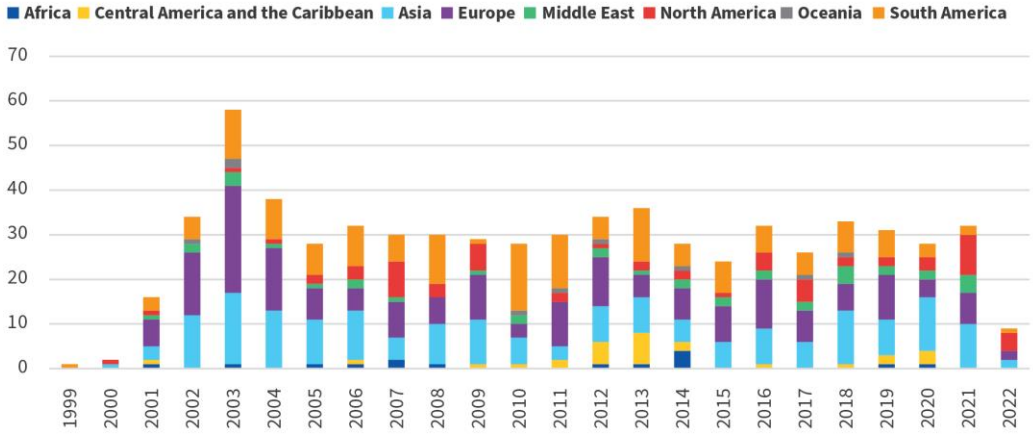
RI.
SE

JOHAN LINÅKER

Procurement and acquisition of Open Source Software

Challenges and Practices from a Swedish context

Figure 2: Regional Distribution of OSS Policies by Year

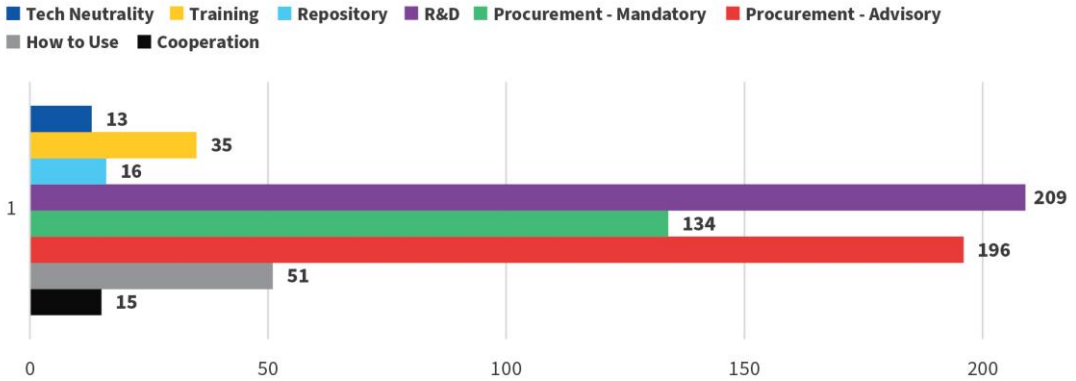


CSIS | STRATEGIC TECHNOLOGIES PROGRAM

- Survey from CISA (2022) show steady cadence of proposal and implementation of Open Source policy.
- Increase expected in recent time due to geopolitical turmoil

- Majority of policies procurement-related
- Most advising on the use and consideration of Open Source, but also some with mandatory requirements

Figure 5: Type of Actions



Source: https://www.csis.org/analysis/governments-role-promoting-open-source-software

CSIS | STRATEGIC TECHNOLOGIES PROGRAM

Designing policies for Open Source adoption

- Policy focus:
  - Public sector vs. Industry
- Policy direction:
  - Inbound vs. Outbound
- Type of intervention:
  - High-level endorsement vs. Advisory vs. Prescriptive
- Form for definition
  - Legislative vs. Government instruction vs. Strategy documents
- Scope of policy
  - National gov. vs. Regional/Local gov. vs. Institution-specific

Full report: https://www.linaker.se/blog/report-software-reuse-through-open-source-software-in-public-sector/

Photo by Elimende Inagella | https://unsplash.com/photos/brown-wooden-chairs-on-gray-concrete-floor-7OxV_qDiGRI

Implementation and support of policy

- Policy is not enough – Example: Italy
- Open Source Program Offices (OSPOs), support functions and centers of competency for OSS and software reuse, emerging on different levels
  - National government OSPOs
  - Regional/Local government OSPOs
  - Association-based OSPOs
  - Institution-centric OSPOs
- Guidelines and support documents
  - Inbound vs. Outbound focus
- Communities of practice
  - Public sector-specific <> External/general

Photo by Mark Potterton | https://unsplash.com/photos/man-in-white-long-sleeve-shirt-and-blue-denim-jeans-standing-on-white-metal-ladder-sNVkn3507Oo

Extant guidelines in Sweden

- ”The public administration's e-services should, as far as possible, be based on open standards and use software that is based on open source code and solutions that gradually free the administration from dependence on individual platforms and solutions.” - E-delegationen
- “[S]oftware with open source code [should] always [...] be considered, provided that an open alternative that meets the specified requirements is available and that the total cost including implementation and any transition is reasonable.” – Swedish Insurance Agency
- ”Software that is developed or procured should normally be published as open source. One reason for this is that the conditions for sharing the software, issues of ownership rights, etc., are thereby regulated in a standardized way.” – Swedish Agency of Digital Government

Photo by Billy Huynh | https://unsplash.com/photos/v9bnfMCyKbg

Does Open Source Software need to be procured?

Upphandlings myndigheten

Start / Frågeportalen / Omfattas öppen källkod av upphandlingsplikt?

Omfattas öppen källkod av upphandlingsplikt?

Publicerad 20 maj 2022 | IT | Upphandlingslagarnas tillämpningsområde

”The procurement laws are only applicable when the procuring organization acquires goods, services, or construction contracts in exchange for compensation.

The use of certain open source solutions and other free trials that can be used without any form of compensation generally falls outside the scope of procurement laws

However, if the procuring organization purchases services such as consulting, development, support, or operations related to it, these are subject to procurement obligations.

Upphandlingsmyndigenheten. (2022). Omfattas öppen källkod av upphandlingsplikt? https://www.upphandlingsmyndigheten.se/frageportalen/2616941/omfattas-oppen-kallkod-av-upphandlingsplikt/

RI. SE

Does Open Source Software need to be procured?

To facilitate the procuring organization in procuring software with open source, it is stated in the appendix 'Requirement Catalog' to the framework agreement that;

The customer may impose a mandatory requirement for specific named software that is fully licensed with one or more licenses approved by the Open Source Initiative (OSI), is free of charge, and free for all suppliers to use (e.g., no reseller status is required).

The customer may impose mandatory requirements on standards if the standard meets the requirements for an open standard according to SOU 2009:86

Kammarkollegiet. (2023). Vägledning för avrop från Programvaror och tjänster. https://www.avropa.se/globalassets/bilagor/1.-aktuella-rao/programvaror-och-tjanster/1.-gem-dokument-1-4/vagledning-for-avrop-fran-programvarorochtjanster231120.pdf?v=8dbe9cefa9a2d00

Adoption of Open Source still limited

- Despite permissive guidelines, call for open source software is still unusual in procurement and acquisition
- Mainly done by a limited number with the necessary capabilities, know-how, and long term horizon

Photo by Filip Kominik| https://unsplash.com/photos/grey-hlalway-IHtVbLRjTZU

But why is it so hard?

- Lack of internal capabilities and dependency on external resources
- Uncertainties and lack of practice in considering OSS in an acquisition and procurement process
- Discoverability of OSS options in the planning-phase of an acquisition (see http://offentligkod.se!)
- Locked-in to a proprietary technology, standard, platform, or ecosystems since before
- Glances at neighboring municipalities and copies procurement structure

Photo by Henri Picot | https://unsplash.com/photos/-X5FNFdq7Uw

But why is it so hard?

- Uncertainties and fear for legal and security-related risks
- Cultural challenges in terms of risk aversiveness, short-term horizons, focus on own municipality
- Lack of sustainable political support and clear policies
- No coordination or organization in acquisition process nor development and maintenance stages
- …

Photo by Henri Picot | https://unsplash.com/photos/-X5FNFdq7Uw

Guidance and support needed

- In what questions to ask, and where to find the answers.
- What can such a process look like?

Photo by Camylla Battani | https://unsplash.com/photos/short-coated-brown-dog-AoqgGAqrLpU

Are there open alternatives?

- In connection with the needs analysis in the preparatory phase of a procurement...
- Investigate the external environment for any existing projects that are available
- Download, test, and match against functional and technical specifications
- Is the missing functionality critical? Is it reasonable in cost to develop? Is it possible to include in the project?
- Example: Luftfartsverket (2019). Report from market research on e-archive products in open source. Diary number: D-2019-184972"

Promotion critical for enabling reuse

- Critical need to show that successful and realistic options are available, and where to find them.
- Several nation catalogues available, e.g., Offentligkod.se in Sweden
- Corresponding available in Italy, France, Germany, Netherlands
- Currently being compiled through an EU federated catalogue using the meta data standard Publiccode.yml

Photo by geralt | https://pixabay.com/illustrations/option-lots-selection-chance-note-112225/

Ensuring interoperability

- Ensure that requirements do not refer to proprietary software directly or indirectly (e.g., via data formats).
- Ensure that all required standards are open and preferably have an open software implementation.
  - Refer to the older list and guidance from Kammarkollegiet [1].
- When referring to other standards, ensure that necessary licenses can be obtained.

[1] Kammarkollegiet. (2014). Öppna standarder - Programvaror och tjänster 2014. https://www.avropa.se/globalassets/dokument/oppna-standarder---programvaror-och-tjanster.pdf

Ensuring transparency and access to data

- The possibility for data sharing and openness is incorporated from the very beginning in systems, processes, and requirement specifications, e.g.,
  - that the supplier must ensure the system can export data in open formats,
  - provide clear documentation, and
  - make it easy to integrate with other systems or publish data openly.

Photo by Claudio Schwarz | https://unsplash.com/photos/a-close-up-of-a-window-with-a-building-in-the-background-fyeOxvYvIyY

How is the OSS project's health?

- How secure and sustainable is the open-source software?
- Do we need to procure support or a packaged solution to guarantee quality of service?
- For checklists, see e.g.
  - https://chaoss.community
  - https://github.com/Forsakringskassan/riktlinje-oppenkallkod
  - https://www.redhat.com/en/resources/open-source-project-health-checklist

Support needed?

- What can we do ourselves? What do we need help with?
  - Services and/or enterprise-packaged solutions?
  - Can the need be fulfilled through any existing framework agreements?
  - Need for a new procurement?
  - Direct procurement to boundary limit to develop missing functionality and build internal competence?
  - Divide customizations and new development in to separate parts?

Photo by Leon | https://unsplash.com/photos/Oalh2MojUuk

Compare and evaluate different options

	A - Kommersiell licens	B - Öppen källkod	C - Paketerad Öppen källkod
Upphandling			
Kräver upphandling	Ja	Delvis	Delvis
Möjlighet att påverka kontraktslängd	Måttlig	Hög	Hög
Delat upphandlingsobjekt	Nej	Ja	Ja
Informationssäkerhet			
Går att kravställa mot utvecklande organisation	Ja	Nej	Delvis
SSC har ansvar för produktens säkerhet	Nej	Ja	Ja
Tillit genom transparens	Nej	Ja	Ja
Kostnad			
Inköpskostnad	Hög	Låg	Hög/Medel
IT-förvaltning	Medel	Medel+	Medel
Organisation			
Förvaltning SSC IT	Standard	Kräver anpassning	Standard
Support	Leverantör av mjukvara	Tredje part	Leverantör av mjukvara
Tekniska kriterier			
Följsamhet mot standard	Medel	Hög	Medel/Hög
Prestandakrav	Kravställs	Testas	Kravställs

Figur 6 - Sammanställning egenskaper

Statens servicecenter (2020). *Förstudie - Etablera en förvaltningsgemensam tjänst för E-arkiv.* Diarienr: 2019-00742-1.7-2.

Example: Italy

- Must according to law consider open alternative (if available)
- Any newly developed software must be released as Open Source Software
- A joint decision model to rank Open Source Software based on:
  - Technical aspects (ex. requirements fulfillment, interoperability, security, personal data management, project health, other administrations that are using it, availability of support...)
  - Total cost of ownership (e.g., installation, integrations, customization, verification, hosting, maintenance, training…)
- See: https://docs.italia.it/italia/developers-italia/gl-acquisition-and-reuse-software-for-pa-docs/en/stabile/index.html

Photo by Mark Tegethoff | https://unsplash.com/photos/I-GmdF7Md0o

Expected value gain?

- What are the expected value gains and drivers for choosing an open alternative?
  - Public money, public code
  - Sustainable management of information
  - Avoid reoccurring shifts of systems at new procurements
  - Benefit from and promote open innovation
  - Customize based on operational needs
  - Possibility to affect development pace
  - Reduce licensing costs
  - Benefits of scale when multiple administrations are involved
  - Increase competition on tenders

Photo by Annie Spratt | https://unsplash.com/photos/QckxruozjRg

Qualification reqs on suppliers?

- Community-first approach for enterprise-packaged solutions
- Beneficial if a supplier can show record of experience of
  - Active participation in Open Source Software projects in general
  - Active participation in the Open Source Software at hand
- Experience should preferably be recent and stretch over a longer period of time
- Supplier should be able to present
  - Accepted code contributions
  - Active participation in technical discussions
- Extra qualifying if supplier is represented in the governance and technical steering of the Open Source Software at hand.

Photo by Dylan Gillis | https://unsplash.com/photos/KdeqA3aTnBY

Qualification reqs on suppliers?

- **1. Relationship with the Software Manufacturer / Community**:
  - Is there a business relationship between the service provider and the software manufacturer or the community and to what extent is there going to be support available from them in the course of a project?
- **2. Ensuring Upstream Publication of Modifications and Patches:**
  - How is the service provider ensuring that modifications of the software and patches are going to be made available upstream to the general public?
- **3. Ensuring High-Quality Level 3 Support:**
  - To what extent is the service provider able to ensure high-quality level 3 support? Do they have the necessary expertise with regard to the source code of the specific open-source product themselves or are they able to ensure the support of the manufacturer?
- **4. Securing the Supply Chain Through Support for Core Components:**
  - Open-source software usually consists of various core components. Does the provider support developers and projects that supply the software components that the provider is integrating in their product? This is also going to become relevant with regard to „supply chain security“ in the Cyber Resilience Act.
- https://osb-alliance.de/english/selection-criteria-for-the-sustainable-procurement-of-open-source-software-press-release

RI.
SE

Help the vendor understand

- Be clear on what you want
- Define ambition levels with clear requirements setting the expectations up-front
  - From Goals, go to ambition levels, to specific requirements
  - Can’t just say, be open source, need to define details
- Maturity – negative -> Ambitions greater > Language is critical, having a neutral language, not framing it negatively
- Be open for vendor feedback, e.g., on license virality

Photo by Dylan Gillis | https://unsplash.com/photos/KdeqA3aTnBY

Source: https://arxiv.org/pdf/2409.01118

Avoiding (Soft) lock-ins - User-driven factors

- Communication
  - Limited communication between municipalities and towards the main supplier. Low level of transparency and awareness of developed functionality and ongoing or planned work.
- Procurement
  - Inconsistent and disqualifying qualification requirements on suppliers limiting the number of potential bidders to incumbents.
- Maintainership
  - Mixed opinions on who is responsible for the maintenance of the OSS project, and for facilitating any open collaborative development and community management.
- Comfort
  - A comfort and risk-aversiveness in preserving status quo. Preference to risks implied by technical debt and soft lock-in, before what could come in an unknown future.

Photo by gaspar zaldo | https://unsplash.com/photos/man-in-black-suit-jacket-wearing-silver-chain-bracelet-7Wg7NJy5JBE/

Avoiding (Soft) lock-ins Technical factors

- Dependency management
  - A limited overview of what dependencies that exist towards third-party components, or what version of dependencies have been included
- Development infrastructure
  - Lock-in to a specific development and build infrastructure.
- Documentation
  - Limited documentation regarding the development, contribution and onboarding, build-environment, and running the OSS project.
- Testing
  - No tests are present in the OSS project code base
- Code quality
  - Lack of visualized automatic analysis of the source code, searching for errors, security issues, etc

Photo by gaspar zaldo | https://unsplash.com/photos/man-in-black-suit-jacket-wearing-silver-chain-bracelet-7Wg7NJy5JBE/

Developing from scratch?

- Be sure of the purpose and value gains that are expected
- Consider costs, risks, and weigh against other alternatives
- Find other stakeholders with the same vision/problem and initiate an open collaboration from start.
- Consider (among other things)
  - Internal vs. acquired development resources?
  - Ownership of copyright?
  - Long-term maintenance and management?
  - Expectations on stakeholders? How can further join?
  - Business opportunities for suppliers?

Photo by Alex Kotliarskyi | https://unsplash.com/photos/QBpZGqEMsKg

Stay Open from Day 1

- The software should...
  - Be developed on an open social coding platform along with an open and transparent infrastructure from the start
  - Be licensed under an Open Source Software license
  - Include or be accompanied by necessary documentation and tooling for anyone to run and develop
  - …
- In other words...
  - Be developed as an Open Source Software project from the start
  - See https://standard.publiccode.net/, https://opensource.guide/, https://joinup.ec.europa.eu/collection/open-source-observatory-osor/guidelines-creating-sustainable-open-source-communities

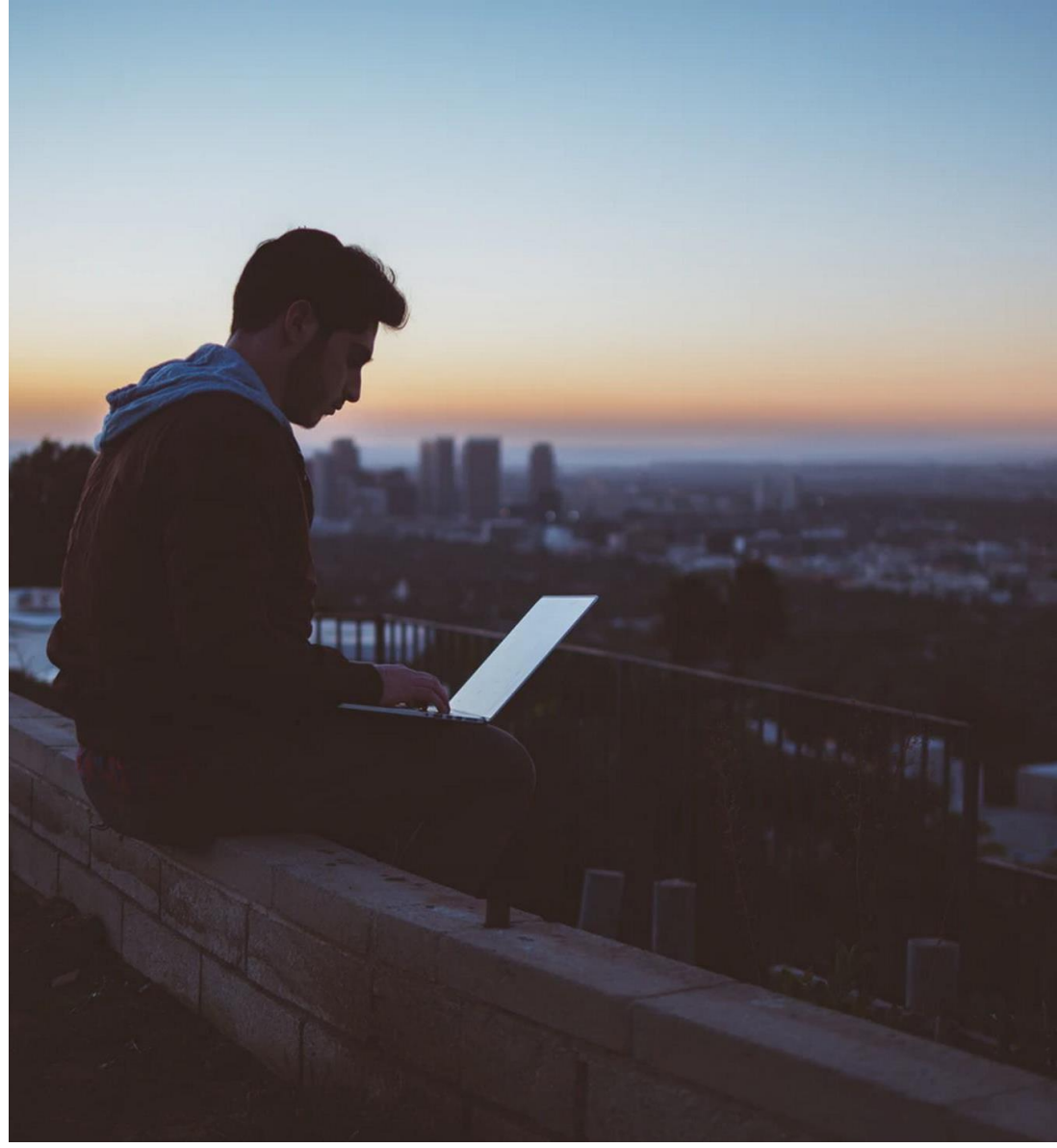
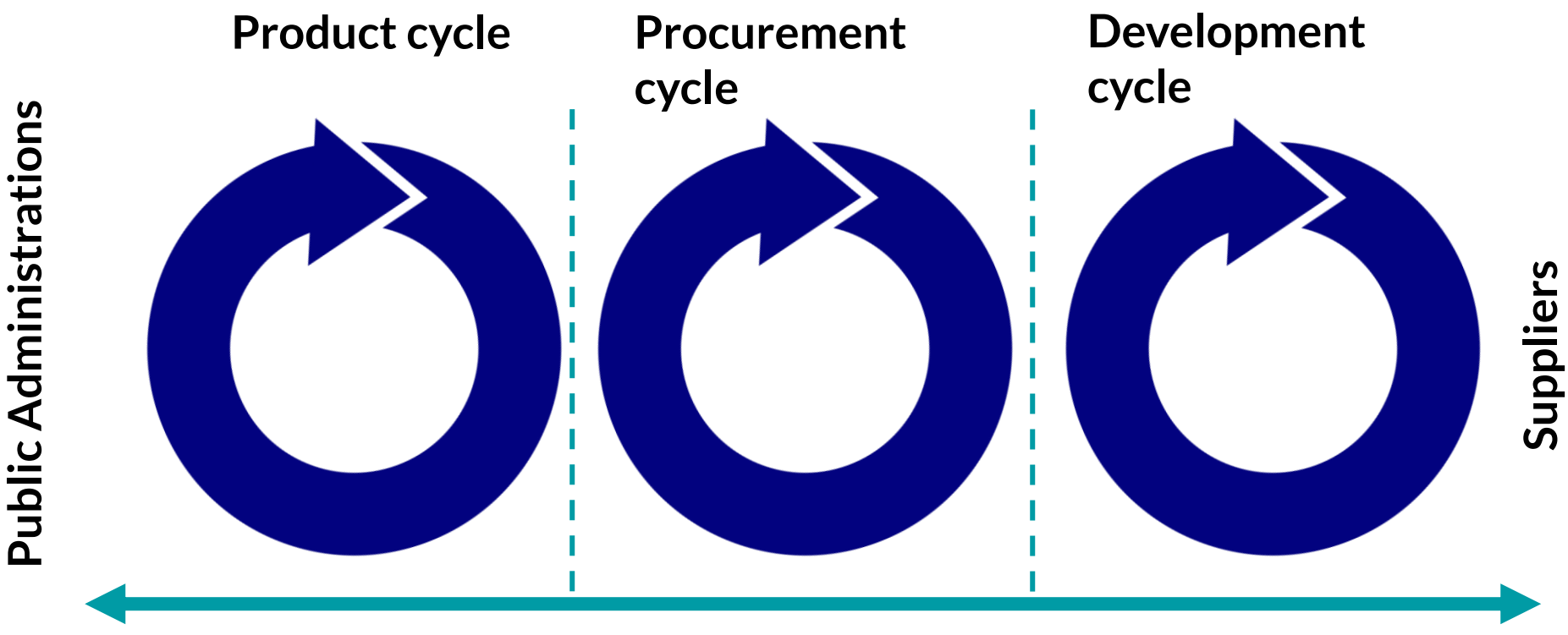


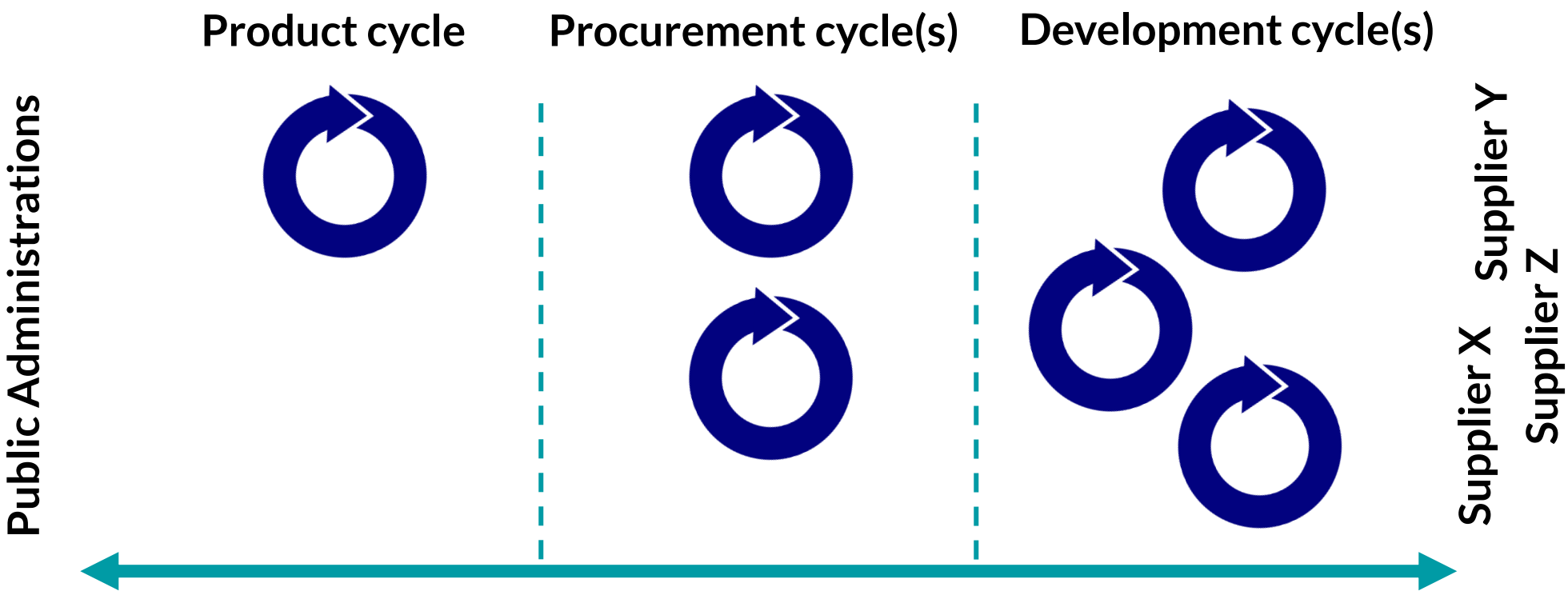
Photo by Avi Richards | https://unsplash.com/photos/Z3ownETsdNQ

Bridging sequential and agile processes

- Procurement is (typically) a one-way sequential (waterfall) process, from requirements specification, to procurement, to realization.
- Development is (nowadays mostly) an iterative (agile) process where development is carried out in smaller cycles.
- How do we bridge these two worlds?

Photo by Chris Liverani | https://unsplash.com/photos/9cd8qOgeNIY


Product cycle
Procurement cycle
Development cycle
Public Administrations
Suppliers


Product cycle
Procurement cycle(s)
Development cycle(s)
Public Administrations
Supplier X
Supplier Y
Supplier Z

Dynamic Purchasing Systems

- An ”open framework agreement” where suppliers, who meet basic qualifying requicements, can join dynamically during the DPS’ lifetime.
- May enable a dynamic and modular development with a bazaar of developers and users
- Tickets as tenders
  - Pull requests as solution proposals
  - Challenge: tool support not mature
- Challenge: tool support not mature
- (+ culture, processes, training, etc.)

Photo by Kazuo ota | https://unsplash.com/photos/5r935yRz8Kg

Need for collaboration and coordination

- Gather resources and expertise
- Coordinated process for requirements management, procurement, and follow-up
- New development
- Maintenance
- Common financing and governance models ...

Photo by Hannah Busing | https://unsplash.com/photos/Zyx1bK9mqmA

Municipal Foundation for OSS

- OS2 –Association of 70+ Danish municipalities
- Projects are initiated by smaller numbers of municipalities with development procured from ecosystem of vendors
- Governance and project development process in place. Vendors sign an MoU.
- Copyright transferred to OS2.
- Technical committee responsible for long term maintenance together with vendor(s). Procurement through municipalites directly.
- Additional municipalites can join at any time. Finacial logic based on size.

Photo by Nick Karvounis | https://unsplash.com/photos/3_ZGrsirryY

Civil Society foundation for OSS

- Open Cities (Otevřená Města) – a Czech non-profit gathering 20 cities in Czech republic to support their digitalizations
- Recieves and hosts OSS projects initaited by public entities. Facilites joint requirements engineering and planning
- Currently hosts three projects, including Cityvizor - an open source tool for transparent municipal management
- Collaborates with civil tech and hacker communities
- Features implemented by municipalities directly. City of Prague a lead user. Collaboration with public service provider.

Photo by Ouael Ben Salah | https://unsplash.com/photos/S2Wek5-5oEc

Lead User leading the way

- Lutece – an e-service platform with an ecosystem of 400+ plugins
- Developed by the City of Paris since beginning of 2000s
- Drives the development through internal IT department. Open to conributions, onboarding new cities and universities
- Pre-packeging containerized solutions and providing as-a-service
- Working to grow community nationally and cross-border

Photo by Chris Karidis | https://unsplash.com/photos/nnzkZNYWHaU

Co-owned service provider

- CommunePlone – a set of tools for civil servants and modules for e-government solutions
- Initiated as grass-roots project initiated by two individuals from two municipalities
- Now hosted and developed through the public service company IMIO, co-owned by 120 Wallonian municipalities.
- Facilitates collaborative requirements engineering process and offering necessary services to operationalize tools developed in CommunesPlone

Photo by Despina Galani | https://unsplash.com/photos/dq2yziptkFU

Evolving from one to the other

- Signalen - a tool for creating, sending, receiving and handling reports about the public space
- Developed in the wild between municipalities supported by civil society Foundation for Public Code
- Transitioned into Dutch Association of Municipalities with formalized governance and technical steering
- Development mainly performed by Amsterdam. Ambition to onboard others and grow community cross-border

Photo by Callum Parker | https://unsplash.com/photos/ClzrGdUNfXc

Follow-up on development

- Open development enables
  - continuous monitoring and follow-up on planning, development, and delivery
  - possibility to engage in requirements discussions and provide a product owner’s perspective
  - review of quality and security aspects as work progresses

Photo by Headway | https://unsplash.com/photos/5QgIuuBxKwM

Overall procurement strategy

- Consideration of how open source software should be taken into account in connection with acquisition and procurement
- Enable synergies between current open source software projects
- Promote interaction between operations and procurement
- Promote the establishment of common management models and forms of collaboration
- Promote increased knowledge and guidance regarding the handling and consideration of open source software

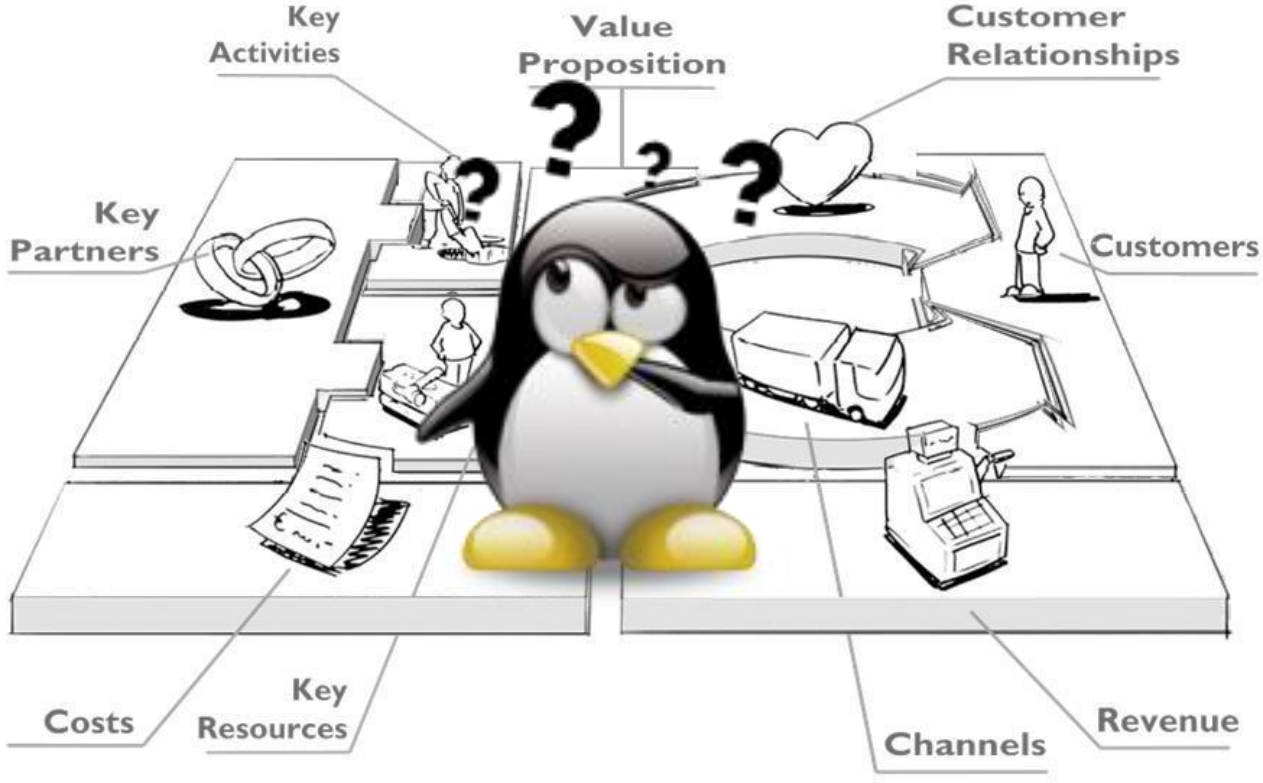
Photo by Heidi Walley | https://unsplash.com/photos/YuAL07NpidY

Need for training, culture, and resources

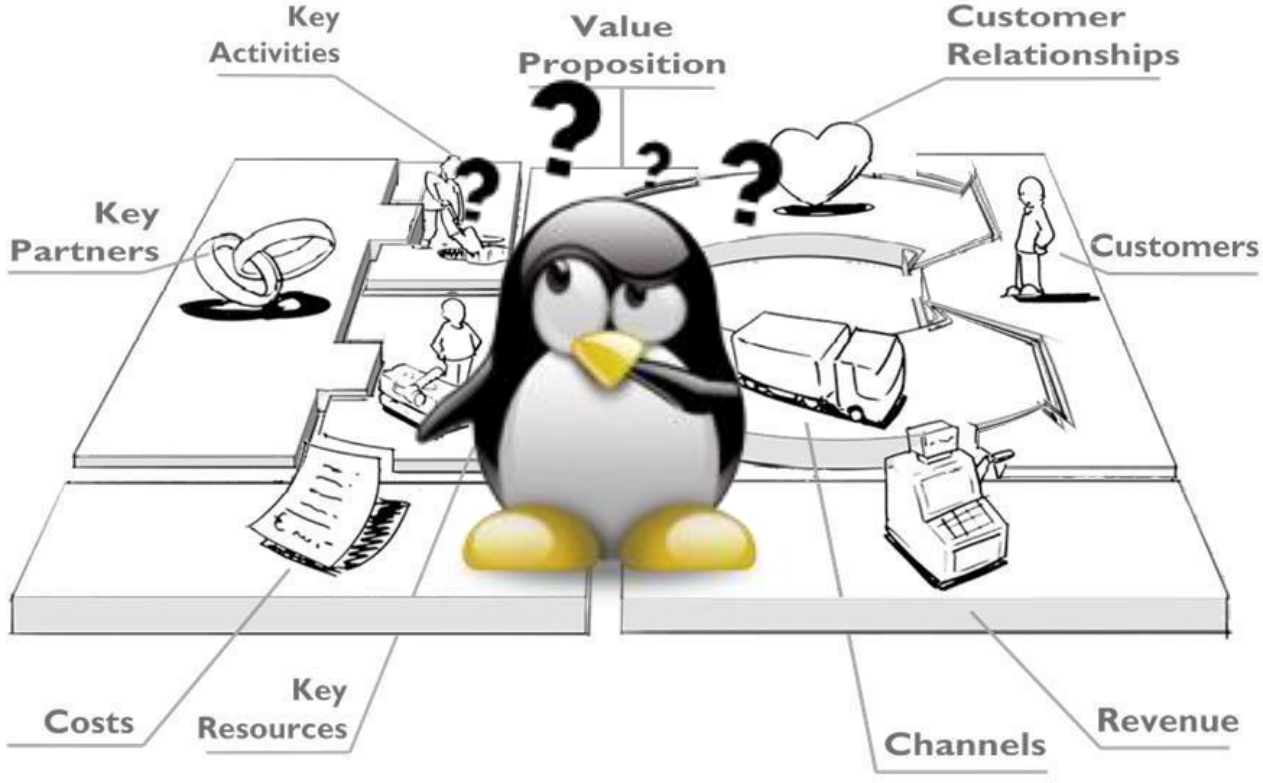
- Catalog of open source software
- Clear process for needs analysis
- Guiding examples of procurements
- Estimation models for value as well as risks and costs
- Example requirements for procurements
- Evaluation models of open source software projects
- Evaluation models of suppliers
- Management and collaboration models for authorities

Photo by Lucas van Oort | https://unsplash.com/photos/_FjlWDrtfmU

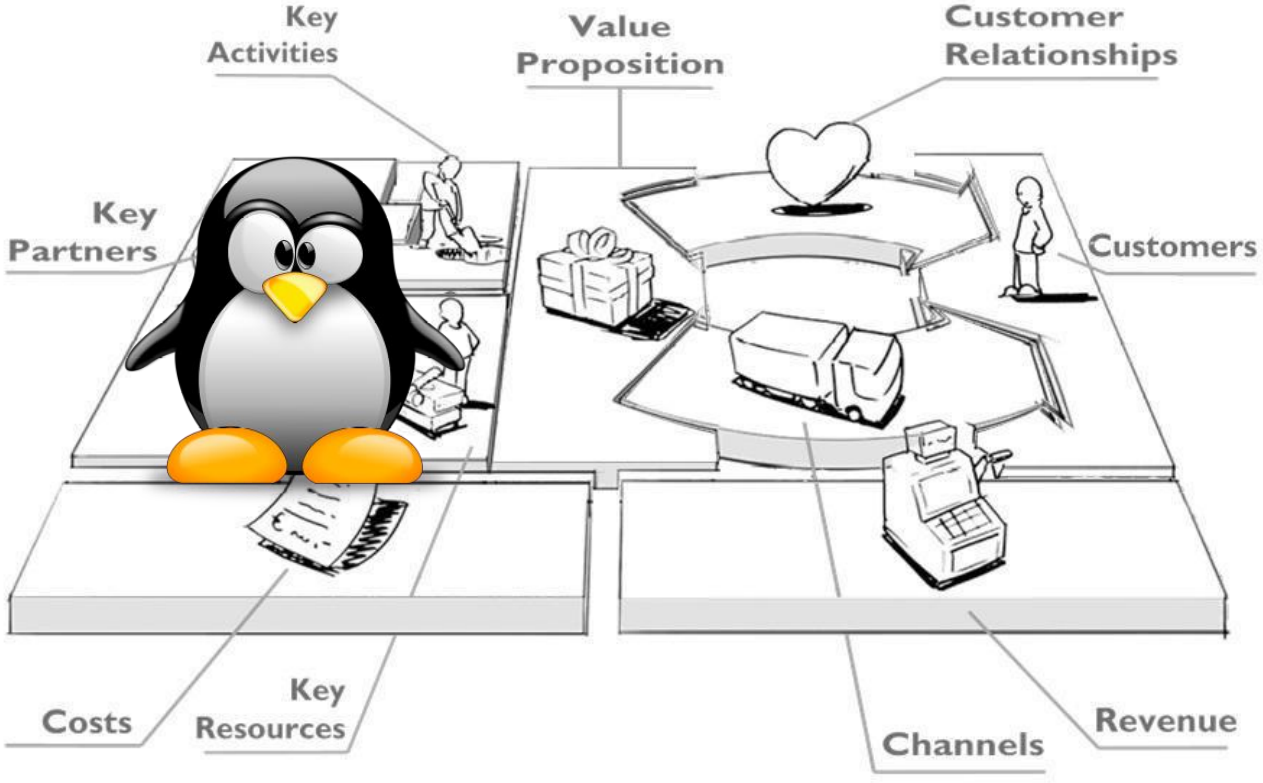
Open Source and Business models



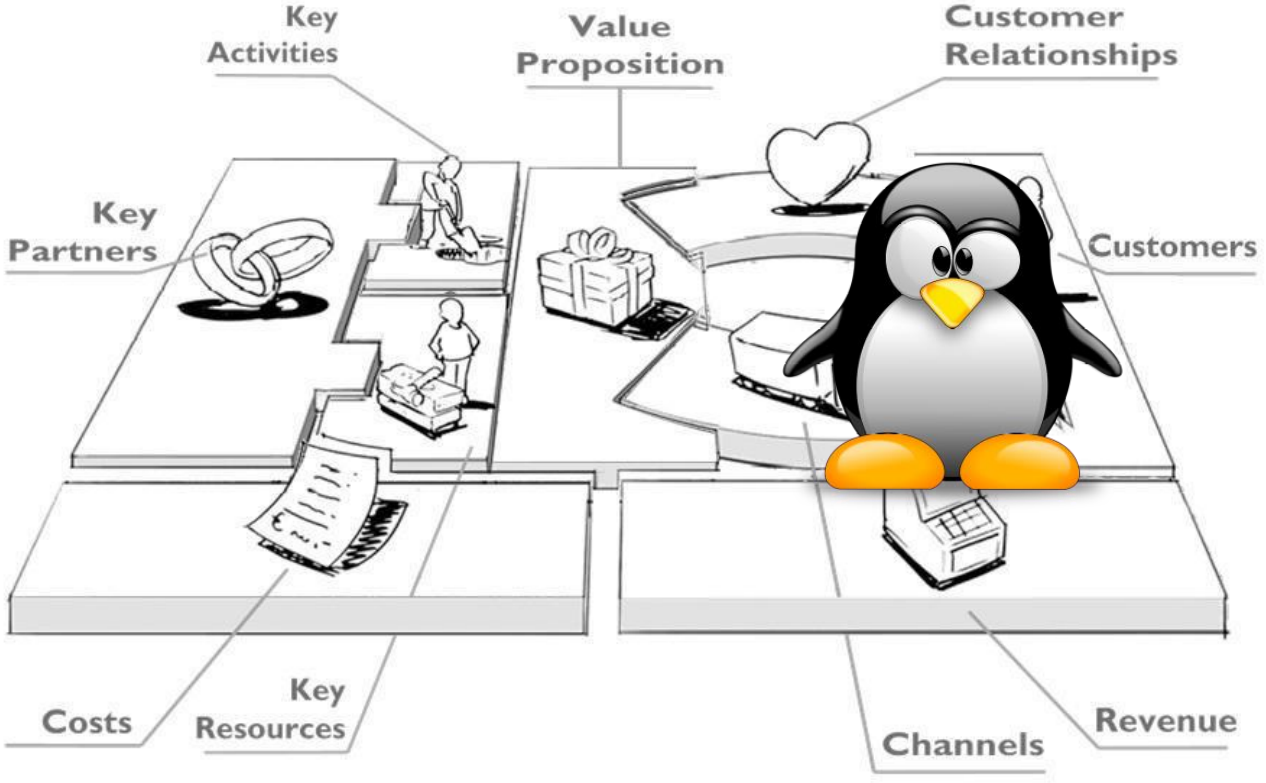
Open Source and Business models



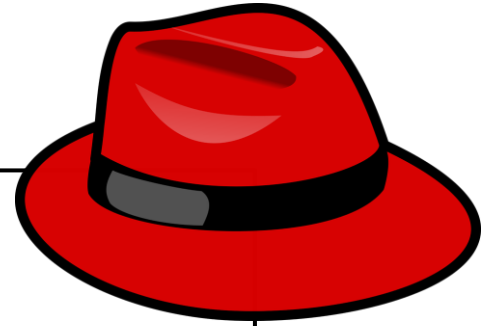
Building block and complement



Distribution, CRM, Sales funnel



Example – Red Hat



Key partners

- Kubernetes community
- Linux kernel community
- ...
- Service providers

Key activities

- 24x7 support
- Open Source development
- Influence communities

Key resources

- Internal engineers
- Open Source projects
- Trust and reputation

Value proposition

- Stable and secure enterprise-grade open source-based software products
- Support- och consultancy
- Slow but stable releases

Customer relations

- 24x7 support
- Communities

Channels

- Redhat.com
- Global organisation

Customer segments

- Large incumbants and public authorities
- Self-serving users

Costs

- Engineers' salaries
- Community building
- Support and maintenance

Revenue streams

- Subscription fees for support and updates
- Fees for consultancy, training and certifications

RI.
SE

Different business model patterns

Support & Subscriptions

- Offers support, operation and maintenance, as well as updates for an open source software
- Guarantee of security and quality in a business-critical environment

redhat.

HORTONWORKS®
POWERING THE FUTURE OF DATA™

CANONICAL

RI.SE

Open Core & Prop. extensions

- Offers additional functionality to the open source software
  - either as modules/plugins for the open source software, or
  - through distributions where the open source software forms the core

RI.
SE

Dual-License

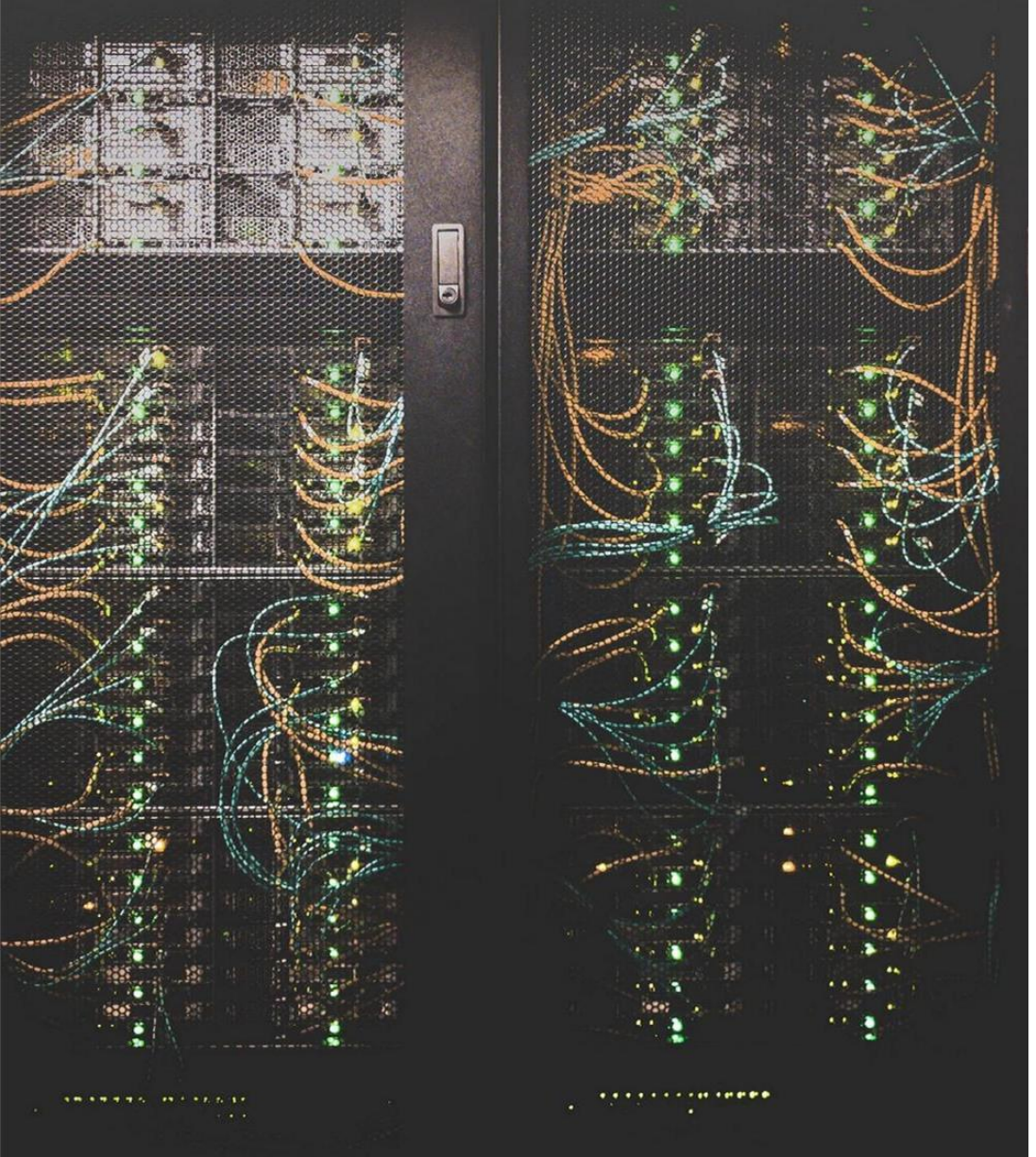
- The company owns all copyright to the open source software and can release it under multiple licenses
- Licenses it with a strict (copyleft) license that requires users to license connected code under the same license
- The customer pays to access the software under a proprietary license without copyleft requirements

mongoDB

MySQL

neo4j

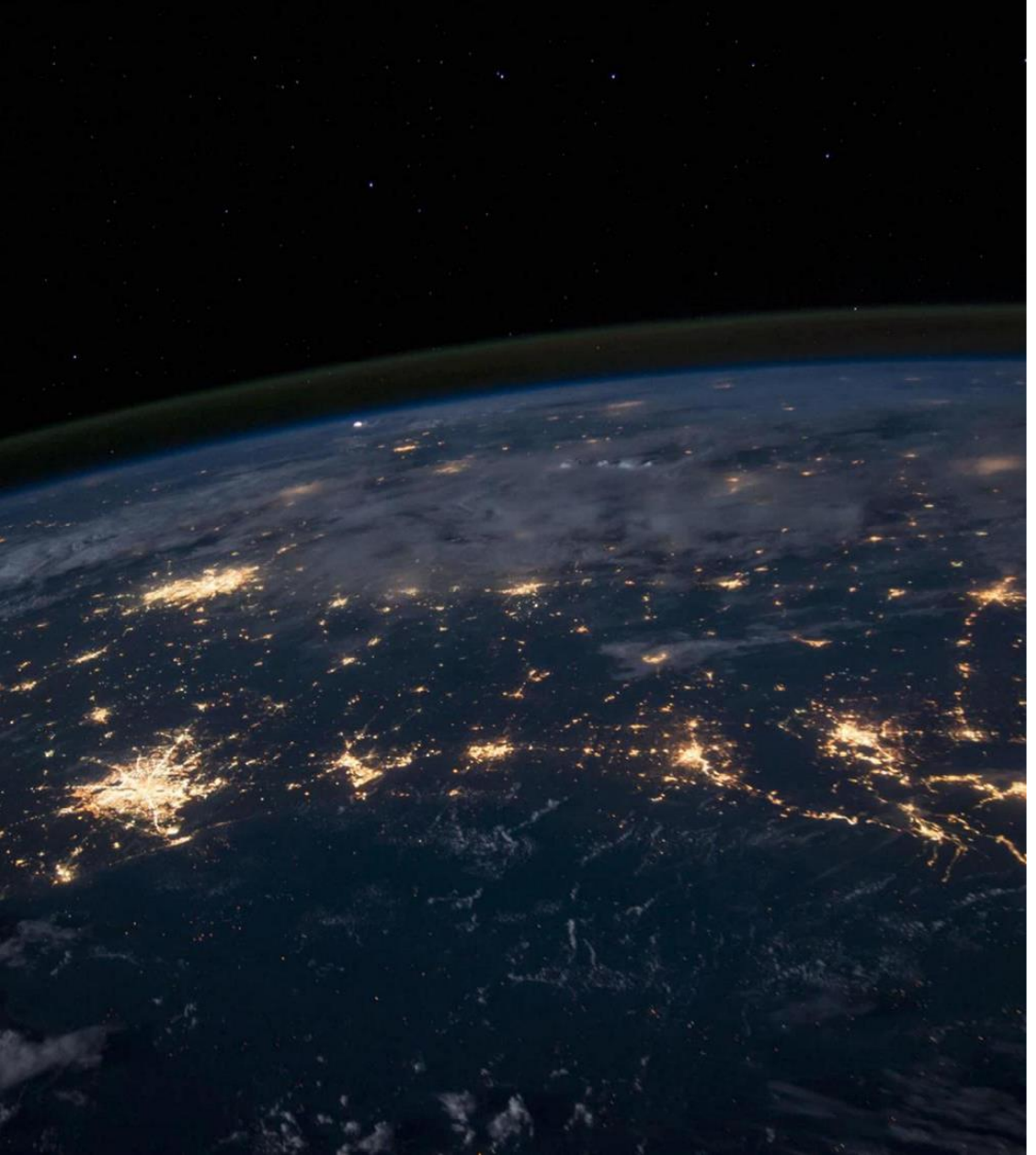
RI.
SE



X-as-a-Service

- Offers the open source software as a cloud service (Software-as-a-Service) or via the customer's own data centers

RI.
SE



Data Driver

- The open source software collects data that can be used for alternative value propositions
- Also called an asymmetric business model

RI.
SE

Product enabler

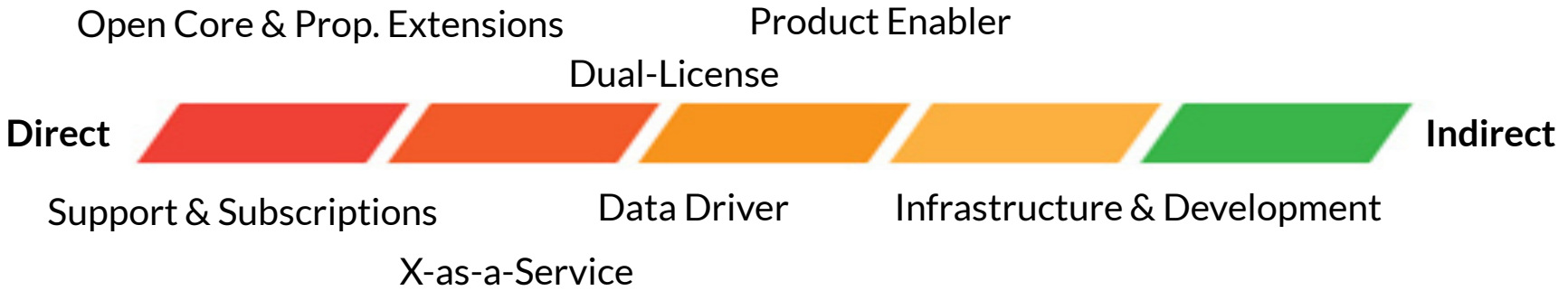
- The open source software is embedded in a hardware product to enable its intended functionality
- Trick question – which products runs on open source such as Linux/GNU?

RI.
SE

Infrastructure & development

- The open source software is used to enable product development, operational activities, and delivery/distribution of a product or service.

RI.SE



Connection to value proposition

- E.g., – Neo4j
  - Dual license + proprietary extensions + SaaS + Services

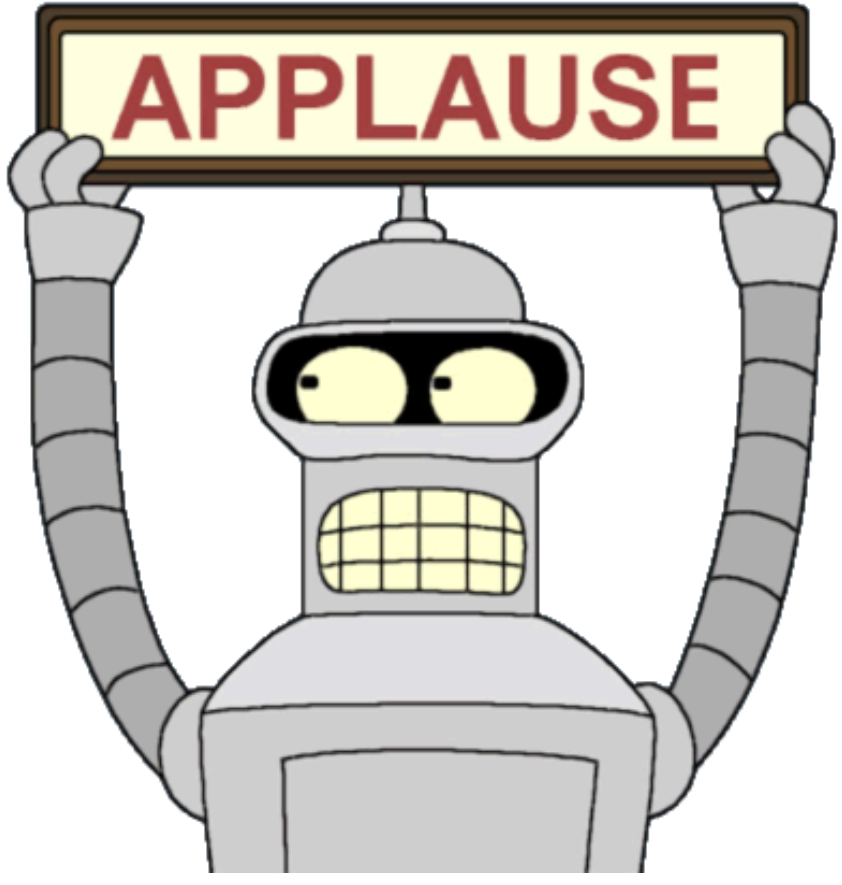
Often used in combinations >>

neo4j

RI. SE

Important to separate between…

- Community vs. Customers
  - Community consists of users, of which some maybe can become customers
- Community vs. Partners
  - Community does not any have contractual obligations
- Projects vs. Products
  - Projects are freely available, products are what you sell



RI.
SE