

TIM Lecture Series

The Business of Open Source

Michael Weiss

“*Today, if you don't think of open source as part of your technology business, you are doing something wrong.*”

Michael Weiss
Associate Professor, Carleton University

Overview

The TIM Lecture Series is hosted by the Technology Innovation Management program (carleton.ca/tim) at Carleton University in Ottawa, Canada. The lectures provide a forum to promote the transfer of knowledge from university research to technology company executives and entrepreneurs as well as research and development personnel. Readers are encouraged to share related insights or provide feedback on the presentation or the TIM Lecture Series, including recommendations of future speakers.

The seventh TIM lecture of 2013 was presented by Michael Weiss, Associate Professor in the Technology Innovation Management program at Carleton University, who examined the business of open source, with a focus on common patterns followed by open source businesses. The event was held at Carleton University on December 12th, 2013.

In the first part of his lecture, Weiss provided an overview of the business of open source, described the key elements of early-stage open source businesses, and presented common patterns followed by open source businesses. In the second part of the lecture, he closely examined late-stage open source businesses, the impact of licensing and architecture, and what the future may hold for open source. The slides from this lecture are available here: tinyurl.com/loz5yof

Summary

Open source has become an integral part of commercial software development. Whereas in the past, open source software development was considered to be

driven by volunteer effort, today most of it is carried out by companies. How companies leverage open source ranges from the adoption of open source development practices, the use of open source development tools, and the integration of open source components into products to active contributions to existing open source projects, and the initiation of their own company-led open source projects. Open source furthermore enables companies to collaborate on the creation of common assets that they can jointly use in product development.

An open source business is a business built around an open source offer. Thus, open source is not by itself a business model; rather, it is used by the business as a strategy to strengthen its business model (Bailetti, 2009; timreview.ca/article/226). Typically, open source businesses use open source to: i) develop new products; ii) build products or services around open source offers; iii) initiate their own open source projects; or iv) leverage open source as a form of co-opetition (i.e., cooperation with competitors). Indeed, close study of the way in which businesses have leveraged open source has led to the identification of common patterns used by open source businesses.

Patterns are proven solutions to common problems, and have been popular in the fields of architecture and software design. More recently, they have also been used to document business strategies, including those used by open source businesses. The patterns that Weiss described in this lecture aimed to provide entrepreneurs, managers, and students of business models with a language for creating new business models around open source, and for incorporating open source into existing business models. Some of these patterns are unique to the stage of a company's engagement

TIM Lecture Series – The Business of Open Source

Michael Weiss

with open source (i.e., from primarily just using open source assets, to contributing to projects, to championing particular development projects, and finally to collaborating strategically within the ecosystem), whereas a given pattern relating to licensing and architecture typically may be used in all stages (Figure 1).

Examples of open source business patterns discussed in the lecture include:

- 1. Bootstrap:** This use-stage pattern refers to the reuse of existing open source components to develop products. This pattern allows a company to shorten the time it takes to create a first version of their product, while keeping costs low; however, it also increases the complexity of the software and the breadth of knowledge needed.
- 2. Contribute back:** By contributing resources (e.g., code, people, money) to the projects they use, companies can: i) build trust with the community, ii) influence the development of the project, and iii) demonstrate their competence.
- 3. Credible promise:** When championing a project, building a critical mass of functionality into the project from an early stage can mobilize contributors to a company's project – it helps the company demonstrate that the project is doable and has merit.
- 4. Feed the community:** A company can build legitimacy with a project community by nurturing the community without any expectation of immediate return, for example by: i) giving to the community (e.g., contributing code, writing documentation, participating in the discussion forum); ii) establishing a clear licensing practice; iii) establishing a clear process for making contributions; iv) making decisions in the open; and v) not treating community members as prospects.
- 5. Sell complements:** An open source product can be monetized through the selling of services (e.g., hardware or support) that complement the open source product.
- 6. Run a tight ship:** To keep control of a project's direction, companies often retain full ownership of the code.
- 7. Dual product (open core):** To entice commercial users to pay for an open source software product, a company may sell a commercial version of the open source produce with exclusive features.

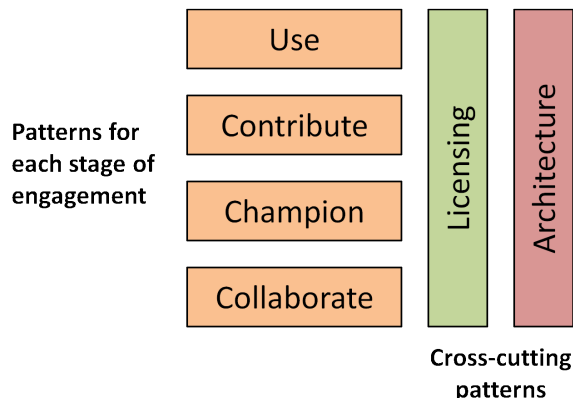


Figure 1. Open source business patterns

- 8. Pool resources:** To optimize the use of resources in the collaboration stage, a company may jointly develop a common stack of open source assets with other companies. Each company can then develop their own differentiated products based on these common assets.
- 9. Foundation:** A company can attract other companies to contribute its open source project by transferring ownership of the code to an independent foundation. A foundation creates an arms-length relationship between the project creator and the project itself, and it centralizes common functions that all members can access (e.g., legal, marketing, project management). Creating a foundation builds trust and facilitates collaboration among the contributors.

Examples of patterns related to licensing include:

- 1. Play by the rules:** The risk of a licensing problem arising after a product is released can be mitigated by ensuring license compliance for open source components that are combined.
- 2. Dual license:** To encourage commercial users to pay for an open source software product, a company can offer the same product under two licenses: commercial and open source. Non-commercial users still benefit from the rights under the open source license, and buyers of the commercial license are released from some of the obligations of the open source license.
- 3. IP modularity:** Alignment of its intellectual property with the product architecture can help a company manage the complexities of having both open source and proprietary versions of a dual product.

TIM Lecture Series – The Business of Open Source

Michael Weiss

Examples of patterns related to architecture include:

1. **Modular architecture:** External contributors may have difficulty contributing if they require deep knowledge of the project. A company can overcome this problem by partitioning the code base so that different parts (or modules) can be worked on and managed independently.
2. **Manage complements:** A governance model and regulatory tools can help manage the quality of complements developed by the community.

To conclude the lecture, Weiss speculated about what the future might hold for open source. The discussion focused on the shift away from desktops and servers onto what are – for open source – non-traditional platforms of mobile, embedded systems, and hardware. The role of open source in the "Internet of Things" (tinyurl.com/5qr2nq) was also explored. Finally, Weiss reflected upon the ongoing growth of "open" beyond software (e.g., open hardware, open data, and open science).

Lessons Learned

In the discussions that followed each portion of the presentation, audience members shared the lessons they learned from the presentation and injected their own knowledge and experience into the conversation.

The audience identified the following key takeaways from the presentation:

1. Patterns help you recognize and find solutions beyond your expertise, and they help you collaborate
2. There are now real and practical examples of companies employing these strategies – open source business is now a well-travelled road.
3. Different patterns are suited to different contexts/opportunities.
4. Although open source can reduce costs, the increased complexity it often brings is a challenge.
5. The value to companies is significant if they share R&D for non-core development; they can then focus their efforts on what makes them different from the competition.
6. Companies go through phases in their engagement with open source. They start out as users, then start contributing. Eventually, they may champion their own project and then collaborate with other companies, perhaps even by forming a foundation. It's a learning curve.
7. Community management only works when you have appropriate governance in place.
8. The value of the code is a function of: i) how quickly you can learn about it; ii) how modular it is; and iii) how many applications it can serve. These factors determine how quickly you can ramp up.
9. Companies have options and can make conscious choices about how to use open source in their strategies and business models.
10. The value of the disruption that open source represents is threefold: i) it allowed companies to make money in new ways; ii) it enabled software to be produced and distributed in completely new ways; and iii) it reduced the barriers to entry while increasing the potential for collaboration.
11. There is a trend towards the creation of open source toolsets that allow for the early and simple creation of applications. This technical capability enables others to quickly solve real-world domain-specific problems.
12. In the future, we need new ways of innovating around business models. The landscape is now different; we need to look forward and not worry about what did or did not work in the past.

TIM Lecture Series – The Business of Open Source

Michael Weiss

Additional Resources

1. Slides from this lecture: tinyurl.com/loz5yof
2. Related papers by Michael Weiss:
 - "Performance of Open Source Projects" (2009; tinyurl.com/l2zf8qz)
 - "Profiting from Open Source" (2010; tinyurl.com/m974nya)
 - "Profiting Even More from Open Source" (2011; tinyurl.com/kt37jy)

This report was written by Michael Weiss and Chris McPhee.

About the Speaker

Michael Weiss holds a faculty appointment in the Department of Systems and Computer Engineering at Carleton University in Ottawa, Canada, and is a member of the Technology Innovation Management program. His research interests include open source, ecosystems, mashups, patterns, and social network analysis. Michael has published on the evolution of open source business, mashups, platforms, and technology entrepreneurship.

Citation: Weiss, M. 2014. TIM Lecture Series – The Business of Open Source. *Technology Innovation Management Review*. January 2014: 28–31.



Keywords: open source software, business models, patterns, engagement, entrepreneurship, licensing, architecture, community