

Software Ecosystems

Week 10



Software Ecosystems - Definitions?

an entity where an **ecosystem owner** provides not simply a software product but an underpinning **platform**.

This platform **offers a set of APIs** 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]

Clear, T. (2020). THINKING
ISSUES: Software Ecosystems:
What do we need to know?
ACM Inroads, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems - Definitions?

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]

Clear, T. (2020). THINKING
ISSUES: Software Ecosystems:
what do we need to know?
ACM Inroads, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems –Maturity Models?

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.

Clear, T. (2020). THINKING
ISSUES: Software Ecosystems:
what do we need to know?
ACM Inroads, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems –Maturity Focus Area?

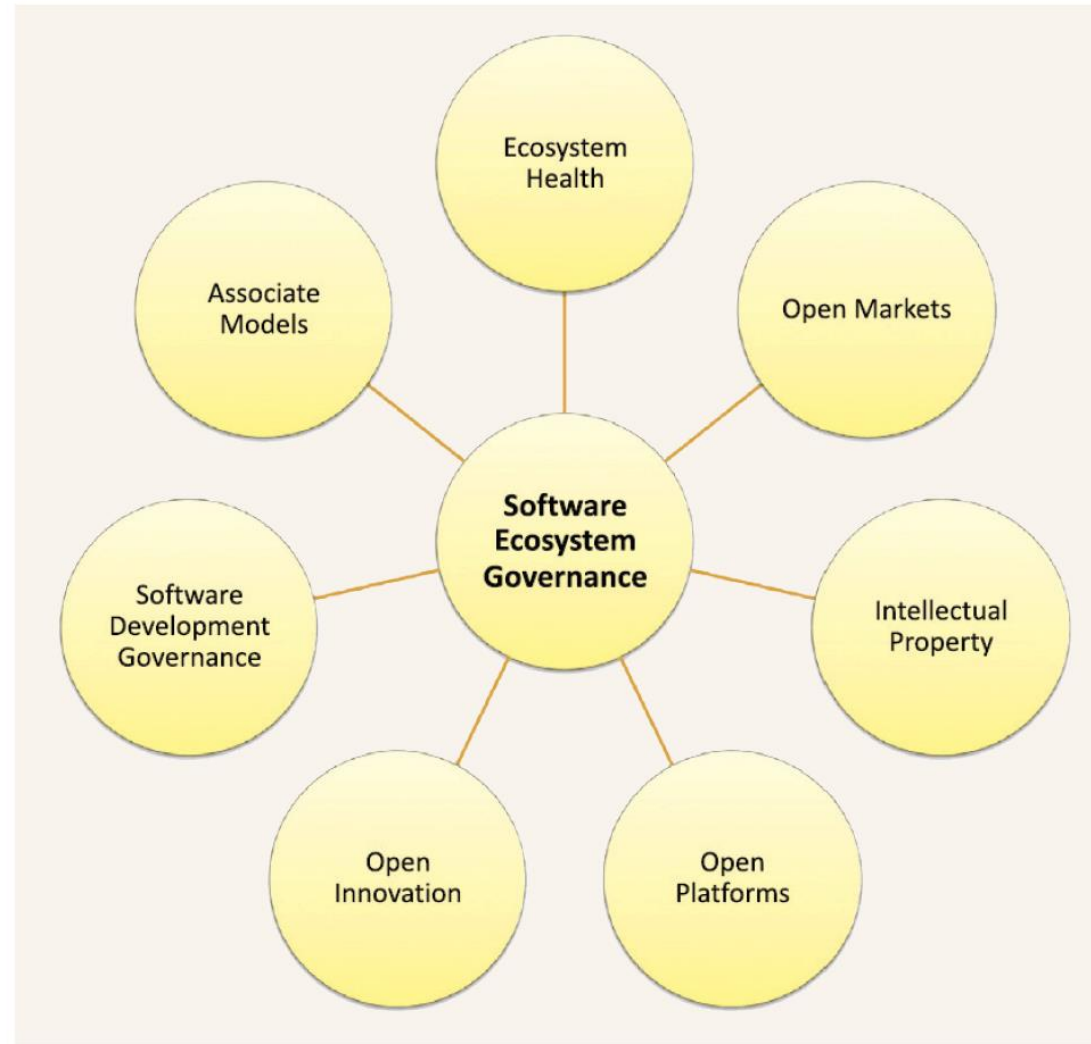


Figure 1: Seven Focus Areas of the *SEG-M²* [5, fig. 2]

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems –Associate Models

- **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.

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Ecosystem Health

- **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 regarding competing ecosystems.

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Open Markets

- **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.

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Open Platforms

- **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.

Clear, T. (2020). THINKING
ISSUES: Software Ecosystems:
what do we need to know?
ACM Inroads, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Intellectual Property

- **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.

Clear, T. (2020). THINKING
ISSUES: Software Ecosystems:
what do we need to know?
ACM Inroads, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Open Innovation

- **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.

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems – Software Development Governance

- **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) about applications and their developers and about supporting developers in helping each other.

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20.
<https://doi.org/10.1145/3395963>

Software Ecosystems Configuration

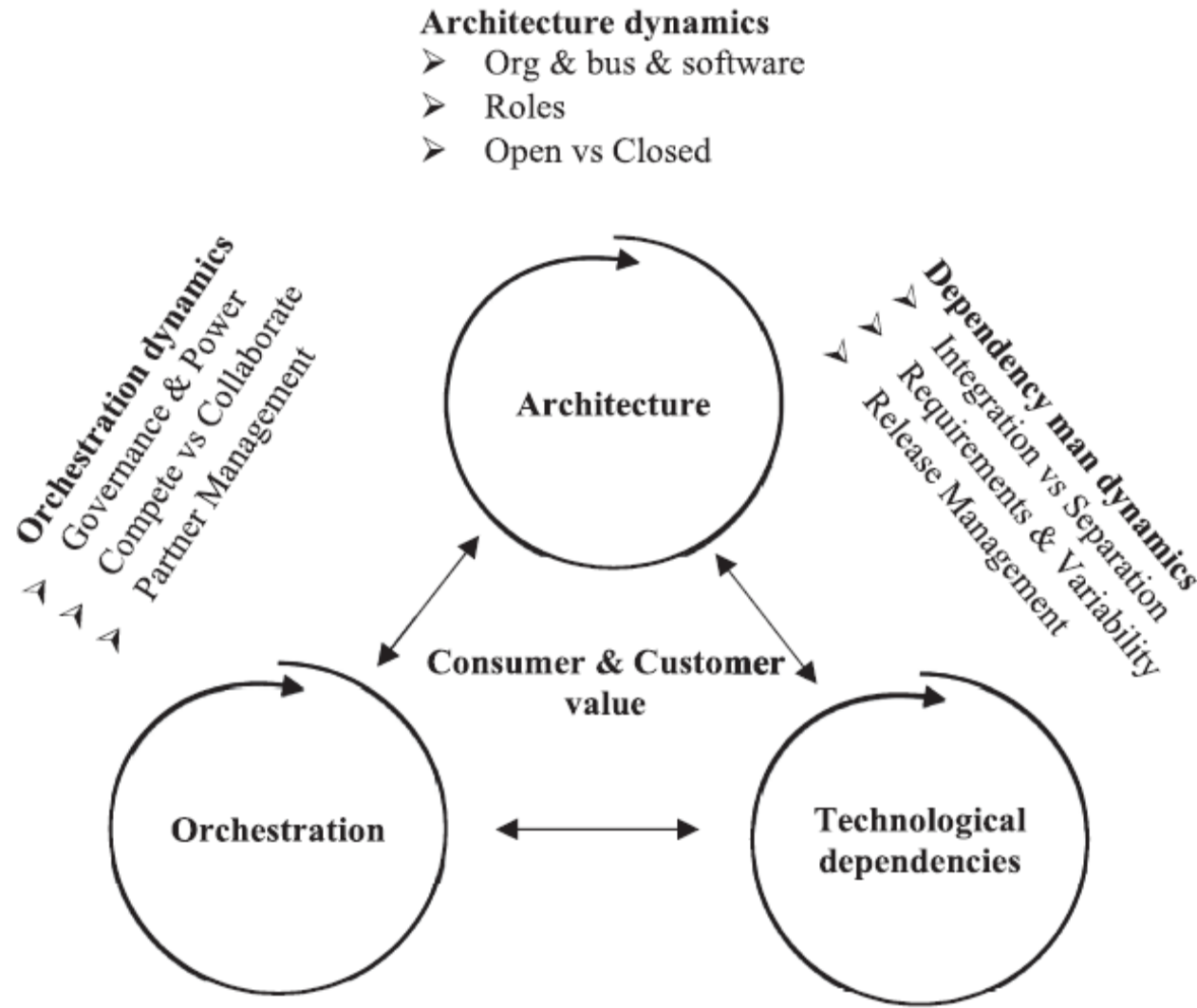


Fig. 1. SECO configuration and contingencies.

Burström, T., Lahti, T., Parida, V., Wartiovaara, M., & Wincent, J. (2024). Software Ecosystems Now and in the Future: A Definition, Systematic Literature Review, and Integration Into the Business and Digital Ecosystem Literature. *IEEE Transactions on Engineering Management*, 71, 12243-12258.
<https://doi.org/10.1109/TEM.2022.3216633>

Software Ecosystems & Open Innovation



BACKGROUND: SOFTWARE ECOSYSTEMS AND OPEN INNOVATION

A software ecosystem (SECO) consists of a set of actors united under a common vision and aiming to solve a common problem, often through the help of an underpinning technological platform. The actors collaborate and potentially also compete in a shared market for software and services.³ There are many examples of successful SECOs with underpinning platforms, both open⁴ and proprietary.⁵ Examples include operating systems (such as Microsoft Windows and Google's Android), web browsers (for example, Google's Chrome and Mozilla Firefox), and smart home assistants (like Amazon's Alexa and Apple's Siri).⁵ To provide access to their underpinning technology and enable complementary services, keystone organizations typically provide access to an open application programming interface (API). Open APIs allow organizations to share functionality, while allowing

their core technologies to remain proprietary, fostering open innovation (OI) within their ecosystems.

OI is an emerging field of research that aims to better understand how organizations "purposively manage knowledge flows across organizational boundaries" for improved organizational innovation.² Chesbrough and Bogers describe three knowledge flows,² modeled in Figure S1:

1. *outside in*, where knowledge flows from external sources to improve internal innovation processes
2. *inside out*, where internal knowledge flows outside the organizational boundaries to external entities for innovation
3. *coupled*, where knowledge flows bidirectionally between the innovating actors.

(Continued)

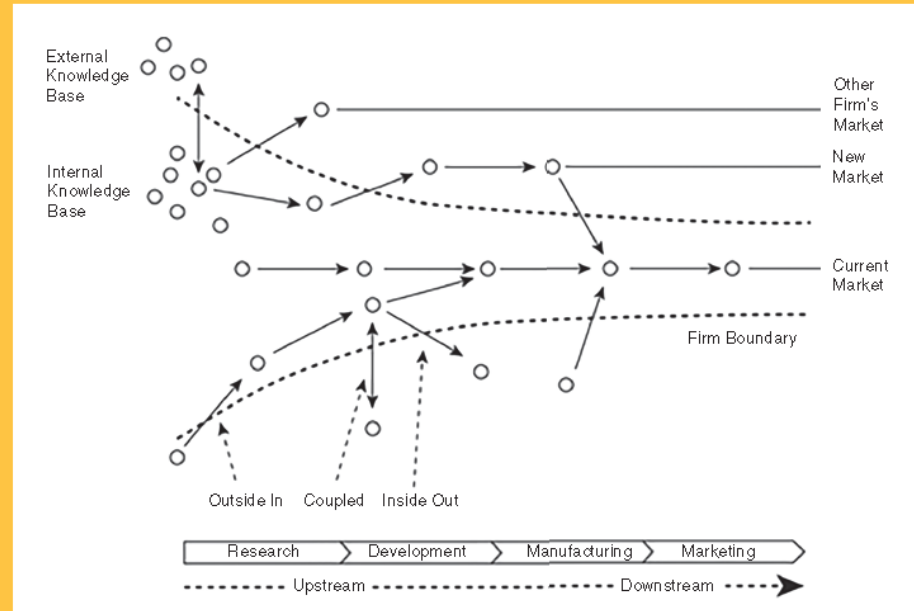
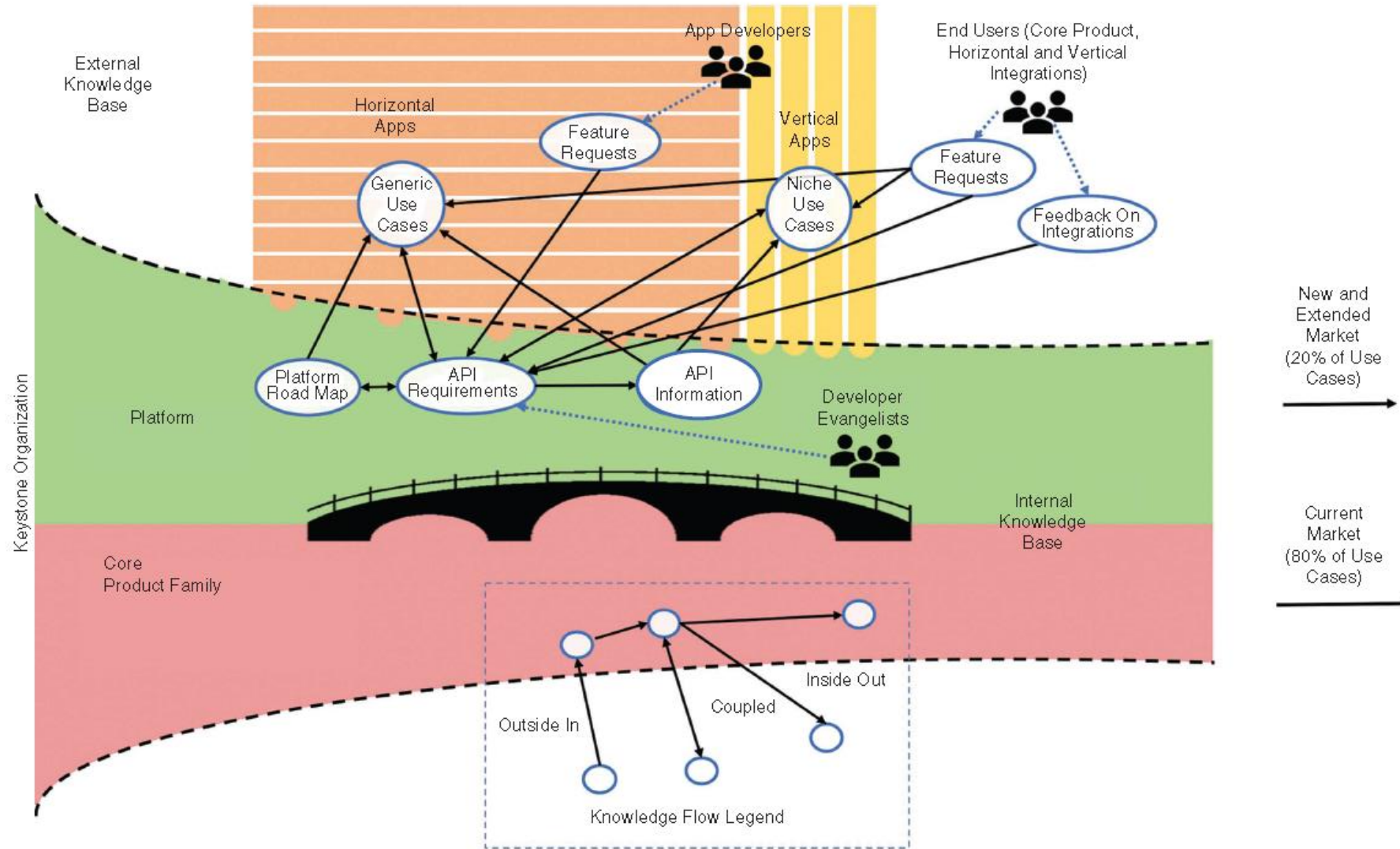


FIGURE S1. The open innovation model by Chesbrough and Bogers,² where the inside of the funnel represents the inside of the company, and the funnel's borders represent the company's wall to the outside through which the different knowledge flows (outside in, inside out, and coupled).

Damian, D., Linaker, J., Johnson, D., Clear, T., & Blincoe, K. (2021). Challenges and Strategies for Managing Requirements Selection in Software Ecosystems. *IEEE Software*, 38(6), 76-87.
<https://doi.org/10.1109/MS.2021.3105044>

Software Ecosystems & Requirements Flow



Damian, D., Linaker, J., Johnson, D., Clear, T., & Blincoe, K. (2021). Challenges and Strategies for Managing Requirements Selection in Software Ecosystems. *IEEE Software*, 38(6), 76-87.
<https://doi.org/10.1109/MS.2021.3105044>

Software Ecosystems –New Competencies Demanded

Table 1: The new skills demanded of developers in software ecosystems

Focus Area	Primary Skill Sets Demanded	Expertise
Associate Models	<ul style="list-style-type: none">• management and coordination of partners• creation of systems that enable partners to communicate with end users	Hybrid – relationship mgt & technical
Ecosystem Health	<ul style="list-style-type: none">• partner health analysis, sharing of market data, and making strategic choices regarding competing ecosystems	Hybrid – strategy & technical
Open Markets	<ul style="list-style-type: none">• extension approval, extension marketing, business model innovation, and app delivery• evenly divided across management and technical boundaries	Hybrid - mgt and technical
Open Platforms	<ul style="list-style-type: none">• creation of a platform, the platforms security, its extension capabilities, and documentation	Technical and documentation
Intellectual Property	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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	<ul style="list-style-type: none">• 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 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

Clear, T. (2020). THINKING ISSUES: Software Ecosystems: what do we need to know? *ACM Inroads*, 10(2), 18-20. <https://doi.org/10.1145/3395963>



#371478045



Questions and Comments....



Tony Clear 2024 S2

