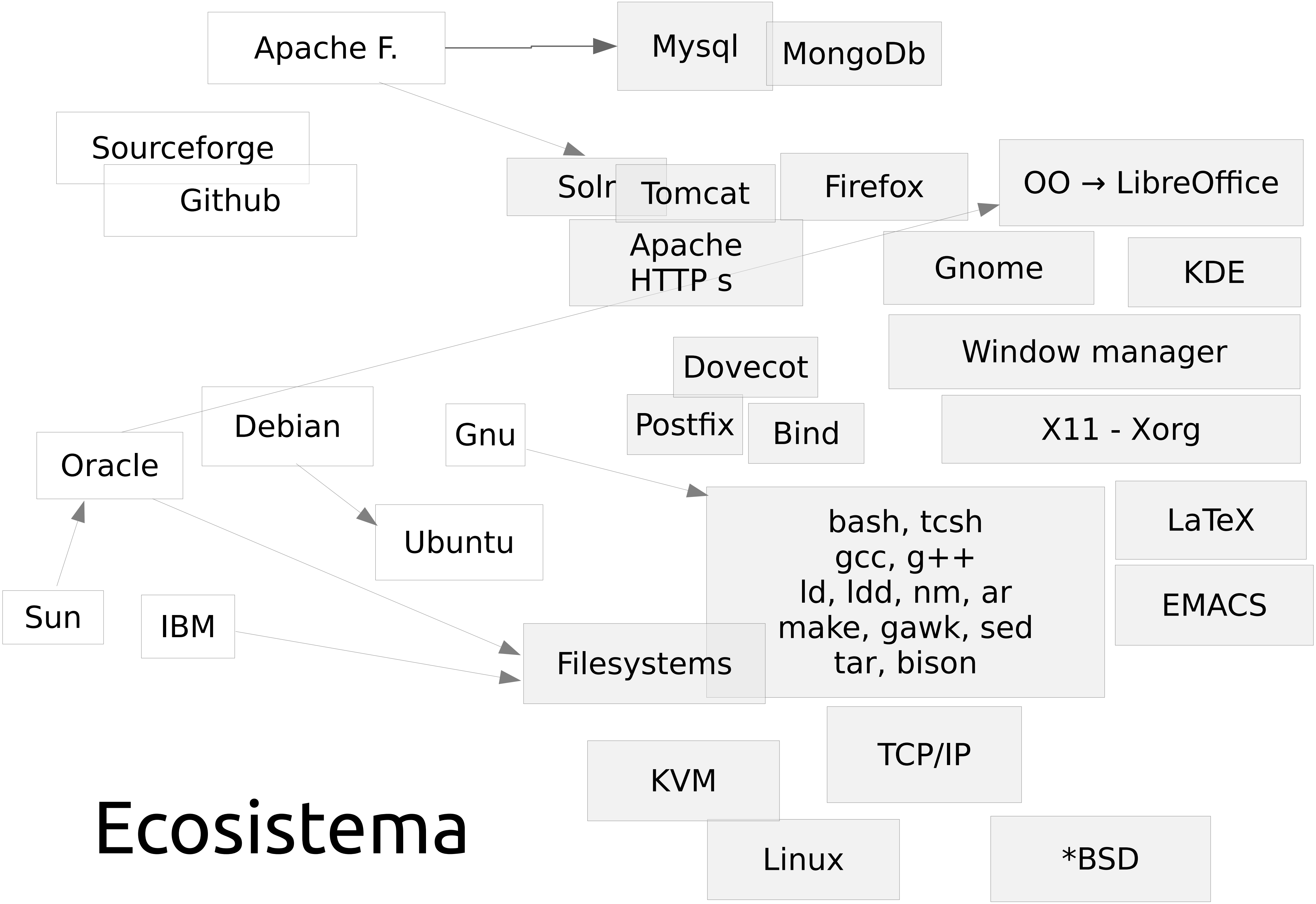
1 - ecosistema f/oss

2 - il mercato del software, regolazione

3 – f/oss e modelli di {

- licenza
- sviluppo
- distribuzione
- (prezzo)

Ecosistema



Apache F.

Mysql

MongoDb

Sourceforge

Github

Solr

Tomcat

Firefox

OO → LibreOffice

Apache HTTP s

Gnome

KDE

Window manager

Dovecot

Postfix

Bind

X11 - Xorg

Debian

Gnu

Oracle

Ubuntu

bash, tcsh
gcc, g++
ld, ldd, nm, ar
make, gawk, sed
tar, bison

LaTeX

EMACS

Sun

IBM

Filesystems

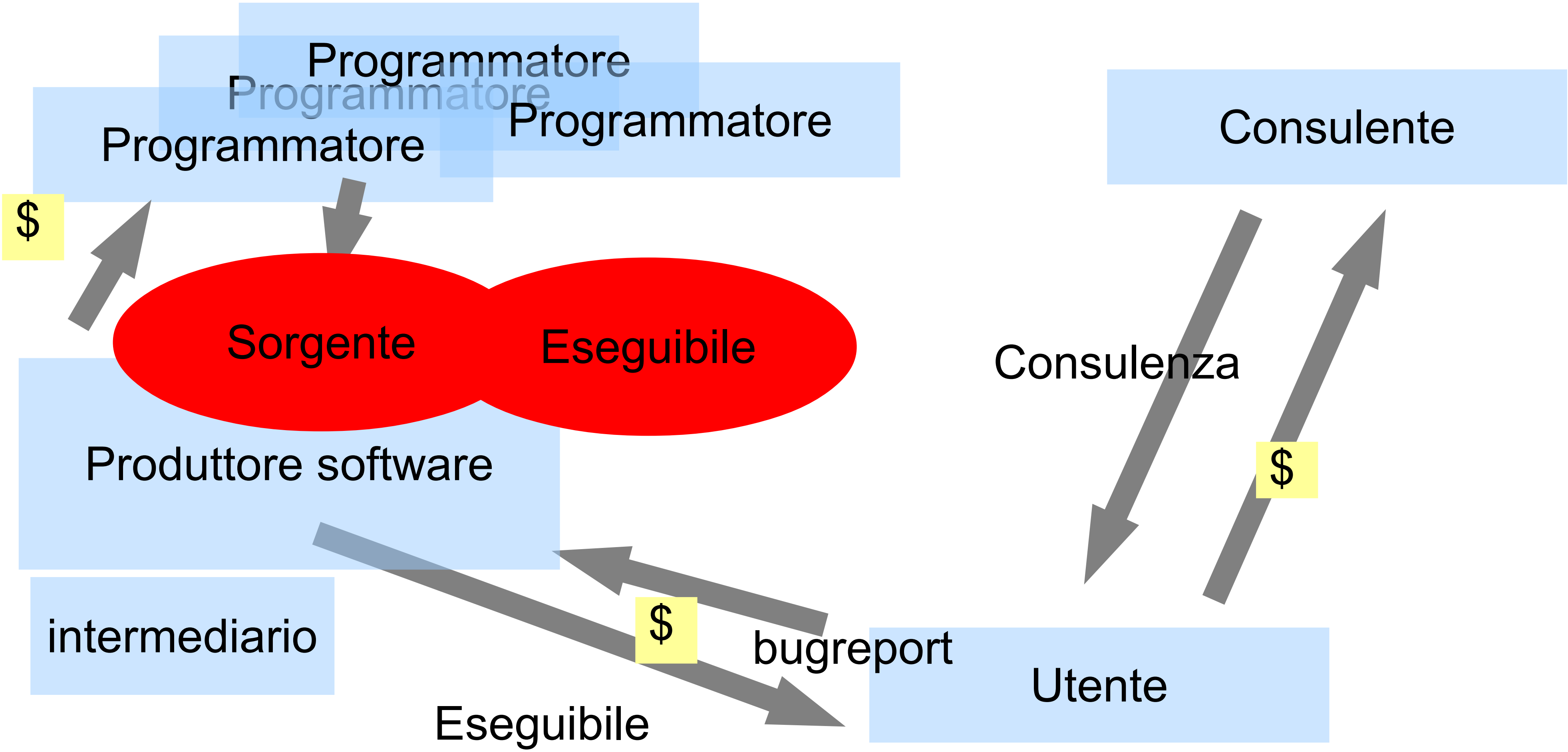
TCP/IP

KVM

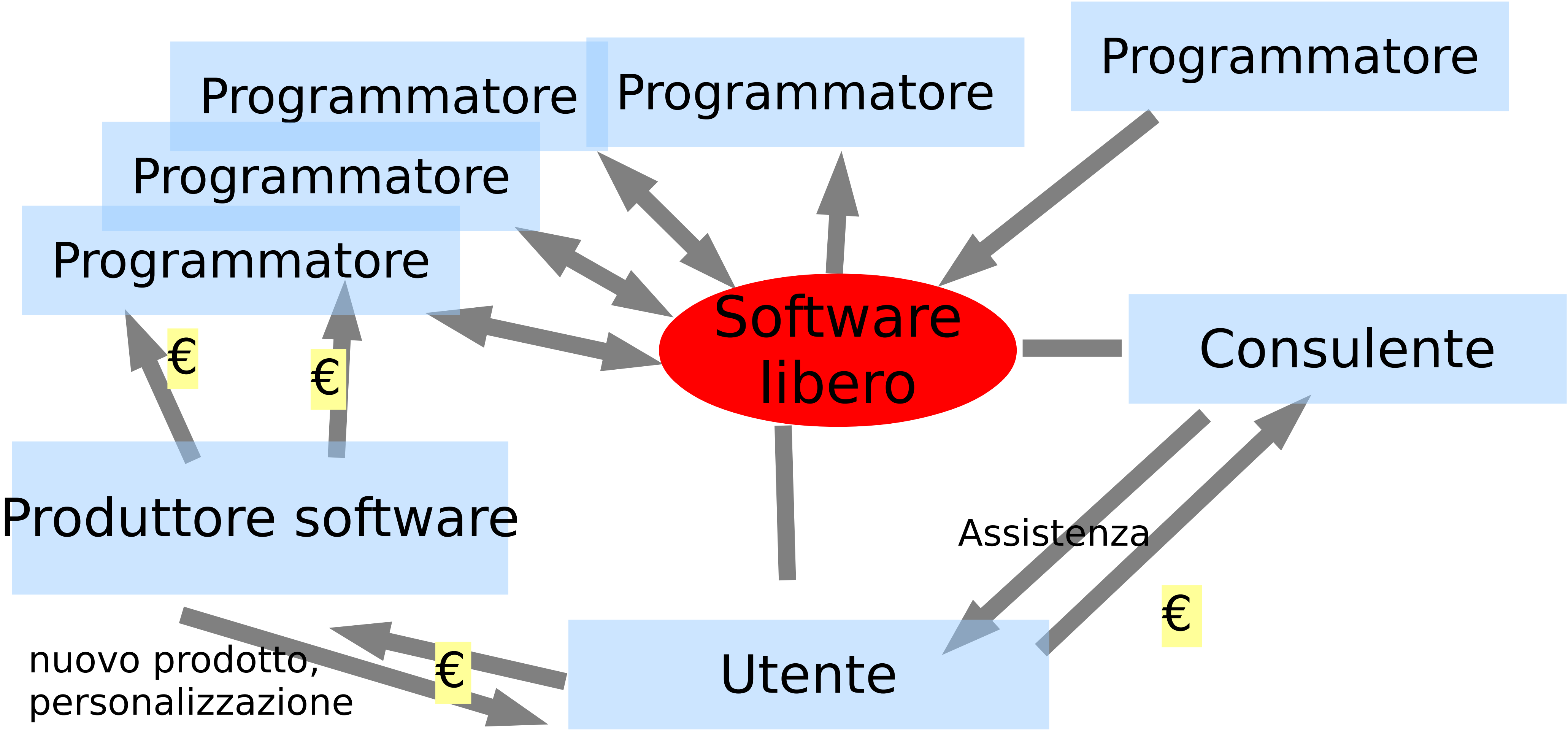
Linux

*BSD

software proprietario



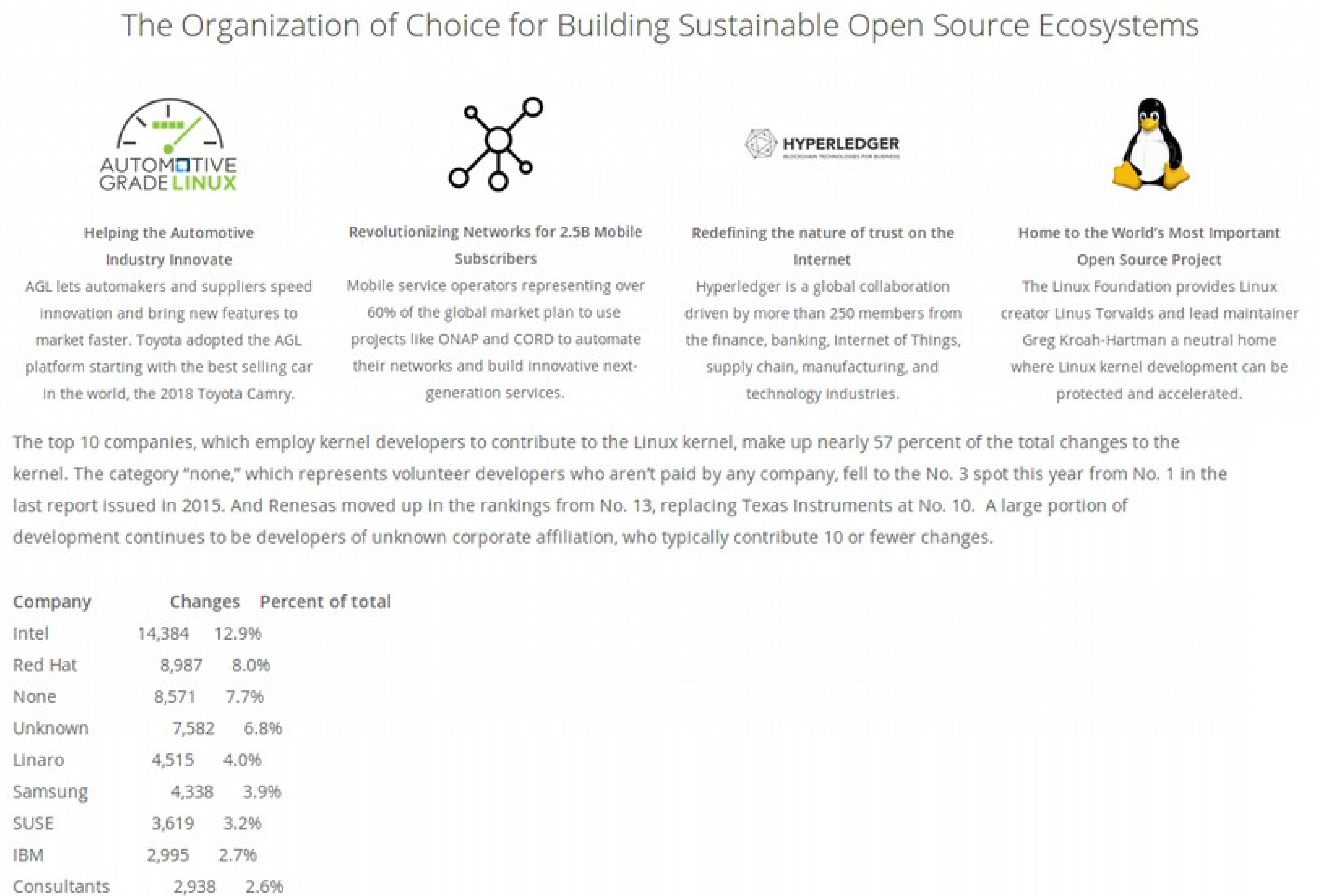
software libero



Perchè le imprese sviluppano F/OSS?

- acquisire la conoscenza necessaria per competere
- influenzare lo sviluppo di prodotti strategici: imporre standard, e poi il *backport* é più costoso
- ridurre rischi: “in proprietary software, it can be too late to back up when you make a mistake”
- Attingere alla creatività della comunità: exploration/exploitation
- SaaS: non vendere il sw, vendere i servizi!

THE LINUX FOUNDATION



The Organization of Choice for Building Sustainable Open Source Ecosystems

AUTOMOTIVE GRADE LINUX

Helping the Automotive Industry Innovate
AGL lets automakers and suppliers speed innovation and bring new features to market faster. Toyota adopted the AGL platform starting with the best selling car in the world, the 2018 Toyota Camry.

Revolutionizing Networks for 2.5B Mobile Subscribers
Mobile service operators representing over 60% of the global market plan to use projects like ONAP and CORD to automate their networks and build innovative next-generation services.

HYPERLEDGER

Redefining the nature of trust on the Internet
Hyperledger is a global collaboration driven by more than 250 members from the finance, banking, Internet of Things, supply chain, manufacturing, and technology industries.

Home to the World's Most Important Open Source Project
The Linux Foundation provides Linux creator Linus Torvalds and lead maintainer Greg Kroah-Hartman a neutral home where Linux kernel development can be protected and accelerated.

The top 10 companies, which employ kernel developers to contribute to the Linux kernel, make up nearly 57 percent of the total changes to the kernel. The category "none," which represents volunteer developers who aren't paid by any company, fell to the No. 3 spot this year from No. 1 in the last report issued in 2015. And Renesas moved up in the rankings from No. 13, replacing Texas Instruments at No. 10. A large portion of development continues to be developers of unknown corporate affiliation, who typically contribute 10 or fewer changes.

Company	Changes	Percent of total
Intel	14,384	12.9%
Red Hat	8,987	8.0%
None	8,571	7.7%
Unknown	7,582	6.8%
Linaro	4,515	4.0%
Samsung	4,338	3.9%
SUSE	3,619	3.2%
IBM	2,995	2.7%
Consultants	2,938	2.6%

modelli di licenza

impresa

diritti di sfruttamento economico

autore

diritti morali
+ economici

licenza

utente

Licenze sw proprietario

concedono:

- esecuzione del codice binario

  (a certe condizioni)

vietano:

- copia, modifica, diffusione
- *reverse engineering*

Licenze sw libero

concedono:

- esecuzione del codice binario, senza condizioni
- modifica, diffusione, del codice sorgente

vietano:

...dipende...

Famiglie di licenze libere

	Reversibile?
1 Public Domain viene ceduto tutto, anche il ©	SI
2 *tipo* BSD (Berkeley Standard Distribution): deve sempre rimanere il © dell'autore → reversibile (può essere reso proprietario)	SI
3 GNU GPL (General Public License): “copyleft” o permesso d'autore: → opere derivate devono usare GPL	NO

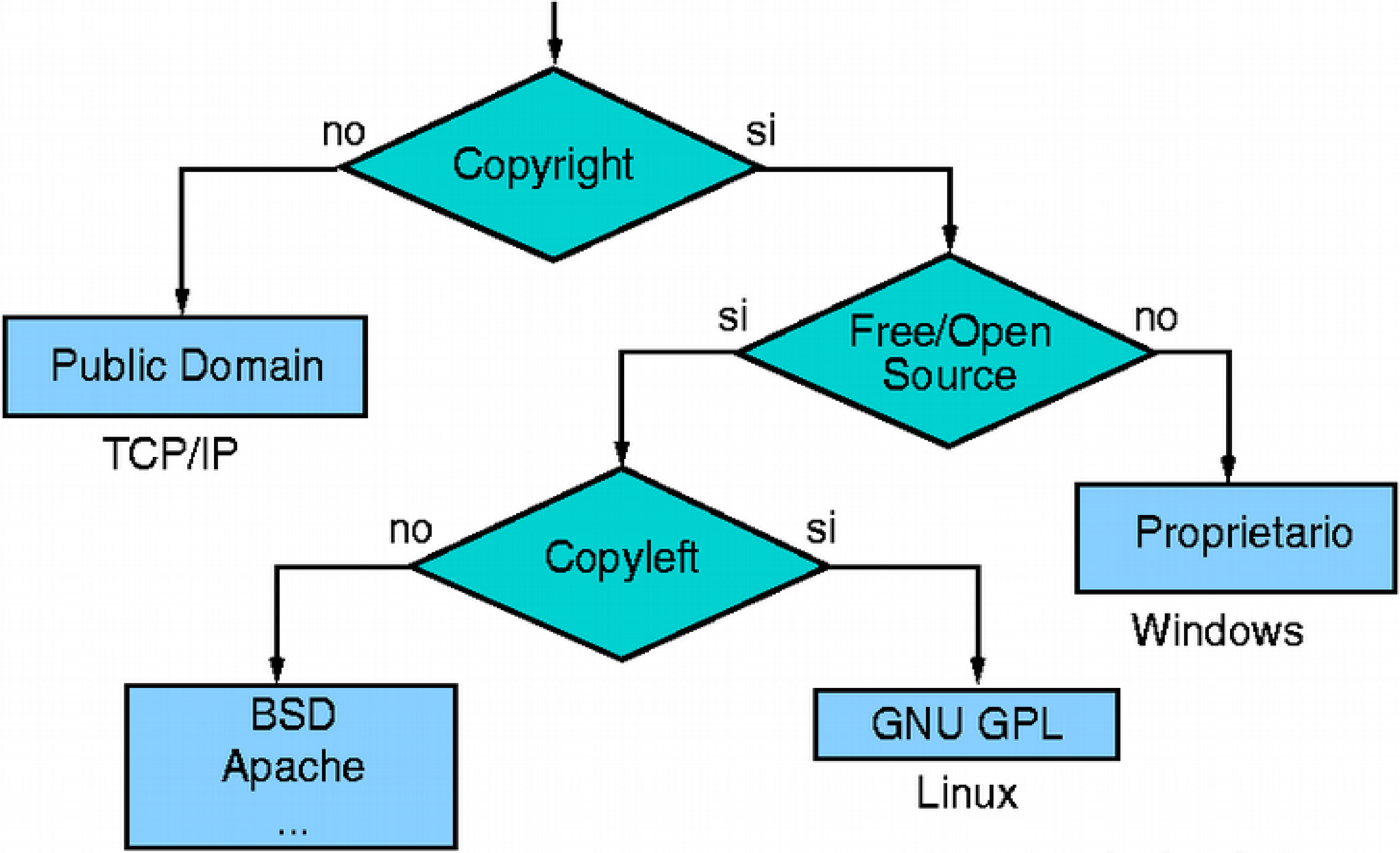
gnu
general public license

Copyleft :

Chi distribuisce copie di un programma coperto da GPL, sia gratis sia in cambio di un compenso, deve concedere ai destinatari tutti i diritti che ha ricevuto.

Deve anche assicurarsi che i destinatari ricevano o possano ottenere il codice sorgente.

E deve mostrar loro le condizioni di licenza, in modo che essi conoscano i propri diritti.



da: Lawrence Lessig: Open Source Baselines, 2002

Modelli di sviluppo

Cattedrale/Bazaar (Raymond, 1999)

Single Guru
Master – Disciple → *Benevolent dictator*
Project team

self-identification (Benkler, 2006)

Modelli di distribuzione del software

~~Tradizionale~~

~~supporto fisico + distributore/rivenditore~~

Internet based

download, “try & buy”, demo/premium

f/oss: distribuzione integrata con sviluppo e supporto: Github, SourceForge

prezzo

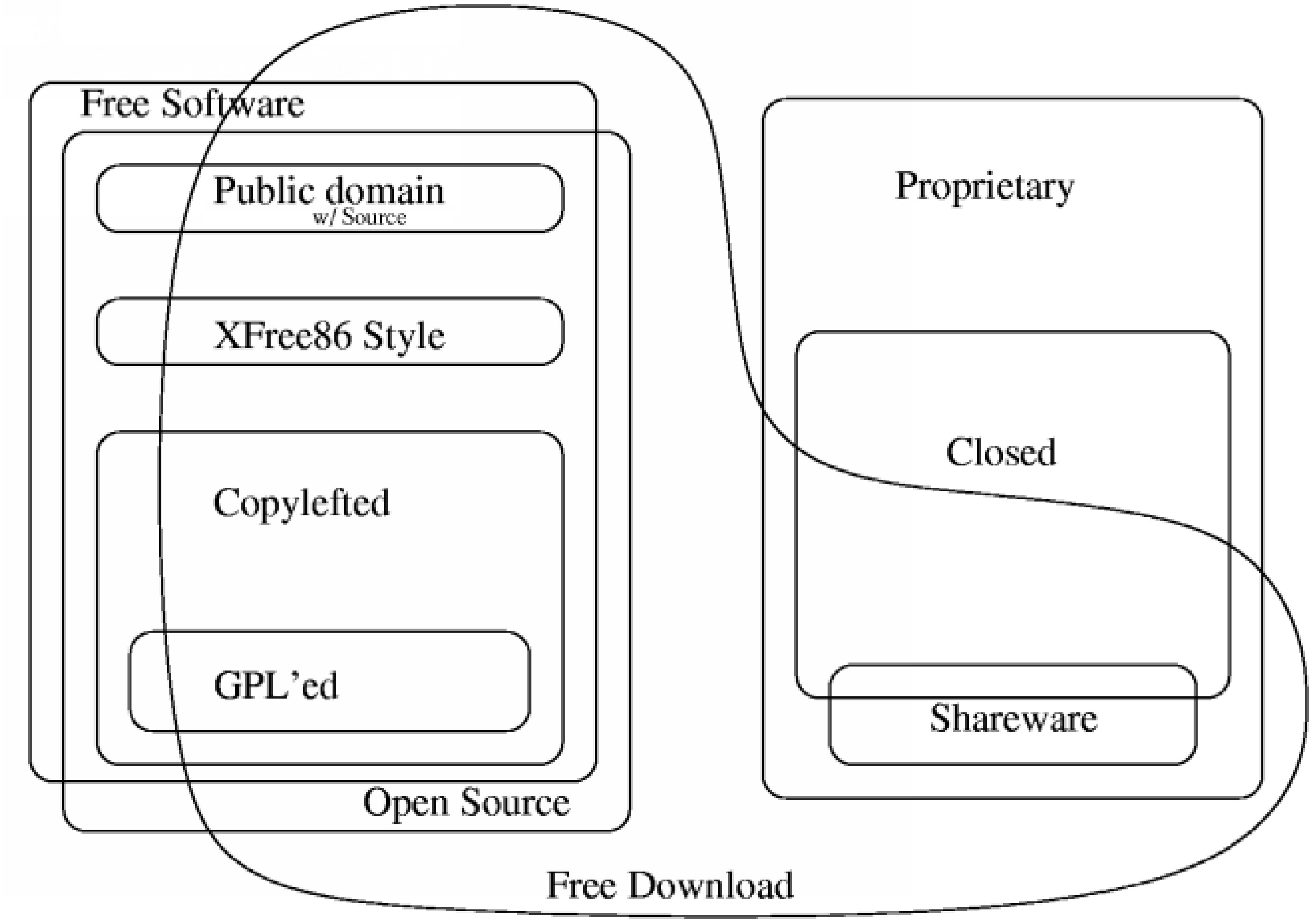
Per essere *free* software
deve essere coperto da licenza libera,

inoltre

può essere gratis
(e spesso lo è)

≠ free software:
freeware (sw proprietario gratuito)

trialware, shareware, demo (proprietario,
scaricabile, a tempo/funz. ridotta)



http://www.gnu.org/philosophy/categories.html codice sorgente

+licenze libere = *free software*

+Internet = *peer-production* e distribuzione

+imprese = *Open Source*

---

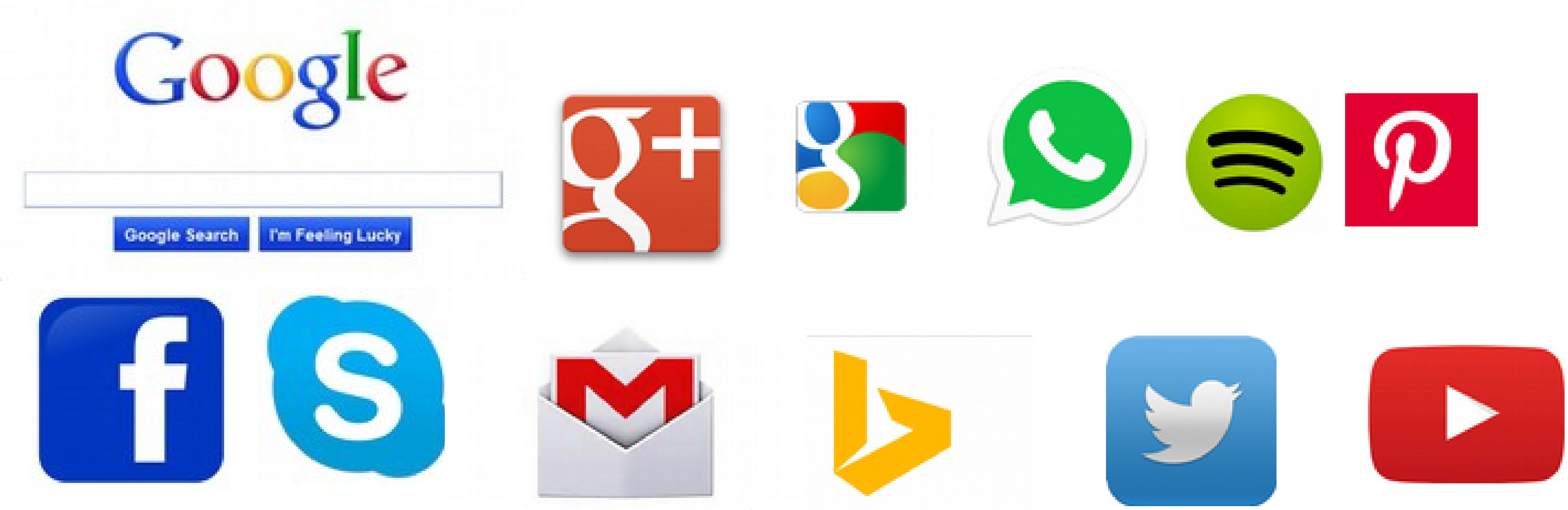
free software prodotto, usato e distribuito grazie ad Internet da privati e imprese

III - prospettive

- Desktop: 88% MS Windows
- Desktop (48%), ↘ Mobile (45%) ↗
- GNU/Linux è lo standard per le **infrastrutture:** cloud, embedded (IoT) e mobile
- Browser war: Chrome 60%, Firefox 12%, IE 12%
- Mobile (bn users):
  Android (2.6), Windows (1.4), iOS/macOS 1.3

https://www.statista.com/statistics/385001/smartphone-worldwide-installed-base-operating-systems/
https://www.netmarketshare.com/operating-system-market-share.aspx
https://www.netmarketshare.com/browser-market-share.aspx

WE WON!



?

0 Libertà di eseguire il programma, per qualsiasi scopo.

?

1 Libertà di studiare come funziona il programma, e adattarlo *alle proprie necessità.*

?

2 Libertà di ridistribuire le copie *in modo da aiutare il prossimo.*

?

3 Libertà di migliorare il programma, e distribuirne pubblicamente i miglioramenti, *in modo tale che tutta la comunità ne tragga beneficio.*

?

Cloud

IBM 2009 - NERCS @ Berkeley Magellan Lawrence Nat'l Lab - Roy Kaltschmidt, photographer

New Tab
Search or enter address
Search the Web

KOTTKE

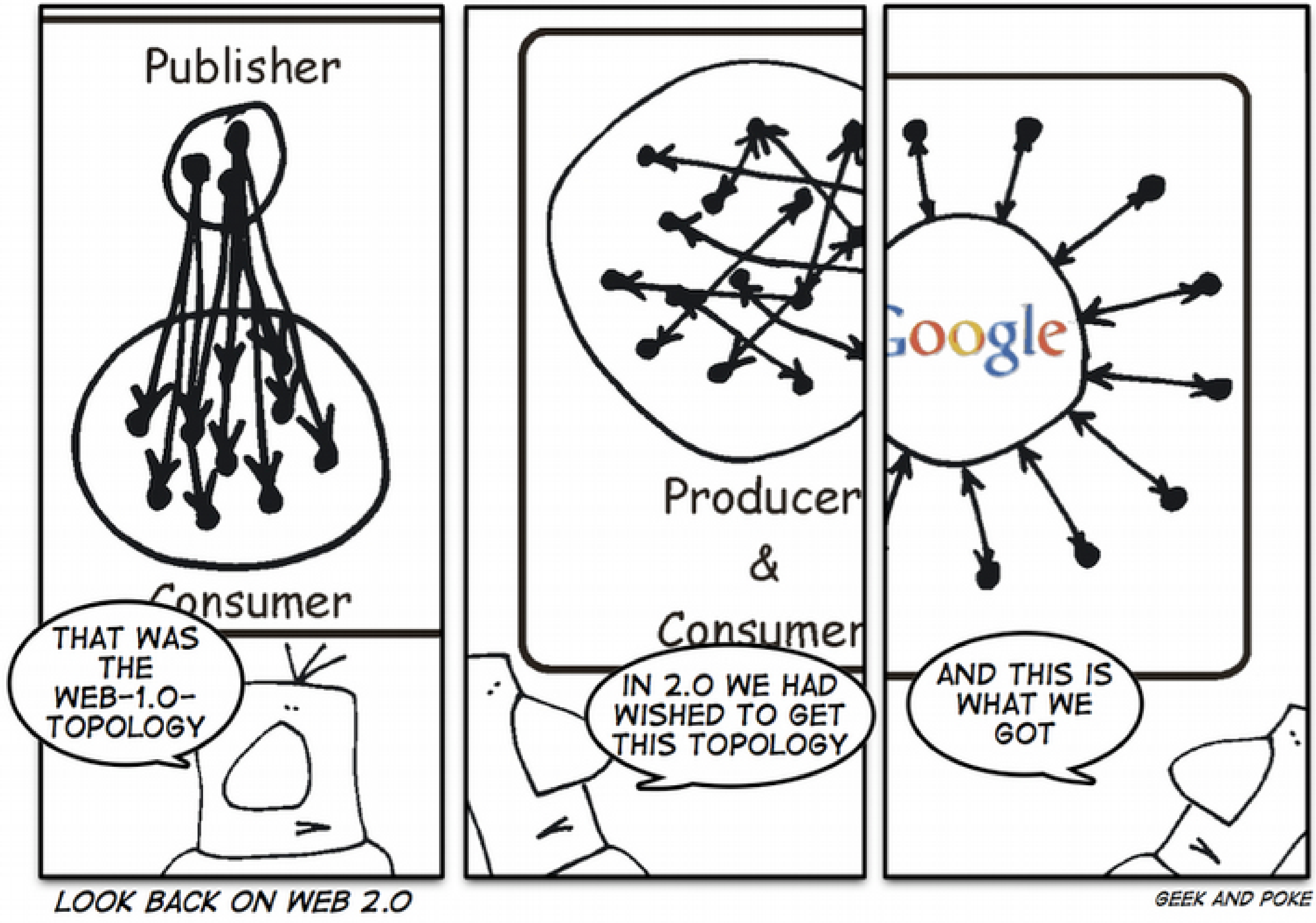
The secret rules of the internet

It's just the beginning of December and the lists of the best books ...

EATER

Maybe Just Don't Drink Coffee

Do not adjust your sets: the parking lot has emptied and it's ready ...



http://geekandpoke.typepad.com/geekandpoke/2006/10/look_back_on_we.html October 21, 2006

Free What?

Infrastructure

open interoperable standards

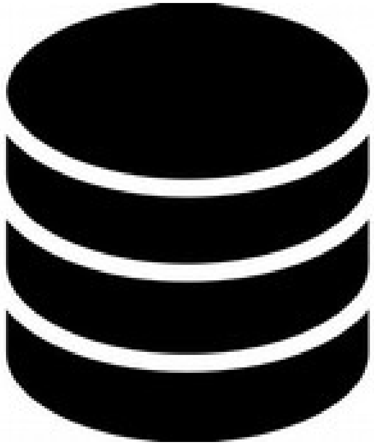
F/OSS O.S., browsers

Application

free/open source licenses

non-centralizzation

device neutrality



Data

data ownership

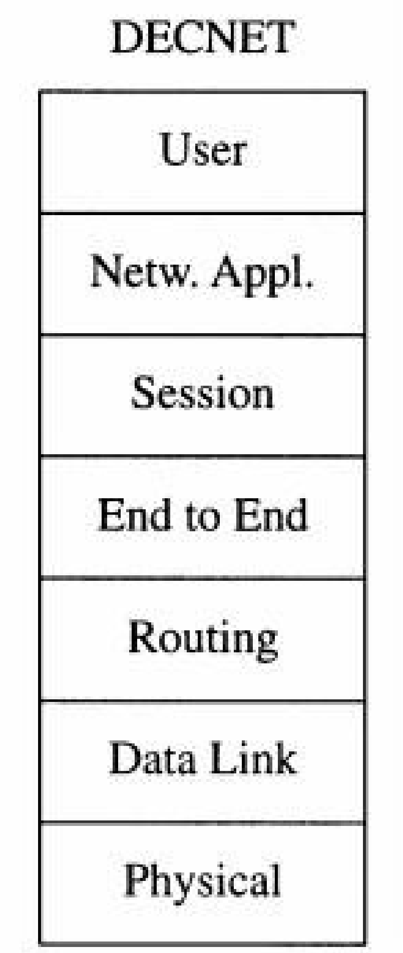
privacy by {design,default}

A/ Infrastructure

digital

IBM

1975 →



DECNET

User

Netw. Appl.

Session

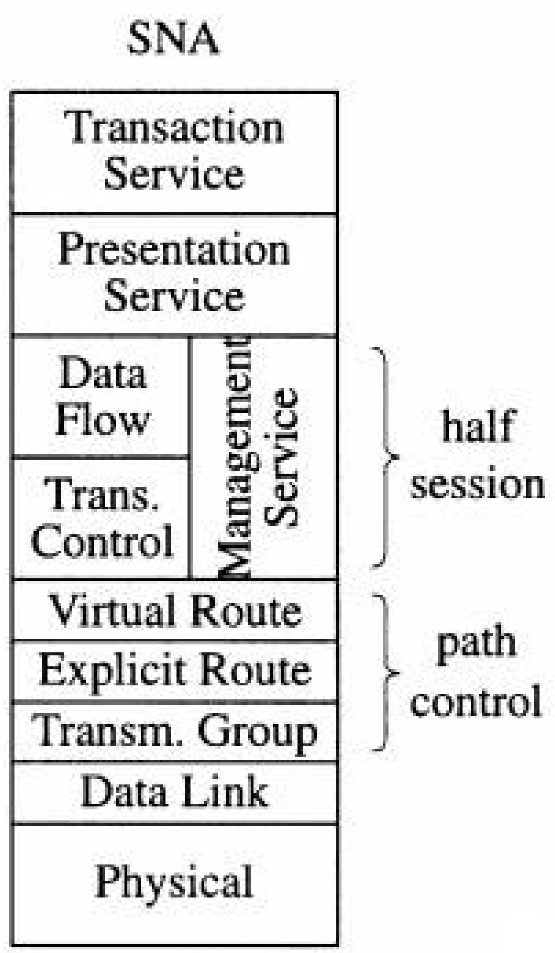
End to End

Routing

Data Link

Physical

1974 →



SNA

Transaction Service

Presentation Service

Data Flow

Trans. Control

Management Service

half session

Virtual Route

Explicit Route

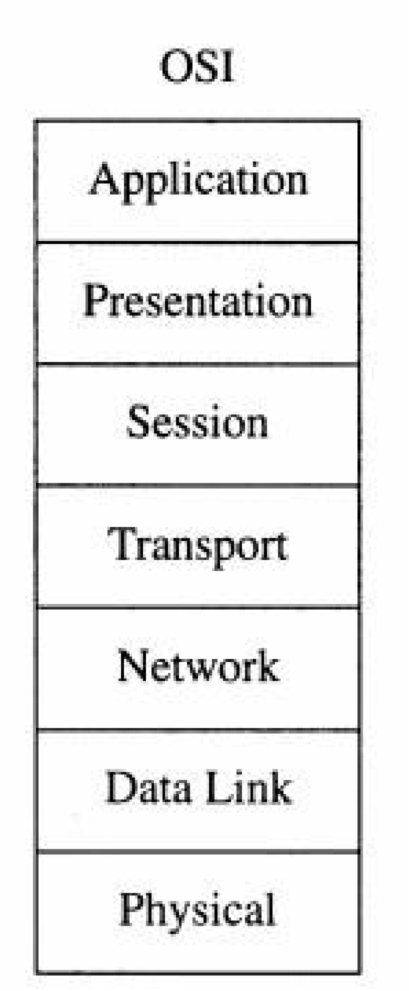
Transm. Group

path control

Data Link

Physical

A/ Infrastructure



OSI

Application

Presentation

Session

Transport

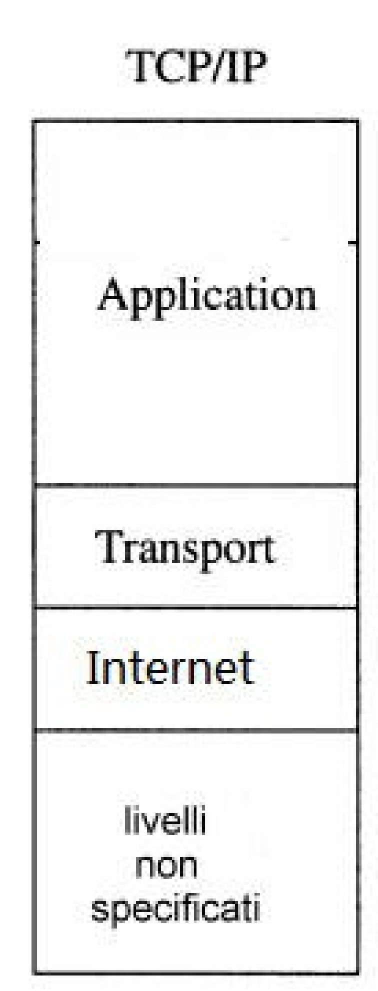
Network

Data Link

Physical

1984

A/ Infrastructure



TCP/IP

Application

Transport

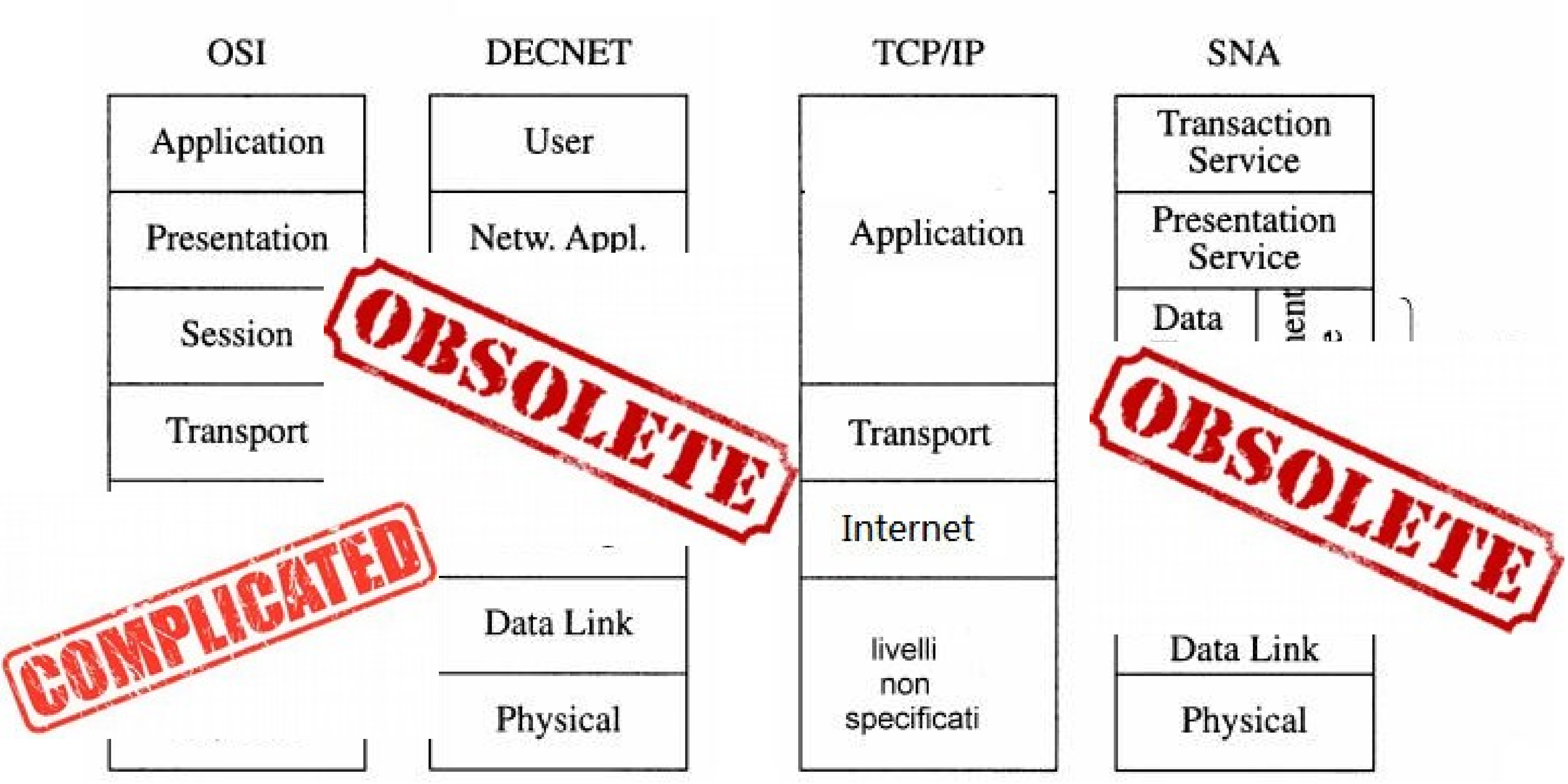
Internet

livelli
non
specificati

A/ Infrastructure

digital

IBM



OSI

Application

Presentation

Session

Transport

DECNET

User

Netw. Appl.

Data Link

Physical

TCP/IP

Application

Transport

Internet

livelli non specificati

SNA

Transaction Service

Presentation Service

Data

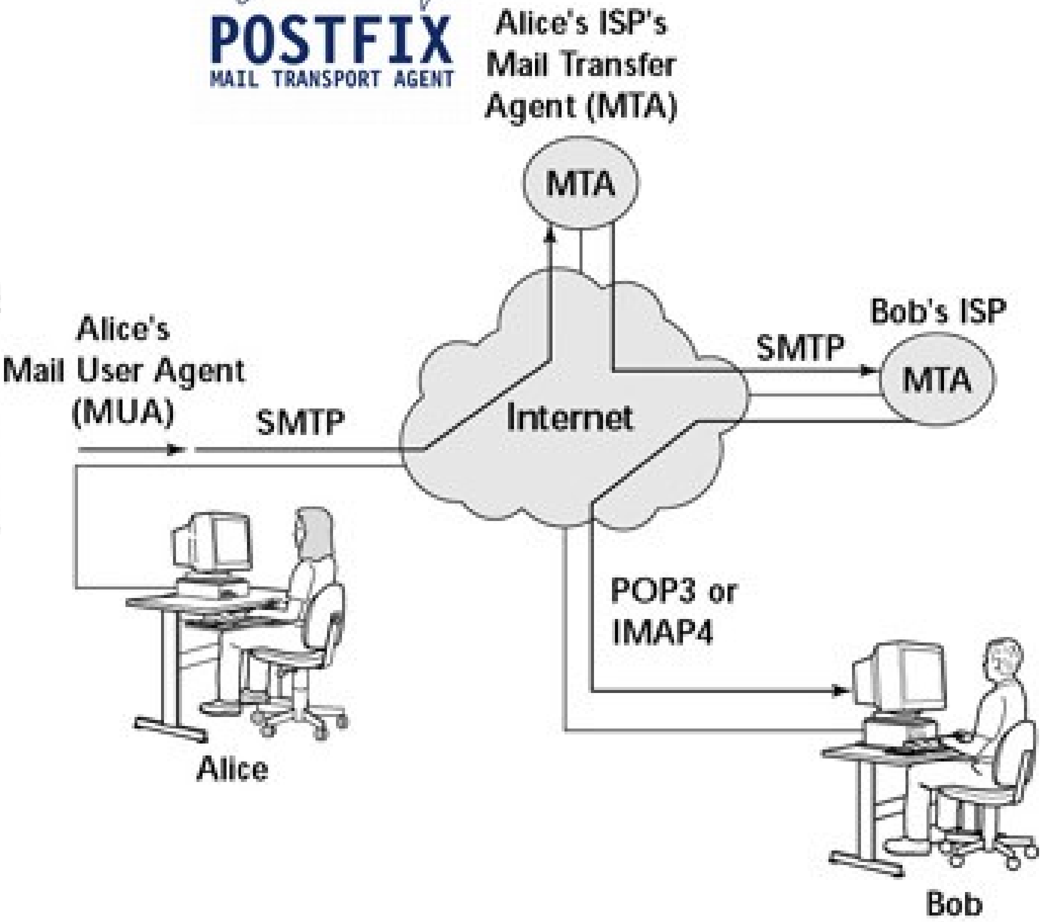
Data Link

Physical

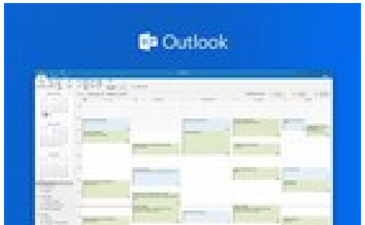
1974→

Es. Email: IP/TCP/SMTP/POP3/IMAP

POSTFIX
MAIL TRANSPORT AGENT

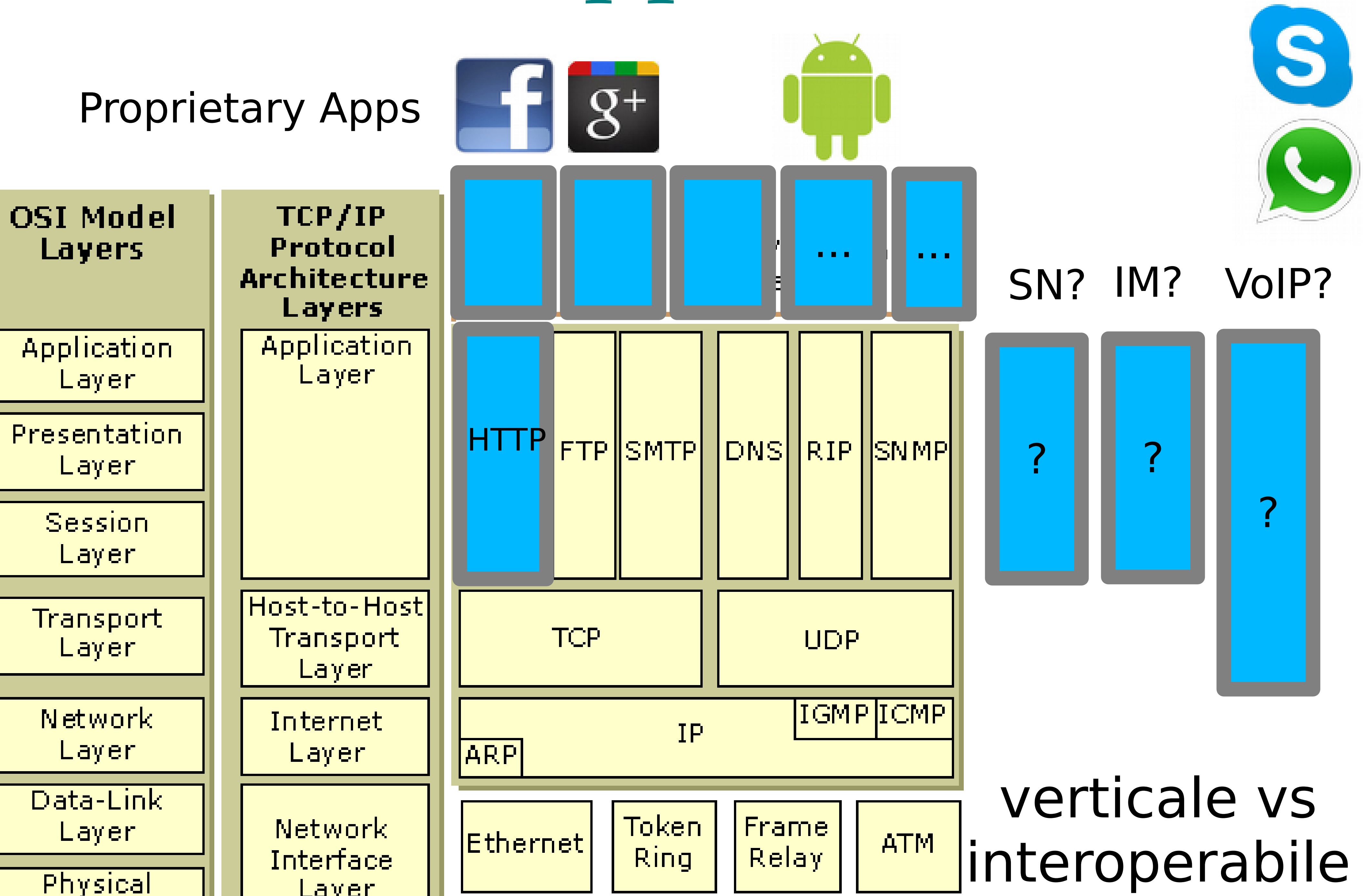


Exchange



B/ Application

Proprietary Apps



verticale vs interoperabile

B/ Application

https://ipfs.io/
La centralizzazione è un problema da risolvere.
→ dati accessibili in base al contenuto, non alla posizione

“The web's centralization limits opportunity
The Internet has been one of the great equalizers in human history and a real accelerator of innovation. But the increasing consolidation of control is a threat to that.
IPFS remains true to the original vision of the open and flat web, but delivers the technology which makes that vision a reality. “

B/ Application

Device Neutrality

Device neutrality **law** states that the users have the right of non-discrimination of the services and apps they use, based on platform control by hardware companies

(Wikipedia)

C/ Data

1) ~~Surveillance~~ Advertising

2) Lock-in

3) Ownership

“Data is the new Oil”

“Data is just like crude. It’s valuable, but if unrefined it cannot really be used. It has to be changed into gas, plastic, chemicals, etc to create a valuable entity that drives profitable activity; so must data be broken down, analyzed for it to have value.

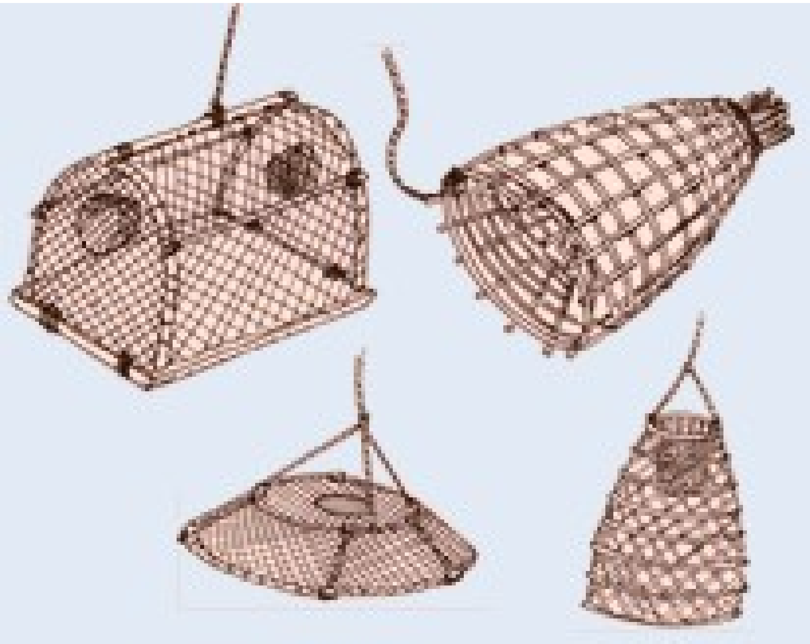
The issue is how do *we marketers* deal with the massive amounts of data that are available to us? How can we change this crude into a valuable commodity ?”

Michael
Palmer
2006

http://ana.blogs.com/maestros/2006/11/data_is_the_new.html

Data *lock-in*

- accesso ai dati:
  - nel *tempo*
  - da *programmi* diversi
  - da *architetture* di tipo diverso



→ formati aperti e interoperabili
Open Document – Open XML

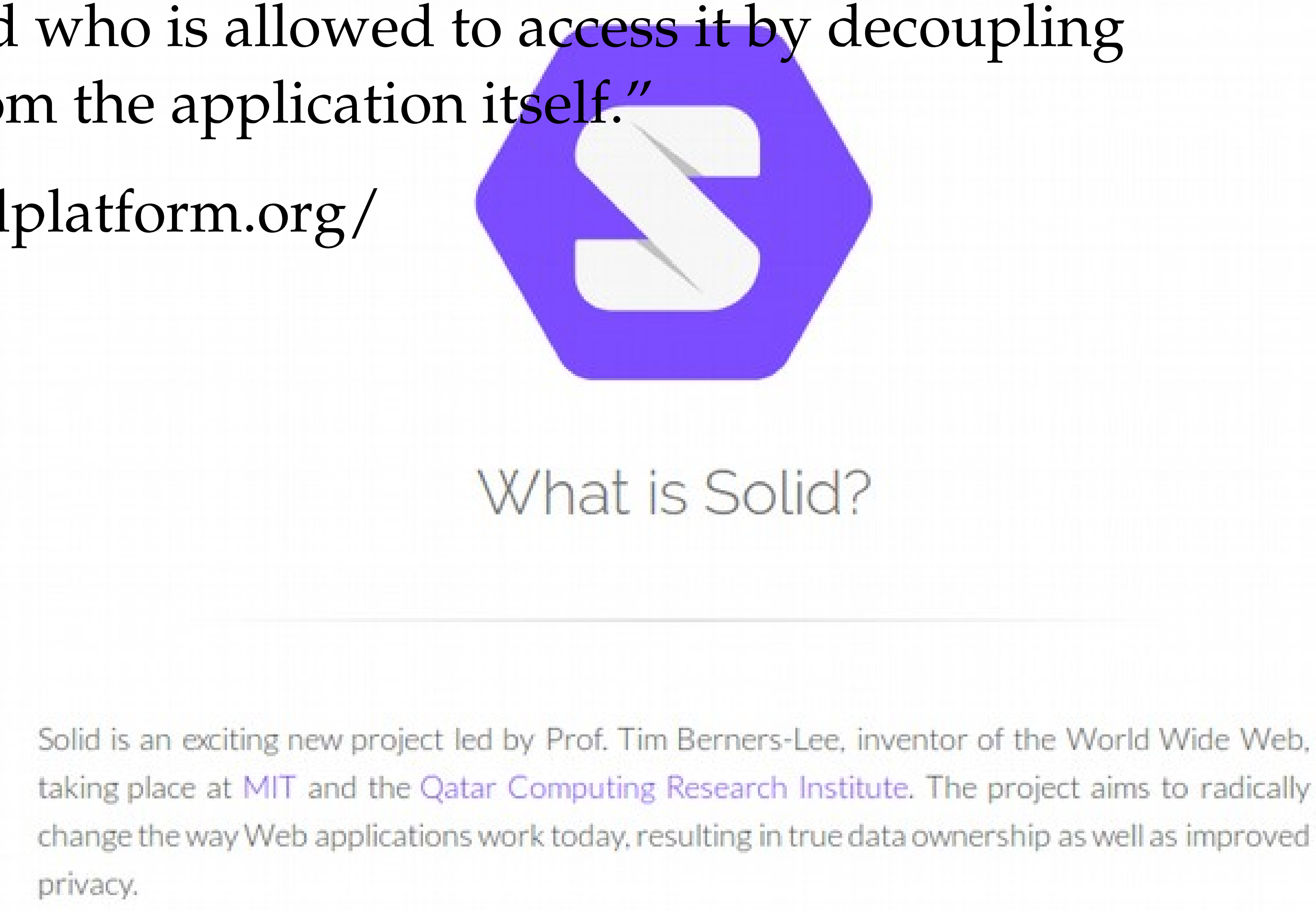
→ GDPR: Right to data portability

The *data subject* shall have the right to receive the personal data concerning him or her, which he or she has provided to a *controller*, in a **structured, commonly used and machine-readable format** and have the right to transmit those data to *another controller* without hindrance from the controller to which the personal data have been provided

Data Ownership

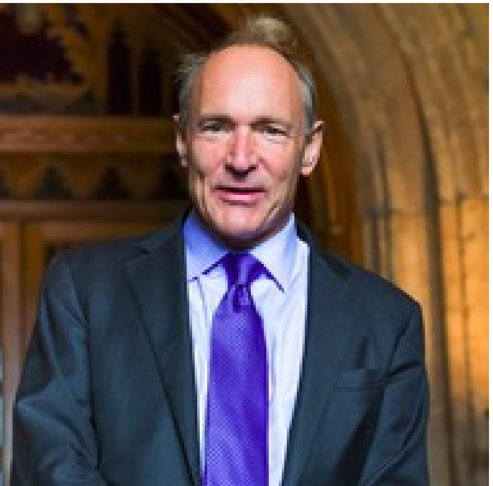
"Users should have the freedom to choose where their data resides and who is allowed to access it by decoupling content from the application itself."

https://solidplatform.org/



What is Solid?

Solid is an exciting new project led by Prof. Tim Berners-Lee, inventor of the World Wide Web, taking place at MIT and the Qatar Computing Research Institute. The project aims to radically change the way Web applications work today, resulting in true data ownership as well as improved privacy.



Thanks and happy hacking!

alberto -at- cammozzo.com
http://cammozzo.com
http://tagMeNot.info