Best practices for helping Public Administrations work with Open Source

Malcolm Bain
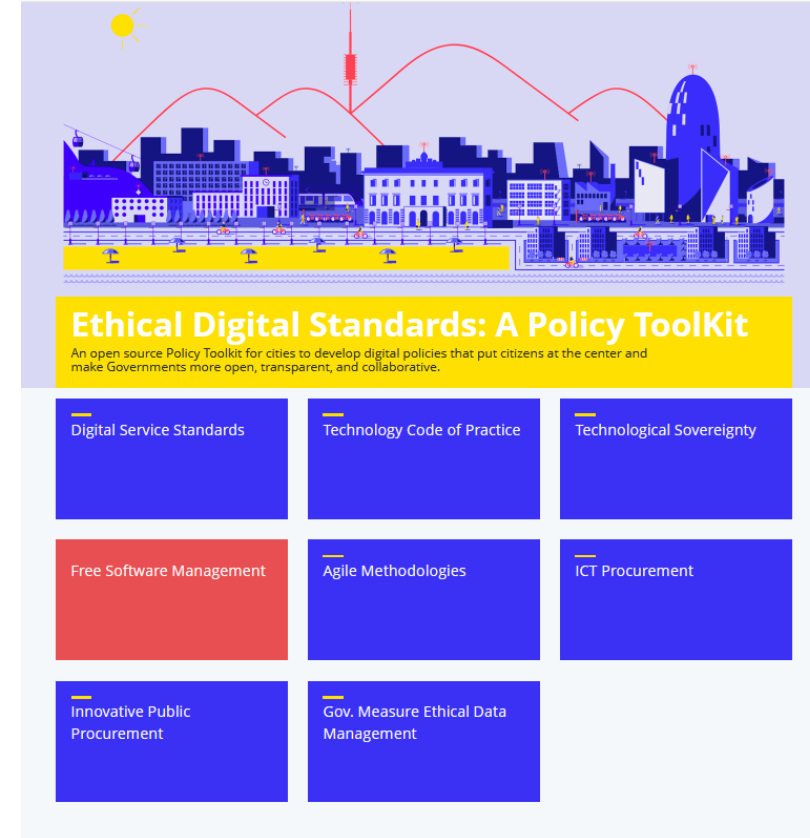
Across Legal

Spain

2 Examples of OS management best practices

OpenChain Compliance Specification

Barcelona Free Software Management Guidelines

OpenChain Compliance Specification... Adapted

FOSS Compliance best practice: OpenChain

OpenChain Specification

Version 2.1

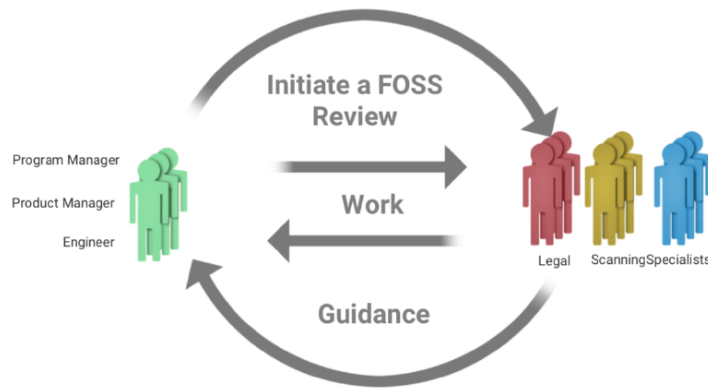OPENCHAIN 2.1

- **ISO/IEC 5230:2020**

Inbound → Training / Policy / Process → Outbound

The Process!

Working through the FOSS Review

OPENCHAIN

Initiate a FOSS Review

Work

Guidance

Program Manager
Product Manager
Engineer

Legal   ScanningSpecialists

The FOSS Review process crosses disciplines, including engineering, business, and legal teams. It should be interactive to ensure all those groups correctly understand the issues and can create clear, shared guidance.

Note: ISO standard now

The Specification!

The OpenChain Specification is about confirming a company has open source processes, policies and training.

The website!

www.openchainproject.org

Open Compliance Program Solutions

Processes

Bill of Materials

Tooling

OPENCHAIN

OpenChain Defines Process Inflection Points

upstream

downstream

inbound

Training

Policy

Process

outbound

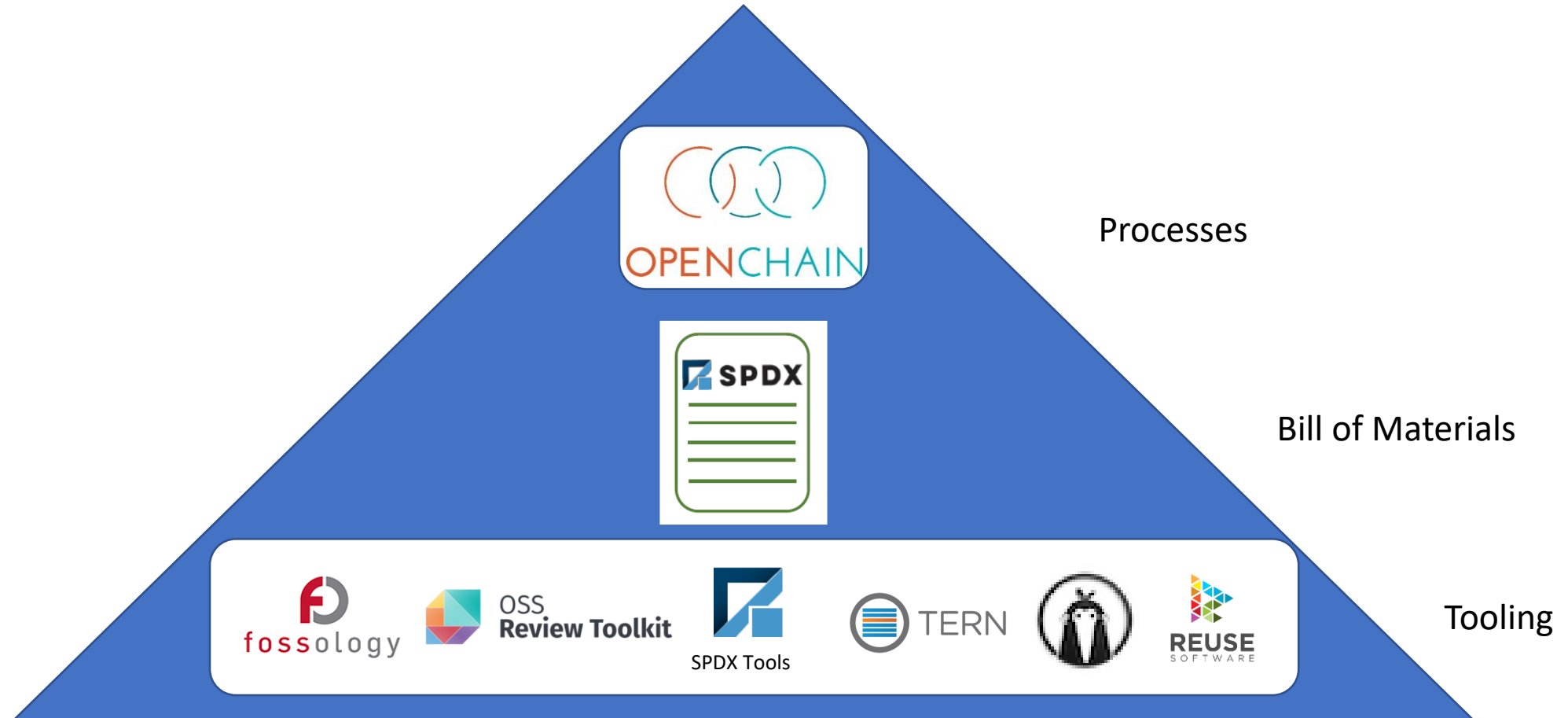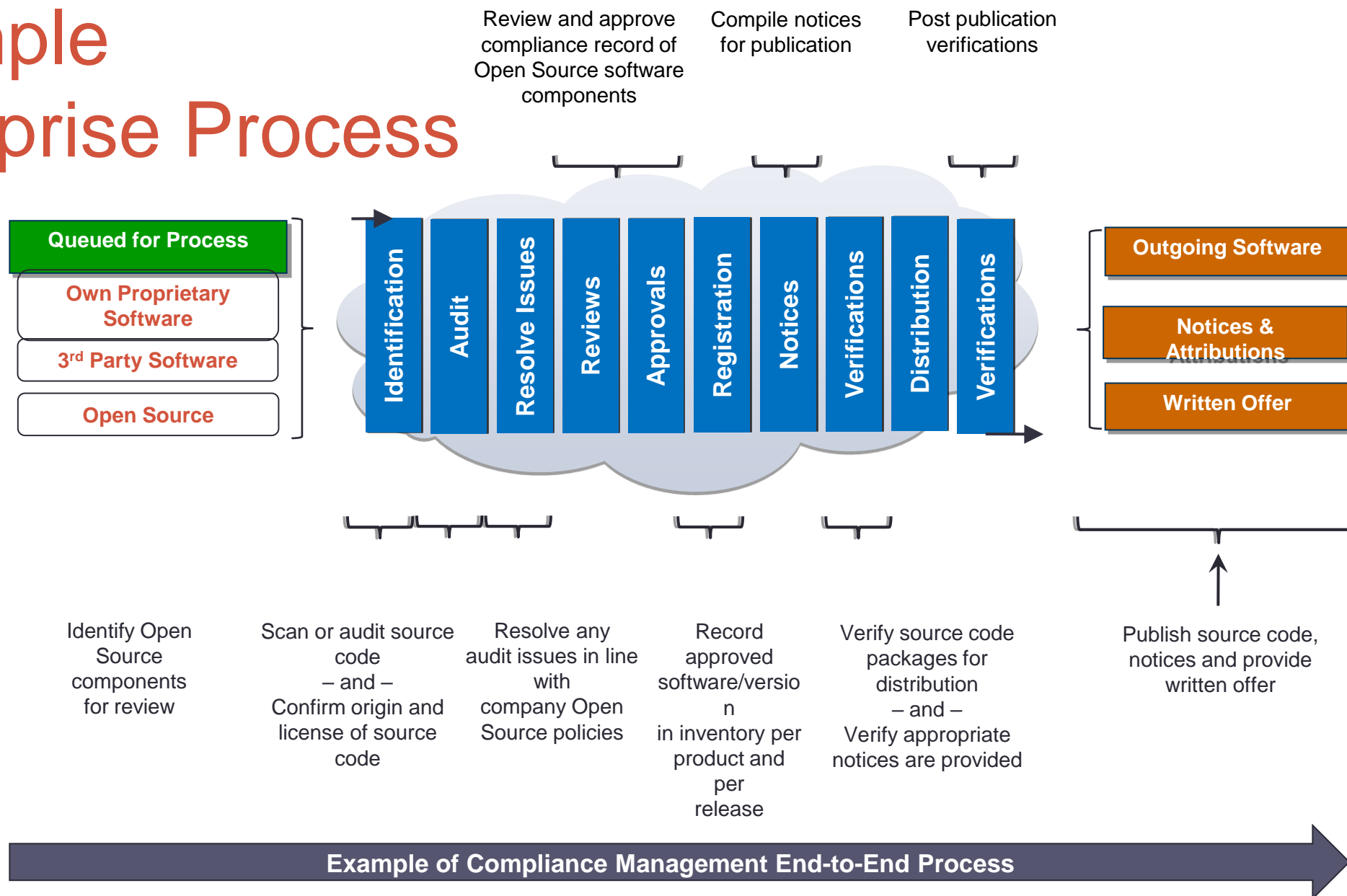OPENCHAIN

Working through the Open Source Review
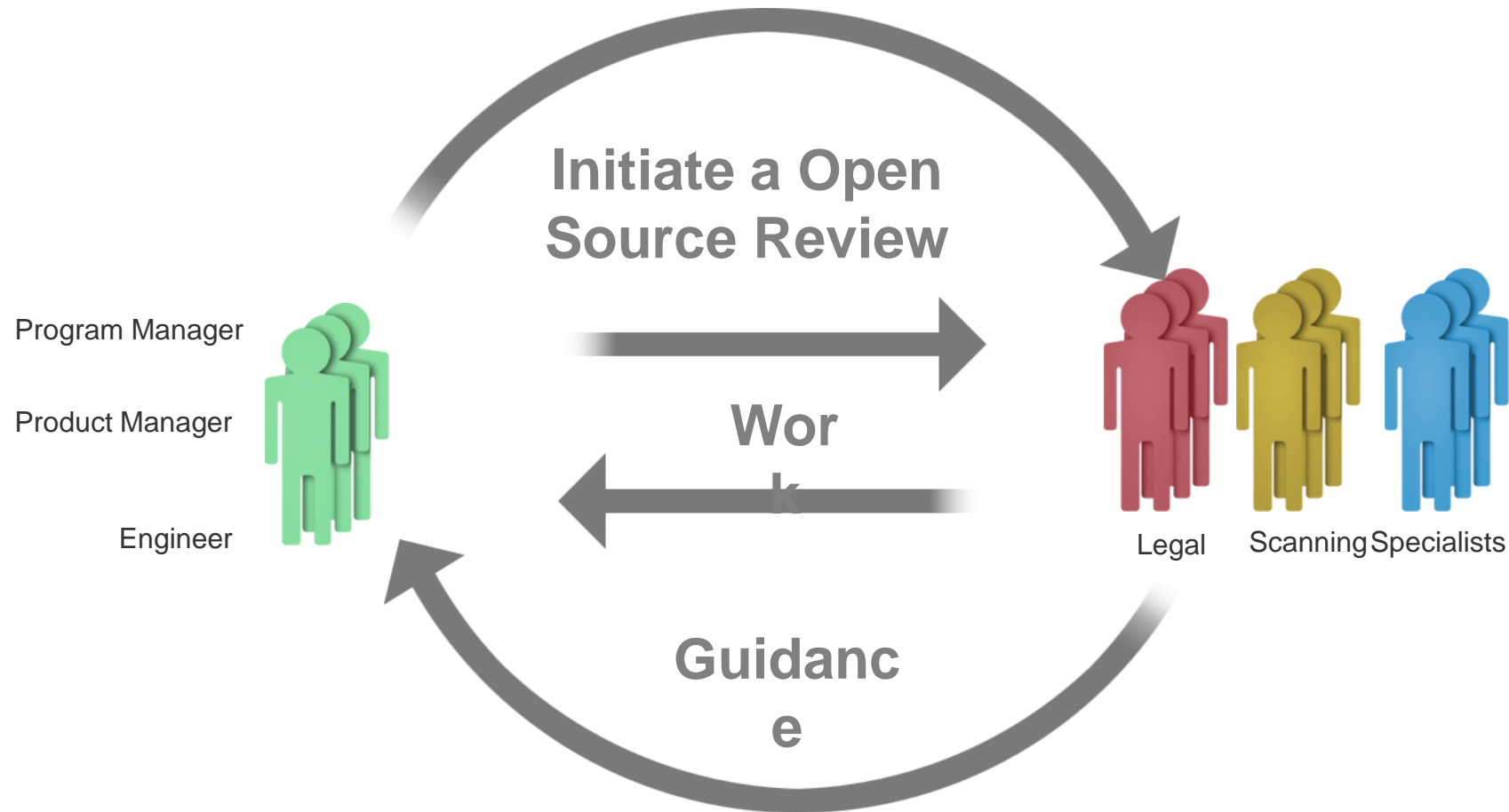
The Open Source Review process crosses disciplines, including engineering, business, and legal teams. It should be interactive to ensure all those groups correctly understand the issues and can create clear, shared guidance.

Example Enterprise Process

Review and approve compliance record of Open Source software components

Compile notices for publication

Post publication verifications

Queued for Process

Own Proprietary Software

3rd Party Software

Open Source

Identification | Audit | Resolve Issues | Reviews | Approvals | Registration | Notices | Verifications | Distribution | Verifications

Outgoing Software

Notices & Attributions

Written Offer

Identify Open Source components for review

Scan or audit source code
– and –
Confirm origin and license of source code

Resolve any audit issues in line with company Open Source policies

Record approved software/version in inventory per product and per release

Verify source code packages for distribution
– and –
Verify appropriate notices are provided

Publish source code, notices and provide written offer

Example of Compliance Management End-to-End Process

Adaptation of OpenChain for Public Administrations

Use OpenChain as schema for the project

- Specification has 5 Requirements + Conformance check
  - **Program Foundation**: FOSS Policy, Competence, Awareness, Scope, Licenses
  - **Roles and Tasks**: Access points, (human) Resources
  - **Review and Approval**: Bill of Materials, License Compliance,
  - **Evidence** (Artefacts): Compliance Artefacts
  - **FOSS Engagement**: Contributions to projects

Adapt to Public Administration

- Simplify processes
- Distinguish R+D/Innovation and production projects
- Set up Open Source Competence Centre/Project Office
- Collaborative sharing among public entities as a "supply chain"?

Implementation Objectives

Overall mission

Establish FOSS Policy and Procedures to enable and support use and development of FOSS and ensure Licensing Compliance in PA FOSS projects

Goals

Establish institutional position and high level policy regarding FOSS	Establish roles, procedures and documents/templates to support the FOSS Policy	Implement policy and verify compliance of existing and new projects	Train of staff in FOSS management	Promote Culture change

Method: Keep if fairly agile: iterations - Pilot projects with staff feedback

PROJECT PLAN – FIRST ITERATIONS

LEVEL	TASKS	DELIVERABLRES
POLICY	Create FOSS Policy Adapt for PA	FOSS POLICY DOCUMENT Start simple, Progressively Updated version
DESIGN	Design, develop all model documents, processes, and identify tools for FOSS compliance Design "competence centre" and FOSS Core Team: knowledge base for FOSS reference in PA	FOSS compliance design and documentation (templates, flow charts, "How-to" and other docs) Documentation for Competence Centre, integrated with existing Dept processes and documentation Tools to embed in PA platforms
IMPLEMENT	Deploy processes and documents and tools iteratively through several projects, fine tuning	FOSS documents, tools and Artefacts for 1 → 3 projects 1st project independently 2 and 3 project nearly in parallel.
TRAIN	Develop adapted training materials for all PA-IT staff Integrate into online learning environment Run courses iteratively (groups of 5-10 IT staff)	Training materials Online course delivered for identified groups Update and Feedback

FIRST ITERATION

Initiate
- Stage 1: Understand FOSS Program Foundation
- Stage 2: Establish FOSS Task definition, Processes, Documentation

Development
- Stages 3 – 5: Implementation of FOSS Program

First deliverables
- FOSS Program Policy and Scope
- FOSS Program procedures
- FOSS Program documents
- FOSS Program Tools and infrastructure

Piloting
- Define first pilots and team

Setup FOSS Competence Centre / Project Office

The creation of the *Open Source Project Office (OSPO):*

- Focus for understanding and awareness of FOSS
- Establish processes and materials
  - Clearance processes
  - Long term ensurance of Compliance (repository,source code,licenses)
  - Training materials and process for researchers
- Q+A for IT and Project management staff
- Coordination among PA centres, University, Private Sector, Community

FOSS Project Office (OSPO)

First iteration:
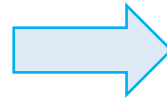4 initial sprints

| Dimensioning and defining the scope of the CC	Defining and setting up the infrastructure	Pilot Projects	Building a community

Infrastructure

- Define the repository
- Define a set of tools for software review and management
- Define set of tools for projects management

→

- Repository set up
- List of operative tools

Piots

- Retrospective with previous pilots
- Identifying new pilot projects and plan
- Working with each project
- Artefacts/evidences productions

→
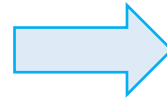
- Project plan
- Bill of materials
- Artefacts
- Compliance certificates

Community

- Email address for FOSS inquiries (external and internal)
- Creating a protocol for addressing inquiries
- Communicating the FOSS compliance protocol
- FOSS forum + Newsletter
- Community infrastructure

→

- Periodic newsletter
- Email address
- Community infrastructure: repository and Slack channel
- Information repo: FOSS protocol, license compatibilities, Training materials

Barcelona Free Software Management Guidelines

Barcelona Free Software Management Guidelines

https://www.barcelona.cat/digitalstandards/en/free-soft/0.2/introduction

- **Introduction**

Barcelona City Council has a policy for using and developing free software which is set out in the Government Measure: Free software and agile development of public administration services, approved in October 2017.

This policy includes a series of principles and guidelines that are outlined in the following documents, annexed to the measure:

- Technology Code of Practice

- Guide to Technological Sovereignty

- Guide to Public ICT Procurement

This Guide spells out these principles and guidelines in a series of measures and recommendations to help staff of the Municipal Institute of Information Technology (IMI), the ICT body of the Barcelona City Council, in their daily work on projects based on free software. It covers aspects such as:

- Looking for and designing solutions

- Service procurement

- Developing, implementing and maintaining information systems

- Participation in free software communities

Document organization

This Guide contains a series of guidelines that we have divided into:

- **Measures** that must be adopted and implemented in all projects.
- **Recommendations** that do not have to be followed but which could be useful in certain situations.

Each measure, recommendation or alternative consists of:

- A unique identifier throughout the document.
- A list of tags.

The **tags** help to restrict when and in what types of projects each measure and recommendation are important. For example, they can serve to generate a checklist of things that project management needs to check.

Tags for Use-Cases and Project Phases

Tag	Description
Integration	**Integrate existing components**: it is necessary to install a new system that can be built on the basis of existing components.
Adaptation	**Modify an external component**: feature needs to be added to a component that is already available, or it needs to be improved by solving defects, adding translations, etc.
Plugin	**Expand a platform with plugins**: new components need to be created that can be integrated into a platform or framework already available and which facilitates this task.
NewProduct	**Create a new component**: a new solution needs to be created from scratch.
Publication	**Publish code from an existing own component**: we have our own code, developed under previous contracts and never publicly distributed, and we want to make it available in a public repository, under a free software license.
Document	**Publish documents not linked to a single project**: we have documentation that is not directly linked to a piece of software but we want to make it freely available in a public repository.

Tag	Description
Preliminary design	Measures to take into account when drawing up preliminary designs.
Procurement	Measures to take into account when drawing up the service procurement specifications.
Day1	Measures to be applied from the first day of the development stage (see the section).
Release	Measures to take into account when a new version of the product is released.

e.g. Day 1 actions

ID	Title	Type	Tags
A_D4F	Link the main repository to a public issue manager	Alternative	Day1; Adaptation; Plugin; NewProduct; Publication
M_F25	Upload a README file to the main repository	Measure	Day1; Plugin; NewProduct; Publication
M_4F5	Publish brief Developer Guidelines	Measure	Day1; Plugin; NewProduct; Publication
R_368	Link the main repository to a free software continuous integration system.	Recommendation	Day1; Adaptation; Plugin; NewProduct; Publication
M_97E	Upload the licence text to the main repository	Measure	Day1; Plugin; NewProduct; Publication
M_A60	Create the main repository in the City Council GitHub public software forge	Measure	Day1; Adaptation; Plugin; NewProduct; Publication
M_A63	Use the GitHub main repository web interface as the project development website	Measure	Day1; Plugin; NewProduct; Publication
M_35A	Link the main repository to a public issue manager	Measure	Day1; Adaptation; Plugin; NewProduct; Publication
R_2D5	Reserve a permanent project URL and always use it to refer to the project	Recommendation	Day1; Integration; Plugin; NewProduct; Publication
M_7EA	Implement and document build and installation procedures with widely used free tools	Measure	Day1; Plugin; NewProduct; Publication
M_CC5	Upload a file with installation instructions to the main repository	Measure	

E.g. Measures

Free Software Management
v 0.2

Introduction

Document organization

▶ Collaborative development

Technical infrastructure

Component management

Free software communities

Legal aspects

Documentation

Document organization

📖 Guides glossary
📄 Download PDF

Procurement for collaborative development

Measure 3.2.A: Award the contract to companies with experience in collaborative development

Tags: Procurement   Adaptation   Plugin   NewProduct

Establish the need for free software development experience as a condition of technical solvency.

However many conditions are included in the contract, if the winning company has no experience of participating in free software projects, it is most likely that the product and the development process will end up not following many free software development conventions. In most cases, there is no reason why that should be the result of bad faith but a lack of knowledge.

Alternative 3.2.B: Enter into a subsidiary independent validation and verification (IV&V) contract

Tags: Procurement   Adaptation   Plugin   NewProduct

Hire a company that does have proven experience of sustained participation in free software projects. This company will act as an external project collaborator and carry out code checks and process analyses, reporting directly to IMI.

In an free software project, what is being contracted is not just the code but also the process.

Add this service to the project technical office.

Alternative 3.2.C: Include experience in free software projects as an award criterion

Tags: Procurement   Adaptation   Plugin   NewProduct

Award a set number of points to companies that can certify experience in projects that have produced free software.

Measure 3.2.D: Ask tenderers to provide evidence of participants' experience in free software projects

Tags: Procurement   Adaptation   Plugin   NewProduct

They must do this by providing references for their individual participation in public repositories and forums (StackOverflow, etc.), from projects they have taken part in.

This can be done as a technical solvency criterion or a performance criterion.

e.g. Legal aspects

grow to achieve the objectives and make it sustainable in the long term.

Measure 7.1.A: Establish Barcelona City Council as the owner of the new source code and the new documentation

Tags: Procurement | Adaptation | Plugin | NewProduct | Publication

That is the default option but not the only one. It forces the contractor to assign ownership of the copyrights
tion that it generates.

It particularly makes sense that the City Council should wish to concentrate all property rights on emblemat
tween public administrations, and on those over which it intends to decide to a large extent the rules of gov

Alternative 7.1.B: Assign ownership of the new source code to the legal owner of the origin

Tags: Procurement | Adaptation

When we adapt an existing product, to accept the inclusion of our code in the original product we may be as
Agreement (CLA) or equivalent contract. If not, the parts developed by the City Council can be owned by it.

Alternative 7.1.C: Assign ownership of the source code to an independent non-profit

Tags: Procurement | Plugin | NewProduct | Publication

There is a third option if the aim is for ownership to eventually go to an external organisation responsible fo
the project in the long term. Given the nature of the function, that organisation will normally take the legal f
profit entity. This makes sense in projects where, besides the City Council, other public, private or third sect
and in decision-making. In this case copyright must be assigned by contract, either directly to the correspon
Council, so the latter can assign copyright to the entity at a later date.

The foundation can manage other projects freed by the City Council or of other entities linked with the publi

Alternative 7.1.D: Establish the supplier as the legal owner of the new source code and the

Tags: Procurement | Plugin | NewProduct | Publication

It makes sense to leave ownership in the hands of the company that develops the software when:

Software components are for general use, including companies and not just public authorities.

The supplier is a company or a cooperative with a track record of publishing free software and running free

Measure 7.2.A: Use the AGPL-3.0 licence as the licence for distributing the project

Tags: NewProduct | Publication

The GNU Affero General Public License v3.0 (AGPL-3.0) has all the characteristics we need for City Council projects:

It is a licence with copyleft, as required by Spanish law for public authorities creating free software products, and as it is reasonable to demand
of them to prevent private appropriation of what has been financed using public money.

For applications where users mainly interact via the internet, the law does not allow the creation of a closed service using software with this li
cence (it establishes internet access as a form of distr

The licence's governing body is the GNU project, the m
it is this group of activists and experts who will design
the legal defence against possible attacks on the free

The main reason for choosing this licence as the defa

- It belongs to the GNU GPL family of licences, which
  will have to spend time researching the licence to d

- Opting for more widely used licences reduces the r
  risk caused by the proliferation of licences and thei

It is written in English. For your information, transla
one.

Alternative 7.2.B: Use the EUPL-1.2 licence as

Tags: NewProduct | Publication

The European Union Public License 1.2 (EUPL-1.2) is a

The advantage it has over licences belonging to the G
https://joinup.ec.europa.eu/page/eupl-text-11-12.[htt
legal diversity of the member states as regards copyr

Just like the AGPL-3.0, it has copyleft and internet cop
softer than those of the AGPL-3.0 and more like those
the European courts. The fine details of the difference
https://joinup.ec.europa.eu/news/eupl-or-gplv3-comp

Using this licence (in its latest version, 1.2) should pos

Measure 7.3.B: Upload the licence text to the main repository

Tags: Day1 | Plugin | NewProduct | Publication

The licence will go in plain text in a file called `LICENSE` (no extension), in the repository's root directory.

The text of the two recommended licences (which should be copied word for word) can be found at:

- https://www.gnu.org/licenses/agpl.txt

- https://joinup.ec.europa.eu/sites/default/files/inline-files/EUPL%20v1_2%20EN(1).txt

The `LICENSE` has to be in English. When using the EUPL-1.2 licence, which has official translations, we have the option of including
`LICENSE.ca.txt` and `LICENSE.es.txt` files. The different translations can be found at https://joinup.ec.europa.eu/page/eupl-text-11-12.
[https://joinup.ec.europa.eu/page/eupl-text-11-12].

Measure 7.3.C: Include a copyright and licence notice in each code file

Tags: Adaptation | Plugin | NewProduct | Publication

Most licences stipulate the inclusion of a licence and copyright notice.

This means at the top of all repository code files (except build script and installation files) there must be a notice that explicitly states which
persons or entities are the code's copyright holders and which licence establishes the distribution terms.

It is important to point out under which specific version of the licence it is distributed, and we recommend stating that this will be automatically
updated to future versions when these are released (usually to adapt to unforeseeable technological and social changes), with no need to up-
date all the code files. In the examples given below, this is shown by clauses such as "either version X of the License, or (at your option) **any
later version**" or "version X or – as soon they will be approved by the European Commission – **subsequent versions** of the EUPL".

Obviously, the notice must go in a comment, using the comment syntax each program language uses. And it must also include all the years
when modifications have been made to the file. This would be an example, if we use the AGPL-3.0 on Java code, assuming the copyright holder
is Barcelona City Council:

```
/* Copyright (C) 2017, 2018 Ajuntament de Barcelona
 *
```

Barcelona Free Software Management Guidelines

- Complete handbook for setting up, running and releasing open source and open source based projects

- Indexed and tagged for easy re.use

- Open content license on GitHub→ please adapt, improve and translate

Conclusions

- Industry is going far far far … faster than Public Administration
- Many tools and best practices out there that are NOT being used by Pas… even best practices created by PAs!
- FOSS is 50% change management and 50% about people and community
  - In PA, a change of government often means change of (top) staff
- To implement any best practices or OSPO
  - Identify the specific issues of your Public Administration
  - Essential to be Agile
  - Have drinks with the boss and work hand in hand with an internal champion
  - Apply change management techniques and look for quick wins
  - Build Community