Building *Enabling Environments*: open source among open systems

Alberto Cammozzo[1]

The Internet, the Web, the Linux kernel and the Debian distribution can be seen as systems, but they can be seen as an ecosystem, an environment for other projects. These open source systems are not only successful examples of cooperatively built and exceptionally large technical achievements, they also enable the development of each other and of many other projects; moreover they can be seen as an example of successful competitive strategy. Key properties of these enabling environments are: (1) their modular and layered structure; (2) the openness of their products, process and governance; (3) having developed a clear strategy along with shared principles among its participants.

Introduction.

A system, according to von Bertalanffy (1967), can be defined as *complex of interacting elements*. Many systemic models, representing the complexity of the relations between components of a whole, are focused on different aspects of the system (input, output, boundary, environment). In economic research, among others, Leontief (1986) analyzed the input/output matrix where "*The effect of an event at any one point is transmitted to the rest of the economy step by step via the chain of transactions that links the whole system together*", while Porter (2001) focused on the throughput process of transformation along a *value chain* happening in a given *competitive context* ; more recently, Chesbrough (2003) stressed on the need of a permeable interface to an environment where innovation leaks from and to innovating firms. Contributions from other research fields put the emphasis on regulation via *feedback* typical of organisms (Wiener 1949) or the presence of a shared channel (and code) between communicating components (Shannon 1948). Open Source, even emerging from a population of heterogeneous players having their own strategy, can be seen – as whole– as a *complex adaptive system*, finding its own way to find coordination thanks to connectivity and interaction (Muffatto, Faldani 2003), *adaptiveness* being the ability of a system to modify itself or the environment when either has changed (Ackoff 1971). According to the the concept of *stigmergy*, the environment can be seen as the place where *communication* takes place in order to allow mass collaboration between components, through its constant modification (Heylighen 2007, Elliot 2007). A trait of complex systems is they are often expressing properties which are not predicted from the knowledge of the parts and their interactions: they show the *emergence* of new properties (Ashby 1957).

In each one of these different views the *environment* is an important element. A definition of environment depends on which components belong to the system and which not: it it is the "*set of elements and their relevant properties which elements are not part of the system but a change in*

1 University of Padova. www.cammozzo.net

which can produce a change in the state of the system" (Ackoff 1971). An ecologic environment is not, as happens with ideal *thermodynamic* systems, a sink or a reservoir of infinite capacity. The terms "environment" and "system" can represent the point of view of the observer more than an objective difference, and the environment can be seen an active system[2] that provides input supply, output takeover and means of communication, command and control (Ashby 1957). Any system is continuously challenged by its environment but at the same time, together with other neighbor systems, builds it.

Any environment has a strong influence on the systems it hosts, and thus systems have to cope with environmental change. As an example Beer's *viable system* (1972) is any organization capable of maintaining its identity in a changing environment, that is, to be *adaptive*. But not only the environment is relevant for the systems it hosts, also the opposite is true. In the organization science research, it has been acknowledged by Daft and Weick (1984) that what they call *enacting organizations* have an active role on the environment: in fact they "*construct their own environments. They gather information by trying new behaviors and seeing what happens. They experiment, test, and stimulate, and they ignore precedent, rules, and traditional expectations"*. Lorenzen and Fredriksen (2005) highlighted the role of the *"market-organized innovation projects*" in the music industry, while Grabher (2002) observes that "*new media evolves as an organizational ecology that fragments into a broad range of activities that furnish the combination of information technology, content, and telecommunications"*. Enhanced communication capabilities allow for more interdependence between the system and its environment.

An open system seeks some sort of equilibrium with its environment, while exchanging elements and components essential to the functioning of the environment itself, which in turn provides essential functions to hosted systems. Interconnection of local networks by means of global transportation and communication networks extended to a global scale the scope of what once was a "local" *milieu*, even if strong local ties are more important than ever. What has been called the "death of distance" can be seen as the merging, overlaying and interconnecting of what once were different environments. Systems and clusters of systems once only loosely coupled have now many common networks, spanning different overlapping environments. The opportunity to exploit existing and latent networks comes along with the threats coming from a larger, more complex and mutable environment. In this scenario, any environmental change is both a threat and an opportunity for organizations: some are unable to react and lag behind and, if overwhelmed, eventually perish. Some are more adaptive and able to cope with environmental challenges and will evolve changing as needed their products, their processes and even governance models and strategies[3]. Only few, like the enacting organizations already cited, are even more active and deploy a strategy aimed at changing and controlling their own environment: not only to be prepared for future change, but to have a gain from competition with other organizations that share an environment they contributed to shape to their needs. This strategy seems specially effective in complex and unpredictable environments. *The* enacting *mode reflects both an active, intrusive strategy and the assumption that the environment is unanalyzable.* [Such organizations] *move ahead incrementally and gain information about the environment by trying behaviors and seeing what works.* (Daft, Weick, 1984)

---

2 Ashby invites the reader to consider, as an exercise: "*When an organism interacts with its environment, its muscles are the environment's input and its sensory organs are the environment's output. Do you agree?*" (Ashby 1952, p.47)

3 As Ackoff (1987) wrote : *Most institutions and enterprises seek what Donald Schon (1971) called a "stable state". Their resistance to change tends to be proportional to the need for it. The more turbulent their environment, the more stable the equilibrium they seek. They fail to realize that the only equilibrium that can be obtained in a turbulent environment -like that obtained by a ship in rough sea- is dynamic. A turbulent environment requires that institutions be ready, willing, and able to change themselves. Without changing themselves, they cannot adapt effectively to external change*

Purpose of the paper

In the past, the economic environment was shaped by nature, governments, industries and scientific research. Recently, other actors were able to so, adhering to large scale projects. The goal of this paper is to identify a set of large cooperative projects so they can be seen themselves as environments, precisely as enabling environments, and to recognize some of their properties. This papers aims to support the statement (A) that some communities, like these developing Free/Open Source Software (F/OSS) projects and those that built the Internet and the world wide web, share the purpose of building, shaping and maintaining the *environments* in which they act rather than just products. This fact could bring important consequences for other organizations, such as traditional firms sharing the same economic environment.

In order to support this statement, the next section contains a qualitative analysis for four projects, chosen for their amplitude and success: the internet, the World wide web, the Linux open source kernel and the Debian GNU/Linux distribution project. For each, I will show that (1) they have a common *structure*, (2) are built cooperatively following a well-defined *strategy*, (3) are similarly *open.* I will point to the key factors that enabled them to shape their environment, being: (1) ability to be *competitive* against other projects of the same open nature or not; (2) a layered and modular architecture, along with (3) open *governance* and coordination, and (4) clear foundation principles helping the emergence of effective *strategies*.

In a greater detail, the analysis will show that all these projects share those key properties:

- they are *modular,* which makes cooperative work easier; they are *layered*, having being incrementally built on successive, progressive technological beds. *Open standards* have been designed on the boundaries, according to good engineering practices;
- they have been pragmatically and incrementally developed from a loose form of *cooperation*, based more often on a set of common understandings and protocols than on central directions;
- their coordination process and project *governance* are open, based upon a set of documents that define a *mission* and provide clear procedures for a shared decision making process focused on consensus among all stakeholders. An analysis of Open Source governance (Markus 2007) recognizes multidimensionality along three paths of action: solving collective action and motivational dilemmas, solving development coordination problems and providing an appropriate *climate* for contribution;
- they are themselves layered in a process of accretion, concurring to create an overall *enabling environment* which is a fertile ground on which other projects can grow; it can be stated that this accretion has continued beyond the four projects examined, increasing the overall potential;
- although these projects are no-profit, they carry relevant economic value, providing an opportunity for business, and motivation for firms to participate following a strategic intent to shape the competitive arena. Individuals, communities and firms cooperate *and* compete through open projects.

As a consequence of this analysis, two other results will emerge: (B) the importance of knowledge being embedded in the environment itself (*stigmergy*) in the process of building the environment, and (C) the simultaneous presence of intrinsic and extrinsic motivations for different actors (Rullani 2006, Amabile 1993): single users, communities and firms in the act of building a common environment.

It is not so clear, though, what will be the fate of the balanced coexistence of these motivations in this complex ecosystem if the elements of reciprocity that allow the present evolution are not maintained.

Projects analysis

1 – Internet

Competitive factors

Theorized in the sixties, geographical computer networks became commercially available in the seventies. The main ones were: SNA (proprietary IBM), DECnet (proprietary Digital), Bitnet (open academic architecture). Fidonet or UUCP were noncommercial networks that would be considered *peer-to-peer* networks.

Internet imposed itself[4] on the other established geographical networks, replacing them and becoming the *de facto* standard for interconnecting local area networks. A major role among the different key factors for its success being the great amount of public spending[5], but also the the openness of technology and its decumentation. As some of the architects of the Internet said (Leiner, Cerf et al 2001):

> *A key to the rapid growth of the Internet has been the free and open access to the basic documents, especially the specifications of the protocols. [...] In 1969 a key step was taken by S. Crocker (then at UCLA) in establishing the Request for Comments (or RFC) series of notes. These memos were intended to be an informal fast distribution way to share ideas with other network researchers. At first the RFCs were printed on paper and distributed via snail mail. As the File Transfer Protocol (FTP) came into use, the RFCs were prepared as online files and accessed via FTP.*

Open technology and open documentation form an environment to which knowledge adheres: technology is used to access the documentation, and documentation is used to enhance technology. The extension of the network without the need of a central coordination had another key role[6], and finally the development of the World Wide Web (WWW) made the Internet a success also among consumers that had never used a computer before.
Although other complex architectures were technically superior, being quite simple and thanks to its flexibility Internet was able to deliver the same level of service and performance of proprietary networks and gradually took the place of its competitors. It should be noted that Free/open source software has been involved in Internet development not only by exerting a technological influence but also by shaping the building process (Benussi 2005).

Layering and modularity

Layered architecture is common to any advanced computer network. Each layer represents an abstraction that offers services to an upper layer without the need to know the implementing details of lower layers. Internet defines only two central layers (IP and TCP), taking advantage of

4 Internet becomes a global project in a few years: *In its 8 1/2 year lifetime, the Backbone had grown from six nodes with 56 kbps links to 21 nodes with multiple 45 Mbps links. It had seen the Internet grow to over 50,000 networks on all seven continents and outer space, with approximately 29,000 networks in the United States.* (Leiner, Cerf *et al*. 2001). Very interesting insights on the history of Internet and the Web can also be found in Ilkka Tuomi (2002).

5 *Such was the weight of the NSFNET program's ecumenism and funding ($200 million from 1986 to 1995) - and the quality of the protocols themselves - that by 1990 when the ARPANET itself was finally decommissioned, TCP/IP had supplanted or marginalized most other wide-area computer network protocols worldwide, and IP was well on its way to becoming THE bearer service for the Global Information Infrastructure.* (Leiner, Cerf *et al*. 2001)

6 The independence of each participating network was among the fundamental Internet's design principles, as outlined by Kahn "*each distinct network would have to stand on its own*" along with the absence of a centralized control "*there would be no global control at the operations level*". (Leiner, Cerf *et al.* 2001)

existing technologies for the lower (physical) layers, and leaving to programmers the definition of upper layers. A tentative of standardizing networking, proposed by the *International Organization for Standardization* (ISO) with the *Open Systems Interconnection* (OSI) initiative (1977), while being very advanced in theory, had seven layers and was heavy and inefficient. The ISO-OSI definition is theoretically so good that it is still being used as a theoretical framework, but the simple design of the Internet and its independence from physical infrastructure allowed its fast adoption without costly investments.

Internet can be defined as a system whose components are independent and autonomous networks: successive components can be added incrementally. Modularity can also be found in its architecture: based on the underlying two layers, each network service (as email, file transfer, printing, web browsing etc.) has its own independent protocol, described in an RFC. On the whole, those services offer the upper layer of the Internet. Yochay Benkler sees modularity as a key aspect of Internet and many other network based projects (Benkler 2006):

> *The core characteristics underlying the success of these enterprises [Internet and peer-production] are their modularity and their capacity to integrate many fine grained contribution.*

The fine granularity is as a key factor for project manageability: for efficiency reasons, the software industry often develops its products using reusable and standard components (Messershmitt, Szyperski 2003).

> *There is growing interest in software reuse and component software as a way of improving the productivity of development organizations, addressing increasing complexity, and improving quality. The idea is to assemble applications largely from existing components, which are modules created independently of any particular system context and constructed for multiple uses.*

Cooperation, coordination and governance

Each network composing the Internet is independently administered, in respect of general interoperability rules and standards (RFCs). Some important central coordination activities are assigned primarily to two entities: the Internet Engineering Task Force[7] (IETF), who evaluates and makes protocol proposals official, and the *no profit corporation* Internet Corporation For Assigned Names and Numbers[8] (ICANN), which regulates domain names and assigns network numbers. Even if formally independent, ICANN depends on the United States government, and its status is currently under debate. IETF, whose home organization is the Internet Society[9], promotes the standards on which Internet is based. Its most important role is regulate the way layers and modules are bound together through interfaces and protocols. Ad-hoc working groups address specific issues, coordinated by a *steering committee*. Inside each group, consensus drives the decision-making process (Bradner 1998):

> *The core rule for operation is that acceptance or agreement is achieved via working group "rough consensus". [...]*
> *IETF consensus does not require that all participants agree although this is, of course, preferred. In general, the dominant view of the working group shall prevail. (However, it must be noted that "dominance" is not to be determined on the basis of volume or*

7 On IETF see: http://www.ietf.org/
8 On ICANN see: http://www.icann.org/general/
9 Om ISOC see: http://www.isoc.org/isoc/

persistence, but rather a more general sense of agreement.) Consensus can be determined by a show of hands, humming, or any other means on which the WG [Working Group] agrees (by rough consensus, of course). Note that 51% of the working group does not qualify as "rough consensus" and 99% is better than rough. It is up to the Chair to determine if rough consensus has been reached.

There is a significant difference between Internet governance infrastructure and traditional coordination of technical organizations such as the ITU (International Telecommunications Union). A good synthesis can be found in an ITU report (Hayashi 2003):

In order to maintain the integrity of the global telecommunications network, standard setting, rule making and monitoring of the system is necessary on the global scale. Such a task has been performed by ITU, ICANN and IETF. ITU is administered as part of the United Nations on the traditional international legal framework, i.e., its membership comprises individual sovereign nations, decisions are made on the unanimity basis, and the agreements are concluded as international treaty between ITU and the participating nations.

On the other hand, the internet technology requires new forms of global organization. For example, the Internet Corporation for Assigned Names and Numbers (ICANN) handles IP address space allocation, protocol parameter assignment, domain name system management, and root server system management functions. The Internet Engineering Task Force (IETF) is a large open international community of network designers, operators, vendors, and researchers concerned with the evolution of the internet architecture and the smooth operation of the internet.

The outstanding characteristics of these new organizations are that they take individuals as members. No corporations or countries have representation in them. They are not inter-national organizations since they do not act on nations. They are trans-national organizations because they involve concerned individuals across countries and regions. Also, they use a non-traditional approach in which participation and compliance is only voluntary.

Principles and strategies

Even being very informal[10], IETF has developed formal rules. Similarly to firms, IETF and many other entities base their strategic action on *statements* and principles, sometimes explicitly formulated as a *mission* or *vision*. IETF goals and objectives are well defined in the document *A mission statement for the IETF* (Alvestrand 2004):

*The **goal** of the IETF is to make the Internet work better.*

*The **mission** of the IETF is to produce high quality, relevant technical and engineering documents that influence the way people design, use, and manage the Internet in such a way as to make the Internet work better. These documents include protocol standards, best current practices, and informational documents of various kinds.*

10 As an example of "formalized informality", this is the *dress code* di IETF:
Dress Code: since attendees must wear their name tags, they must also wear shirts or blouses. Pants or skirts are also highly recommended. Seriously though, many newcomers are often embarrassed when they show up Monday morning in suits, to discover that everybody else is wearing t-shirts, jeans (shorts, if weather permits) and sandals. http://www.ietf.org/tao.html#2.3

The same document draws the process and identifies resources to serve the purpose while remaining in the scope of the mission; this kind of behavior can be seen as a strategy (Alvestrand 2004):

> *The IETF will pursue this mission in adherence to the following* ***cardinal principles****:*
>
> ***Open process*** *- any interested person can participate in the work, know what is being decided, and make his or her voice heard on the issue. Part of this principle is our commitment to making our documents, our WG mailing lists, our attendance lists, and our meeting minutes publicly available on the Internet.*
>
> ***Technical competence*** *- the issues on which the IETF produces its documents are issues where the IETF has the competence needed to speak to them, and that the IETF is willing to listen to technically competent input from any source. Technical competence also means that we expect IETF output to be designed to sound network engineering principles - this is also often referred to as "engineering quality".*
>
> ***Volunteer Core*** *- our participants and our leadership are people who come to the IETF because they want to do work that furthers the IETF's mission of "making the Internet work better".*
>
> ***Rough consensus and running code*** *- We make standards based on the combined engineering judgement of our participants and our real-world experience in implementing and deploying our specifications.*
>
> ***Protocol ownership*** *- when the IETF takes ownership of a protocol or function, it accepts the responsibility for all aspects of the protocol, even though some aspects may rarely or never be seen on the Internet. Conversely, when the IETF is not responsible for a protocol or function, it does not attempt to exert control over it, even though it may at times touch or affect the Internet.*

On the *multi-stakeholder governance* of the Internet the United Nation has produced an extended report. (MacLean 2004). Given the growing economic and political importance of the Internet, these principles are more and more challenged, and more traditional forms of regulation are likely to shape the future of the Internet. The debate is mainly centered around the issues of *network neutrality* regards to the contents, of government censorship and the on balance between security and privacy.

2 – The World Wide Web

Competitive factors

Hypertext origin goes quite far in time: "oNLine System" was made by Doug Englebart[11] in the 1970s, and Ted Nelson's *Xanadu* planned network access to documents[12]. The story of the competition that brings the present Web is complex. In 1990 Tim Berners-Lee tries to address loss of information issues in Geneva's CERN with a distributed hypertext system. University of Michigan around 1991 developed another friendly system for accessing internet information, called "gopher", which was not made for use with graphical workstations but only on traditional terminals. At the beginning the gopher code was free software, but it was closed from 1993, probably as a response to the distribution of first graphic browser, Mosaic[13]. Mosaic was publicly funded by

11 See video recordings: http://sloan.stanford.edu/mousesite/1968Demo.html

12 See http://xanadu. Ted Nelson eventually released Xanadu source code under the name Udanax in 1998: http://www.udanax.com/

13 Mosaic was then developed by the firm Netscape Communication and became the well-known Netscape Navigator, ancestor of the present open source browser Firefox, which was distributed as *open source* as a response to the

USA's National Center for Supercomputing Applications and its code was freely available. The graphic, hypertext and multimedia Web was adopted first by academic and research organizations, then, by the nineties, by all others, given that Mosaic was available for the graphical operating system Windows.

> *[...] Mosaic could match the functions of the Gopher protocol and additionally offer added functions such as hyper linking from within HTML files which brought together related pages more efficiently than Gopher, there was no longer a compelling reason to choose the Gopher system. Another advantage the early Web had over Gopher was the decision of the University of Minnesota not to definitively rule out the option of exercising its intellectual property rights over the Gopher protocol, for any other organisation deciding whether to devote time, effort, and expense to adopting one of the systems the possibility of getting locked into a technology that they could then find themselves being charged for was good reason to prefer the World Wide Web. Most of the files and databases that had been available on Gopher were converted into HTTP compatible formats and made available on the Web[...]* (Codeghost 2007)

The intricate story of the browsers continue with the *browsers war*: commercial rights for Mosaic went to Spyglass inc. which gave them also to Microsoft to develop Internet Explorer (IE). Andreessen started Netscape, the most used browser before the introduction of IE. In order to compete, Netscape released its browser as *open source*. In both the Gopher-Netscape and IE-Netscape competitions, the openness of the technology was a competitive factor. A similar story can be told for the Web servers competition.

Layering and modularity

Basically the web is a client-server architecture based on two layers: the first is the language that prescribes the way pages are composed (HTML - *hyper text markup language*) and the other regulates information flow between web servers and client browsers. In this second layer the most used protocol is HTTP but many others are used for different functions (as FTP for file transfer): all those protocols can be seen as modules. Both layers are becoming more and more complex as the Web evolves and more protocols are added in a cumulative process (Tuomi, 2002).

Search engines can be seen as a further layer, that makes it possible to access information through its *content* and not only through an *address* that should be known in advance.

The thing called “Web 2.0”, even being quite elusive, is for sure a *platform*, a layer upon which modules (and other layers) can be built. The compact definition given by Tim O'Reilly (2005) says:

> *Web 2.0 is the network as platform, spanning all connected devices; Web 2.0 applications are those that make the most of the intrinsic advantages of that platform: delivering software as a continually-updated service that gets better the more people use it, consuming and remixing data from multiple sources, including individual users, while providing their own data and services in a form that allows remixing by others, creating network effects through an "architecture of participation," and going beyond the page metaphor of Web 1.0 to deliver rich user experiences.*

As said for the Internet, web modularity can be found in its *technologies* (different protocols for different functions) but also in its *resources:* independent web sites form networks linking their

---

browser wars with Microsoft Internet Explorer.

information together with the same coalescence process seen in the building of the Internet.

Cooperation, coordination and governance

The cooperation enabled by web resources (such as wikis) can be seen as another form of information modularity. This process has been analyzed by many, among which Benkler (2006) which defines it *peer-production*. The Web offers an environment, a context, in which information becomes accessible, usable and can become knowledge. Information sticks to an environment where it becomes accessible also thanks to search engines[14] that try to extract and provide meaningful information. Semantic web and *folksonomies*, and in general projects based on users producing and tagging information in a cooperative way will assume a growing importance, complementing search engines.

Principles and strategies

The goal of the international World Wide Web Consortium (W3C) is defining technical standards. Even if open to anyone, W3C activities are aimed mainly at organizations and firms. W3C issues *recommendations*, not mandatory standards.

The declared *mission* of W3C is:

> *To lead the World Wide Web to its full potential by developing protocols and guidelines that ensure long-term growth for the Web.*[15]

The governance process requires the maximum attention to any stakeholder:

> *All stakeholders can have a voice in the development of W3C standards, including Members large and small, as well as the public. W3C processes promote fairness, responsiveness, and progress: all facets of the W3C mission.*[16]

Developing an open Web explicitly designed for sharing knowledge is one of W3C's goals (Bratt 2006):

> *The social value of the Web is that it enables human communication, commerce, and opportunities to share knowledge. One of W3C's primary goals is to make these benefits available to all people, whatever their hardware, software, network infrastructure, native language, culture, geographical location, or physical or mental ability.*

It important to note that W3C will approve only recommendations that can be implemented with royalty-free patents.

3 – The Linux Kernel

The Gnu/Linux operating system ecosystem has many components: in addition to the Linux *kernel* there is a collection of system programs developed by the GNU project, and many other programs from different other projects. Therefore different GNU/Linux *distributions* reflect different ways to pick, merge and blend those different elements. The process of building a distribution is the object of the next section, while this section is focused on the Linux *kernel*.

---

14 Google holds a position of substantial monopoly in the search engines marketplace. The importance of the process of information selection and the lack of transparency of Google's ranking algorithms raises concern.

15 http://www.w3.org/Consortium

16 http://www.w3.org/Consortium/process.html

Competitive factors

Linux is a product among others in a segmented operating system market. As its main competitor MS Windows, it has explicitly developed specific strategies to compete in the different segments of the market. The subject in charge to work out those strategies is The Linux Foundation[17] who collects its members from the whole ITC industry and "*promotes, protects and standardizes Linux by providing unified resources and services needed for open source to successfully compete with closed platforms[18]*".

Even if Linux is quite weak in the desktop system vendors segment, where computers are usually sold with Windows already installed[19], servers can be bought also with one of the GNU/Linux most popular commercial distributions. Reportedly, GNU/Linux server installations are growing more than its competitor (Shanckland 2004, IDC 2008).

In the *equipment suppliers* segment (appliances, industrial control systems, phones, and other embedded systems) Linux competes with, among others, VxWorks, QNX, and Microsoft (which has a share similar to others). Equipment suppliers often install different operating systems for different products, so they don't seem to have a precise direction. However interest for Linux appears to be growing (Turtley 2005), especially in the mobile segment where the Linux kernel is at the core of many projects from many firms, including Nokia, Google, Motorola.

Different *Linux distributions* providers are competing between them *and* against other proprietary competitors: some are given for free (as Debian), while other are commercial (RedHat and Novell are the best known). Many offer a product (a specific GNU/Linux distribution) along with maintenance services, while several firms offer maintenance alone.

It has been observed that once free/open source software enters a market, it is very difficult to find competitive strategies to expel it, as users tend to stick to it fearing proprietary lock-in (Lindman 2004). More will be said on the competition of firms involved in Linux in the section "principles and strategies".

Layering and modularity

Operating systems are, as general rule, engineered in layers. Moreover, the core (kernel) is a layer on which other elements of the system are based, and is modular in most operating systems, both for technical reasons and to ease its development. As in networks, modularity appears at two levels: the kernel as a *product* is modular, as the *process* to build it.

It has been observed that modularity helps managing innovation and differentiation but at the expense of coordination (Devetag, Zaninotto 2001):

> *Modularity seems better suitable than other approaches when the adaptation to the demand variety and the speed of search in a given problem space (i.e., the rate of technological innovation) are more important than tight co-ordination. If this is the case, division of labour must occur in such a way that considerable space for local adjustment and experimentation is assured, unlike the case of a traditional assembly line.*

In the Linux *kernel*, as in other projects, modularity is a way to minimize complexity generated by

17 *"Founded in 2007 by the merger of the Open Source Development Labs and the Free Standards Group, it sponsors the work of Linux creator Linus Torvalds and is supported by leading Linux and open source companies and developers from around the world":* http://www.linux-foundation.org/en/Main_Page

18 https://www.linuxfoundation.org/about

19 On having GNU/Linux installed by PC vendors see Vaughan-Nichols 2007

the interdependence of components. The goal for modularizing is (Narduzzo, Rossi 2005):

- manage uncertainty and variability in problem solving,
- ease task and product decomposition,
- widen product variety, on a differentiation strategy.

Given the extreme division of tasks that a free/open source project is likely to face, modularity is not only a technical need, but a development model (Narduzzo, Rossi 2005).

> *We argue that modularity, which can be regarded as an innovative manufacturing paradigm for the design and the production of complex artifacts, is a key element in explaining the development and the success of many F/OSS projects, and it offers a comprehensive explanation of many key issues in F/OSS development, such as how division of labor takes place within developers, how coordination is achieved and how code testing and integration is deployed, how innovation occurs, and so on.*

Modularization sometimes fail: knowledge embedded in interfaces must be carefully planned in advance, and – as system evolves – the boundaries between its modules must fit tightly to each other (Narduzzo, Rossi 2005).

> *[...] Unfortunately, this neat description of modular design sometimes does not succeed; most of the times, after the integration of the independently developed modules, inconsistencies come up on and the system does not work properly. The most common reason for this failure is the emergence of some interdependencies which were left out at the beginning, at the time of architecture and interfaces definition.*

Cooperation, coordination and governance

Linus Torvalds, first author of Linux, still coordinates and has the “last word” on decisions regarding the project, which is still Internet- and Web-based. But Linux has also become a huge business: since year 2000 the firms most involved in the project have founded a no-profit corporation, the Open Source Development Laboratories (OSDL), now part of The Linux Foundation. While the development continues in the community, the Foundation has the purpose of *promoting, protecting, and standardizing Linux and open source software*. Even individuals can register as affiliates for a modest fee[20], but no member has any right on property or assets of the corporation[21].

Principles and strategies

Such a large coordination between firms and a large community of users and developers around a single project under a common strategy is unprecedented and raises some interesting issues.

The original OSDL declared *mission*[22] *was*:

> *To accelerate the deployment of Linux for enterprise computing through:*
> *- Enterprise-class testing and other technical support for the Linux development community.*
> *- Marshalling of Linux-industry resources to focus investment on areas of greatest need thereby eliminating inhibitors to growth.*
> *- Practical guidance to our members - vendors and end users alike - on working*

20 http://www.linuxfoundation.org/about/join/individual/join
21 Bylaws, section 3.8 http://www.linuxfoundation.org/about/bylaws
22 http://www.osdl.org, now http://old.linux-foundation.org/about_osdl

effectively with the Linux development community.

These are essentially strategic goals: identify and eliminate threats in the market and weakness points in product, develop efficient production processes, exploit opportunities. Members[23] of the Foundation are often competing firms, nonetheless they contribute developing common strategies around a product that, staying free, allows them to develop competitive advantages. Why do firms competing between them invest in a innovation project they don't even *own*? Because of the importance of the competitive environment they share. In *Corporate Philanthropy* (Porter, Kramer 2002), this kind of action is called the *construction of the competitive context*. Authors identify four fronts:

> *A company's competitive context consists of four interrelated elements of the local business environment that shape potential productivity: factor conditions, or the available inputs of production; demand conditions; the context for strategy and rivalry; and related and supporting industry.*

In detail, those four fronts are present in the building of the competitive context for Linux.

1. *Factor conditions.* One single product, Linux, is a common infrastructure technology available for each actor, in a context where the only conditions needed for production are net access, available to anyone at the same conditions, and the ability to innovate.
2. *Context for Strategy and Rivalry.* The competitive context is being defined through a common policy, set by the GPL license, which is valid for each stakeholder: programmers, firms, users, all are in the same conditions with respect to code.
3. *Related and Supporting Industries.* A standard product allows firms to work in similar conditions on the same product, leveraging their strategies and creativity to compete.
4. *Demand conditions.* Linux *sophisticated users* not only are able to anticipate the demands of most users, but can enter in the production cycle adapting the product to their needs and submitting their innovative proposals to the project.

The following statement seems suitable (Porter, Kramer 2002) to describe what happens around free software:

> *Philanthropy can often be the most cost-effective way for a company to improve its competitive context, enabling companies to leverage the efforts and infrastructure of nonprofits and other institutions.*

Another point of view, focused on innovation rather than the competitive context, may come from the concept of *open innovation* (Chesbrough, 2003). Even if never abandoning a “traditional” view over *property* of knowledge, Chesbrough pictures effectively the extreme difficulty and definitive obsolescence of the internal (closed) innovation paradigm, due to erosion factors that *have loosened linkage between research and development*.

> *Ideas can no longer be inventoried on the shelf, because they will leak out to the broader environment over time. A company that fails to utilize its technology may later see variants of those ideas exploited by other firms. At the same time, these erosion factors collectively create rich variety of possible research inputs available outside the firm.*

The solution is to let firms be open to a constant flow of knowledge (both in input and output) with

23 http://www.linuxfoundation.org/about/member

the environment.

> *Companies must structure themselves to leverage the distributed landscape of knowledge, instead of ignoring it in the pursuit of their own internal research agendas.*

Chesbrough also recommends that firms develop a suitable business model to *claim a sufficient portion from the [value] chain to justify its participation*.

4 – The Debian project

Installing one of the tens of thousands different free/open source programs whose source code is available around the net may be complicated even for a skilled system manager. *Distributions* are collections of programs already compiled in *packages*, ready to be installed, designed to reduce the hassle of compiling the source code.

The Debian project's work is to produce and maintain the widest and most complete GNU/Linux distribution, which contain more than 20'000 different programs: it is probably the largest operating system ever realized (Amor-Iglesias et al, 2005). According to a more comprehensive and recent research (Gonzalez-Barahona, Robles et al, 2009) is defying the manageability of such a project.

> *With respect to the absolute figures, it can be noted that Debian 4.0 is probably one of the largest coordinated software collections in history, and almost certainly the largest one in the domain of general-purpose software for desktops and servers. This means that the human team maintaining it, which has also the peculiarity of being completely formed by volunteers, is exploring the limits of how to assemble and coordinate such a huge quantity of software.*

Competitive factors

Linux system providers offer different distributions in competition between them. Some of them use the software packages from Debian distributions, while other use their own criteria, as RedHat and Novell. Distributions are differently targeted on users, the variety of programs they offer, the set of services offered and their price[24]. Debian is the base of the popular distribution Ubuntu, with which it competes and collaborates[25].

Layering and modularity

In a system installed from a distribution, the single package can be seen as a module. Designing good interfaces between those modules and reporting all the relations between them is the key for a stable system and a successful distribution.

> *The modularity of Debian, understood as a large collection of components, is allowing developers to build powerful, yet small applications, that gain advantage of using tens of other packages. (Gonzalez-Barahona, Robles, et al, 2009)*

Layering in a Debian system emerges from the interdependence between packages: some of them rely on others, in fact imposing a hierarchy of layers .

> *[...] packages are highly interrelated, and as Debian evolves, the total number of dependencies grows quickly. We have also seen how packages with the interpreters for*

24 For a review of available distributions see: http://en.wikipedia.org/wiki/Comparison_of_Linux_distributions
25 http://www.ubuntu.com/community/ubuntustory/debian

some scripting languages, Perl and Python, are among those being used by more packages, and that the C run time library, libc6, is being required by almost every package. (Gonzalez-Barahona, Robles, et al, 2009)

Cooperation, coordination and governance

Based on the *Debian constitution*[26], Debian has a complex division of tasks between technical roles achieved by volunteer *developers* recruited[27] with a severe selection process, and coordination roles carried out by *project leaders*, being elected by developers. Debugging, writing documentation and user support activities rely on the wider community of Debian users. An elected *leader* of the whole project holds mainly coordination and communication functions. The Debian project's property is owned by *Software in the Public Interest inc*[28] (SPI), which also provides legal support.

Principles and strategies

There are a number of documents describing Debian's goals and strategy. The *manifesto* states that the whole work process is centered on the community's needs (*The Debian Linux Manifesto*[29]):

The Debian design process is open to ensure that the system is of the highest quality and that it reflects the needs of the user community.

The *Debian Social Contract*[30], along with the *Debian Free Software Guidelines*, define the project's traits and what features its products should have, outlining a long term *strategy*. Point 4 of the *social contract* is very clear on how the project mission should create an open *environment* for users.

4. Our priorities are our users and free software
We will be guided by the needs of our users and the free software community. We will place their interests first in our priorities. We will support the needs of our users for operation in many different kinds of computing environments. We will not object to non-free works that are intended to be used on Debian systems, or attempt to charge a fee to people who create or use such works. We will allow others to create distributions containing both the Debian system and other works, without any fee from us. In furtherance of these goals, we will provide an integrated system of high-quality materials with no legal restrictions that would prevent such uses of the system.

The *Debian Free Software Guide Lines*, which are part of the Social Contract, are at the base of the *Open Source Definition*[31], being accepted also by the industry as a definition of Open Source (Di Bona et al 1997).

Common elements and the role of knowledge

All of these projects, even if they don't have a strong central coordination, are developing strong identity and strategy: they sometimes behave like firms and compete successfully with them. Those kind of projects, thanks to the key factors explained above, flank each other and sometimes

26 http://www.debian.org/devel/constitution
27 The process has been studied by Gabriella Coleman. See (Coleman 2005) .
28 http://www.spi-inc.org
29 http://www.debian.org/devel/join/newmaint
30 http://www.debian.org/social_contract
31 http://www.opensource.org/docs/definition.php

combine in layers forming environments that enable others projects to grow upon them. Sometimes positive feedback enhance this enabling effect: free software and hypertexts existed before the Internet, but Linux and the Web owe their success to its commercial diffusion. Internet demand, in turn, exploded thanks to the diffusion of the Web, *etc.*

Each project is an *enabling environment* in which knowledge is a key factor: all of them embed knowledge of the project in the project itself in the form of code, comments, documentation, mailing lists conversations, practices, websites; this way it “speaks for itself and by itself” to newcomers and allows its own reproduction. Moreover all the projects can be seen as components of a whole system which is an environment on which other projects can rely on.

These are the common elements in synthesis:

1. Layered and modular **structure.** Knowledge plays a major role in the division between horizontal layers and vertical modules:
   - each level or layer needs a different kind of knowledge or competence. At a given layer, the knowledge needed at another layer can be ignored;
   - different modules inside a layer address different problems or projects, but require the same “set of knowledge”;
   - layers and modules relate to each other through well defined, standardized interfaces and protocols, that address and regulate every aspect of the dialog between them;
   - process cooperation and formalized interaction between modules becomes possible on the basis of those standards.
1. **Cooperative** model. All subjects developing and taking care of the system modules cooperate using predefined interfaces and tools. Coordination of the cooperative process (*governance*) can happen on loosely hierarchical basis, with more or less democratic procedures, but in general based upon rough consensus. Product modularity is reflected in a cooperative, and modular, *production* process.
2. **Openness** on all levels:
   1. *use and distribution*: no limitation for use of the product or project. Anyone can use it, without barriers on purpose or intent;
   2. *product architecture*: no barrier of technical, normative, economic nature is opposed to the access to information that describes and defines the project. Anyone can suggest and try to implement new elements or modules, or modify and improve system architecture or process;
   3. *governance*: project management and decision process are potentially open to any stakeholder. Nearly anyone with enough technical competence and necessary consensus can access to the coordinating roles.
3. **Shared principles.** All stakeholders generally share some general principle: a mission, guidelines, ethical code or some other foundation document. The more the participants become involved in the project, the more they account those principles as important. The foundation documents include not only some ideal principle, but also a *vision*, operative guidelines and future directions. These determine the way to respond to stimuli (threats and opportunities) coming from the environment and how to shape it, and can be regarded as a strategy, or at leasta part of it. Those principles give an order (in both meanings of “order”: *command* and *regular arrangement*) to a structure that otherwise leaves to individuals and groups the largest autonomy inside the single modules. This looks like a way by which letting chaos produce the maximum creative yield without failing to attain prearranged objectives and maintaining consensus.

Conclusions: the building of an *enabling environment*.

An "*enabling context*" is defined as (Von Krogh, Ichijo, Nonaka 2000) "*what drives knowledge creation"* and is linked to the concept of *Ba* as an organization's shared space that facilitates interaction, exchange and sharing of knowledge and creativity. The examples examined in this paper cannot be seen as a space that is *internal* to some system, nor anybody *owns* them entirely. Instead of being a closed space where actors meet to build some other thing, they are an "external" context, an enabling environment that actors contribute to build *from wihin*. A free software project as Linux, for instance, cannot be imagined as being inside something, not even its own community (while the contrary can be said). It can be seen as if it is "floating" in the net, where it grows and contributes providing its own environment, as long as it is used. Building, enhancing and empowering the environment is the *objective* of the action but also – simultaneously – the *means* by which the activity takes place: it is a production factor of itself, being made of spaces, relations, infrastructures, code, projects, frozen knowledge. The goal of building the environment is never completed: it is incessantly manipulated and has new modules and layers continuously added in a state called *permanently beta* (Neff, Stark, 2002). Such a project cannot be completely planned in advance, but still their actors "*move ahead incrementally and gain information about the environment by trying behaviors and seeing what works"* (Daft, Weick, 1984).

Many firms are learning to take advantage of those environments as *a place where innovation happens* (Chesbrough 2003) and where to contribute with innovation and investments. An open environment is effectively *enabling* towards *any* actor: anyone can become involved enhancing the environment and contributing to its development, and all users of the environment can potentially be co-workers. Alongside of cooperation, also *competition* is present, providing the conditions of selection by which only the best solutions are widely adopted and evolve: it wasn't technical superiority to give to the projects analyzed in this paper a competitive advantage, but a good modular and layered design that made production easy, clear goals that help a people in a community to work stick together ad provide them identity, and open processes and governance.

The building of a common environment allows the coexistence of the prevalently intrinsic motivations of single individuals adhering to communities with those, more extrinsic and profit-oriented, of firms; this coexistence of motivations happens in a shared environment. Such a vision gives an account of what Perroux (1981) called *enthusiastic, lasting support for some great common undertaking.*

After an initial lack of trust, industry is now embracing happily open source software and cooperative modes of producing and organizing production, made possible by an highly comunicative environment. New slogans are: "lets' throw it to the community" or even "let's build a community for it". Attention should be paid: communities are part of an "environment" which is not a reservoir of infinite capacity, but an ecosystem. Delicate balances, especially of motivations, should be respected.

Aknowledgements

The central part of this work has been presented to OSSEMP07 workshop in Limerick in the context of the Open Source Systems conference. I wish to thank reviewers, organizers and participants for attention, precious suggestions and insights: the workshop has proved to be a stimulating *environment.*

References

- Ackoff, Russell L., *Towards a system of systems concepts*, Management Science, vol 17 n. 11, July 1971
- Harald T. Alvestrand, *RFC 3935, A Mission Statement for the IETF*, October 2004
- Amabile, Teresa M. "Motivational synergy: Toward new conceptualizations of intrinsic and extrinsic motivation in the workplace." *Human Resource Management Review* 3, no. 3 (1993): 185-201.
- Juan-José Amor-Iglesias, Jesùs M. Gonzàlez-Barahona, Gregorio Robles-Martìnez, and Israel Herràiz-Tabernero Measuring Libre Software Using Debian 3.1 (Sarge) as A Case Study: Preliminary Results; http://www.upgrade-cepis.org/issues/2005/3/up6-3Amor.pdf
- Beer, Stafford. *Brain of the firm*. Herder and Herder, 1972.
- Benkler, Yochay, *The wealth of networks: How Social Production Transforms Markets and Freedom*, Yale University Press, 2006; http://www.benkler.org/wealth_of_networks/index.php/Main_Page
- von Bertalanffy, Ludwig, *General System Theory: Foundations, Development, Applications*, Braziller, 1973, New York
- Benussi, Lorenzo, *Analysing the technological history of the Open Source Phenomenon: Stories from the Free Software Evolution*, Working paper 2005, http://opensource.mit.edu/papers/benussi.pdf
- Bradner, Scott, ed, *RFC2418: IETF Working Group Guidelines and Procedures*, September 1998; http://rfc.net/rfc2418.html
- Bratt, Steve, *Worldwide Participation in the World Wide Web Consortium, W3C*, May 2006, http://www.w3.org/2005/02/W3C-Global-Focus.html
- Chesbrough, Henry, *Open Innovation*, Harvard Business School Press, 2003
- Code Ghost. "The Internet Gopher." http://www.codeghost.com/gopher_history.html.
- Daft, Richard; Weick, Karl, *Toward a model of organization as interpretation systems*, Academy of management review, 1984, Vol 9, n.2
- Devetag, Giovanna; Zaninotto, Enrico; *The imperfect hiding : some introductory concepts and preliminary issues on modularity*. ROCK Working Papers. 2001 http://eprints.biblio.unitn.it/archive/00000388/01/DZModularity.pdf
- DiBona, Chris, Sam Ockman, and Mark Stone. *Open sources*. O'Reilly, 1999.
- Elliott, Mark. "Stigmergic Collaboration: A Theoretical Framework for Mass Collaboration." http://mark-elliott.net/blog/?page_id=24.
- González-Barahona, Jesús M. ;Ortuño Pérez, Miguel A. ; de las Heras Quirós, Pedro ; Centeno González, José ; Matellán Olivera. Vicente ; *Counting potatoes: the size of Debian 2.2* http://opensource.mit.edu/papers/counting-potatoes.html
- Gonzalez-Barahona, Jesus, Gregorio Robles, Martin Michlmayr, Juan Amor, and Daniel German. "Macro-level software evolution: a case study of a large software compilation." *Empirical Software Engineering* 14, no. 3 (June 1, 2009): 262-285.
- Grabher G, 2002, *Fragile sector, robust practice: project ecologies in new media*, Environment and Planning A 34(11) 1911 – 1926
- Hayashi Toshihiko, *Fostering globally accessible and affordable ICTs*; International Telecommunication Union, 2003; http://www.itu.int/osg/spu/visions/papers/accesspaper.pdf
- IDC - Press Release - prUS21114208, 2008, *Worldwide Server Market Experiences Modest Growth in Fourth Quarter as Market Revenues Reach Seven-Year High in 2007* http://www.idc.com/getdoc.jsp?containerId=prUS21114208
- Lindman Juho, *Effects of Open Source Software on the Business Patterns of Software Industry*, Helsinki School of Economics Department of Management, 2004, http://opensource.mit.edu/papers/lindman.pdf
- Leiner, Barry M, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, and Stephen Wolff. "A Brief History of the Internet." http://www.isoc.org/internet/history/brief.shtml.
- Leontief, Wassily W., Input-Output Economics. 2nd ed., New York: Oxford University Press, 1986.

- Krogh, George Von, Kazuo Ichijo, and Ikujirō Nonaka. *Enabling Knowledge Creation*. Oxford University Press US, 2000.
- Lorenzen, M. and Frederiksen, L. (2005) *Management of projects and product experimentation: examples from the music industry*, European Management Review, 2(3), pp. 198–211.
- Markus, M. "The governance of free/open source software projects: monolithic, multidimensional, or configurational?." *Journal of Management and Governance* 11, no. 2 (May 7, 2007): 151-163.
- Muffato, Moreno; Faldani, Matteo ; *Open Source as a Complex Adaptive System.* Emergence, 2003, Vol. 5 Issue 3, p83-100
- Narduzzo A., Rossi A., *The Role of Modularity in Free/Open Source Software Development*, in S. Koch (ed by), 2005, http://opensource.mit.edu/papers/narduzzorossi.pdf
- Robles Gregorio, Gonzalez-Barahona, Jesus M. ; Michlmayr Martin; *Evolution of Volunteer Participation in Libre Software Projects: Evidence from Debian;* http://www.cyrius.com/publications/robles_barahona_michlmayr-evolution_participation.pdf
- Rullani, Francesco, *Dragging developers towards the core. How the Free/Libre/Open Source Software community enhances developers' contribution*, December 2006; http://freesoftware.mit.edu/papers/rullani_comsocenv_v5.pdf
- MacLean, Dan; ed., *Internet Governance: A Grand Collaboration*, United Nations ICT Task Force, 2004
- Messerschmitt, David; Szyperski Clemens, *Software Ecosystem – Understanding An Indispensable Technology and Industry,* MIT Press, 2003
- Neff, Gina; Stark, David; *Permanently Beta: Responsive Organization in the Internet Era*, September 2002 http://opensource.mit.edu/papers/neff-stark.pdf
- O'Reilly, Tim, Compact Web2.0 definition, 2005, http://radar.oreilly.com/archives/2005/10/web_20_compact_definition.html
- Perroux Francois, *Pour une philosophie du nouveau développement*, Aubier/Les presses de l'UNESCO, 1981.
- Porter, Michael E., *Strategy And the Internet*; Harvard Business Review, March 2001
- Porter, Michael E., Kramer Mark R., *The Competitive Advantage of Corporate Philanthropy*, Harvard Business Review, December 2002
- Sawhney, Mohanbir, Prandelli Emanuele, *Communities of Creation, managing distributed innovation in turbulent markets,* California Management Review, vol.42, n.4, summer 2000
- Tuomi Ilkka, *Networks of creation*, Oxford University Press, 2002
- Vaughan-Nichols Steven J. *Mr. Dell opens up about Desktop Linux*, March 7th 2006 http://www.desktoplinux.com/news/NS3822185143.html