Acquisition and Development of Open Source Software in the Public Sector

Johan Linåker

RI.
SE

Guidelines in Sweden

- Soft and implicit guidelines compared to e.g., Italy, France, and Estonia

- *"The public administrations' e-services should, as far as possible, be based on open standards and use open source software and open source software-based solutions to gradually remove lock-in to individual platforms and solutions."* - Swedish E-delegation

- *"Open Source Software should always be considered pending that it fulfills all requirements and the total cost of ownership is reasonable."* - Swedish Insurance Agency

- *"Software that is developed or acquired should (in first hand) be published as open source software."* - Swedish Agency for Digital Government

Reuse of existing
Open Source Software

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

Are there open alternatives?

- In the preparatory phase of an acquisition...

  - Investigate existing alternatives

    - Software catalogs, networks, RFIs

  - Download, test, and cross check against requirements specification

  - Missing requirements critical? Can they be developed at a reasonable cost? Is it possible to upstream?

Project health?

- How secure and sustainable is the Open Source Software?

- Do we need to procure support or a packaged service to guarantee quality and availability?

- For checklists, see:

  - *https://chaoss.community*

  - *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?

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

Qualification requirements 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

Weighing the different options*

	Proprietary option	OSS community ed.	OSS enterprise ed.
Procurement			
Security			
Cost			
Organization			
Technical reqs			

- Comparison from a pre-study of e-archival solutions by the Swedish Governmental Service Center
- (Record ID: 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*

Development of a new
Open Source Software

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

Costs, risks, and other complexities

- 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
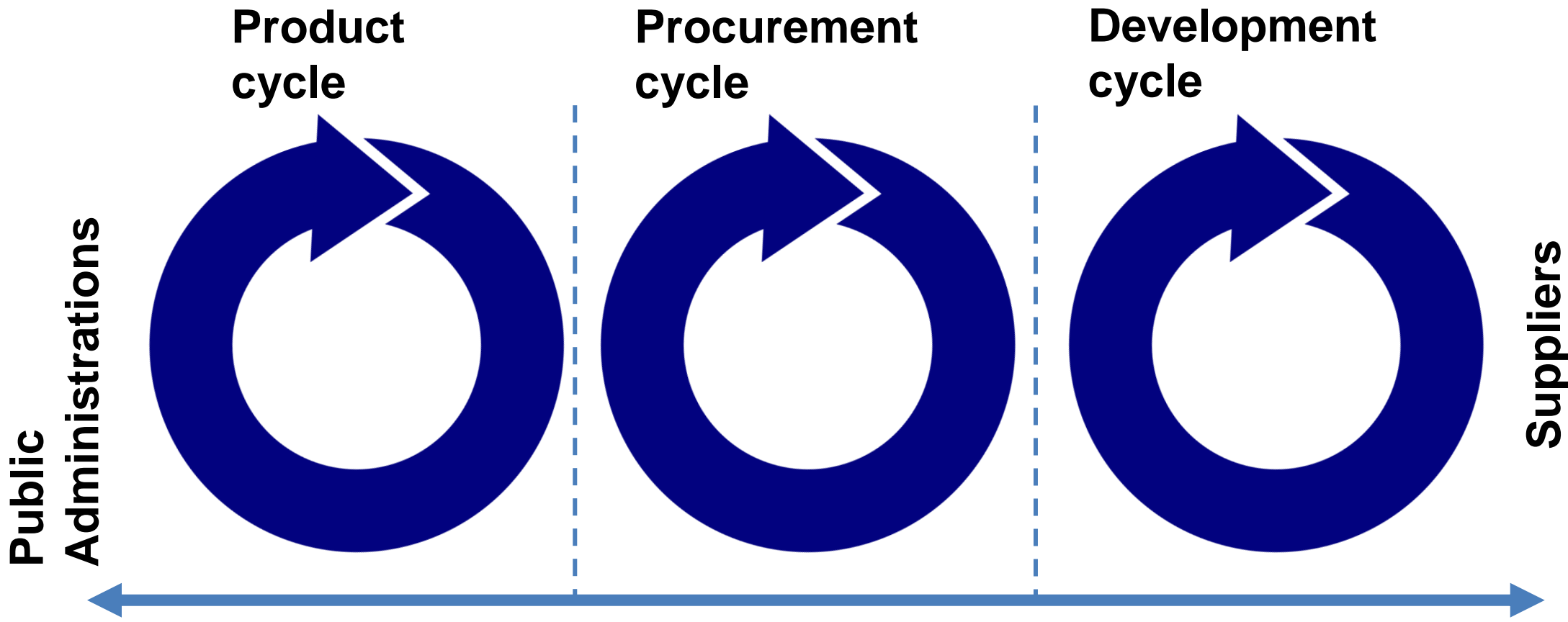
Be Open from the start

- 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*
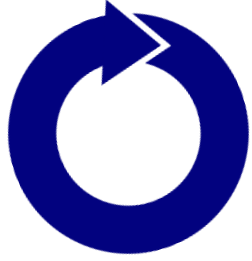
Bridging sequential and agile processes

- 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.
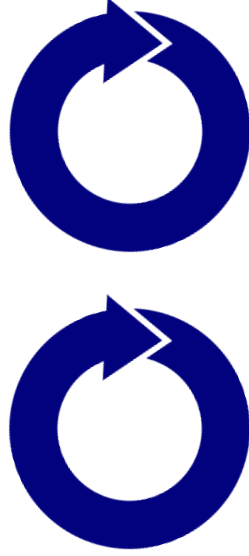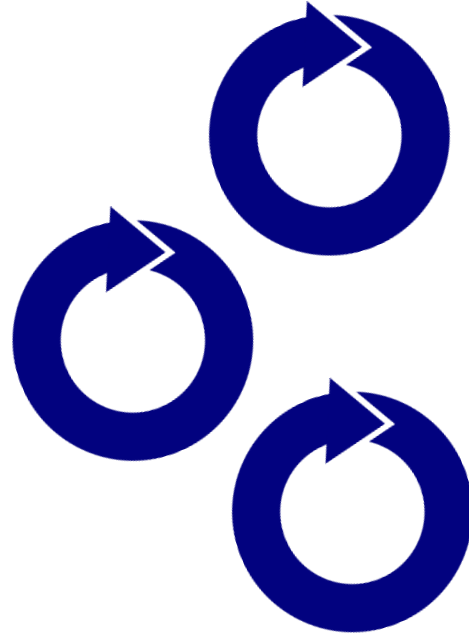
- How do we bridge these two worlds?

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

- (+ culture, processes, training, etc.)

Need for collaboration and coordination

- OSPO-model: National and regional competence centers in Italy

- Foundation-model: OS2 in Denmark, a (mainly) municipal collaboration

- Network-model: NOSAD in Sweden, an open network for public sector to share and develop knowledge on how to use and collaborate on Open Source Software

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

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

- General strategy for how to consider Open Source Software during an acquisition process that enables

  - synergies between projects

  - collaboration between operations and procurement office

  - common governance models and forms of collaboration

  - training and knowledge sharing

Need for training, culture, and resources

- Catalog over Open Source Software

- Process for procurement planning

- Real-world examples and case studies

- Cost, and risk evaluation models

- Example requirements for tenders

- Evaluation models for Open Source Software projects

- Evaluation models for suppliers

- Maintenance and collaboration models for public administrations

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