

Analysis of a Discourse Community

Colbyn Wadman

June 1, 2020

There exists a discourse community quite unlike anything you have probably seen from past students. Still in it's infancy, this discourse community has nevertheless become a foundation of the IT industry. As it matures it will become a cornerstone of efficiency gains in all STEM fields. As our collective knowledge of all technical matters get reduced to mere functions -mapping problems to solutions- this discourse community will become the librarian, troubleshooter, curator, documenter and coordinator of such knowledge.

This discourse community -the printing press of computable knowledge- is the rather humble open source community and industry.

For a brief overview for the layperson: the open source community -as the name implies- is a community built upon open source software, where open source software may be regarded as software built from publicly accessible source code. The open source community may be regarded as the 'human aspect' of open source software. As in communication facilities between the producers and consumers of such software. As well as mechanisms for

updating the source code of such software.

My journey in the open source industry began when I worked at a former startup called upLynk (now absorbed into what's called Version Digital Media Services). At upLynk there was a rule of only using free and open source software, as is common in the IT industry^[2]. This wasn't necessarily because it's free (the company was immensely profitable as is the industry overall), but due to a multitude of factors. For instance, open source software can be a means of industry standardization. Furthermore, for software that upLynk relied upon, my boss even encouraged me to submit any bug fixes I found in such software back to the original author. This is because it's generally in the companies interest to do so, because otherwise their version of this software component will be 'out of sync' with the official version, so any updates from the 'official version' will therefore need to be merged into this nonstandard version, if they wish to benefit from such updates^[5].

Overall the open source community is informal and decentralized. Nevertheless according to GitHub, in the last year alone I have made 1,281 contributions (at the time of this writing) to the open source community at large^[6], while in e.g. 2019 I made 1,323 contributions. These contributions may be a multitude of things, but the predominate contribution will be source code freely released to the world for others to build upon.

To put things in perspective from December 2016 to December 2017, there is a statistic counting total developer contributions on GitHub^[9]; the 256th person contributed 1322 total contributions^[9], I literally beat the 256th person by a single contribution.

So let us conclude that I am certainly a member of the open source community,

and am very active therein. So from my rather unique background and experience, let us proceed to the my thesis.

I propose that the open source community is in some respects, akin to the printing press, for they both facilitate widespread use of their respective mediums^[1]. Furthermore I propose that the production of abstract intellectual goods, when facilitated by the open source community, facilitates emergent phenomena that therein leads to greater sophistication of such products over time.

To explain myself regarding emergent phenomena, consider the following hypothetical model of a system of processes, where each process is a cell in some space (the following are rules for some cellular automata):

- Builder cells take neighboring discrete units and produce non-discrete units
- Abstraction cells take neighboring non-discrete units and produce discrete units
- Unit transitions represent development of a new unit; some development will stall unless a given prerequisite development has been made by neighboring units
- If neighboring conditions aren't satisfied the the cell stalls.
- Cell types can randomly swap places
- physical layout is never optional, so most activity dependent upon such will be stalled.

Physical layout and related settings is based on the starting configuration of the system.

(Abstraction is integral to unit transitions, as I believe abstraction is integral to our modern world, due to it's inherit complexity.)

We can interpret this model as a system manifesting “emergent phenomena that therein leads to greater sophistication of such products over time”: discrete units build upon other discrete units and therein produce more complex non-discrete units. The system permits for abstraction in some manner, so complex non-discrete units may be abstracted into simple discrete units. Thereafter this process of production and abstraction enables further production and abstraction and so forth. Each iteration or generation may be considered to be more sophisticated than prior generations, given that each generation is a product of prior generations. From this analogy, you can imagine these bottom up and cumulative processes will eventually give rise to very sophisticated products, and perhaps one day, akin to how emergence gives rise to the complexity found in nature.

From personal experience, my [imager.io](#) project wouldn't be possible without the various open source components it's built upon. Simply because my time is finite, and especially because lower-level encoding details is just too complicated for me to understand and implement on my own. I am nevertheless able to compose such components into a larger and more sophisticated end product, from the preexisting output of resources and information from the global open source community. Overall added value that may be considered to be greater than the sum of its components, and therefore emergent in a manner of speaking.

Furthermore given the above example, I am a mere node in my hypothetical cellu-

lar automata. What is important is not any given individual, but the actions from millions of such cooperating individuals. Again, consider my example at the scale of millions of other developers working in a similar manner.

Furthermore, given the generality of my model, the processes may be regarded as labor from any STEM field, and the units as abstract intellectual goods or products. For I propose that most STEM fields will eventually become acquainted with the open source community in some form or fashion. Be it a biologist sharing automation routines for otherwise monotonous processes in a lab, or some other biologist asking for help, trying to figure out how to apply the automation scripts the former biologist wrote.

As I begin to conclude this essay, consider this: while a book may describe a life's work in mathematics and abstractions therein, the medium is itself rather passive. A book may describe a life's work in applied mathematics, yet a mind is required to manifest it's application. Whereas, imagine a medium where the most knowledgeable of experts can record their understanding of a given domain as functions that map problems to solutions, in a manner that can be utilized by any layperson, and thereafter this record can be reapplied, reused, and so forth, forever thereafter. This is software, while the open source community is what facilitates the overall adoption, use, and further development of such products.

Furthermore I propose that this is significant -to society and therein ourselves- because humanity is quite finite, and it's production capacity is likewise finite, and since humanity must produce that which it wants, humanity is therefore bounded. We are just as bottlenecked as the abstract processes in my model. Nevertheless just as the industrial

revolution introduced a force multiplier of human muscle, so too does abstraction introduce a force multiplier of the human mind.

So in conclusion: I propose that there exists a discourse community quite unlike anything you have probably seen from past students and furthermore this discourse community will effect your life in perhaps perhaps unimaginable ways, and this discourse community is the open source community and therein: the production of abstract intellectual goods, when facilitated by the open source community, facilitates emergent phenomena that therein leads to greater sophistication of such output over time.

This essay has been open sourced, and is available on GitHub:

<https://github.com/colbyn/DC-Essay>

- [1]: Schrape, Jan-Felix. (2017). Open-source projects as incubators of innovation: From niche phenomenon to integral part of the industry. Convergence. 10.1177/1354856517735795.
- [2]: <https://techcrunch.com/2019/01/12/how-open-source-software-took-over-the-world/>
- [3]: <https://www.zdnet.com/article/its-an-open-source-world-78-percent-of-companies-run-open-source-software/>
- [4]: <https://heartbleed.com>
- [5]: <https://security.googleblog.com/2014/01/ffmpeg-and-thousand-fixes.html>

- [6]: <https://github.com/colbyn>
- [7]: <https://github.com/colbyn?tab=overview&from=2019-12-01&to=2019-12-31>
- [8]: <https://github.com/imager-io/imager/graphs/contributors?from=2019-10-27&to=2020-05-24&type=c>
- [9]: <https://gist.github.com/paulmillr/2657075/>