

INSIGHT: Thinking Issues

Tony Clear

Software ecosystems: what do we need to know?

In my last column I discussed the educational and curriculum challenges posed by scaled agile software development approaches [3]. These continue to arise as the discipline and the profession increasingly encompass broader global and business focused dimensions. Extending that line of argument, this column turns to the phenomenon of *software ecosystems*. What are *software ecosystems*, and what implications do they have for our students? What will they need to learn and how may they practice as future developers of software within a software ecosystem?

We can think generally of a *software ecosystem* as an entity where an *ecosystem owner* provides not simply a *software product* but an underpinning *platform*. This platform offers a set of API's through which external developers can connect and build applications. Common examples can be seen in "App Marketplaces", such as those provided by Apple through its *App Store*, Google its *Play Store* and, in the New Zealand context, Xero the accounting software company through its *APP Marketplace*.

The key goal of "software product and platform producing organizations...is to run an innovative continuous software business with propensity for growth"[5]. The construction of an ecosystem around a platform aims at achieving what Cusumano has dubbed *Staying Power* [4], "by minimizing risk, increasing innovation, increasing revenue, and creating a healthy network of partners around the business"[5].

More formally a *software ecosystem* has been defined "as a set of businesses functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts." [6]

This topic is of particular interest to me, as we have been awarded a Royal Society of New Zealand Catalyst International Leader's grant [*Leading the Way in Software Ecosystems for NZ*], to host my colleague Professor Daniela Damian of University of Victoria, Canada, in an investigation of software ecosystems in a New Zealand context. As we scope our work and deepen our investigations, the questions for education and the demands on software professionals in this new arena become increasingly to the fore.

A useful map of the domain encompassed is given in recent work by Jansen [5]. This work has been investigating the question of governance in software ecosystems, and developing a corresponding maturity model. The general idea of a "focus area maturity model" develops the idea of well known software engineering (SE) maturity models. Paulk[7] for instance, presented the Capability Maturity Model, demonstrating a level of achievement in SE, but here Jansen [5] does so with a specific focus area. So a *focus area maturity model* presents a set of areas of focus, which contain capabilities, which in turn contain a set of

practices, within maturity levels and result in functional domain capabilities. These can be implemented as levels of achievement (maturity) and institutionalized in an organizational context.

For the domain of software ecosystem governance, Jansen [5] has outlined “*the software ecosystem governance maturity model – SEG-M²*”.

Figure 1. Seven Focus Areas of the SEG-M² [ex. 5, fig.2]

Each of the seven focus areas in the SEG- M² is represented by the four core values emphasized above by Jansen [5] when improving *Staying Power*. In this representation of “*the software ecosystem governance maturity model – SEG-M²*” seven practice areas are identified, summarized from Jansen below.

- **Associate Models** - All practices to do with management and coordination of partners. It contains practices such as the creation of partnership models, partner training, and consultancy and sales partner support. One of the more technical aspects of associate models is the creation of systems that enable partners to communicate with end users, such as approval systems in app stores or SAP’s customer partner connection center, that enables partners to share ticketing systems with customers and SAP itself.
- **Ecosystem Health** – the practice area that regards the ecosystem as a living ecosystem that can be analyzed as a whole, also contrasting itself with other potentially influencing ecosystems. The practices in this focus area are concerned with partner health analysis, sharing of market data, and making strategic choices in regards to competing ecosystems.
- **Open Markets** – the practice area that concerns itself with the creation of an open market for services and applications. The practices belonging to extension approval, extension marketing, business model innovation, and app delivery are part of the open markets focus area. The area evenly divides itself across management and technical boundaries.
- **Open Platforms** - All practices related to the creation of a stable solid and open platform belong to the open platforms focus area. It is concerned with the creation of a platform, the platform’s security, its extension capabilities, and documentation.
- **Intellectual Property** - The practices to do with patent management and intellectual property management within the ecosystem. At the lowest levels it is concerned with innovation sharing across the ecosystem. At the higher levels it is concerned with patents, licenses, and stimulation of ecosystem health by co-creation.
- **Open Innovation** – the practice area concerned with sharing knowledge across the ecosystem to feed external developers with new possibilities for improvement, also known as niche creation. At the lowest levels it is concerned with sharing development practices and innovations with partners. At higher levels it is concerned with creating shared innovations and ecosystem standards.
- **Software Development Governance** - all practices concerned with observing, supporting, and enabling software developers. The practices are concerned with domains such as testing, road mapping, shared requirements. At the lowest levels the focus area is concerned with opening up to developers and enabling them to develop third-party extensions. At higher levels it is concerned with collecting data (software operation knowledge, or SOK [28]) about applications and their developers and about supporting developers in helping each other.

Under the seven ecosystem management focus areas 168 practices have been identified. These practices have been collected into an ecosystem management maturity model, with the goal of providing ecosystem managers with a road to improvement and achievements of higher levels of maturity.

The further categorisation of the seven focus areas in Table 1, breaks out the primary skill sets demanded. Thus, software development utilising an ecosystem model poses a new set of demands on software professionals. Furthermore software ecosystems present new hybrid career paths which extend the developer skill sets that have traditionally been demanded. The technical demands of platform and app. development are now complemented by market data sharing and management strategy. So, this software approach demands a developer skill set which encompasses business awareness, communication and negotiation skills, and navigating strategy for product and partner relationship development.

Focus Area	Primary Skill Sets Demanded	Expertise
------------	-----------------------------	-----------

Associate Models	management and coordination of partners, creation of systems that enable partners to communicate with end users	Hybrid – relationship mgt & technical
Ecosystem Health	partner health analysis, sharing of market data, and making strategic choices in regards to competing ecosystems	Hybrid – strategy & technical
Open Markets	extension approval, extension marketing, business model innovation, and app delivery. Evenly divided across management and technical boundaries.	Hybrid - mgt and technical
Open Platforms	creation of a platform, the platforms security, its extension capabilities, and documentation	Technical and documentation
Intellectual Property	innovation sharing across the ecosystem. At higher levels concerned with patents, licenses, and stimulation of ecosystem health by co-creation	Hybrid – relationship mgt, strategy & technical
Open Innovation	At lowest levels concerned with sharing development practices and innovations with partners. At higher levels concerned with creating shared innovations and ecosystem standards.	Hybrid – relationship mgt, strategy & technical
Software Development Governance	concerned with observing, supporting, and enabling software developers. Concerned with domains such as testing, road mapping, shared requirements. Opening up to developers and enabling them to develop third-party extensions. At higher levels it is concerned with collecting data about applications and developers and about supporting developers in helping each other	Hybrid - relationship and data mgt strategy, Technical and documentation

Table 1. The new skills demanded of developers in software ecosystems?

The narrow technical focus of many of our CS degrees seems poorly prepared to meet these needs, risking our graduates being relegated to a dark room far from much of this dynamic new action. In addition to an extended hybrid skill set encompassing technical expertise about how to build systems and platforms, our graduates will need to practice a broader human and social set of capabilities about sensing what to build and for whom. So, the curriculum needs to adopt an approach which at least builds awareness of the hybrid nature of skills increasingly expected in professional settings, and the increasing limitations of a wholly technical skill set. At the postgraduate level in the AUT Master of Computer and Information Sciences programme, as one example, we have long included a paper “Service Relationship Management”. Separately, aspects of the role of computers in society, project management, communication and professional ethics have been covered at the undergraduate level. As an example of some of the broader strategic challenges now being posed for extended development models, our graduates working in software ecosystems, will also need to raise their sights to the wider social issues of privacy (cf. Figure 2), ethical practice and competition policy, as highlighted in the developments with which this column concludes.

From a New Zealand perspective, reviewing the year of 2019, Paul Brislen on behalf of IT Professionals NZ, has ruefully criticised Facebook’s behaviour as a major Ecosystem owner: *“The absolute lack of compassion from the likes of Facebook over the mosque attacks and their refusal to change their business models to make it harder to publish video footage of mass murder is one trend that has dominated the year. While it’s one thing to accidentally publish materials, as Google has done with its zeitgeist email, it’s something else to monetise terrorism the way Facebook has, and its refusal to back down may well be the final straw that breaks the regulators’ backs. Countries around the world are calling time on Facebook’s unwillingness to abide by local law, pay local tax, support local regulations and operate alongside the rules of society. It’s not just the live video feed, it’s also the lack of transparency around its advertising models and lack of responsibility around user privacy that may well see the company knocked off its perch in a number of jurisdictions”*. [1]

As evidence of such developments, a recent report by the UK’s Competition and Markets Authority, on the regulation of online platforms, has “concluded that new approaches needed to be taken to regulating platforms, that relying solely on enforcing existing competition law was not sufficient and that introducing pro-competitive rules and regulations was necessary” [2].

So a broader perspective on discerning how, what and for whom software will be built, the role and place of software platforms, their limits and how they should be used will increasingly face developers. Our curricula and graduates will need to adapt!

References

1. Brislen, P. (2019, 13 December). Brislen on Tech: The Ghost of Christmas Past. *ITP Techblog*. Retrieved from https://techblog.nz/2017-Brislen-on-Tech?utm_source=email
2. CMA. *Online platforms and digital advertising: Market study interim report*. Competition and Markets Authority, London, 2019.
3. Clear, T. THINKING ISSUES: Global software engineering and scaled agile: pushing the boundaries of the discipline? *ACM Inroads*, 9, 3 (2018), 38-39.
4. Cusumano, M. A. *Staying power: Six enduring principles for managing strategy and innovation in an uncertain world (lessons from Microsoft, Apple, Intel, Google, Toyota and more)*. Oxford University Press, 2010.
5. Jansen, S. A focus area maturity model for software ecosystem governance. *Information and Software Technology*, 118, (2020), 106219.
6. Jansen, S., Finkelstein, A., and Brinkkemper, S. A sense of community: A research agenda for software ecosystems. In *31st International Conference on Software Engineering-Companion Volume* (Vancouver, Canada: IEEE, 2009), 187-190.
7. Paulk, M., Weber, C., Garcia, S., Chrissis, M. and Bush, M. *Key Practices of the Capability Maturity ModelSM, Version 1.1* [Online available <http://www.sei.cmu.edu/pub/documents/93.reports/pdf/tr25.93.pdf>. Accessed 07/03/2005]. CMU/SEI-93-TR-025 ESC-TR-93-178, Carnegie Mellon University: Software Engineering Institute, Pittsburgh, 1993.