

# Sustainability and Governance in Developing Open Source Projects as Processes of In-Becoming

Daniel Curto-Millet

*“It’s not coordinated, it can’t be, because it’s all left-field stuff. So GPSoC I knew nothing about. And I mean quite honestly that’s the way I would want it to be because I think a thousand blossoms blooming is really the nature where we’re at, at the moment, because we are not going to know exactly where the whole thing will resonate and where it will add value. We had no idea that somebody in Cambodia was going to download Opereffa and build a TB national alert system.”*

An openEHR Board Member

Sustainability is often thought of as a binary state: an open source project is either sustainable or not. In reality, sustainability is much more complex. What makes this project more sustainable than that one? Why should it be assumed in the first place that sustainability is a prolonged state of an ingraced project? The threads are pulled from their yarns in many directions.

This article attempts to reconceptualize some assumed notions of the processes involved in developing open source software. It takes the stance in favour of studying the fluctuant nature of open source and the associated artefacts, not as well-defined objects, but as commons that are continually built upon, evolved, and modified; sometimes in unexpected ways. Further, the governance of these commons is an ongoing process, tightly linked with the way in which these commons are allowed to further develop. This perspective of "in-becoming" is useful in understanding the efforts and processes that need to be provided to sustainably govern the development of open source projects and the advantages for managing requirements derived therein.

## Introduction

Studying sustainability in open source software is complex because the practices sometimes contradict what we know from traditional software engineering (Crowston et al., 2011; [tinyurl.com/alglebm](http://tinyurl.com/alglebm)). There are still few theories and only Elinor Ostrom ([tinyurl.com/pcxroc](http://tinyurl.com/pcxroc)) has left us with something akin to a grand theory of the governance of the commons. By theorizing about this phenomenon using the very tools and paradigms that had previously obscured them, she established the notion of the commons and open source as an accepted and worthy field to study. She did this by giving the field a tradition, a rhetoric, and a history that we could all use to legitimize our own work.

This article attempts to problematize sustainability by contrasting notions of order and structure and by veering away from binary state definitions. In analyzing openEHR ([openehr.org](http://openehr.org)), an open source software project focused on electronic health records (EHRs) and related systems, this article emphasizes that the governance of an open source project *is a verb*, and *not a noun*, and that the way in which the processes are governed (or otherwise managed) is integral to the ongoing sustainability of the project. Using Ostrom's concept of commons and the theoretical underpinnings of *becoming* from Deleuze and Guattari (1987; [tinyurl.com/awujgdr](http://tinyurl.com/awujgdr)), it is argued that the principal commons in open source are the very processes that create those commons, which are not static resources, but *in-becoming*. Finally,

## Sustainability and Governance as Processes of In-Becoming

Daniel Curto-Millet

the concept of sustainability in the governance of open source projects is associated with that of change (Tsoukas and Chia, 2002; [tinyurl.com/b7xbfof](http://tinyurl.com/b7xbfof)). Further, problematizing sustainability as chaotic vectors with potentially unexpected developments reveals efforts that otherwise would remain invisible and could be helpful in governing its processes effectively.

### Methodology

openEHR is an open source project that defines clinically meaningful concepts and describes in which ways these are to be defined. In so doing, it presents itself as a potential standard for the interoperability of EHRs. openEHR exemplifies the new frontiers explored by open source bridging many different expert communities: academics, patients, clinicians, computer engineers, and politicians. Further, it places itself in the still largely unstable area of EHRs (Black et al., 2011: [tinyurl.com/bdogg8r](http://tinyurl.com/bdogg8r); Hayrinen et al., 2008: [tinyurl.com/bz6bqzq](http://tinyurl.com/bz6bqzq); Hovenga and Garde, 2010: [tinyurl.com/a2t96rx](http://tinyurl.com/a2t96rx)).

The case study presented here is part of ongoing doctoral research focused on requirements engineering and development processes in open source. The findings are derived from 10 open-ended interviews with core members of the openEHR project. Most core members are also board members; others have built strong reputations through work on satellite projects and prolonged involvement. The overall objective was to understand what core members understand by requirement processes in open source, leading to questions of governance, sustainability of processes and community, sourcing of ideas, control, etc.

### Why Open Source Governance Should Be a Verb

Typically, it is understood that a project is open source if its license conforms with criteria set by the Open Source Initiative (OSI; [opensource.org](http://opensource.org)). At its foundation, open source is a static, legal definition describing what can be done with the source code (Perens, 1999: [tinyurl.com/27jxjg7](http://tinyurl.com/27jxjg7); Raymond, 2001: [tinyurl.com/9fsvzua](http://tinyurl.com/9fsvzua)). Whether this matters at all to the general public, or whether it has any immediate effect beyond the development team, is a problem known as the "Berkeley Conundrum" (Fitzgerald, 2003; [tinyurl.com/93g4adm](http://tinyurl.com/93g4adm)). It matters when considering the reaches that code as law can have, the specific mechanisms of social control it can induce, and the implications to democratic values (Lessig, 2006; [tinyurl.com/qyzuhj](http://tinyurl.com/qyzuhj)). Lessig's view is more political, less static. His concern turns from legal code to one of con-

sumption, production, and social responsibility. These issues are even more relevant given the new domains, into which open source has entered and which are far away from its academic and hacker origins (Fitzgerald, 2006: [tinyurl.com/9hnxdgv](http://tinyurl.com/9hnxdgv); Lindman and Rajala, 2012: [timreview.ca/article/510](http://timreview.ca/article/510)). When discussing open source and archetypes (abstract representations of meaningful clinical concepts in EHRs), a board member says:

*"When I hear open source I tend to think software rather than knowledge so it's quite different. So the philosophy is, the issue is how do you know when an archetype is good. How do you know, the phrase is, how do you quality assure a model? That your fellow colleagues, the developers, that national governments know that this archetype is safe, doesn't contain manifestly bad practice, whatever. And most people take the view that of a kind of waterfall approach, so you get the great and the good and the wise say what the requirements are, some clever people [...] develop the archetypes, and then we pass this off to a standards body, [...] have some experts, blood pressure experts who say oh yes, yes, yes, that's right, they tick the boxes, they will have some formal criteria against which they'll be marking the archetypes and they will pull in experts, maybe cardiologists. Now I just don't think that's going to happen. I don't think it is possible to know **when an archetype is good enough.**" [emphasis added]*

This quotation evidences the new nature of open source software, away from the code, in the knowledge realm; it shows some of the values and goals associated with using an open source approach (quality through edition, diffusion, and acceptance of the archetypes; open source rivalling with standards bodies as an institutionalizing power; and continual and acceptable change. "When is an archetype good enough?" Who can answer that other than someone following an open source process?

Open source, and the artefacts it engenders, are definitions in the making, processes, arguments, and particular engineering models. The knowledge engendered is not a thing, a static good eventually catalogued, it is potentially embedded in a continual process of being made, of evolving. Given that an open source commons is an ongoing construction that can never be considered "finished", it can be difficult to place a commons in time and ask the question: "When is an open source commons?" To answer the question, and to understand why it is so difficult to answer, we must study the nature of these commons.

## Sustainability and Governance as Processes of In-Becoming

Daniel Curto-Millet

### Open Source Commons Through Open Source Collective Actions

Ostrom's work was framed by the economics of resource scarcity. Notably, one of the spin-offs of her framework helped inspire a framework on social-ecological systems (SES) (McGinnis, 2011; [tinyurl.com/9r5f849](http://tinyurl.com/9r5f849)). How can Ostrom's work be useful in a field where what is abundant or scarce is not one of the usual resources that we think of (Anderson, 2009; [tinyurl.com/a9mkngf](http://tinyurl.com/a9mkngf))?

What is an open source commons? According to the static, legal definitions of open source, the code is the foremost of commons. It is the central artefact to which people are contributing. However, focusing only on the source code is limiting, because it does not take into account the entirety of what Ostrom calls the "action situation", where actors interact and evaluate outcomes (Ostrom, 2011; [tinyurl.com/akfmd3w](http://tinyurl.com/akfmd3w)). In open source, this is not one physical space, but many interrelated ones (e.g., presence in the code, in the mailing lists, in the documentation, in the IRC channels, in the annual conferences, even in the press). It is useful to see that open source is not just online coding, but that it occurs in a wide variety of different media. The rules and engagements are likely to be different in each media, and the ownership of those different spaces depends on various rules and norms of engagement. Ciborra would probably say that these technologies carry different "necessities of hospitality" (Ciborra, 2004; [tinyurl.com/addfljo](http://tinyurl.com/addfljo)).

The notion of an open source commons is also a fleeting one, with the increasing range of domains into which open source is entering (Fitzgerald, 2006; [tinyurl.com/9hnxgdv](http://tinyurl.com/9hnxgdv)). The complexity of what a common is, and therefore, the ownership of those commons is much more complex than it used to be. As an example, openEHR could be said to have several layers of commons. The project's goal is to become a standard in health by defining and creating archetypes that in turn define meaningful clinical concepts. These archetypes are based on a reference model that has become an established standard. The reference model was principally inspired by the efforts of openEHR and other previous projects in which the core members participated. Archetypes are potential clinical requirements for any system that adopts openEHR; they describe clinical concepts such as blood pressure. To define archetypes, the openEHR foundation proposes two editors ([tinyurl.com/byksdmk](http://tinyurl.com/byksdmk)), one from a company with goals closely aligned to the foundation, and another from Linköping University. The editors themselves are based on parsers that understand the Archetype Definition

Language (ADL). When archetypes are drafted, they are placed in the Clinical Knowledge Manager (CKM; [openehr.org/knowledge/](http://openehr.org/knowledge/)), which is a repository of archetypes where they can be discussed, analyzed, reviewed, approved for publication, translated, etc. On top of that come templates, which are supposed to instantiate the deliberately generalized and generalizable archetypes to particular contexts of use.

Now, all of these are resources in the making. All these layers can have their own licensing, and, maybe more importantly, have their own interrelated *action situation*. How could this complexity be managed without undue reduction and simplification? How should these "crops" be studied? What should an archetype look like? Who decides what it should accomplish? Once again, because of the continual, in-becoming nature of knowledge-commons, we fall back to the true commons in openEHR, and in many other open source projects: processes of creation. The processes involved and the knowledge created are so entangled that it is difficult to distinguish the assemblage of actors from the processes they are driving that not only try to reshape the world, but come to a collective understanding of their own collective actions. Since there is no "when" bounding the creation of knowledge-commons to a specific, well-defined time, the next logical step is to study how these knowledge-commons are created, and what are the processes that sustain them.

### The Sustainable Processes: Creating Abundant Commons

Sustainability in open source refers to the project's ability to support itself over time (Chengalur-Smith et al., 2010; [tinyurl.com/ckvgaff](http://tinyurl.com/ckvgaff)). It has already been studied, especially through the lens of the community, free-riding, and project size (Lerner et al., 2000; [tinyurl.com/a8oygl3](http://tinyurl.com/a8oygl3); 2006; [tinyurl.com/9bp78rm](http://tinyurl.com/9bp78rm)).

Recent efforts have looked at processes instead of static commons. Studies have shown that power relations are important in the process of contributing to the source code (Iivari, 2009; [tinyurl.com/a4bsl27](http://tinyurl.com/a4bsl27); 2010; [tinyurl.com/ajheqtw](http://tinyurl.com/ajheqtw)). Also, values, culture, and organizational shifts have been identified as key issues in the adoption of open source into corporate processes (Lindman and Rajala, 2012; [timreview.ca/article/510](http://timreview.ca/article/510); Shaikh and Cornford, 2009; [tinyurl.com/9oyo3hv](http://tinyurl.com/9oyo3hv)). Finally, technology has been seen to play a role in the way it enables collaboration in a distributed scale (Laurent and Cleland-Huang, 2009; [tinyurl.com/a2vh9kq](http://tinyurl.com/a2vh9kq); Noll, 2010; [tinyurl.com/9x8subn](http://tinyurl.com/9x8subn); Scacchi, 2009; [tinyurl.com/ab6ynwx](http://tinyurl.com/ab6ynwx)).

## Sustainability and Governance as Processes of In-Becoming

Daniel Curto-Millet

It is difficult to place a taxonomy to the current study of open source precisely because of the evolving understanding of its complexity. Open source is a negotiated concept, and the processes of creating open source software can be competitive and conflictive, and it can disrupt technologies beyond expert walls. It is *becoming* an abstract political machine, shifting itself to accommodate new ideas, pushing for changes (Deleuze and Guattari, 1987; [tinyurl.com/afzx9r4](http://tinyurl.com/afzx9r4)). Open source becomes a way to diffuse innovation and to act upon it. When asked about the use of open source in developing software in multiple-expert-domain, an interviewee said:

*"Well, you could argue that you don't need open source to build that relationship, but the thing about it is that, if you want to build an ecosystem of clinicians and developers all collaborating around the same software, let's say around the NHS [National Health Service], then it needs to be generally open source, or at least the clinical models need to be open source."*

Open source becomes an enabler and an enactor of ecosystems. Through its links to its rooted academic history, to the hacker folklore which is slowly dissipating, to the corporate worlds it is entering, to the legal definitions that impose obligations and grant rewards, and so many other links, it creates a viable alternative to the development of worldly projections. Some would say that it has created itself as an obligatory passage point, an indispensable question that has to be asked when thinking about developing a new software project (Callon et al., 2009; [tinyurl.com/8zgbqzc](http://tinyurl.com/8zgbqzc)). "Should we go open source?" is implanted in practice, just as the software engineering norm "don't reinvent the wheel" has been impressed into every computer scientist. Another interviewee said:

*"And the rigour bit, for me, in the scientific world, most of physics couldn't exist without open source software, because that's the way people, you know, software is extraordinarily complex, unless you've actually got it in your hand and you work with it, you don't really know. And there's so much software around in the world that nobody really knows that... And it gets sold for millions and millions of pounds and then it turns out to be not what people wanted. We really need the practitioners in the field to be much more grounded."*

This quotation emphasizes another aspect of open source development processes: it is sustainable through the scientific, rigorous, transparent values that it enacts by the publishing of the artefacts. This defini-

tion encapsulates the requirements engineers' philosopher stone: how to build the correct system above building it correctly (van Lamsweerde, 2009; [tinyurl.com/a4d3tl3](http://tinyurl.com/a4d3tl3); Letier and van Lamsweerde, 2004; [tinyurl.com/8n9bj43](http://tinyurl.com/8n9bj43)). In other words, how can the proper processes be employed that will ensure that a useful system is built? This brings the discussion full-circle back to the governing of open source.

### Governing Open Source: Sustainability is a Process

If the processes are so important in defining continuous knowledge-commons that spring out of open source, how should their management be studied? Can they be managed at all? Clearly, knowledge-commons require continuous efforts, but how can their processes be sustainable? Going back to Ostrom's work, the cited interviews lead us to wonder how sustainable processes can be governed in open source. Actors, it is argued, are one of the principal elements.

In institutional theory, a major component of sustainability is the shared meaning given to norms (Ostrom, 2011; [tinyurl.com/aqlanst](http://tinyurl.com/aqlanst)). A shared meaning, even inside open source contexts implies some form of stability. As Schweik and English (2012; [tinyurl.com/9tl3myo](http://tinyurl.com/9tl3myo)) put it: "Institutions – social norms and formalized rules along with established mechanisms of rule creation, maintenance, monitoring and enforcement – are the means through which direction control and coordination can occur." This assertion presupposes an establishment of stabilizing forces throughout open source projects. It is worth wondering to what extent these are accepted, if not debated openly. In open source, even basic and fundamental assumptions are put into question, forming part of the learning process (Dueñas et al., 2007; [tinyurl.com/aytv5x7](http://tinyurl.com/aytv5x7); Shaikh and Cornford, 2004; [tinyurl.com/bzzzjr4](http://tinyurl.com/bzzzjr4)). Also, given the usually informal nature of open source software development, how can the invisible, tacit rules be taken into account (Iivari, 2010; [tinyurl.com/ajheqtw](http://tinyurl.com/ajheqtw))? Are stability and sustainability too strongly associated? Are order and routine erroneously taken as the norm (Tsoukas and Chia, 2002; [tinyurl.com/b7xbfof](http://tinyurl.com/b7xbfof))?

Ostrom's work helps when analyzing institutional norms and dysfunctions in the governance of commons. Through the identification of dysfunctional institutions, we can ponder on the tensions between sustainability and stability in open source. Usually, taken together, dysfunctional institutions give the im-

## Sustainability and Governance as Processes of In-Becoming

Daniel Curto-Millet

pression of instability, and open source can be seen as such a disruptive force (Carlo et al., 2010; [tinyurl.com/an9gdue](http://tinyurl.com/an9gdue)). Could instability contribute to maintaining sustainability? If projects become too stable, they could end up losing momentum. Shaikh and Cornford (2004; [tinyurl.com/bzzzjr4](http://tinyurl.com/bzzzjr4)), found that the learning processes in open source vary depending on the community, technology, code, and even basic, concepts that are questioned but induce collaboration and conflict. Thus, stability is, just as sustainability, a relative concept. This is on par with Ostrom's research, where actors evaluate previous outcomes of an *action situation*, opening the door to much more chaotic and irregular evolutions. When defining actors, Ostrom limits the set to "single individuals or as a group functioning as a corporate actor" (Ostrom, 2011; [tinyurl.com/akfmd3w](http://tinyurl.com/akfmd3w)). This is somewhat limiting, but given that the intended audience were economists, as information systems scientists, we can enlarge the population of actors to other entities, agential or not.

In Ostrom's work, actors play an important role in the governance of projects. The actors form an integral part in shaping the processes of creation. What actors contribute to the sustainability of open source processes, and in what degree? This is a difficult question to answer, and will likely depend on the project. But the list is much more varied than it would otherwise seem. Entities understood as economic resources are much more than static objects devoid of action. Some even attribute sentiments to them (Ciborra, 2006; [tinyurl.com/chqp8rx](http://tinyurl.com/chqp8rx)). Commons (knowledge or otherwise) are not only resources, they are living entities to which are assigned and which themselves assign centres of gravity and inscribed behaviours. Their properties, their beings are infused with materiality. They are well and alive, and they have an enormous influence, despite being "things". Take another quote from an openEHR board member in a recent project board meeting:

*"The analogy that comes to mind is the interaction between publishers and librarians. In the context of librarianships, you have national repositories [...] you have some kind of governance framework around the numbering and cataloguing [...] and you have an ecosystem of publishers. You need a new kind of governance which recognises the curation, the librarianship, the skills, is an analogy related to books, there's going to be a correlate of that in the context of archetypes, templates, and there's also going to be a world of publishers."*

The recent debates in the publishing world have provided an analogy to explore and understand how the project itself could or should evolve. It is a new exploration to different types of worlds that appear unexpectedly open. Who could have thought at the beginning of the project that it could learn from the publishing industry?

The knowledge-commons mentioned in this quote (i.e., the archetypes and templates) are thought to be static resources, which, in a matter of seconds, are disembodied, reshaped, and reformed into academic papers, peer-reviewed journals, processes of governance, and curation of books that are seen to merit their emulation. The properties of these knowledge-commons are so flexible in their definition and their processes, that they escape the static view attributed to actor-objects to such an extent that they evade the boundaries between objects and subjects. This is relativism. The knowledge-commons, what they are and what they could be, tear and pull at the project members, influence their view on their governance, and even demand curation.

In this sense, governance is not merely accepted and established institutions, rules, and norms, but also projections of possible worlds, competing values that define the project's essence, historicity, past arguments, motivations, and many other fleeting concepts (Scacchi, 2002; [tinyurl.com/bqcwrgn](http://tinyurl.com/bqcwrgn)). And hence, the difficulty to understand open source projects only through the evaluation of outcomes, when the worlds that are projected are so difficult to grasp. What IP license should be applied to what artefacts? What effect will the licenses have on the uptake by future community members? These evaluations depend, not only on outcomes, but also on the chaotic projections of possible worlds; they may lead to a positioning of the project in some way or another, yet will remain malleable and subject to change. What might be interesting to study are the efforts to cement those evaluations, to create those fleeting institutions, or as Callon would put it, to objectify them (Callon et al., 2009; [tinyurl.com/8zgbqzc](http://tinyurl.com/8zgbqzc)).

Thus, the evaluation and exploration is not without consequences. Opening the project to the outside and questioning its internal processes is crucial in the sustainable governing of open source. The actors, intentionally or not, initiate a tentative alignment with possible worlds, embracing uncertainty so as to better cope with it. Does openEHR have the necessary mech-

## Sustainability and Governance as Processes of In-Becoming

*Daniel Curto-Millet*

anisms to "curate" archetypes? What would a curated archetype look like? What are the processes that need to be implemented for allowing the archetypes to fit unknown requirements? Is the project relevant in this new world? A seemingly invisible negotiation takes place to align the project to possibly relevant new actors.

### Conclusion

This article tried to problematize the nature of sustainability in open source software. Given the open properties of the knowledge-commons present in open source, their management is much more amenable to changes. Continuous efforts are made to sustain the project and adapt it to the demands of the changing environments, sometimes questioning basic concepts, their purpose and meaning. Sustainability, therefore, is much more than a simple binary state, but the continuous efforts of assemblages of varied actors (technologies, democracy, quality through diffusion and adoption), unexpected alliances (publishing world), collective efforts, and values. These actors are crucial to the sustainable governing of open source. Their efforts and enactments, supported by their values, forces open source projects to explore worlds it had not imagined, in turn questioning its own place and purpose, and the adequateness of its processes.

Sustainability in open source, therefore, is not a one-time buy-in option. It is a continuous process of engagement, of negotiation of even basic concepts. Sustainability depends on the evolution of commons created by project communities. New communities can, sometimes unexpectedly, become integrated into the project, redefining the contexts of use, processes, and purpose. The project, then, is introduced into arenas that were not anticipated. In this article, I have attempted to make apparent that the development of open source projects needs a considerable amount of rethinking in terms of governance, which can only be achieved through the understanding of specific open source concepts: principally, the *in-becoming* nature of commons and underlying processes of development, particular values, fluctuating contexts of use and boundary-less sourcing of requirements.

### Acknowledgements

The author would like to acknowledge the support and cooperation he has received from the openEHR board and other members of the project. People approached have been universally accessible and candid in providing information. This openness has been essential in conducting this research. The author also would like to thank the Fundación Ramón Areces for their financial support.

### About the Author

**Daniel Curto-Millet** is a PhD student at the London School of Economics and Political Science. His research focuses on studying requirements engineering and innovation in open source contexts from new perspectives. He has presented his work at a number of international conferences including the Academy of Management conference and the European Conference of Information Systems Doctoral Consortium. He has a background in Software Engineering from University College London and has worked for the DG DIGIT of the European Commission.

**Citation:** Curto-Millet, D. 2013. Sustainability and Governance in Developing Open Source Projects as Processes of In-Becoming. *Technology Innovation Management Review*. January 2013: 40-45.

