Collaboration and Innovation Dynamics in Software Ecosystems: A Technology Management Research Perspective

den Besten, M.; Amrit, C.; Capiluppi, A.; Robles, G.

DOI
10.1109/TEM.2020.3015969

Publication date
2021

Document Version
Final published version

Published in
IEEE Transactions on Engineering Management

Collaboration and Innovation Dynamics in Software Ecosystems: A Technology Management Research Perspective

Matthijs den Besten, *Member, IEEE*, Chintan Amrit, *Member, IEEE*, Andrea Capiluppi, *Life Fellow, IEEE*, and Gregorio Robles, *Senior Member, IEEE*

***Abstract*—In a widely used definition, a software ecosystem (SECO) is a "*collection of software projects which are developed and evolve together in the same environment*." The objective of this article is to explore how SECOs foster collaboration among their key players to achieve innovation within individual software projects. Thus, in this article, we merge two prevalent views from past research, namely, the organization and the project-centric views on innovation management into a concept map. We then describe how these views complement each other. The resulting concept map contains new concepts that provide better coverage of how innovation currently takes place within ecosystems and shows how internal and external drivers are key to the innovation in SECOs. With this article, we want to open the floor to further discussion on how internal and external drivers can be effective in propelling innovation of both traditional (closed) and open-source SECOs. We solicit future work to expand on this view and to achieve a more complete understanding of how SECOs are connected to innovation.**

***Index Terms*—Collaboration, innovation, open source software, software ecosystems (SECOs).**

I. Introduction

SOFTWARE ecosystems (SECOs) are very important socio-technical systems that mould IT infrastructure all over the world. SECOs are a very nascent field, where change is the order of the day. While the definition of ecosystems in biology stems from the 1960s (and has not changed since then), the definition of SECOs has been continuously evolving since it first appeared in 2007. We observe that SECO research encompasses open-source software project research as well as research into commercial SECOs. Our aim in this article is to explore the innovation and collaboration dynamics in SECOs. In the next section, we describe the literature and definitions of SECOs as well as the collaboration aspects of ecosystems (see Section II-A) and the methods used in SECOs research (see Section II-B). This is followed by a discussion of the collaborative innovation that is carried out in ecosystems (see Section III-A) and the innovation drivers that affect it (see Section III-B). We use both dynamics and drivers to produce a concept map of how innovation is created and managed in SECOs and a call for future research (see Section IV). Section V concludes this article.

Manuscript received June 1, 2020; revised July 28, 2020; accepted July 30, 2020. Date of publication September 14, 2020; date of current version July 2, 2021. The work of M. Besten was supported by the French Government (Labex Entreprendre, ANR-10-Labex-11-01). The work of G. Robles was supported in part by the Government of Spain through Project "BugBirth" under Grant RTI2018-101963-B-100 and of the Madrid Regional Government through the e-Madrid-CM Project under Grant S2018/TCS-4307, co-funded by the European Structural Funds (FSE and FEDER). Review of this manuscript was arranged by Department Editor T. Daim. *(Corresponding author: Chintan Amrit.)*

The authors are with the Montpellier Business School Montpellier, 105384 Occitanie, France (e-mail: m.denbesten@mbs-edu.fr; c.amrit@uva.nl; a.capiluppi@rug.nl; grex@gsyc.urjc.es).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TEM.2020.3015969

II. Literature Background

We articulate this section in two parts: the first presents a short literature survey on SECOs; the second is a summary of the empirical approaches that have been attempted in the past to analyze SECOs.

A. SECOs and Collaboration

Messerschmitt and Szyperski proposed the oldest definition of SECO in the found literature referring to the book on SECO published in 2005: "*Traditionally, a software ecosystem refers to a collection of software products that have some given degree of symbiotic relationships*" [1].

Bosch *et al.* takes domains and domain experts into a more prominent role when considering SECOs: "*A software ecosystem consists of a software platform, a set of internal and external developers and a community of domain experts in service to a community of users that compose relevant solution elements to satisfy their needs*" [2], [3].

Lungu *et al.* present a different definition of the SECOs that is adopted by many papers: "*A software ecosystem is a collection of software projects which are developed and evolve together in the same environment*" [4]. Given these definitions, we now explore what role collaboration plays in innovation in SECOs.

Ecosystems are built on a common technological platform that enables different actors and stakeholders to collaborate and contribute to the ecosystem [5]. Such a system is also a socio-technical system and can be either an open-source software ecosystem (OSSECO) or a proprietary SECO. OSSECOs are essentially built through collaboration processes of different developers. As the developers who participate in OSSECOs do so voluntarily, they need to optimize their collaboration to reduce individual effort and maximize the contribution toward the collective project [6]. On the other hand, proprietary SECOs mostly focus more on the organizational and business perspective and less on the collaborative processes [5]. However, there are many commercial OSSECOs, that blur the line between purely commercial ecosystems and OSSECOs. These also borrow the collaborative aspects from OSSECOs and the organizational and business perspectives from purely proprietary ecosystems.

B. Approaches Used in SECOs Research

In their systematic mapping study on SECOs and open innovation for embedded systems, Papatheocharous *et al.* [7] analyzed 260 publications among others on the type of research that is conducted.

They found that a majority (37%) of the research falls in the category of evaluation research (i.e., an investigation of a problem in practice or an implementation of a technique in practice), followed by 23% by philosophical papers (i.e., that sketch a new way of looking at things, a new conceptual framework, among others), 11% by solution proposal, 11% opinion papers, 10% by validation research (i.e., research that investigates the properties of a solution proposal that has not yet been implemented in practice), 4% from experience reports (i.e., the paper contains a list of lessons learned from the author's own experience), and almost 2% as unknown [7].

We do not have, however, data on the research methods used in those studies. Nonetheless, we argue that research on OSSECOs is especially well-suited for empirical research methods. There are two main types of research paradigms having different approaches to empirical studies [8].

On one hand, we have qualitative research methods [9], which means investigating ecosystems in their natural setting. By interacting with persons involved in those ecosystems and understanding their views, researchers try to discover causes—an example of this approach can be found in [10].

On the other hand, we have quantitative research [11], mainly focused on quantifying a relationship or to compare two or more groups. Although in many scientific areas, quantitative research involves controlled experiments, this is difficult to perform in environments such as SECOs, where researchers cannot manipulate them to extract conclusions. Observational research is therefore better suited [12], as performed in [13] and [14].

Nevertheless, and especially in OSSECOs, there is a wide range of publicly available data ready to be retrieved and analyzed, as has been already done in [13]–[15]. The availability of large amounts of data on SECOs is both an advantage and a challenge. Big graph and network analysis techniques are therefore among those techniques that have been applied to the gathered data in many investigations, for example, in [16].

SECOs are nonetheless about technological or human relations and their interactions, resulting in collaborative innovation. But, researchers may be however overwhelmed by the magnitude of data (and may suppose they have the *whole picture*, which is in general wrong as many interactions may not be recorded, and risk to perform superficial or erroneous analyses [17]. In this context, case studies, normally aimed at tracking a specific attribute or establishing relationships between different attributes [18], offer a good approach, as already shown in [16], [19]–[21].

III. Dynamics and Drivers of Innovation

We divide this section in two parts: Section III-A analyses which medium- and long-term dynamics are at play within SECOs dealing with innovation and see Section III-B analyze what (internal or external) drivers play a discernible role for innovation, concerning SECOs. This is by no means an extensive list, but a way to open up the floor to other researchers and practitioners to comment and add to our list. Both dynamics and drivers will be used to enhance our conceptual map of innovation in SECOs (see Section IV).

A. Innovation Dynamics in SECOs

We take innovation to be *"the implementation of a new or significantly changed product or process"* [22]. A product can be a good or a service, and processes can concern production, delivery, organization, or marketing. Innovation dynamics, then, concern changes in the ability of SECOs or its constituents to implement the changes. With regards to these changes, we are interested in explaining long-term and medium-term trends and also in exploring what drives short-term variations.

Taking the long-term view, the innovation dynamics of a SECO are determined by the strategic decisions of its stakeholders. In the case of embedded software, for instance, these stakeholders are mainly software producing organizations, whose software is part of a product platform that defines the ecosystem [7]. Platform providers might not only provide a platform where third-party vendors can innovate, but also participate in the innovation process, by leading the innovation process [23]. There is a variety of processes, which software producing organizations carry out, including research and development (R&D), marketing and sales, consulting and support, and much else [24]. Innovations in the business model or software being produced depending on the capabilities of these organizations [25]. To study long-term innovation dynamics, then, one should assess the capabilities of organizations and the choices they make that reinforce or weaken these capabilities. At a macrolevel, this is reflected by the organizations' R&D expenditures [26], while the microfoundations can be found in the procedures that organizations develop to guide the processes they carry out [7]. From this perspective, a crucial question is how well the design of participating organizations fits with the changing innovation needs of the ecosystem and how easily the design can be adapted or the organization be replaced [27].

In the medium term, an element that helps an organization to become more responsive to its environment is to organize itself around projects [28]. In the context of SECOs, these projects can be interorganizational in addition to intraorganizational. Consequently, the dynamics of innovation are driven by the ability of the ecosystem to incorporate new projects and rejuvenate existing ones. In case of open-source software, where stakeholders can include individual developers and communities as well, the innovative capacity of projects depends on their ability to attract talent and resources and to organize for growth, while the innovative capacity of the ecosystems depends on the ease with which projects can be disbanded when they are no longer innovative [29]. Zaggl *et al.* [30] identify patterns of the dynamics in the ecosystem and propose a framework of the dynamics of openness while describing a shift from a manufacturer based to a user-based innovation.

Finally, short-term variations stem not so much from organization or project life cycles, but interactions among developers and users. For instance, the technical interdependencies among software packages within the ecosystem may create bottlenecks and innovation may suddenly spike once the bottleneck is resolved [31]. Besides, the adoption of software solutions by end-users, and hence the ability of developers to attract resources, is to some extent determined by the quality of exchanges between both groups [32]. The next section looks at more innovation drivers of this type.

B. Innovation Drivers in SECOs

Various innovation drivers underpin the achievement of new products or processes within ecosystems. These drivers can be either "internal" or "external." In turn, they have the potential to attract new projects to the ecosystem or to boost the productivity (or other health indicators) of the software projects contained [33]. In the following sections, we discuss a few of the internal and external drivers that have played a significant role in the advancement of innovation. As mentioned in the discussion section below, we have isolated the most prominent ones from researching the current literature. This article intends to solicit further research on other internal or external aspects that are not included below.

1) Internal Drivers to Innovation—Project Governance: The advantages of a large ecosystem of software projects have been described at length [34]–[36]. Below, we provide and justify several factors, internal to an ecosystem that can act as powerful drivers toward a fast innovation pace.

1) *Project Agility:* there is a large literature on project governance. In the context of SECOs, most "good practices" seem to be related to agility [37]—and it is claimed that this aspect is an innovation driver [10].
2) *Ecosystem Manager and Release Planning:* especially for ecosystems like OpenStack, GNOME, Linux. More regular releases improve project members' proficiency in managing releases [38]. This, in turn, affects to boost innovation as long as release dates are planned.
3) *Feature Freeze:* In several SECOs, it has been reported that all creative development must be completed by the feature freeze milestone date specified in the release proposal [21]. This allows development and innovation to work on deadlines and identified boundaries.
4) *Competition Between Alternative Packages:* OSSECOs have been reported to be innovation-friendly: for example, both the *npm* [13] and *JavaScript* [14] ecosystems frequently adopt external packages and value making it easy to contribute and publish packages. Most importantly, both show a healthy competition between multiple equivalent packages to solve any single problem. In turn, this competition boosts the amount of innovation that the combined packages will produce for one particular ecosystem.
5) *GitHub Branches and Pull Mechanism:* in a distributed version control system (VCS), every member of the software development team has a working copy of the entire project and logs (i.e., source code or files) locally. In this case, a member of the software development team is not required to be online to commit or make any changes in the source code. Members of the software development team can *pull* a request with any other team member in the software project. It has been reported that continuous integration improves the productivity of project teams, who can integrate more outside contributions, without an observable diminishment in code quality [39].
6) *Forking Mechanism:* forks in distributed VCSs are public copies of repositories in which developers can make changes, potentially, but not necessarily, to integrate those changes back into the original repository [15]. Compared to other VCSs (e.g., CVS, SVN), the most recent distributed VCS (e.g., Git, Mercurial, Darcs) have the potential to foster a more sustained degree of innovation, allowing more developers to effectively join the development of a software project.
7) *Public Code Reviews:* Code reviewing is a process in which a reviewer decides on somebody else's contribution. Several OSSECOs use that mechanism publicly, to improve the quality of software products. Modern code reviews provide a wide spectrum of benefits to software teams, such as knowledge transfer, team awareness, and improved solutions to problems [19]. These, in turn, can have a beneficial impact on the rate of innovation within a software project.

2) External Drivers to Innovation—Technology: We could also identify factors that might be able to influence the rate of innovation of software projects in SECOs, but that is not necessarily contained within the SECO itself. We describe the main ones as follows.

1) *Coopetition:* the term coopetition describes the industrial scenario, where companies collaborate with other companies to produce innovation that would be slower (or impossible) if a single company ventured without further support. The case of the OSSECO described in [16] shows that the OpenStack ecosystem receives contributions from many external companies (Rackspace, Canonical, IBM, HP, Vmware, Citrix and others are cited). The coopetition between external companies within the same OSSECO has been demonstrated to have a positive impact on the rate of innovation, for example, in [20]. Also, partner network size, interconnectedness, and diversity impact individual firm value [40].
2) *Disruptive Technologies:* Radical innovations refer to *highly discontinuous (technological) changes*. Such innovations are characterized by totally new features, high

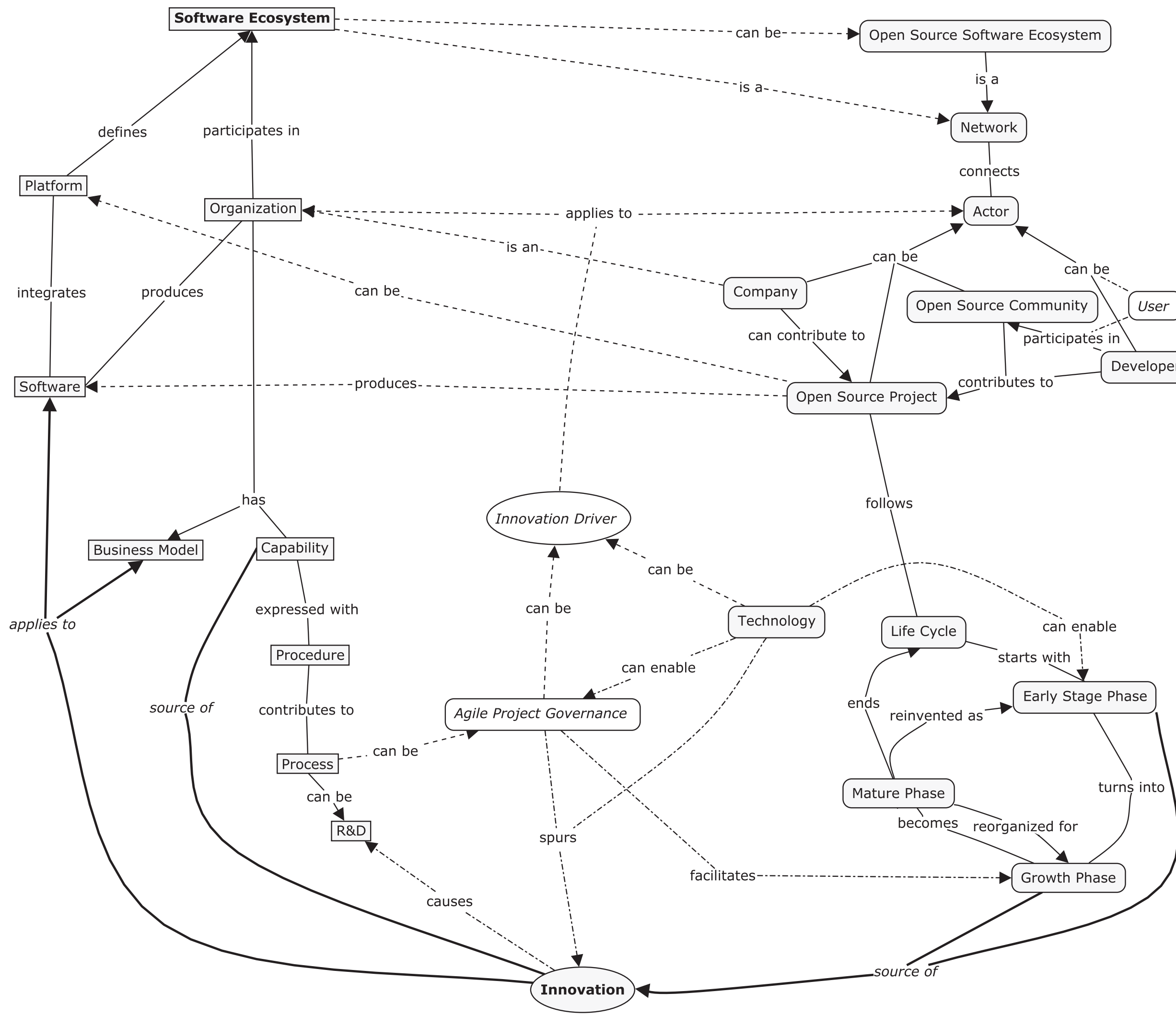

Fig. 1. Face of innovation—Concept map depicting how SECOs can bring about innovation.

uncertainty and the necessity for existing companies, or whole ecosystems, to acquire new capabilities to fully exploit emerging opportunities [41]. SECOs are most likely associated with disruptive change: one of such examples was recorded in the financial sector that in 2010 was completely transformed by the newly emerging bitcoin ecosystem [42].

IV. Toward a Conceptual Map of Innovation

Leaving aside the question of how innovative ecosystems emerge [43], we have focused on the long- and medium-term innovation dynamics within an existing ecosystem. The description of internal and external drivers, as well as the methods used in past research on SECOs, can be used to enrich the existing models that link ecosystems and innovation. This way, the dynamics within ecosystems emerge as interactions between actors, processes, and technologies. Fig. 1 shows a map of concepts relevant in this context. Concept maps are graphs representing conceptual knowledge [44], actual, or imagined. They can be considered as the first step in the construction of ontologies [45] and as such might be useful pretheoretical tools. Several scholars have employed concepts maps to enrich literature reviews [46], [47], but to our knowledge, just one other concept map has been proposed previously with regards to SECOs: a map that tries to explain the place of OSSECOs within a wider typology of SECOs [48]. Note, however, that with its focus on the relation between SECOs and innovation, our map has a different aim.

In the map that we propose (see Fig. 1), we use the following notation.

1) Concepts that stem from an organization-centered view of SECOs are represented by nodes with a *rectangular* shape.
2) Concepts related to a project-centred view are represented as *rounded* blocks.
3) The *oval* shape is reserved for an innovation and its driver.
4) The proposed map introduces *new* concepts that have not been explicitly put forward by the two base-line/reference

studies. These have a *white background* and an *italic* font.

5) Similarly, the links that have been added to the two existing frameworks are represented by *dashed lines*.
6) Links between SECO concepts and innovation are in *bold*, where they have been established in either study.

The left part of the map in Fig. 1 represents the long-term, organization-centric perspective, originally developed with regards to embedded SECOs [7]. The right part of the map represents instead the medium-term, project-centric perspective, that was originally developed to describe OSSECOs [29].

Both perspectives contain concepts that are closely related: we have indicated some of these associations with dotted lines. Also, we have added additional concepts, which we feel are of particular use for explaining short-term variations in innovation dynamics.

V. Conclusion

SECOs and OSSECOs share the possibility to enact one or more of the innovation drivers to foster innovation, through both the governance and the technology enablers. Within OSSECOs (but less so in traditional SECOs) the phases of their life cycle are directly and visibly related to certain aspects of the innovation drivers. The "technology" aspects are fundamental enablers of the transition between the early phase and growth phase; whereas the Agile governance aspects are visible propellers of the growth phase.

We do not consider the map complete, for example, the concept map currently does not explicitly mention the influence of policy, regulation or finance on the innovation dynamics of ecosystems. Moreover, in the guise of a research agenda, we propose the corroboration of the links proposed on the map, as well as the addition of new concepts to complement or substitute the existing ones. A particular fecund avenue for future research is the exploration of the various innovations drivers that spur innovation as well as their interaction with elements of the SECO.

References

[1] D. G. Messerschmitt and C. Szyperski, *Software Ecosystem: Understanding an Indispensable Technology Industry*. Cambridge, MA, USA: MIT Press, Dec. 2005, vol. 1. [Online]. Available: https://ideas.repec.org/b/mtp/titles/0262633310.html

[2] J. Bosch and P. M. Bosch-Sijtsema, "Softwares product lines, global development and ecosystems: Collaboration in software engineering," in *Proc. Collaborative Softw. Eng.*, 2010, pp. 77–92.

[3] J. Bosch and P. Bosch-Sijtsema, "From integration to composition: On the impact of software product lines, global development and ecosystems," *J. Syst. Softw.*, vol. 83, no. 1, pp. 67–76, 2010.

[4] M. Lungu, M. Lanza, T. Gîrba, and R. Robbes, "The small project observatory: Visualizing software ecosystems," *Sci. Comput. Program.*, vol. 75, no. 4, pp. 264–275, 2010.

[5] K. Manikas, "Supporting the evolution of research in software ecosystems: Reviewing the empirical literature," in *Int. Conf. Softw. Bus.*, 2016, pp. 63–78.

[6] W. Scacchi, "Socio-technical design," *Encyclopedia Human-Comput. Interact.*, vol. 1, pp. 656–659, 2004.

[7] E. Papatheocharous, J. Andersson, and J. Axelsson, "Ecosystems and open innovation for embedded systems: A systematic mapping study," in *Software Business* (Lecture Notes in Business Information Processing), J. M. Fernandes, R. J. Machado, and K. Wnuk, Eds. Cham, Switzerland: Springer, 2015, pp. 81–95.

[8] C. Wohlin, M. Höst, and K. Henningsson, "Empirical research methods in software engineering," in *Empirical Methods and Studies in Software Engineering*. Cham, Switzerland: Springer, 2003, pp. 7–23.

[9] N. K. Denzin and Y. S. Lincoln, *The Sage Handbook of Qualitative Research*. New York, NY, USA: Sage, 2011.

[10] M. Brand, V. Tiberius, P. M. Bican, and A. Brem, "Agility as an innovation driver: Towards an agile front end of innovation framework," *Rev. Managerial Sci.*, to be published.

[11] P. Leavy, *Research Design: Quantitative, Qualitative, Mixed Methods, Arts-Based, and Community-Based Participatory Research Approaches*. New York, NY, USA: Guilford Publications, 2017.

[12] R. Wieringa, N. Condori-Fernandez, M. Daneva, B. Mutschler, and O. Pastor, "Lessons learned from evaluating a checklist for reporting experimental and observational research," in *Proc. ACM-IEEE Intl. Symp. Empirical Softw. Eng. Meas.*, 2012, pp. 157–160.

[13] A. Trockman, S. Zhou, C. Kästner, and B. Vasilescu, "Adding sparkle to social coding: an empirical study of repository badges in the npm ecosystem," in *Proc. 40th Int. Conf. Softw. Eng.*, 2018, pp. 511–522.

[14] E. Wittern, P. Suter, and S. Rajagopalan, "A look at the dynamics of the JavaScript package ecosystem," in *Proc. 13th Int. Conf. Mining Softw. Repositories*, 2016, pp. 351–361.

[15] S. Zhou, B. Vasilescu, and C. Kästner, "What the fork: A study of inefficient and efficient forking practices in social coding," in *Proc. 2019 27th ACM Joint Meet. Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng.*, 2019, pp. 350–361.

[16] J. Teixeira, G. Robles, and J. M. González-Barahona, "Lessons learned from applying social network analysis on an industrial free/libre/open source software ecosystem," *J. Internet Serv. Appl.*, vol. 6, no. 1, 2015, Art. no. 14.

[17] J. Howison, A. Wiggins, and K. Crowston, "Validity issues in the use of social network analysis with digital trace data," *J. Assoc. Inform. Syst.*, vol. 12, no. 12, 2011, Art. no. 2.

[18] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case Study Research in Software Engineering: Guidelines and Examples*. Hoboken, NJ, USA: Wiley, 2012.

[19] A. Bacchelli and C. Bird, "Expectations, outcomes, and challenges of modern code review," in *2013 35th Int. Conf. Softw. Eng.*, 2013, pp. 712–721.

[20] A. Osarenkhoe, "A coopetition strategy–a study of inter-firm dynamics between competition and cooperation," *J. Database Marketing Customer Strategy Manag.*, vol. 17, pp. 201–221, 2010.

[21] Z. A. Khalil, E. Constantinou, T. Mens, L. Duchien, and C. Quinton, "A longitudinal analysis of bug handling across eclipse releases," in *Proc. IEEE Int. Conf. Softw. Maintenance Evol.*, 2019, pp. 1–12.

[22] F. Gault, "Defining and measuring innovation in all sectors of the economy," *Res. Policy*, vol. 47, no. 3, pp. 617–622, Apr. 2018.

[23] K. Kim and J. Altmann, "Platform provider roles in innovation in software service ecosystems," *IEEE Trans. Eng. Manag.*, to be published, doi: 10.1109/TEM.2019.2949023.

[24] S. Jansen, S. Brinkkemper, J. Souer, and L. Luinenburg, "Shades of gray: Opening up a software producing organization with the open software enterprise model," *J. Syst. Softw.*, vol. 85, no. 7, pp. 1495–1510, Jul. 2012.

[25] W. L. Miller and L. Morris, *Fourth Generation R&D: Manag. Knowledge, Technology, and Innovation*. New York, NY, USA: Wiley, 2008.

[26] B. Guloglu and R. B. Tekin, "A panel causality analysis of the relationship among research and development, innovation, and economic growth in high-income OECD countries," *Eurasian Econ. Rev.*, vol. 2, no. 1, pp. 32–47, Jun. 2012.

[27] G. Westerman, F. W. McFarlan, and M. Iansiti, "Organization design and effectiveness over the innovation life cycle," *Org. Sci.*, vol. 17, no. 2, pp. 230–238, Apr. 2006.

[28] T. Lappi, T. Karvonen, L. E. Lwakatare, K. Aaltonen, and P. Kuvaja, "Toward an improved understanding of agile project governance: A systematic literature review," *Project Manag. J.*, vol. 49, no. 6, pp. 39–63, Dec. 2018.

[29] N. Jullien, K.-J. Stol, and J. D. Herbsleb, "A preliminary theory for open-source ecosystem microeconomics," in *Towards Engineering Free/Libre Open Source Software (FLOSS) Ecosystems for Impact and Sustainability: Communications of NII Shonan Meetings*, B. Fitzgerald, A. Mockus, and M. Zhou, eds. Singapore: Springer, 2019, pp. 49–68.

[30] M. A. Zaggl, T. G. Schweisfurth, and C. Herstatt, "The dynamics of openness and the role of user communities: A case study in the ecosystem of open source gaming handhelds," *IEEE Trans. Eng. Manag.*, vol. 67, no. 3, pp. 712–723, Aug. 2020.

[31] J. Howison and K. Crowston, "Collaboration through open superposition: A theory of the open source way," *MIS Quart.*, vol. 38, no. 1, pp. 29–50, 2014.
[32] C. Cheruy, F. Robert, and N. Belbaly, "OSS popularity: Understanding the relationship between user-developer interaction, market potential and development stage," *Systemes d'information Management*, vol. 22, no. 3, pp. 47–74, Nov. 2017.
[33] S. Jansen, "Measuring the health of open source software ecosystems: Beyond the scope of project health," *Inform. Softw. Technol.*, vol. 56, no. 11, pp. 1508–1519, 2014.
[34] P. J. Williamson and A. De Meyer, "Ecosystem advantage: How to successfully harness the power of partners," *Calif. Manage. Rev.*, vol. 55, no. 1, pp. 24–46, 2012.
[35] M. G. Jacobides, C. Cennamo, and A. Gawer, "Towards a theory of ecosystems," *Strategic Manag. J.*, vol. 39, no. 8, pp. 2255–2276, 2018.
[36] B. Dattée, O. Alexy, and E. Autio, "Maneuvering in poor visibility: How firms play the ecosystem game when uncertainty is high," *Acad. Manag. J.*, vol. 61, no. 2, pp. 466–498, 2018.
[37] J. Warsta and P. Abrahamsson, "Is open source software development essentially an agile method," in *Proc. 3rd Workshop Open Source. Softw. Eng.*, 2003, pp. 143–147.
[38] M. Michlmayr, B. Fitzgerald, and K.-J. Stol, "Why and how should open source projects adopt time-based releases?" *IEEE Softw.*, vol. 32, no. 2, pp. 55–63, Mar./Apr. 2015.
[39] B. Vasilescu, Y. Yu, H. Wang, P. Devanbu, and V. Filkov, "Quality and productivity outcomes relating to continuous integration in GitHub," in *Proc. 10th Joint Meet. Found. Softw. Eng.*, 2015, pp. 805–816.
[40] R. C. Basole and H. Park, "Interfirm collaboration and firm value in software ecosystems: Evidence from cloud computing," *IEEE Trans. Eng. Manag.*, vol. 66, no. 3, pp. 368–380, Aug. 2018.
[41] M. Latzer, "Information and communication technology innovations: Radical and disruptive?" *New Media Soc.*, vol. 11, no. 4, pp. 599–619, 2009.
[42] D. Wörner, T. Von Bomhard, Y.-P. Schreier, and D. Bilgeri, "The bitcoin ecosystem: Disruption beyond financial services?" in *Proc. ECIS*, 2016, paper 33.
[43] O. Dedehayir, S. J. Mkinen, and J. Roland Ortt, "Roles during innovation ecosystem genesis: A literature review," *Technol. Forecast. Soc. Change*, vol. 136, pp. 18–29, Nov. 2018.
[44] J. D. Novak and A. J. Caas, "The theory underlying concept maps and how to construct them," Florida Institute for Human and Machine Cognition, Pensacola, FL, USA, 32502, Tech. Rep. 1, 2006.
[45] R. R. Starr and J. M. Parente de Oliveira, "Concept maps as the first step in an ontology construction method," *Inform. Syst.*, vol. 38, no. 5, pp. 771–783, Jul. 2013.
[46] P. Sisson and J. J. Ryan, "A knowledge concept map: Structured concept analysis from systematic literature review," *J. Entrepreneurship, Manag. Innov.*, vol. 13, no. 3, pp. 29–70, 2017.
[47] I. Islam, K. M. Munim, S. J. Oishwee, A. N. Islam, and M. N. Islam, "A critical review of concepts, benefits, and pitfalls of blockchain technology using concept map," *IEEE Access*, vol. 8, pp. 68 333–68 341, 2020.
[48] O. Franco-Bedoya, D. Ameller, D. Costal, and X. Franch, "Open source software ecosystems: A Systematic mapping," *Inform. Softw. Technol.*, vol. 91, pp. 160–185, Nov. 2017.

Matthijs den Besten (Member, IEEE) received the M.Sc. degree in AI from the University of Amsterdam, Amsterdam, The Netherlands, in 2000, and the Ph.D. degree in computer science from the Darmstadt University of Technology, Darmstadt, Germany, in 2004.

He is currently an Assistant Professor with Montpellier Business School (MBS), Montpellier, France, with a focus on digital entrepreneurship.

Prof. Besten is a founding member of the public research center Montpellier Research in Management (MRM), University Montpellier. He is a member of the Entrepreneurship and Innovation Chair, which is part of LabEx Entrepreneurship, University of Montpellier, France.

Chintan Amrit (Member, IEEE) received the master's degree in computer science from the Indian Institute of Science, Bengaluru, India, in 2000, and the Ph.D. degree in management information systems from the University of Twente, Enschede, The Netherlands, in 2008.

He is currently an Associate Professor with the Department of Operations Management, University of Amsterdam, Amsterdam, The Netherland. His research interests include business intelligence (using machine learning), open source development and mining software repositories. He is particularly interested in exploring network structures and their impact on software development and logistics.

Andrea Capiluppi (Life Fellow, IEEE) received the master's degree in management engineering and the Ph.D. degree in software engineering from the Politecnico di Torino, Turin, Italy, in 2000 and 2004, respectively.

He is currently an Associate Professor with the University of Groningen, Groningen, The Netherlands. His research focused on empirical software engineering.

Gregorio Robles (Senior Member, IEEE) received the master's degree in telecommunication engineering from Technical University, Madrid, Spain, in 2001, and the Ph.D. degree in computer science from the Universidad Rey Juan Carlos, Madrid, Spain, in 2006.

He is currently an Associate Professor with the School of Telecommunication Engineering, Universidad Rey Juan Carlos, Madrid, Spain. He is specialized in free/open source software research, basically by mining software repositories. Finally, he investigates as well how to assess the development of Computational Thinking skills. He is one of the founders of Bitergia, a software development analytics company.