Open Source vs. Open Data[1]

François Bancilhon

Data Publica and INRIA/ISM

Abstract

This paper addresses the differences and commonalities of open source and open data. It is based on my personal experience and current understanding of both fields. I first briefly recall the definitions of open source and open data, then I discuss the relationship of program and data, finally I focus on open source vs. open data. I do this analysis by listing characteristics and features where they differ or are alike.

Acknowledgments: my warmest thanks to Serge Abiteboul, Claire Gallon and Patrick Moreau for reading an earlier version of the manuscript and contributing many corrections and new ideas, and to Sophie Pellat for suggesting to write the article.

Introduction

I have spent six years running an open source company (Mandrakesoft/Mandriva). These have been difficult and challenging years. I am now running, with co-founder Christian Frisch, an open data company (Data Publica). I hope it won't be as difficult as Mandriva. I find the open data space very exciting and Data Publica is clearly the most interesting company I have run. My shrink (if I ever had one) would probably comment about my obsession of openness.

I asked myself at some point what are the similarities, commonalities and differences of these two fields. I put my thoughts on paper. This was, as usual, a difficult and painful exercise. This is the result.

I briefly recall the definition of open source and open data. Then I look at program vs. data. Finally I discuss open source vs. open data. I pick a number of characteristics and analyze similarities and differences.

Caveat 1: This is not a presentation of open source, nor is it a presentation of open data. I just provide a quick summary for the reader. But I assume she/he is somewhat familiar with both fields.

Caveat 2: I use the term "open source" to cover both free software and open source software.

Therefore, I use it to describe software developed and distributed under some FSF license, GPL or similar. Yes, I know some people will not like this. I just do this because open source vs. open data sounds much better than free software vs. open data.

Definitions[2]

Open-source software (**OSS**) is computer software available in source code form: the source code and certain other rights normally reserved for copyright holders are provided under an open-source license that permits users to study, change, improve and at times also to distribute the software.
Open source software is very often developed in a public, collaborative manner. A report by the Standish Group states that adoption of open-source software models has resulted in savings of about $60 billion per year to consumers.
The Open Source Definition, notably, presents an open source philosophy, and further defines the terms of usage, modification and redistribution of open source software. Software licenses grant rights to users which would otherwise be reserved by copyright law to the copyright holder. Several open source software licenses have qualified within the boundaries of the Open Source Definition. The most prominent and popular example is the GNU General Public License (GPL), which "allows free distribution under the condition that further developments and applications are put under the same license" – thus also free. While open source distribution presents a way to make the source code of a product publicly accessible, the open source licenses allow the authors to fine tune such access.

Open data is the idea that certain data should be freely available to everyone to use and republish as they wish, without restrictions from copyright, patents or other mechanisms of control. The goals of the open data movement are similar to those of other "Open" movements such as open source, open content, and open access. The philosophy behind open data has been long established, but the term "open data" itself is recent, gaining popularity with the rise of the Internet and World Wide Web and, especially, with the launch of open-data government initiatives such as Data.gov.

Programs vs. data

Open source is about programs and programming. Open data is about data, and data management. So let's start with a quick look at programs and data.

Theory and models

A program can be modeled by a mathematical function. It typically takes an input and produces an output. Computer science has produced models for programming, such as the Turing Machine. The Turing machine includes a program, reads data on an input tape and writes data on an output tape. There is a large body of theory about programs, and most of theoretical computer science is indeed devoted to their study. We know many things about programs and many theoretical results are available.

In the Turing Machine model, data are just sequences of symbols, e.g., zeros and ones, written on a tape. This is not a very exciting description of data. A popular way to model data comes from first-order predicate logic. This is the one adopted by the relational model (introduced by Codd and popularized by, e.g., Oracle, DB2 or mySQL). For instance, the data about a person with address, age and revenue may be modeled by a logical statements of the form "John lives in NYC, is 32 and makes $55K a year". In an RDF style, a standard for the semantic Web. This may be captured using three

[2] Those two definitions are straight out of Wikipedia

statements: leaves-in(John, NYC), has-age(John, 21) and makes (John, 55000).

Of course, many other data models have been proposed. Actually, the ability of the programming language community to generate new programming languages is almost matched by the ability of the database community to generate new data models. The religious wars between believers in the various programming languages are of the same strength as those between the believers in the various data models.

Production

Human work is needed to produce a program. Typically, a programmer is given a specification (e.g. “write a program that takes a list of numbers and produces the corresponding ordered list of numbers)”. This specification can be more or less formal, but in any case there is a task description. The programmer writes code, compiles it, tests it, and convinces him/herself that the program indeed performs the task described in the specification. Programs can be right or wrong. The verification process is a highly intelligent task because it is in fact equivalent to proving a theorem. That is one of the things we learn from theoretical computer science. And by the way, this is one of the arguments in support of open source: mathematical knowledge belongs to everyone.

Because work is needed to produce a program, there is an associated cost. Thus a program has an economic value, because there is a cost to produce it and because it may serve some economic purpose.

Work is also needed to produce data. But this work is of a very different nature. Data, like programs have a specification: they represent information from the real world. There are many ways of producing data : identifying the sources, extracting it from the sources, transforming it, cleaning it, etc. Thus data also have some production cost. Thus, like programs, data end up have an economic value, both because there is a production cost and also because they fulfill some useful purpose.

Life

Programs have a life. They have a life because they keep changing over time. They are alive because developers keep changing them for fixing bugs and improving them. Programs also need to be updated to be adapted to new hardware and they use external components that keep changing. A lot of the business of software companies is linked to this life. An important part of software revenues goes into software maintenance and evolution. Maintaining programs alive has a cost, there again it has an economic value.

Data also are alive, but in a different way. Data represent real life facts, and those facts keep changing over time. Meteorological data, traffic data, population data, legal data, economic data, pricing data, etc. keep changing over time at different rate from milliseconds to years. Thus data need to be maintained to take into account those changes. Maintaining data alive has a cost and there again an economic value.

Technology and tools

Both programs and data have technology and tools and they are indeed different

- for programs, we have programming languages, compilers, debuggers, checking, development environments, workflow systems, etc.
- for data, we have database systems, modeling, query languages, query optimizations, big data technology, hadoop, map reduce, NoSQL, etc.

A large part of the difficulty in application design and development lies at the frontier between programs and data. For instance, large scale Web applications (such as e-commerce web sites) rely on two critical large pieces of software: a workflow system (on the program side) and DBMS's (on the data side). A key difficulty is to have them cooperate smoothly. In particular, if we start understanding program verification (as already said, a complex task), we know little about verification when databases are considered.

Source and object

Programs have source code and object code. If you give me object code, I will be able to execute the program, but unless I use some heavy-handed reverse engineering, I will not be able to modify it nor understand how it works internally. Which brings us to the whole open source idea: open source people believe you should give me both the object and the source code (and the right to change it).

Data have no such distinction. There is the so-called physical and logical description of data, but whatever form you give me, I am in general able to figure out the data and use it[3],[4].

Intellectual property and licenses

There is intellectual property (IP) about data just as about programs and the license situation looks very much the same to me.

The license associated to a program or to data is the contract that defines the rights and duties of the licensor (the person who developed the program (or the data)) and the licensee (the person who receives the program (or the data) from the licensor).

Roberto DiCosmo worries that when someone provides a license on a data set, he/she might want to provide a license on the information in the real world represented by this data set. I don't believe this is correct: when you license data, you actually just license the data itself, and the fact that they model the real world information, but not the real world information they attempt to represent. So I can grant or revoke rights to the use of the map of the Paris Metro which I have built, but I cannot grant or revoke rights to the actual facts they represent.

There is a similar relationship between a program and its specification than between a data set and the real world facts that this data set models. So you might license a specific piece of code that implements an array sort, but you cannot license the idea of sorting numbers. This is not in contradiction with the fact that you can license an algorithm (you license the fact that the algorithm performs a specific task). Nor is it in contradiction with the fact that you can license a specific DNA sequence: there again you license the fact that this sequence corresponds to a specific gene, or is specific Aptamer.

[3] The use of PDF could be a counterexample. It is used exactly for this by many people: they put their data (text, tables, etc.) in pdf format so that nobody can change it (which is by the way not true). It is however different from source code. By reading your object code, it's usually very hard to figure out what the program does. By reading your pdf presentation, it's very easy to understand your data.

[4] It can also be argued in other cases: in scientific data, one also explains how the data has been obtained, to validate the data and give you the opportunity to repeat the experience. One can also see something similar in citation (citing sources) and one starts considering, in explanation (explaining some deduction).

Patents

Programs can be patented. This has not always been the case. But since the 80's, people have started patenting algorithms and programs. So now, very large amounts of money are spent on expensive lawyers to develop and fight these patents. Open source people are very much against patents : it goes against their general philosophy and vision of the world, and it is incompatible with the business model of open source.

To the best of my knowledge, data are not and cannot be patented, and I sincerely hope it will stay this way.

Open Source vs. Open Data

I now turn to the difference and similarities between open source and open data.

Age

Open source is a reasonably mature and established movement. The “open source” term was coined in 1998.

Open data is more recent. Even though the fight of citizens for government data, or of human beings for knowledge can be traced back to Adam and Eve (actually more to Eve), the occurrence of the world and the movement goes back to 2009 and is fairly recent.

Open loves open

Most open source believers have embraced open data quite naturally. This is why some open source technology and products are often found in may open data solutions.

Not all open data believers consider that open source necessarily goes with open data. Thus, some open data operations have been based on purely proprietary software. However, if you consider that open data is aiming at cost reduction, data syndication, sharing, accessibility, co-development and transparency, then you could argue that open data should lead to open source.

Standards

Both open source and open data support and requires open standards. Open source favors open standard languages and tools. Open data favors data in open format languages, which allows easier and more open access.

Development

Open source software is a new way of developing and distributing software (collaboratively, transparently and with the source code).

Open data is not a new way of developing data. It is primarily the process of making the data you have available under an open license and re-usable format.

Maybe the closest to open source development in the data world is actually crowd-sourcing: it is done in a collaborative way and the license is in general open.

Public vs. private

Open source has been traditionally orthogonal to the public vs. private distinction: it can apply to software produced by private or by public parties. Initially, the public sector has embraced it more readily than the private sector, but it also widely adopted by private players now[5]. Open source proponents push for a systematic use of open source for all types of programs, public or private.

Open data is currently pushed mainly for public data, and associated to PSI (Public Sector Information). There are good reasons to extend the idea of open data to the private sector and many open source advocates push for this. But, to my knowledge, no one is advocating the idea that all private data should be made public. Open data from private corporation is unusual at this stage, but you can start hearing it here and there. Extending the notion of open data to the private sector makes a lot of sense in a number of situations. But clearly the motivation for the private sector is going to be different : transparency is an image issue, and open data in the private sector will have to be motivated by business reasons, such as developing an ecosystem around your company.

Laws

There are no laws about open source (at least not to my knowledge). The closest one can find are regulations stating that open source should be treated the same way as proprietary software. These were passed following challenges to call for tenders by open source proponents.

Most democracies around the world have or will have laws about open data. And some not so democratic countries have or will have them also. Open data is forced on many administrations and public bodies by laws (or rules or directives). These define what must be open, sometimes under which licenses, and specifies the structure of the open data directories which must be set up. These laws and regulations can occur at the continent (e.g., EU), nation, region or city level (see for instance http://www.fastcompany.com/1701410/san-francisco-passes-first-open-data-law)

Money

I found the same schizophrenic relationship to money in both communities.

Open source and open data activists both have a strong distrust of money and everything having to do with money. Personally, in both worlds I have been looked upon with suspicion every time I explained I was running a company whose goal was to to create value, jobs, revenue and profit. The ultimate sin was: “He wants to make money with open data.”

In the open source space, it is critical to create an economic activity which allows open source developers to earn a living and to bring quality products to the user community. In the open data space, one of the arguments for open data is the creation of economic activity, thus it is essential that we support such economic value creation.

I strongly believe that in both spaces, one can build companies which both respect the base values of openness, bring value and contribute to the economic wealth of the world.

I am encouraged in that thinking by the fact that many entrepreneurs have been able to build healthy businesses in the open source world. In the open data space, I see many new start ups emerging and expect the resulting ecosystem to be thriving.

[5] Even Microsoft is using it now.

Licenses

There are many open source licenses. They have reached a stage where they are understood. There is a large body of knowledge about them. There is a large group of specialists of the issue who can explain, educate and provide consulting. The general speech on the four fundamental freedoms of open source is clear, pedagogical.

In the open data world, things are more recent. Whereas the GPL license has established a reference and a standard, there is no dominant reference license for open data. We are still at the stage of evangelisation and soul searching. Every presentation I have heard describing open data licenses has left me slightly unsatisfied. We are still in the process of producing new licenses (eg the Open License in France[6]) and of proving and explaining things about them.

The issue of Share Alike or Copyleft, i.e., the transitivity of the license is an integral part of the open source model: if you redistribute the open source software, you should distribute it under an open source license. It is not at this stage part of all open data licenses. The ODBL license includes the share alike clause, and I personally think it is a showstopper for business. The difference between GPL and LGPL does not have an equivalent in the data world, because we are able to distribute software in independent modules, while we do not do this about data[7].

Going Open

Several categories of people can use open source : software developers who choose to develop and distribute software in open source mode, users who choose to use mainly or exclusively open source software (linux + libre office + lamp, etc.), re-users who benefits from open source components to develop more software and contributes (or not) by making updates and derived work available to others.

Several categories of people can use open data: users who benefits from the available open data for your own use, re-user who makes use of open data for business purposes, for research or for data journalism, “data openers”, who contribute by making more public data available. These can be politicians prompting their administration to open data, civil servants opening new data sets, some data, activists pushing other players to open data or opening directly through crowd-sourcing.

Motivation

Going open source is making a choice to develop and distribute the software you produce in a specific way. Motivations vary : they can be an act of faith (I believe software should be free and open), a business reason (I believe developing and distributing software through an open source license will help me develop and grow my business), or a strategic decision (I believe contributing to this software will help grow the market in which I am operating), or a matter of visibility (I want to be a guru in a community.)

Going open data is making the choice for those who have developed data sets to make them available to other people. Motivations vary: transparency (government data should be shown to citizen as a way of enforcing democracy), fairness to the taxpayer (data developed with taxpayer money, belong to the

[6] I pick many of my examples in France, because it is the market I know the best, I apologize to the international readers, and will gladly add other examples if they are proposed.

[7] One way to bring the same situation to the open data world is to build “data mash ups” through web services. This forces the data set builder to deliver data sets in a more complex way, but it could be a way to turn around the ODBL problem.

taxpayer), economic enabler (public data can help generate new businesses and enhance the existing ones), idealism (scientific data should be opened to the entire world for the progress of science).

Alternatives

Open source is a way of developing and distributing software: if you develop software, you have the choice of doing it open source or proprietary (with a spectrum of possibilities in between). The alternative to open source is therefore proprietary. In a proprietary program, usage is limited to the running the program, and the ability to see the source code and modify is not given to the user.

Open data is not a way of developing and distributing data. It is the idea of making the data you have developed available to others. So the alternative to open data is data with a license which prohibits reuse, or data which you have to pay for, or data that is simply not available.

Gurus and Activists

The open source space has gurus such as Richard Stallman and some heroes, such as Linus Torvald and Eric Raymond. All of them are geeks, by the way, but all of them public figures and advocates of their vision of the open source movement.

Open data does not have gurus, nor heroes, I have seen many aspiring gurus or heroes, but I don't think they quite qualify yet. Of course Tim Berners-Lee is a guru, but I view him more as a semantic web guru, than an open data guru[8]. Maybe we could nominate Barack Obama and Vivek Kundra as open data heroes.

The open source movement is organized by groups such as the Free Software foundation. In France, because we love to argue, we even managed to have two different groups April and Aful, both doing interesting work.

At this stage, we have seen no such thing for open data. Of course, there are many activist and citizen groups supporting open data such as the Sunlight foundation and OKFN in the Anglo-Saxon world and, Regards Citoyens or Libertic in France, but none of them is claiming at this point to be the meeting point or spoke-person for the field.

I am not advocating that we need such gurus. The groups above seem to be doing quite an efficient job. However, in this days where communication is essential, I do think that one of the most efficient way to make an idea successful is to have it represented by actual people.

Culture

The open source world has the healthy habit of pretty rough language. During my 6 years in that space, I have seen and heard many insults and the Godwin point can be reached very quickly.

The open data world looks more gentle. They don't insult each other. They may be very opinionated and may be quite talented at ostracizing opponents. But they behave socially and I do hope it stays this way.

Open source is an issue mainly of interest to techies. Open data can potentially be of interest to the general public. Ask someone on the street whether every citizen should be able to access public files: chances are this person will say you yes. Ask the same person whether program users should be able to

[8] I hope I don't get in too much trouble for this one.

access the source code of programs and modify it, chances are the person will either not understand the question or not be interested

Opposition

In the early days, the open source movement did face a strong opposition. There were very strong opponents to the idea. Just remember the intense anti-open source lobbying from Microsoft. As time went by and open source software became ubiquitous, the idea became pretty much mainstream and accepted by all.

Open data never had strong vocal opponents. Even those who actually fight it, claim this is a good idea. Of course a lot of people will drag their feet about it. But basically no one really argues that it is a bad idea.

Politics

When I gave presentations on open source, I always asked the question whether the open source movement had a political bias and whether it was more leftist than conservative. And my feeling has been that the answer was yes to both questions, and I used to base this on statements made by various political figures on each side of the spectrum. As open source is becoming more and more mainstream, this is perhaps less and less true.

Of course, because open data is a concern closer to the general public, it will generate interest in wider circles and could become more political. Politicians have discovered that with open data you can look good without spending much. In France, in the socialist primary election, least one candidate made it an issue. It was also an issue in the current presidential election. In the US, data.gov was launched by a democratic administration. In the UK, it was launched by a labor government, but the Tories took it up and kept it moving. In France, nothing has happened till 2011 under a conservative government and the actual opening of data in cities (Rennes, Paris and Nantes) has occurred in socialist controlled cities (but most cities are ruled by the left at this moment). Then in 2011, the conservative government put in place a data.gouv.fr and took several measures in clear favor of open data. So I believe that in most democratic countries, there will be a general consensus around the idea of open data. Of course, this is much less true for nondemocratic countries.

Open data is a tool. Therefore it can be used for a liberal purpose (privatizing public services for instance) or for a progressive purpose (participative democracy). The two approaches converge in the current situations of a state budgetary crunch.

Open source development could be labeled progressive: sharing of capital, and valuing real value add rather than return on capital by charging for the software license.

Producer size

An individual in a garage can produce open source code, he/she is unlikely to build a large and rich data set.

Typically, the rich data sets are expensive coming typically from the government or major players. Communities can (and have) build large data sets. So the model for data sets is more large open source development such as Linux or Apache that small actions.

Business models and ecosystems

The traditional proprietary software business was long established before the open source players stepped in. It included software editors, service companies and users.

On the open source side, four types of members of the open source ecosystems can be found:

- service companies: they provide services around open source software products and components. Some open source services companies have been pretty successful, whether they are pure players (Smile, Linagora, Alterway, in France) or existing software companies that have set up open source practices (Atos Origin, Cap Gemini, Logica, etc.)
- software editors: they develop, maintain and market open source software products. Nothing says that you can't charge for open source software, but in general editors don't charge for open source software. Some open source editors have become quite successful (Red Hat, MySQL, Talend, Jboss, etc.). Several business models have been adopted by open source editors
  - *The simple model*. For proprietary software you sell license, support, training, maintenance, set up and customization. For open source, you sell the same, but the license is free. Most software editors have a tight control over the software they edit, so they are legitimate to provide the software.
  - *The freemium/premium model.* A part of the product is distributed and available under a free (as in free beer) license. That part is sufficient to actually do some real stuff. Then more elaborate versions are available at a cost.
- users and contributors: they use open source software and components, contribute to their development by fixing bugs or by developing contribution which they make available to the community
- users only: they use open source software and in general understand the word “receive” much better than the word “give”.

There is a well established data business. In France alone, it is evaluated at €1.6 billion annually, organized in nine silos, and consisting of about 120 companies, large and small. It is estimated that more than half this data originates from public sources. These data companies usually operate in three phases

1. data collection (gathering data from various sources free or charged, through various means)
2. data structuring (cleaning up, integrating)
3. data analysis or usage in specific applications

The open data community has not yet generated such a new ecosystem.

- In the UK, a group of small companies revolve around data.gov.uk.
- In the US, a number of companies have been launched by the VC community whether they are setting up data market places (Factual, Infochimps, Data Market, etc.) or helping public organizations to put in place their open data policy (eg Socrata).
- In France a number of companies are starting in that space (including Data Publica and OpenDataSoft).

The established business will be disrupted by

- the apparition of new technologies
- the open data phenomenon
- new usage for data (applications and optimization)

My belief is that we will see emerge a new ecosystem, which will include

- service companies that help public and private organizations open their data,
- data production activity, which will be distributed among users producing data for themselves, pure players producing and selling data, service companies producing data for others,
- data market place companies which will organize the sale, acquisition, barter and syndication of data,
- data usage activity (data analysis, applications based on data, data visualization, data journalism, etc.), once again distributed among users and providers.

Conclusion

My conclusion is that the two fields have a lot in common and differences. A lot of differences should tend to shrink when open data matures, but the differences which relates to the intrinsic nature of data will remain. The two fields have enough in common that some lessons learned in the open source space can be applied to the open data space. But they have enough differences that we should be very careful in applying them. Clearly, there are some synergies between them, which we should identify and leverage.