Designing for an open source culture

Joel Huang

Abstract— In this paper, we discuss the design philosophies, challenges, and strategies needed to create and sustain an open source culture, examining its promises and analyzing its perils on the on-campus, university community as a microcosm of larger institutions, in the hopes of expanding this culture far and beyond its walls.

Open source and its related ideas often sport a technological front, but have deep, embedded socio-political and ethical complexities. We look at two areas where open source culture could make a huge difference, in governance and in knowledge, and examine how open source culture could complement the rigidities of existing closed and ordered hierarchies, how legacy structures can accommodate the push towards openness, and how we can begin to integrate open source designs in these systems.

I. THE MOTIVATIONS BEHIND OPEN SOURCE

The open source movement has its roots in the late 1980s, starting with the release of the GNU software project by Richard Stallman. It was built around the idea that source code (code that can be compiled into computer programs), should be freely exchanged and modified, letting anyone contribute to its development, modification and improvement.

It is a movement that could only have been born and grown in the Anthropocene - out of resistance against, and grown by the technologies of the computer megacorporations of the 1980s. The US Copyright Act of 1976 allowed manufacturers to retain the exclusive use of software on their computers. They stopped distributing source code, and started using copyright and restrictive software licenses to limit or prohibit copying and redistribution, a trajectory that would threaten to make closed-source the norm.

II. WHAT EXACTLY IS AN OPEN SOURCE CULTURE? WHY IS IT IMPORTANT?

A. What qualifies as open source?

Open source follows the incomplete design paradigm. Garud et al. [1] write that incompleteness acts as a trigger for what they term generative engagement. Generative in two ways: in the creation of new ideas by the empowered community, and creation of new avenues of engagement via existing avenues of engagement, which in turn attracts diverse contributors to join the community.

Julien Pénin describes a generalization of the open source model beyond the realms of software and computers. He places open source innovation at the far right of a spectrum of openness and interactivity [2], in contrast to closed innovation on the extreme left, and open innovation [3], which sits between the two. The distinction is that open source innovation features a continuous and collective knowledge production and exchange, while information flow in open innovation is semi-permeable and sporadic.

Once distinctive to the software world, open source has since become a part of the broader notion of open collaboration [4]. More than its technological front, a flux of political and ethical ideas churn deep beneath the surface of the open source movement, always guiding and transforming its systems and communities. Within this flux, a number of grand, perhaps lofty ideas, emerge as enduring concepts the community has come to recognize and embrace as core to the philosophy of open source. Technological or not, open source design seeks to align itself with these ideas. Open source systems are designed to accelerate progress through collaboration and free exchange, sometimes decentralizing power. Governed by mechanisms that make them transparent and trackable, participation and selection of tasks are made voluntary [5], and conversations are open to everyone in the community.

B. Open source ideals and promises

Is open source like anarchy, but in a good way? David Fidler [6] draws parallels between open source ideals and anarchy: a sort of constructivist interpretation might be useful to see that community agents can directly affect the nature of the movement. "Ideational ebb and flow", he writes, "not material power and capabilities, determines the nature of anarchy". In other words, its malleability arises from its transitional and incomplete quality, and change emerges from the interactions of its agents. Open source might partake in the ideas and qualities of Fidler's anarchy, but cannot be reduced to it; open source fails as a system should it ever turn anarchistic. Fundamentally, open source processes may be regarded as open and incomplete systems that necessitate a certain degree of order designed into them.

Open source promises to be adaptive, evolutionary and representative of its community. In his recent resignation letter, Alibaba founder Jack Ma highlights the importance of design for a future without the self. This enables continuity: future generations are likely to enjoy the same experiences, and since generative engagement is possible, they are able to embrace and facilitate change for the generations after them - not just generations of families, but of people within institutions and organizations with finite terms; nobody stays forever.

C. Pitfalls of open source

Open source systems contain distributed production processes, but the distribution of contribution is hardly homogeneous. With the Linux kernel's development as an example,

some hundred central members contribute most of the code, while a few thousands of peripheral collaborators contribute intermittently and indirectly. Within those circles that collaborate, ethnic, gender and class diversity is uncommon. We also must acknowledge that designing open source can be wicked. Continuity is both the motivation and the enemy of open and incomplete design. With open source meta-architectures, systems architecture is slightly more flexible, but should there be a need to overhaul the system, the inertia that we might choose to design into this system can be irreversible.

D. The technicalities of open source

Open source culture empowers and is reinforced by its technological manifestations. A stellar example - from a university, unsurprisingly - is NUSMods [7], a student initiative to build a timetabling platform at the National University of Singapore hosted on GitHub, a code-hosting platform popular with the open source community. The medium-sized project has been ongoing for some six years. There were 2,600 commits (changes to source code) since August 2012 from 49 contributors. In total, 460 issues, or discussion threads, were raised. Common users are invited to visit the developer community through a link at the footer of the page, clarifying the open source nature of the project.

III. DESIGNING TOWARDS OPEN SOURCE CULTURE

A. A brief on the OpenSUTD movement

We started the OpenSUTD [8] movement in April 2018, with a vision of an open source culture at the Singapore University of Technology and Design (SUTD). The movement aims to create a culture of open collaboration and co-creation within the University. A study on the difference between current and desired cultures used the Organizational Culture Assessment Instrument (OCAI) to achieve community consensus on the prevailing and desired organizational culture. While the current culture was found to be hierarchy, respondents from all across the hierarchy desired a culture of clan and adhocracy [9]. Independently, our team noticed similar sentiment during our early conversations and participatory research efforts. The community expressed a desire for change, and adhocracy - more decentralized processes across bureaucratic lines to capture opportunities and solve problems [10].

How could this knowledge be useful in designing culture? Attempting to build an overlying open source culture is particularly hard because each design towards it is open-ended, incomplete, and overlaps in the intention to motivate the generation of new goals and directions [11]. As a start, by examining the functions and prevailing culture of the University, we decided to design for open source governance and knowledge, realizing that these were functions the community could rally around.

B. The reality of open source

In the following sections, we look at specific real-world examples of open source in governance and knowledge and examine how well their strategies are able to complement the existing, and contribute to building an open source culture, in general and in institutions like SUTD.

C. Open source in governance

In a groundbreaking move towards open legislation, the Washington, D.C. Council began hosting the digital source for its laws on GitHub. The hosted material is not merely a copy of the DC law - it is an authoritative source [12]. The big change here is in the way the law can be accessed. Amendments to laws are made through changes to the digital source, and published in ways better and faster than ever before, allowing citizens to see changes in the laws that govern them as they are being amended. Software and systems can now automatically adapt to changes in laws by directly referencing the digital source. However, it is worth noting that behind the change was a five-year long joint effort between activists, lawyers and the Council to de-commercialize the source from contractors that were naturally copyrighting it, a task made easier by an extremely receptive Council and its lawyers [13].

But why should institutions introduce open source governance? It would be audacious to suggest that anyone - even the majority - could contest and rewrite policy anytime they choose! Our institutions and politics cannot accommodate a change this huge, and how then, would we govern, and be governed? Ultimately, to understand why we should introduce open source governance, we must recognize that there exists a design failure in the status quo, and dispel the notion that integrating elements of open source in how we govern is equivalent to anarchy. Institutional policy seeks to govern by regulation, but in doing so, suffers from rigidity, bureaucracy, and misplaced intentions. We want to move away from playing catch-up and from tying up loose ends, to a adaptive, well-fitted policy that truly represents the needs of the community, while recognizing that this brings not only existing power structures, but deep socio-political ideas like community membership, identity, and social contracts into conversation. In other words, the design problem here is to make policy and discourse (in the Foucault sense) more adaptive through open source mechanisms, while not undermining their power and purpose. The overall result should align with the purposes of the organization, else there would be little point in implementing it.

The University is the best place to demonstrate how institutions may begin to adopt open source policy, priding itself as a center of co-creation, frequent knowledge sharing, and a curiosity for the contrary. Institutional policy design has always been rigid, contemplative and prescriptive, and for good reason; but for institutions and organizations willing to break the monolith and embrace participatory policy design, legislation and policy can be worked on as if it was a piece of open source software - the technology to facilitate rapid, accountable open design is now here, although packaged with a complex, interleaved web of political and ethical issues that we have to design for.

Surprisingly, closed, hierarchical systems might actually find it easier to adopt open source policy than existing, more open, systems. From our experience with prototyping OpenSUTD as a knowledge repository, we found that for the project to remain at healthy levels of contribution, collaborators must be engaged and made to feel their contributions matter. Here, the need to facilitate engagement is less necessary, as policy directly engages individuals through its impacts, and the value of contributions to the community is immediate and apparent. As both J. Lee's [9] and our research has suggested, communities in closed, ordered organizations might desire a move away from a hierarchical culture to adhocracy. This makes the fact that power has to be decentralized less of a concession - some Taylorized, bureaucratic efficiency is traded for adaptability and community engagement. In order to maintain legitimacy of policy, sources must be kept authoritative, and there must be a clear distinction between those who can suggest change, and those who can approve change. In our case, the existing hierarchical structure of the University makes it easy to introduce this separation. To truly be able to scale and manage the numerous policies that an institution might pass, critical and repeatable tasks such as review and tracking and license compliance should be automated through the use of technological tools [14]. A caveat is that under this framework, community discussion on policy is never filtered, despite its tendency to be anecdotally and emotionally driven. Policy managers must address issues systematically and transparently, even for these cases.

D. Open source in knowledge

Openness in knowledge is based on the belief that free access to knowledge should not be restricted by a lack of opportunity - we look at some examples of efforts by institutions and individuals towards open knowledge. In the space of knowledge sharing, free digital course repositories have emerged in the last two decades, projects like MIT OpenCourseWare, Stanford Online, and numerous others, providing legacy materials to students worldwide.

Thousands of authors are starting to bypass traditional publishing to write open textbooks in the spirit of knowledge sharing. A small number of authors choose to directly publish textbook copies or digital source, but most choose distribute through print as well, which makes it illegal for the source to be shared - a recent textbook on Deep Learning [15] by Goodfellow, Bengio, and Courville was published in its entirety online, in addition to its print release.

Cornell University's electronic print archive arXiv [16] provides an open-access, moderated repository of scientific research to authors and researchers worldwide. There has been a tremendous shift in the computer science, engineering and mathematics fields to publish pre-print before they are officially accepted and published in journals. This makes the research community very agile and ensures new knowledge is published to the world the moment the research is complete. Version updates are then tracked, and edit history can be viewed by anyone.

If we consider Pénin's trichotomy of openness [2], these efforts seem merely to be achieving open knowledge and not open source knowledge - knowledge transfer is monodirectional and sporadic. In the space of open source knowledge, Wikipedia cannot go unmentioned. Wikipedia is open source on two layers, the more obvious being that all knowledge on Wikipedia is crowd-contributed, and the less being that the technological architecture of Wikipedia is also open source. This is how internet communities and fandoms easily co-create knowledge bases, or wikis, for the topics they love. A common criticism of open source knowledge bases is that knowledge is of poor quality, with back-and-forth editing and simple fixes prioritized over complex ones because of the way edits are credited. Knowledge on open source bases is not authoritative and should not be evaluated as so [17], a contrast to the slightly official nature of knowledge in the three examples of open knowledge efforts.

We hope to introduce our own flavor of open source knowledge, transforming mere publishing and sharing into a community orchestration of knowledge, balancing authoritativeness and quality with the benefits of co-creation. Similar to the design of open source policy described above, we again need the moderator-contributor paradigm to impose constraints on the openness of the system. The University is no stranger to this concept, already having entire live courses with evolving material moderated by its professors, including classes on design theory and ethics (Prof. Jeffrey Chan), blockchain technology (Prof. Paweł Szałachowski) and data science (Prof. Dorien Herremans).

It seems that incomplete systems must, at very least, incorporate some sort of design trace system in order to be feasible. During one of our open discussions, Assc. Provost Pey Kin Leong and Prof. Max Colla were interested in the ideals of open source knowledge - Max fondly remembers "muddy notes" during his time in school, where a student would annotate notes and photocopy them, and another student would photocopy those notes and add their own annotations, ad infinitum. Clearly, this also means that things could quickly go out of control, if editors had malicious intent to deface or to remove content that others have painstakingly contributed. Thankfully, there is a technological solution in Git, a version control system invented by Linus Torvalds, which enables the systematic management of branches, changes and reversions, even allowing large repositories and multiple files to be edited simultaneously by any number of users while maintaining version integrity.

A repository of open source material was posted on GitHub in April 2018, as an underground validation experiment. This was publicized within a sophomore Computer Science cohort of about 120 members. There was strong contribution over 2-3 months, where contributors were actively engaged in uploading, curating and modifying knowledge material for different courses they had previously taken using the Git version control system. However, when we stopped engaging these early collaborators, contribution rates dropped drastically. We theorize that there needs to be a two-tier mechanism, where faculty, teaching assistants, or course

alumni in the University are the stewards of the material, and assist in curating and providing direction for specific pieces of knowledge. Currently, there exists no design for engagement within the open source knowledge system, and neither moderator nor collaborator are incentivized to make the first move, unlike the open source policy system. The need for champions, community architects, and evangelists within the community is critical to raise awareness of the potential promises of an open source knowledge base.

IV. Conclusion

We have discussed the design of two open source systems as part of the overall effort to create open source culture in a university. If successful, implementing open source policy as an open governance mechanism makes policies visible, and easy to fact-check. A robust system with a community of managers and collaborators are able to curate, debate, modify and evaluate policy at the source. On a systemic and cultural level, participatory design of policy puts us a step up on Arnstein's ladder [18]. An open source knowledge system creates open access to knowledge material across student or faculty roles, across pillars or departments of the university, and across time; and the sense of continual contribution from everywhere in the community.

If we are correct in our assessment that there is sentiment in most ordered hierarchies that adhocracy is desired, we can start to create alignment around what wild success looks like, facilitate change, and apply this to institutions in general.

Open and incomplete systems often require facilitation, a highly unintuitive method to strategists working in predictable and closed systems, but a routine task for designers. In strategy, planning is kept short and based on optimization and full knowledge of the problem space. Facilitation for change instead demands clarity on the personal motivations of stakeholders, while never fixing on a goal definitely. If open source culture is indeed a possible direction for existing closed hierarchies, perhaps a new evolutionary line of the institution will emerge in the foreseeable future, one that places the community and the ideals of change and design at the center of its development and progress.

References

[1] Garud, R., Jain, S. & Tuertscher, P. (2008). Incomplete by Design and Designing for Incompleteness. Organization Studies, vol.29, no.3, pp. 351-371. doi:10.1287/orsc.2013.0872

[2] Pénin, J. (2011). Open source innovation: Towards a generalization of the open source model beyond software, Revue d'économie industrielle, vol.36, pp. 65-88.

[3] Chesbrough, H. (2003). Open Innovation: The New Imperative for Creating and Profiting from Technology.

[4] Levine, Sheen S., & Prietula, M. J. (2013). Open Collaboration for Innovation: Principles and Performance. Organization Science, doi:10.1287/orsc.2013.0872.

[5] Weber, S. (2004). The Success of Open Source. Harvard University Press.

[6] Fidler, D. (2008). A Theory of Open-Source Anarchy. Indiana Journal of Global Legal Studies, vol.15, issue 1, article 11.

[7] NUSMods. https://github.com/nusmodifications.

[8] OpenSUTD. https://github.com/opensutd.

[9] Lee, J. (2011). A Computer-based Group Discussion Support Tool for Achieving Consensus and Culture Change using the Organisational Culture Assessment Instrument (OCAI): an Action Design Research Study (Doctoral dissertation). Retrieved from https://ris.utwente.nl/ws/portalfiles/portal/6050331/thesis_J_Lee.pdf.

[10] Waterman, R. H. (1990) Adhocracy: The Power to Change.

[11] Simon, H. A. (1996). The Sciences of the Artificial. Third Edition. Cambridge, MA: The MIT Press. [Pp. 162-164]

[12] Tauberer, J. (2018). How I changed the law with a GitHub pull request. Retrieved from https://arstechnica.com/tech-policy/2018/11/how-i-changed-the-law-with-a-github-pull-request.

[13] Delgadillo, N. (2018). GitHub Is A Programming Website. So What Is D.C.'s Legal Code Doing On There? Retrieved from https://dcist.com/story/18/11/28/github-is-a-programming-website-so-what-is-d-c-s-legal-code-doing-on-there/

[14] TODO Group. https://github.com/todogroup/guides/blob/master/tools-for-managing-open-source-programs.md.

[15] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press. Retrieved from https://deeplearningbook.org.

[16] arXiv. Cornell University Library. https://arxiv.org.

[17] Carr, N. (2005). Let Wikipedia be Wikipedia. Retrieved from http://www.roughtype.com/archives/2005/12/let_wikipedia_b.php.

[18] Arnstein, S. R. (1969). A Ladder of Citizen Participation. JAIP, vol. 35, no. 4, pp. 216-224.