THE SHARING AND REUSE OF IT SOLUTIONS FOR IT MANAGERS

Specific Knowledge Sharing Module 2

OUTLINE

1. INTRODUCTION
2. USE COMMON REGISTRIES TO SHARE AND REUSE IT SOLUTIONS
3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS
4. ENSURE THAT YOUR IT SOLUTION RESPECTS LEGAL REQUIREMENTS
5. ENSURE THAT YOUR IT SOLUTION IS EXTENSIBLE
6. MAKE YOUR IT SOLUTION INTEROPERABLE
7. PROVIDE MAINTENANCE AND SUPPORT

1. INTRODUCTION

INTRODUCTION

This knowledge sharing module was produced under the ISA² programme of the European Commission, specifically aimed at IT managers within public administrations.

It is based upon the Sharing and Reuse Framework for IT Solutions, a set of recommendations providing guidance to help public administrations share or reuse IT solutions (software and services).

This specific knowledge sharing module is part of a threefold information package, which also includes a generic module and a specific module for policymakers and central body representatives.

LEARNING OBJECTIVES

This knowledge sharing module for IT Managers will guide you in implementing the recommendations of the **Sharing and Reuse Framework for IT solutions**.

By the end of this knowledge sharing module, you will have an understanding of:

- Existing registries of solutions that can be used.
- The basics of the legal requirements that your solution needs to respect.
- How to make sure that your solution can be extended and reused.
- The value of providing maintenance and support to your solution.

MOTIVATION

Why is sharing and reusing IT solutions important for IT managers?

Lower costs

Avoiding to start from scratch each time an IT solution is developed saves time and money. Collaborative development means splitting costs between contributors.

Efficiency

Organisations benefit from reusing existing IT solutions such as generic building blocks instead of "re-inventing the wheel", saving time and encouraging reuse.

Sustainability

Communities formed through joint development of, or existing around, open source software have a positive effect on the long-term sustainability of a solution.

BARRIERS

- Public administrations face various barriers in sharing and reusing IT solutions:




Communication **Legal** **Technical** **Organisational**

- In this context, the success of an IT solution depends on overcoming these barriers.

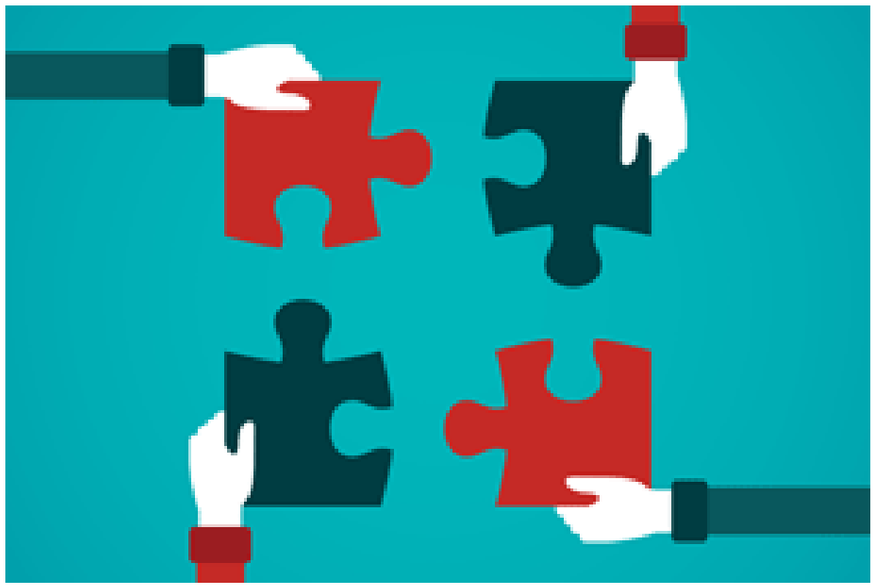
- IT project managers need to do so through making a **conscious effort** to **share their IT solutions** and **improve their reusability**.

European Commission

2. USE COMMON REGISTRIES TO SHARE AND REUSE IT SOLUTIONS

2. USE COMMON REGISTRIES TO SHARE AND REUSE IT SOLUTIONS

- Reusing and sharing IT solutions are separate pieces of the same puzzle
- Public administrations can reuse IT solutions that already exist or act as providers of IT solutions.



European Commission

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

Problem statements

- To be able to reuse an IT solution, one has to know it exist.

- Poor solution descriptions and ambiguous information about support structures may have a demotivating effect on potential reusers.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

Solutions

- A common registry is a system devoted to the proper description, structuring and publishing of IT solutions, together with the relevant documentation and process description in an integrated schema.

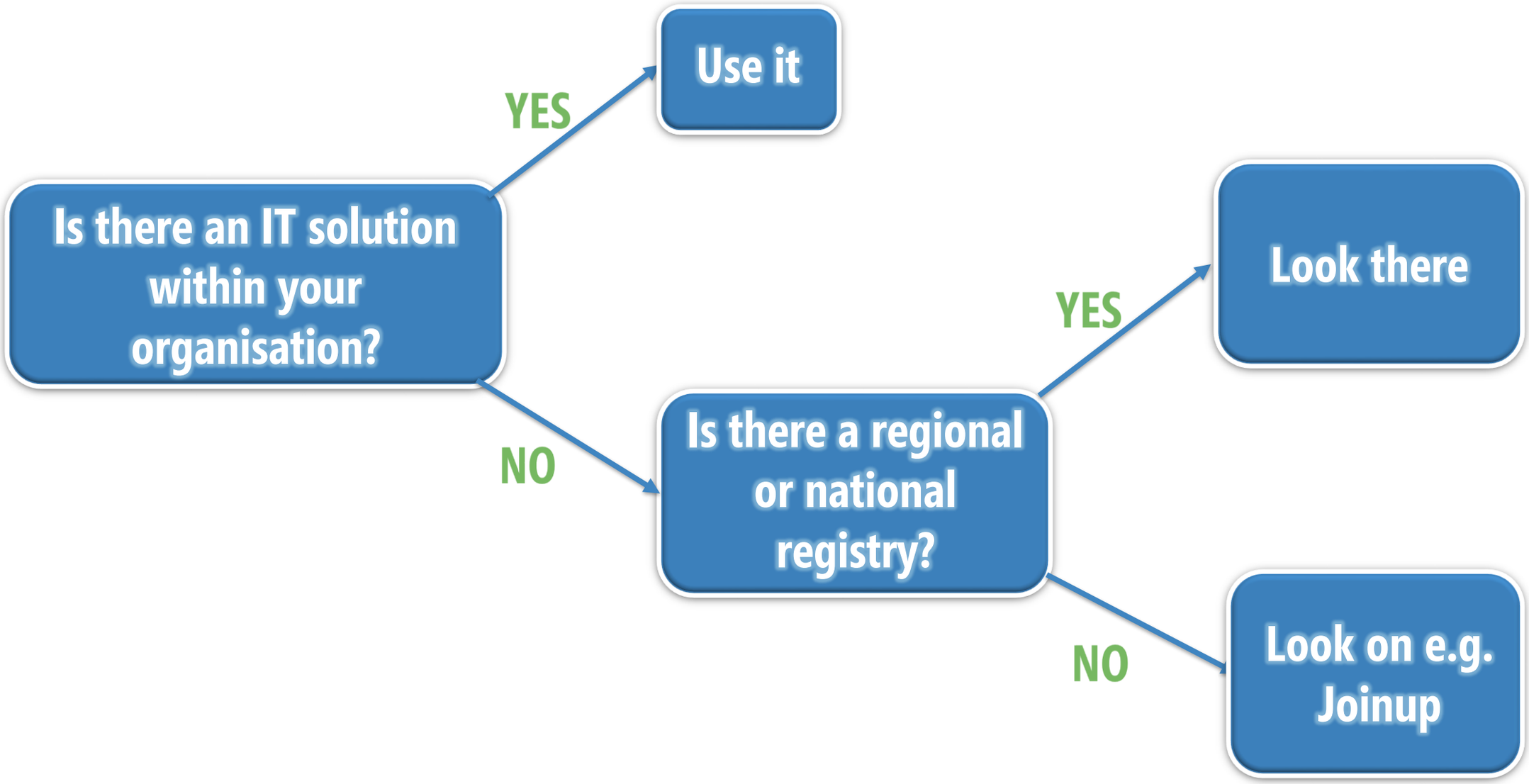
- Public administrations should consult common registries with a view to **reuse** existing IT solutions to save time and resources before even considering creating a new one.

- Use common registries to find IT solutions - by having a detailed description of IT solutions, you can compare them and find the one that best fit your needs.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

Solutions

- Use common registries to find IT solutions that you need.



Example of a EU registry

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Where can I look for IT solutions ?

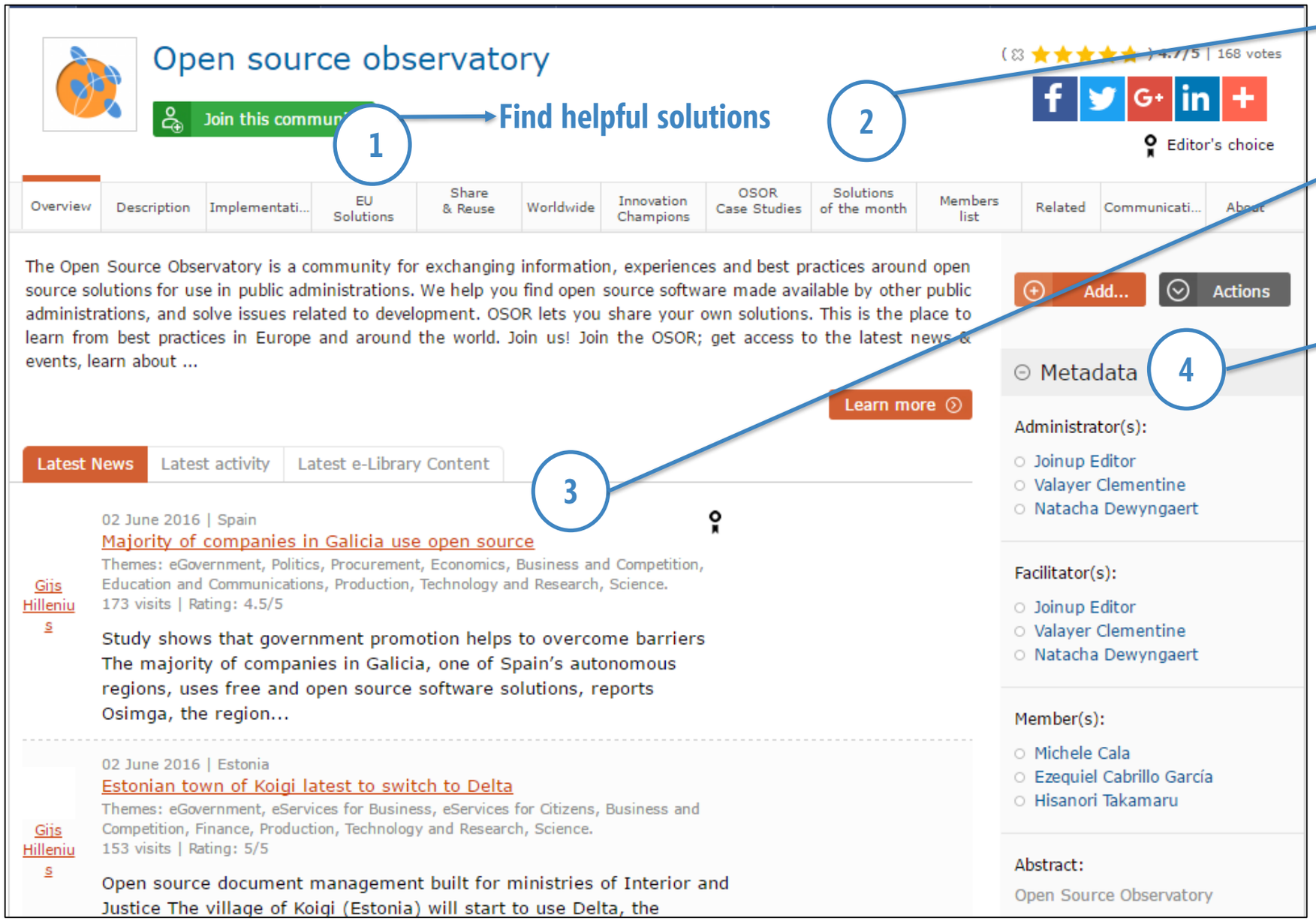
IT solutions can be found at following locations:

- Federated registries (Joinup)
- National registries (CTT, Adullact)
- Repository as a service (GitHub, Sourceforge, Bitbucket) – used by many public administrations
- Organisation registries (Apache, OW2)

Example of a Member State's public administrations active on GitHub. Click on the picture to explore more details. https://government.github.com/community/

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Open Source Observatory Repository (OSOR) on Joinup



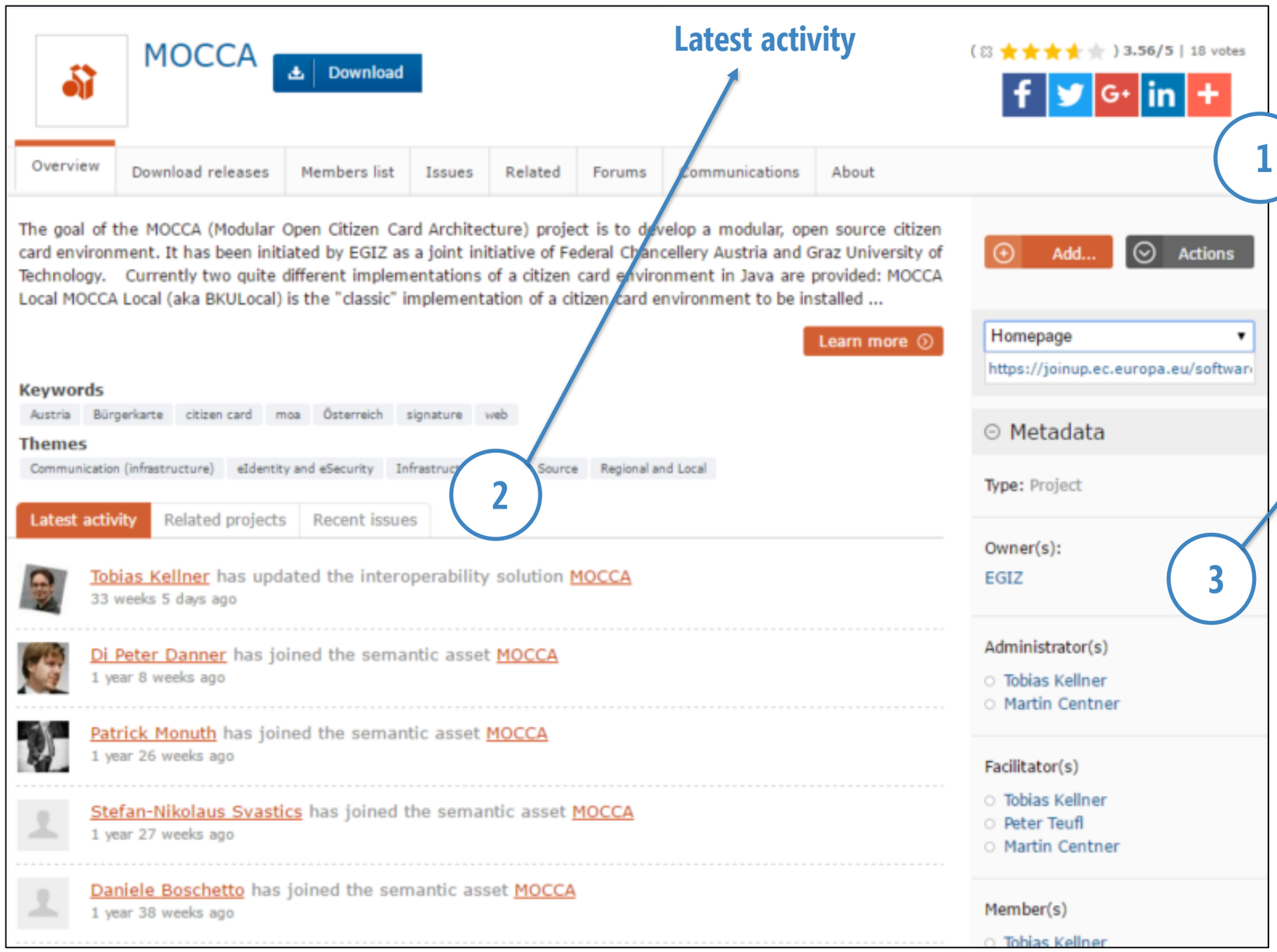
Check recent activities

The **OSOR community** on Joinup showcases the most important EU solutions, news, events and case studies around open source.

Users can consult the most recent news items and contact the facilitators to be involved in the community.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Joinup Project Description (e.g. MOCCA)



Latest activity

Check all the resources (mailing list, code, etc.)

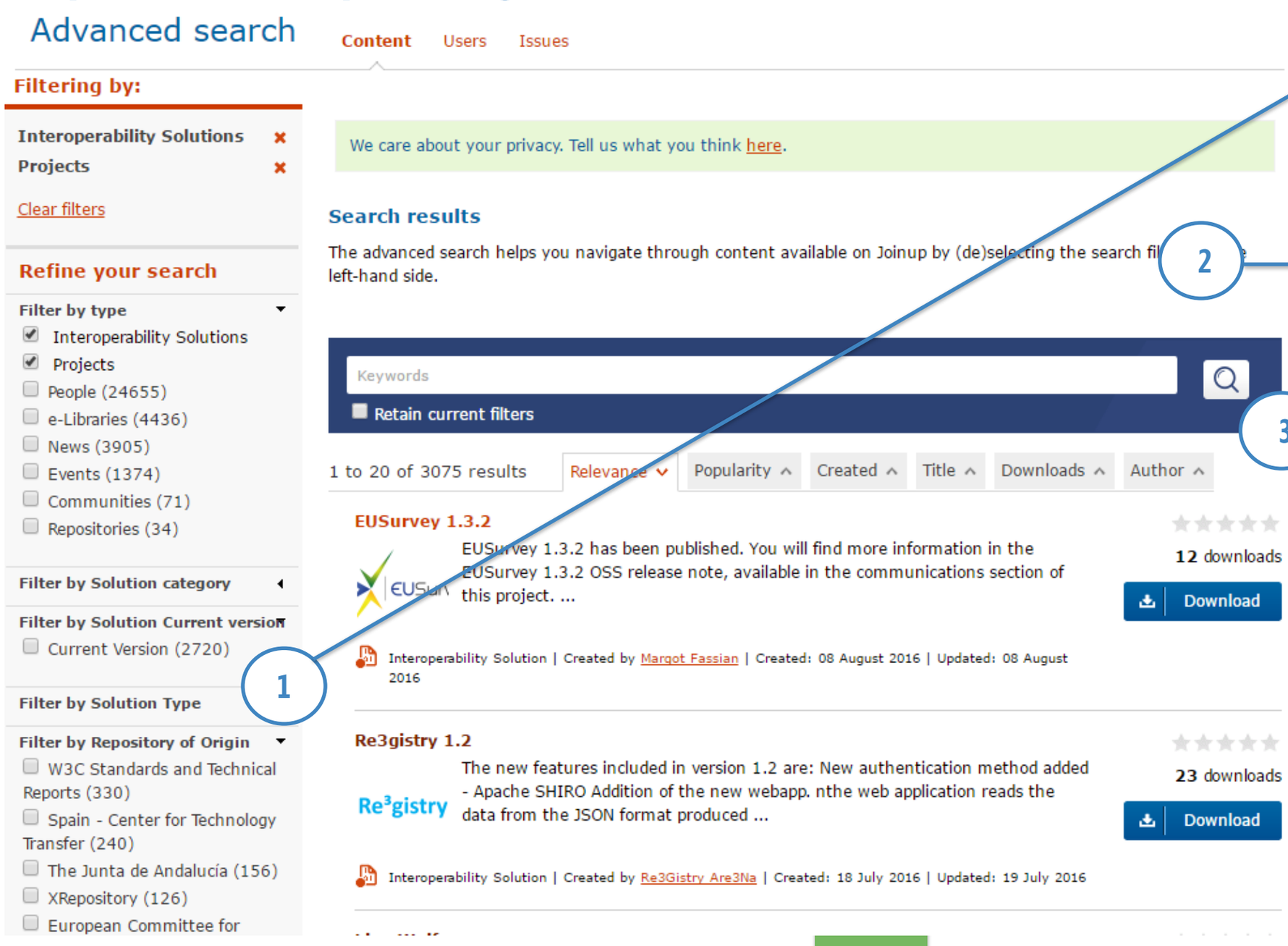
Contact

Each project on Joinup has a detailed description including all resources and related points of contact.

Users can learn about the recent activities of a project and download the latest releases.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Joinup Catalogue



Refine your search by looking through over 30 registries.

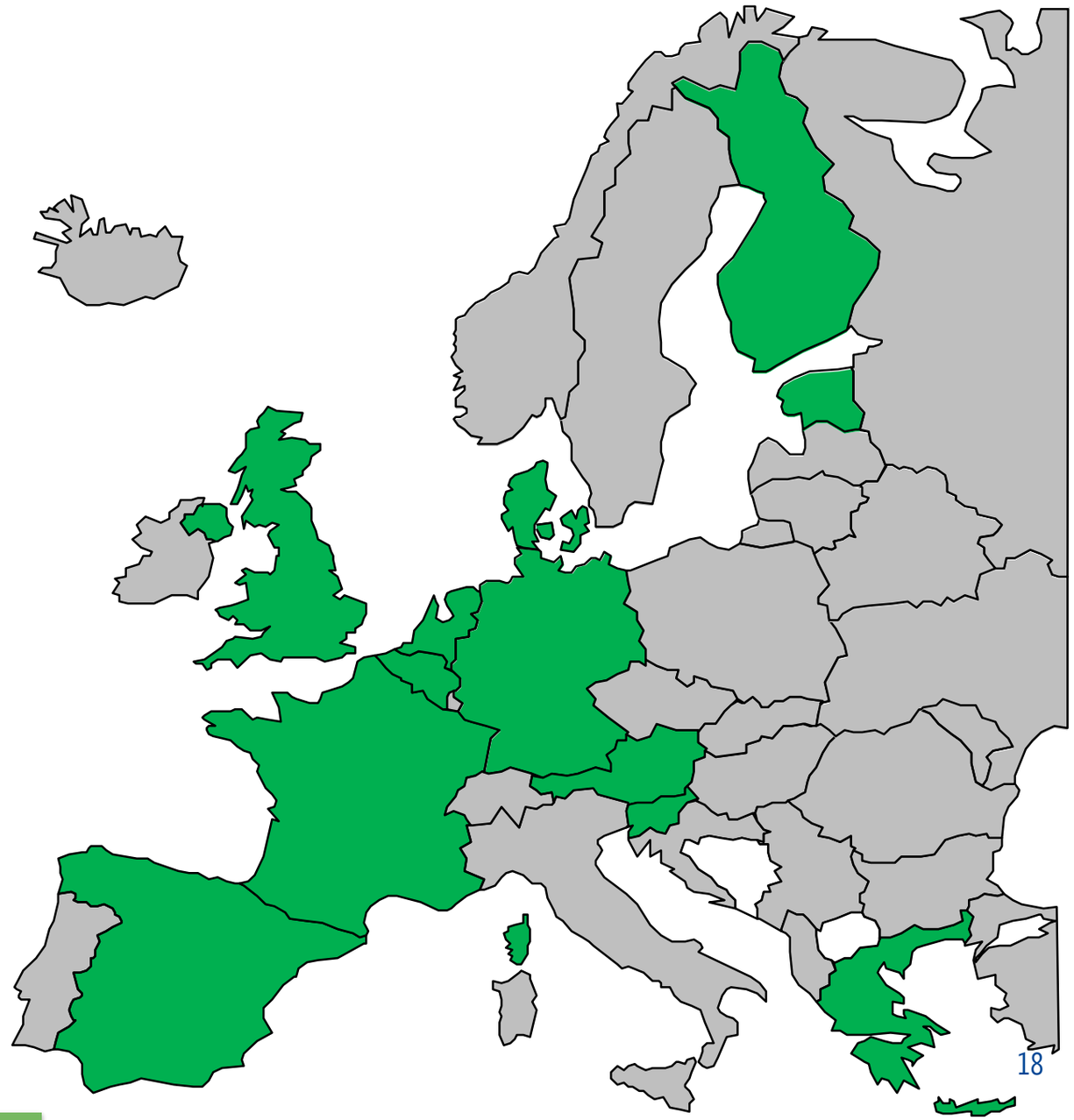
Search among over 3000 solutions directly in the Catalogue

Order the results in the way that best fits your needs

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Joinup Federation

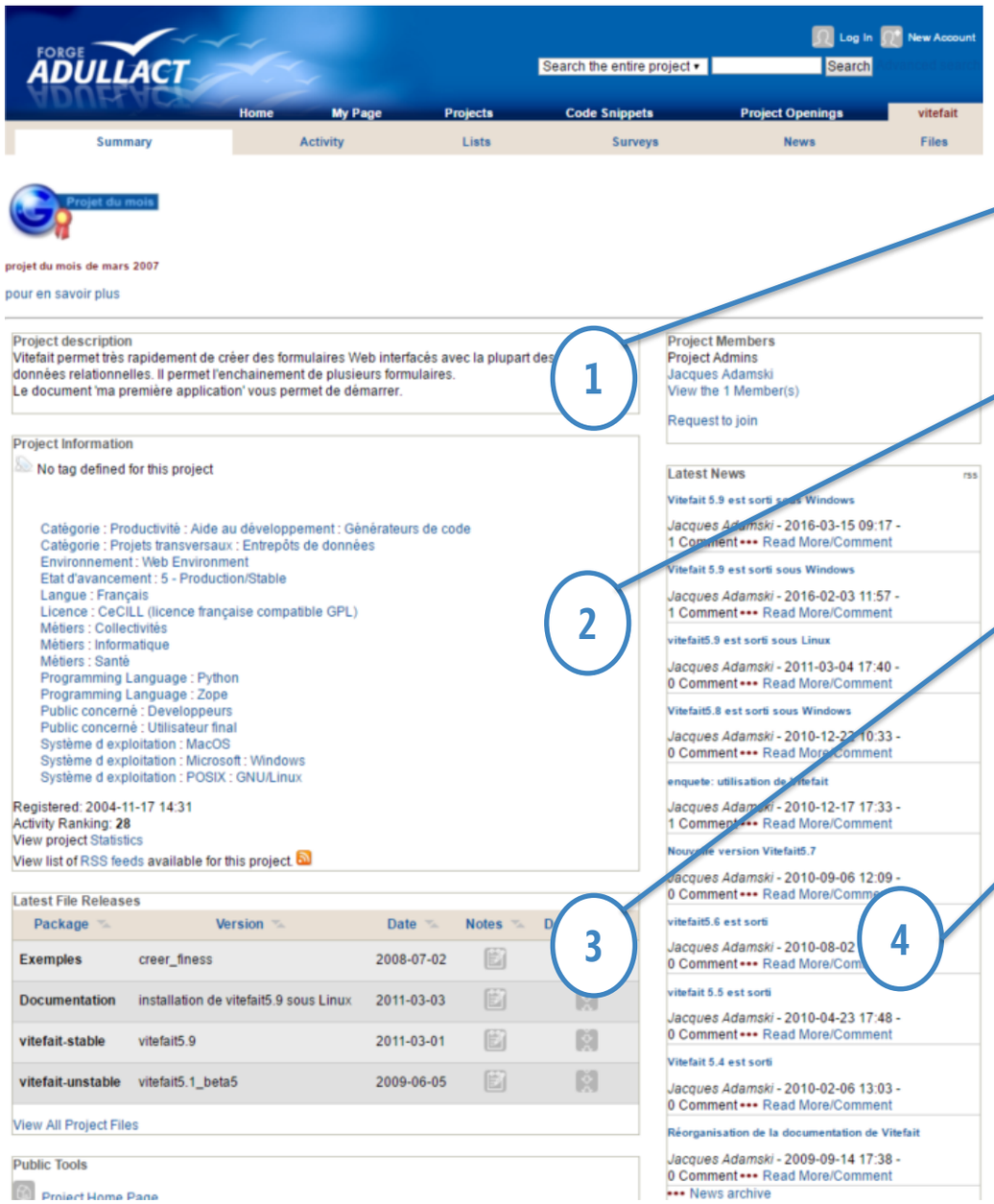
Federated Repository	Country
GBA Thesaurus	Austria
Belgian Interoperability Catalogue, OSLO	Belgium
Digitaliser	Denmark
RIHA	Estonia
Avoindata	Finland
Adullact	France
Xrepository	Germany
E-GIF	Greece
Dutch Standardisation Forum	Netherlands
CTT, Junta de Andalusia, Forija de Rediris	Spain
NIO	Slovenia
ESD Standards	UK



European Commission

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: National Registry Example: Adullact (France)



Project information

Categories

Downloads area

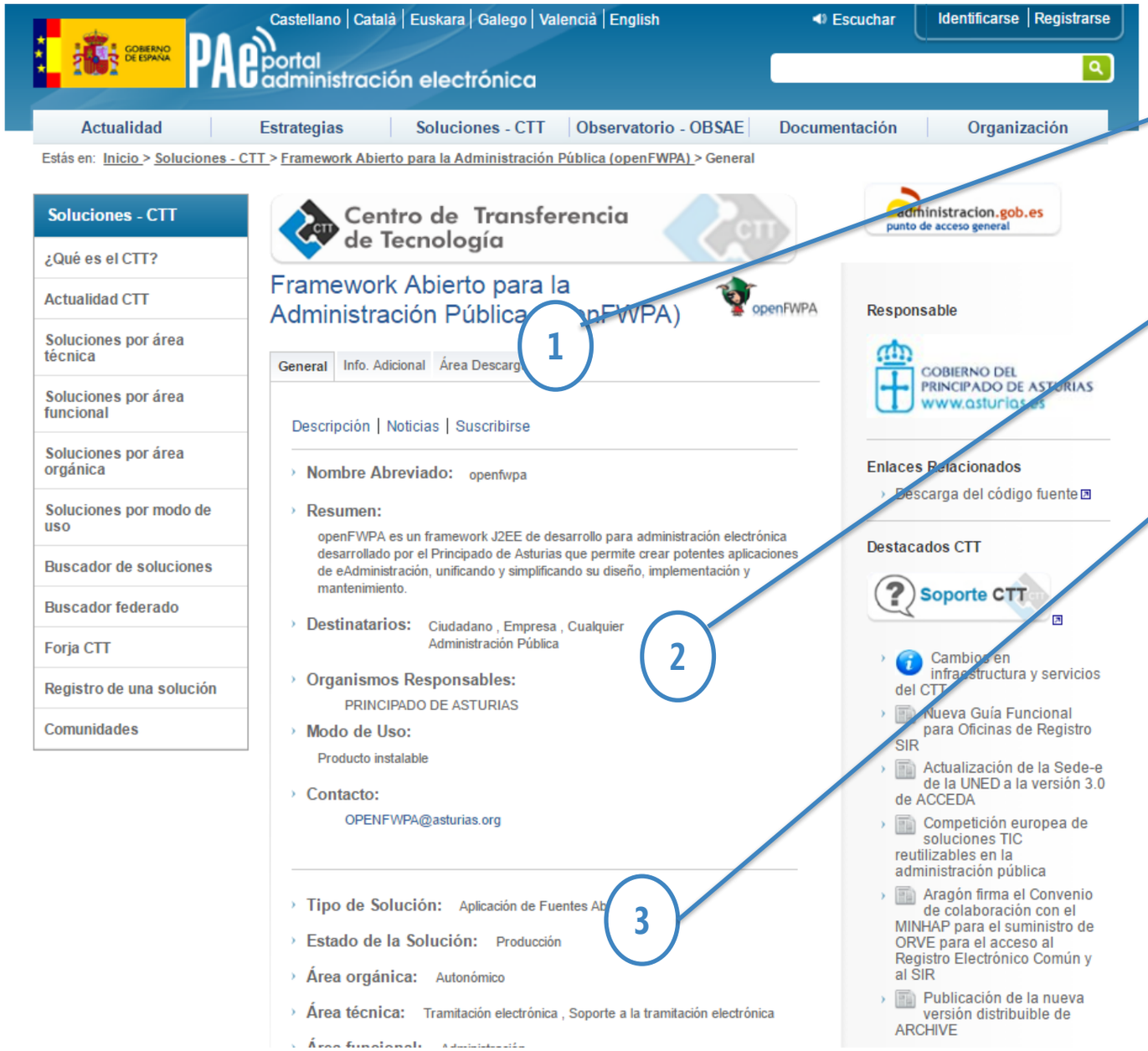
Latest news

Adullact provides similar project information to Joinup, including:

- Description
- Categories
- Download(s) of the last release(s)
- Recent project activities

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: National Registry Example: CTT (Spain)



Downloads area

Project organisation

Categories

The **Centro de Transferencia de Tecnologia** (Technology Transfer Centre) provides similar project information to Joinup, including:

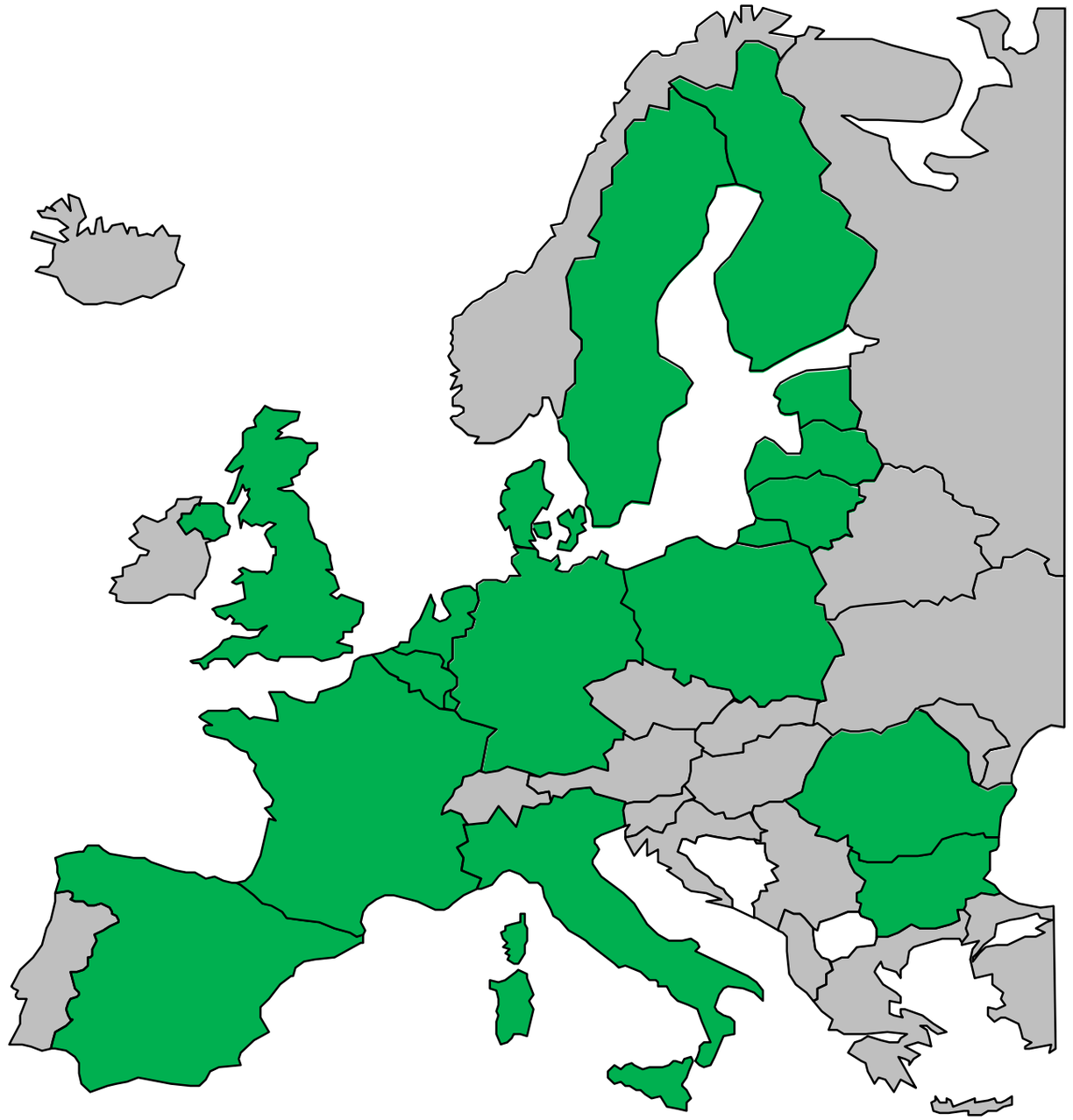
- Description
- Categories
- Download(s) of the last release(s)
- Recent project activities

European Commission

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: GitHub Public Repositories

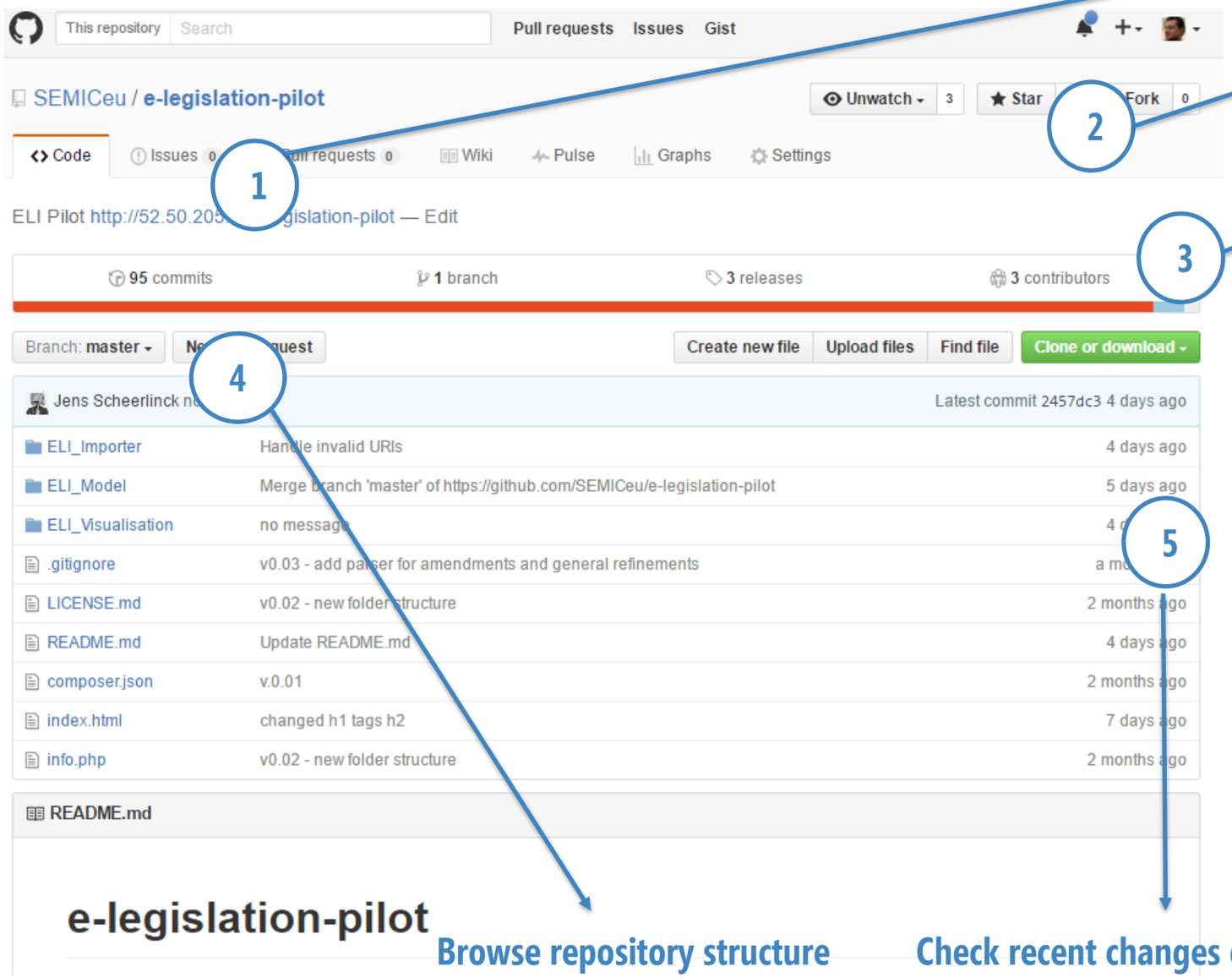
Country	Repositories
Belgium	5
Bulgaria	2
Denmark	1
Estonia	6
Finland	13
France	13
Germany	4
Italy	5
Latvia	1
Lithuania	2
Luxembourg	2
Netherlands	13
Poland	2
Romania	3
Spain	9
Sweden	12
U.K.	79



Data from: https://government.github.com/community/

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: GitHub (repository at the centre)



GitHub is aimed at developers with a minimal set of utilities for project description:

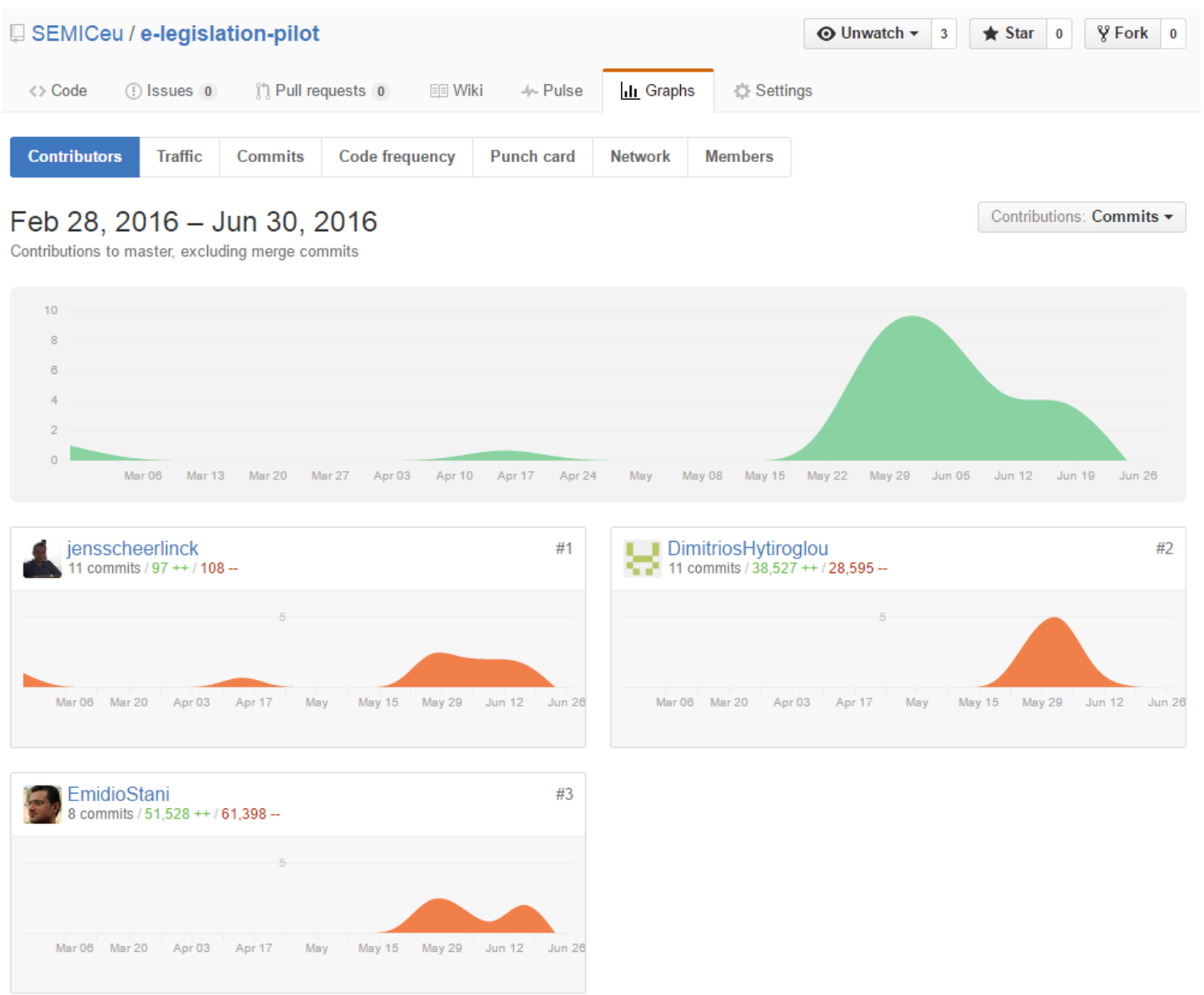
- Minimalistic project description
- No classification
- Minimalistic wiki

Furthermore, it supports collaboration between developers by using:

- Pull requests
- Allowing users to fork and rate the project

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: GitHub (project analytics)

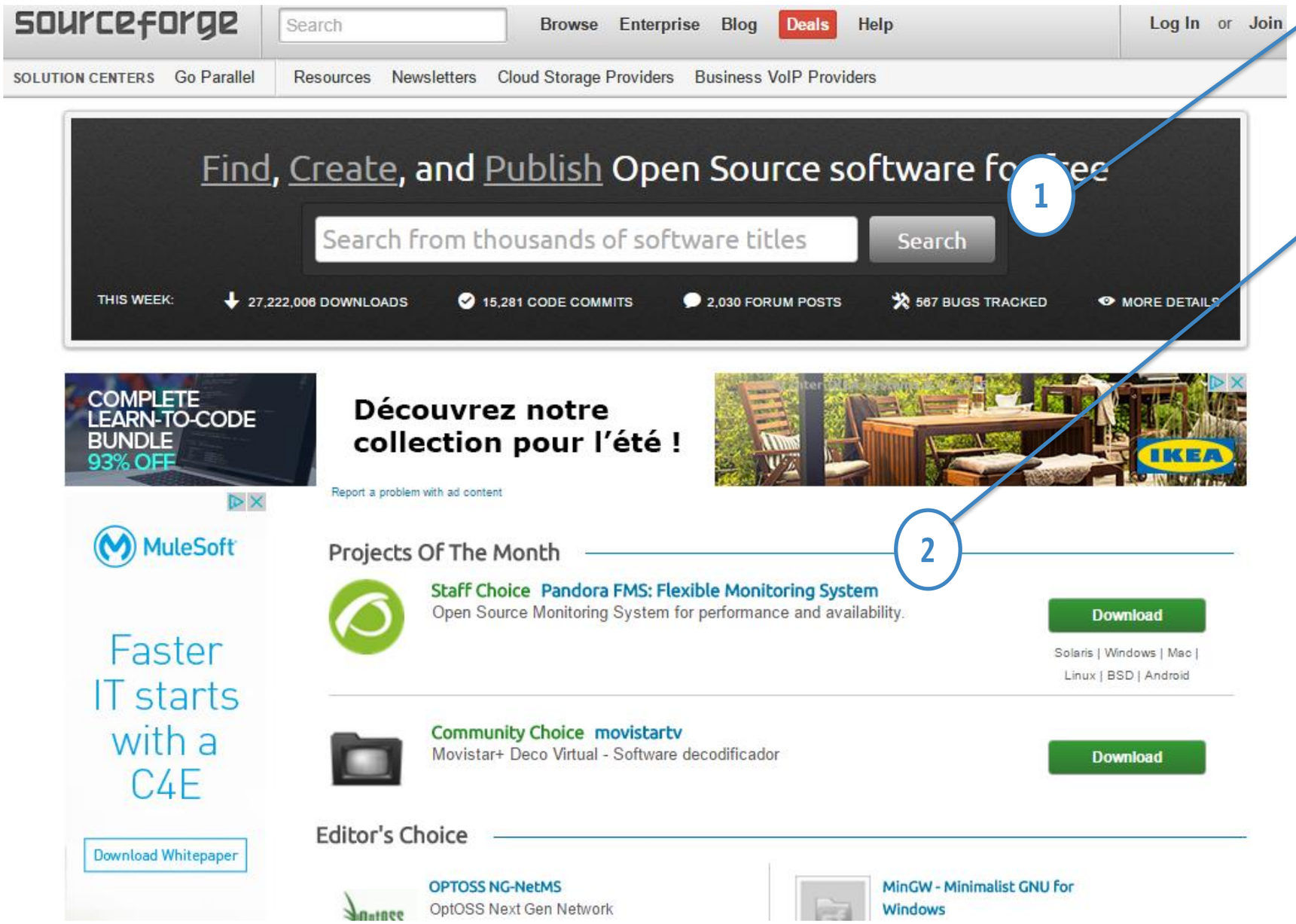


[GitHub](#) also provides a lot of statistics around a software:

- Contributors' activity
- Traffic
- Commits
- Punch card (weekly working hours)
- Number of forks
- Star ratings

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Sourceforge



Search

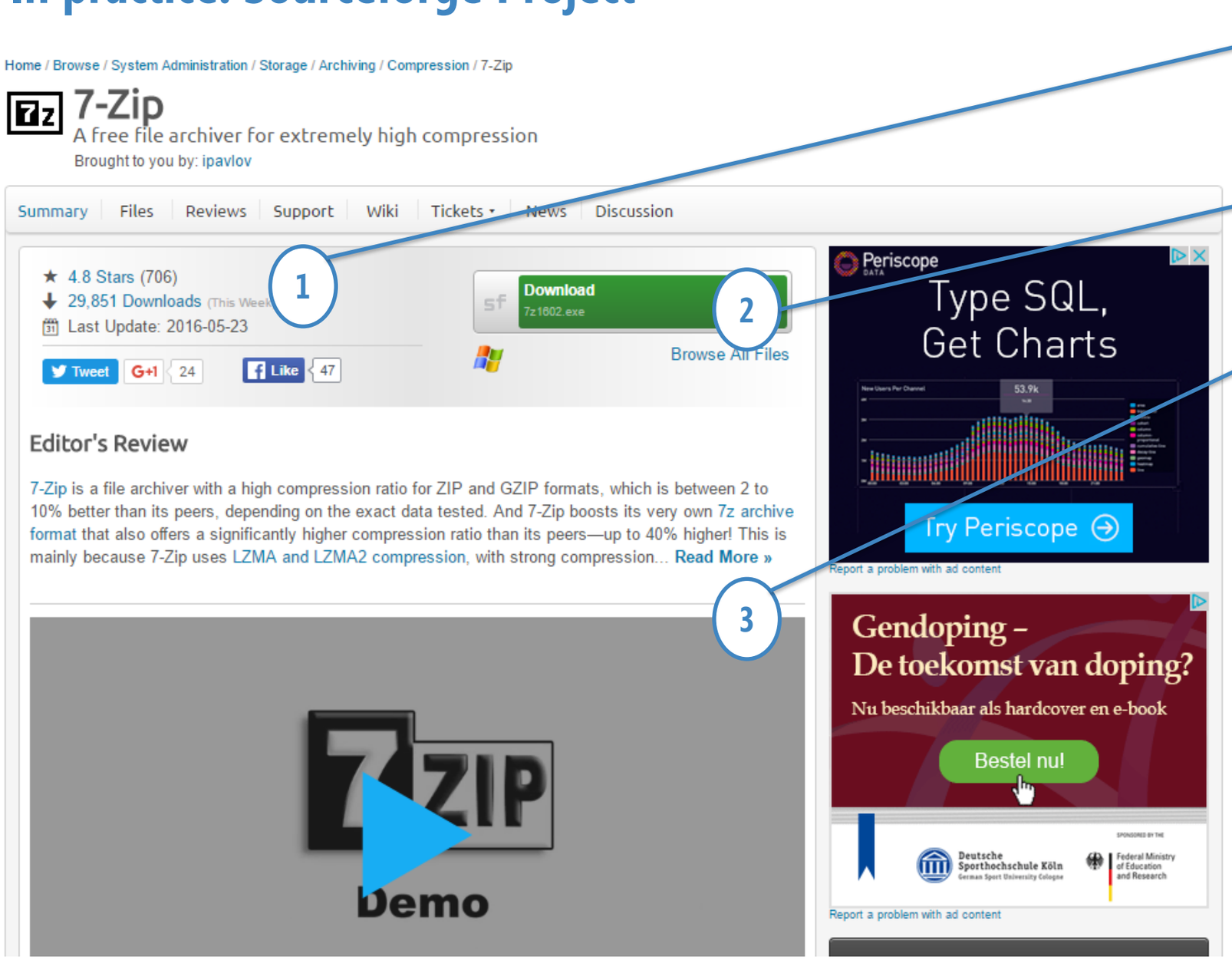
Highlighted projects

Sourceforge allows users to find thousands of projects and highlights the most important ones on a monthly basis (similar to the OSOR community).

The Sourceforge business model is based on advertising.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Sourceforge Project



Last updates

Download the last version

Sourceforge video

Sourceforge allows users to download the last version of the software easily.

The Sourceforge editorial team provides related videos explaining how to use the hosted software.

European Commission

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

Problem statements

- To share an IT solution properly, one has to know which means exist to make it searchable.

- Publishing an IT solution on a local website makes difficult for a user to find the required information (description, author, licence, release note, etc.).

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

Solutions

- A common registry is a system devoted to the proper description, structuring and publishing of IT solutions, together with the relevant documentation and process description in an integrated schema.

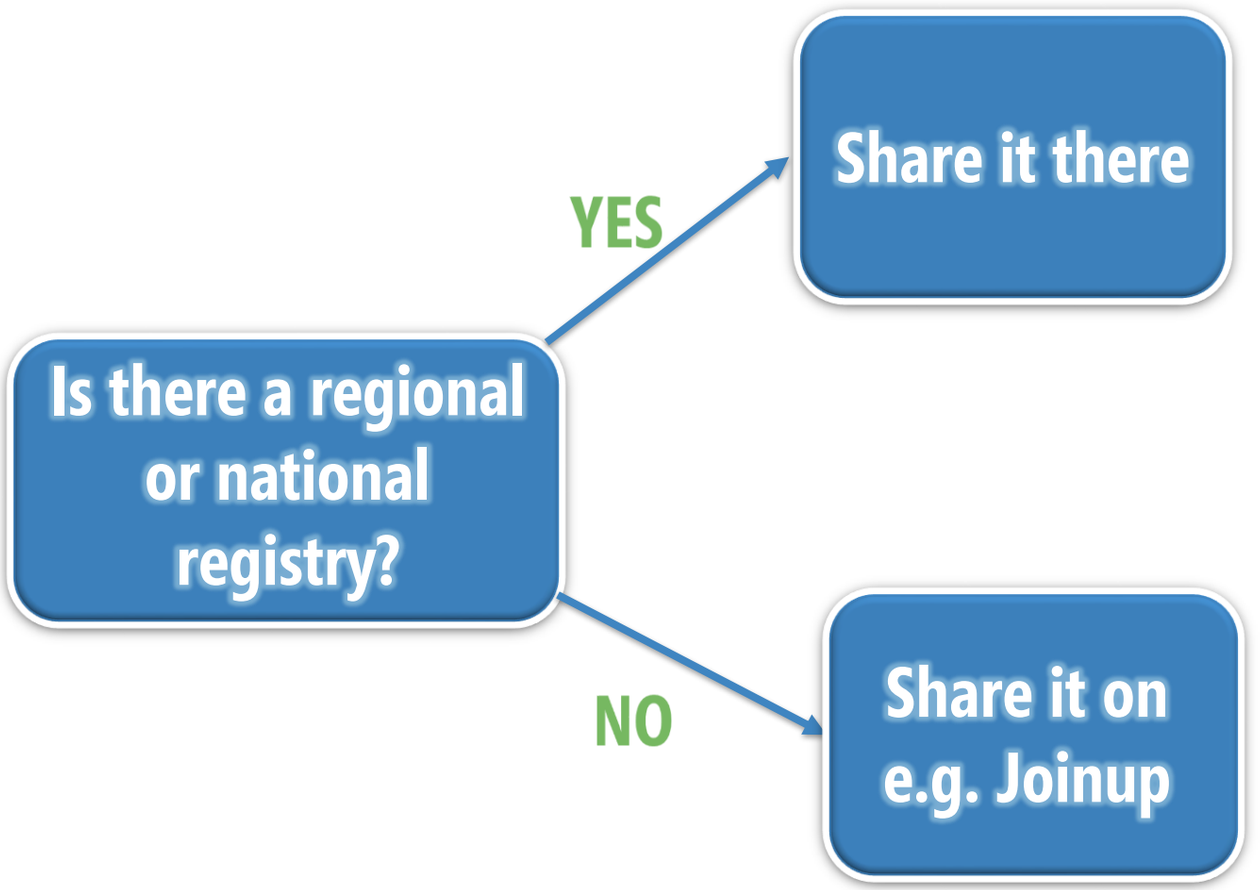
- Public administrations should publish their IT solutions on common registries instead of limiting such publication to their own websites.

- Use common registries to publish IT solutions – putting IT solutions in common registries increases their potential to be reused.

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

Solutions

- Use common registries to share* IT solutions that you have developed.



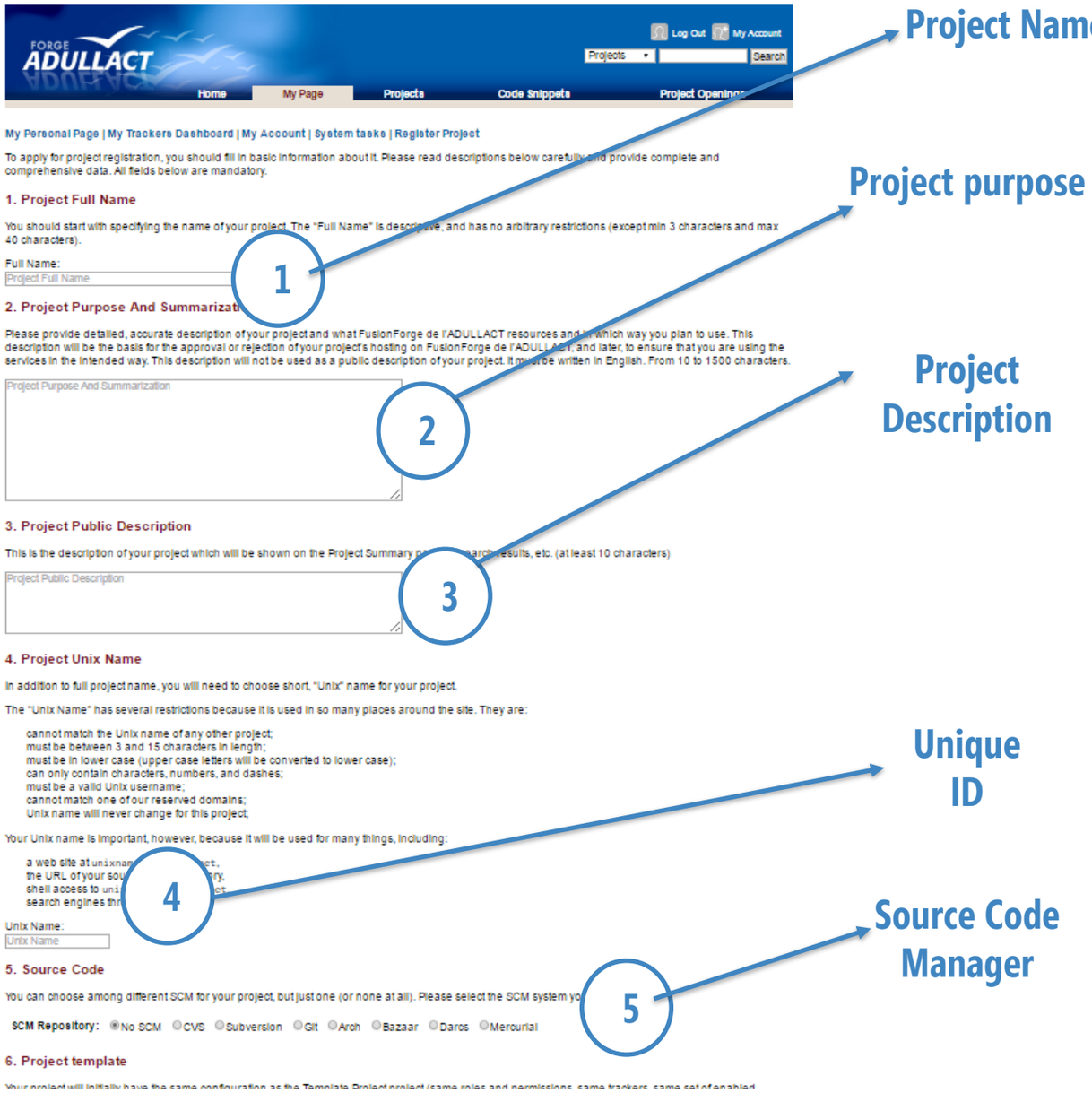
Example of a EU registry

*The source code of the IT solution could be hosted in any public repository such as GitHub or SourceForge.

European Commission

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

In practice: National Registry Example: Adullact (France)

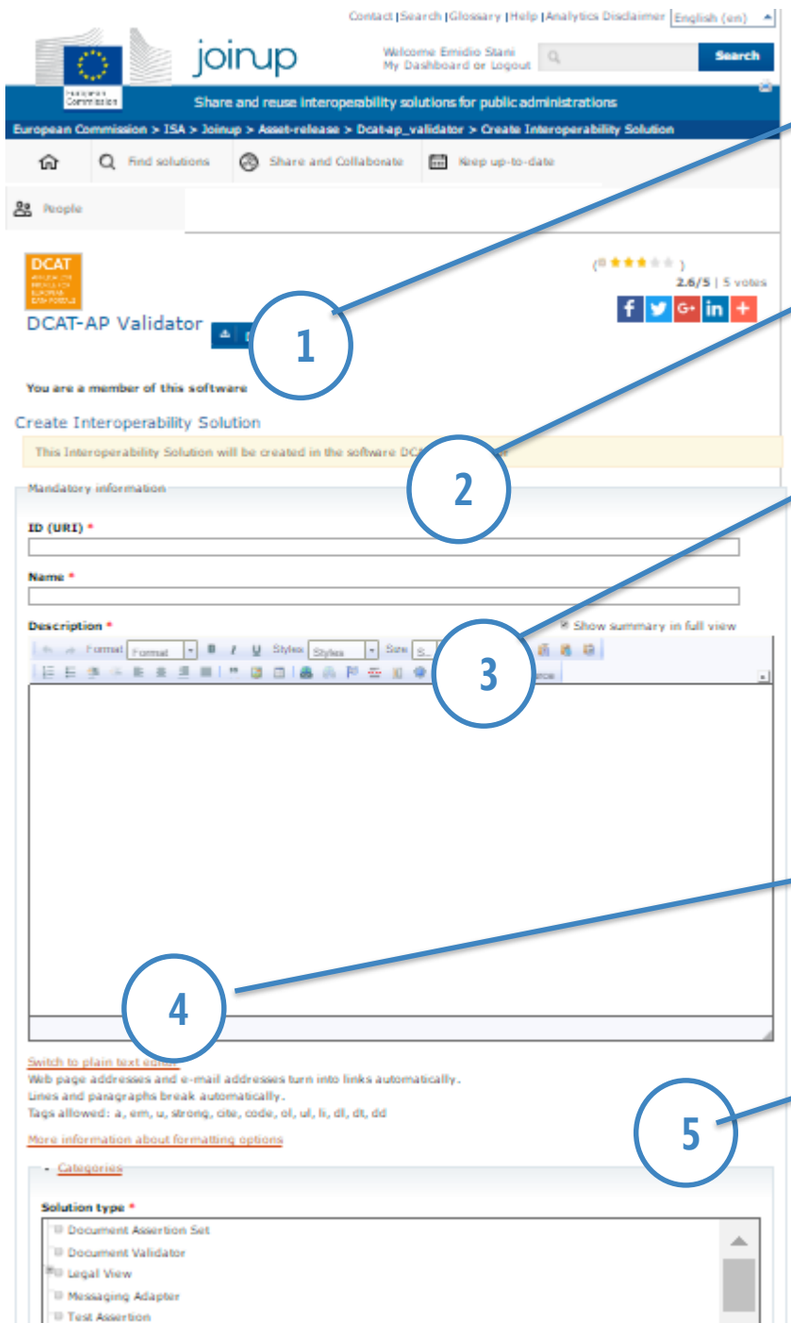


Adullact allows IT solution creators to describe their solutions and provide means, such as Source Code Management (SCM), to manage code.

It is a collaboration platform where users can share their code.

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

In practice: Joinup



Solution ID

Solution Name

Solution Description

Solution Type (classification)

Solution keywords

Joinup allows IT solutions to be created easily by completing a few mandatory fields such as the "Solution Type" (classification), which allows end users to find easily your project.

European Commission

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

Problem statements

- Public administrations require a minimum level of trust before they commit to reuse an IT solution.

- IT solutions that are not clearly described might not be trusted for reuse.

- Beyond trust, public administrations have to be sure that an IT solution is suited to their needs. The description of an IT solution plays an important role here as well.

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

Solutions

Public administrations should try to reuse existing and widely accepted standards where possible instead of creating new ones. By using an existing standard, a solution will benefit from the spread of the standard by becoming more extensible so that other solutions can take advantage of it. Widely accepted standards include UML, XML, HTML, SOAP, HTTP, etc.

For instance

- By describing the architecture of an IT solution, the use of **UML** diagrams makes it easy to understand the structure of its components.

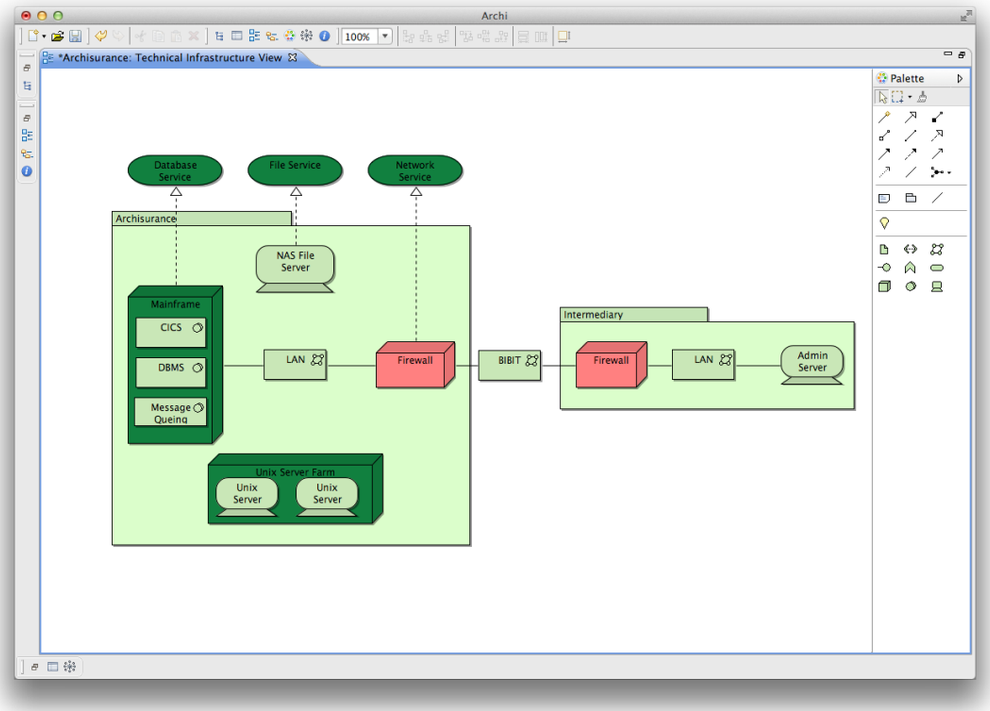
- Representing a data model in **XML** format will support easy transformation to other formats.

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

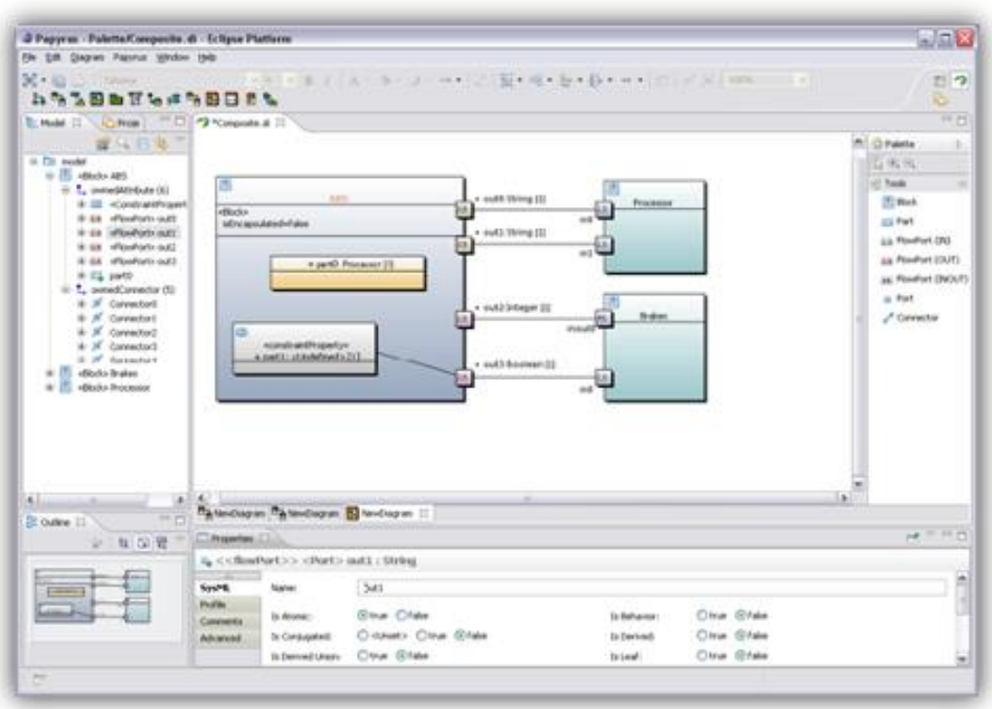
In practice

Sharing an **architectural model** allows other public administrations to better understand and improve IT solutions. For instance, this could be performed by sharing:

- Archimate diagrams using **Archi**: http://www.archimatetool.com/
- UML diagrams using **Papyrus**: http://www.eclipse.org/papyrus/



Archi diagram example



Papyrus diagram example

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

In practice : examples in the EU

The European Interoperability Reference Architecture (EIRA) provides a reference architecture based on Archi:

- ISA² Action on e-Documents and e-Files uses Archi to describe a reference architecture on e-Documents and Papyrus to describe class diagrams of Public Forms.
- Denmark, Estonia and Netherlands have started to use EIRA.

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

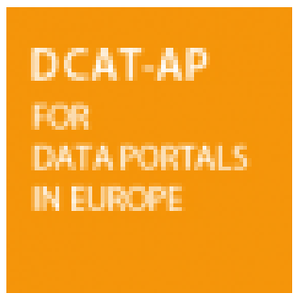
In practice

- ADMS

The **Asset Description Metadata Schema (ADMS)** is a vocabulary to describe interoperability assets, making it easier for those interested to find such assets.

- DCAT-AP

The **DCAT application profile** for data portals in Europe is a specification based on the Data Catalogue vocabulary to describe public sector datasets in Europe. It enables cross-data portal search for datasets, making public sector data more searchable across borders and sectors.



3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

In practice: examples in the EU

- ADMS is adopted in the Joinup portal and Publication Office.

Estonia, Denmark, Germany have already started to adopt ADMS

- DCAT-AP has been extended in MS such as Belgium, Italy and Norway,

and by other standards such as StatDCAT-AP and GeoDCAT-AP (INSPIRE)

4. ENSURE THAT YOUR SOLUTION RESPECTS LEGAL REQUIREMENTS

European Commission

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

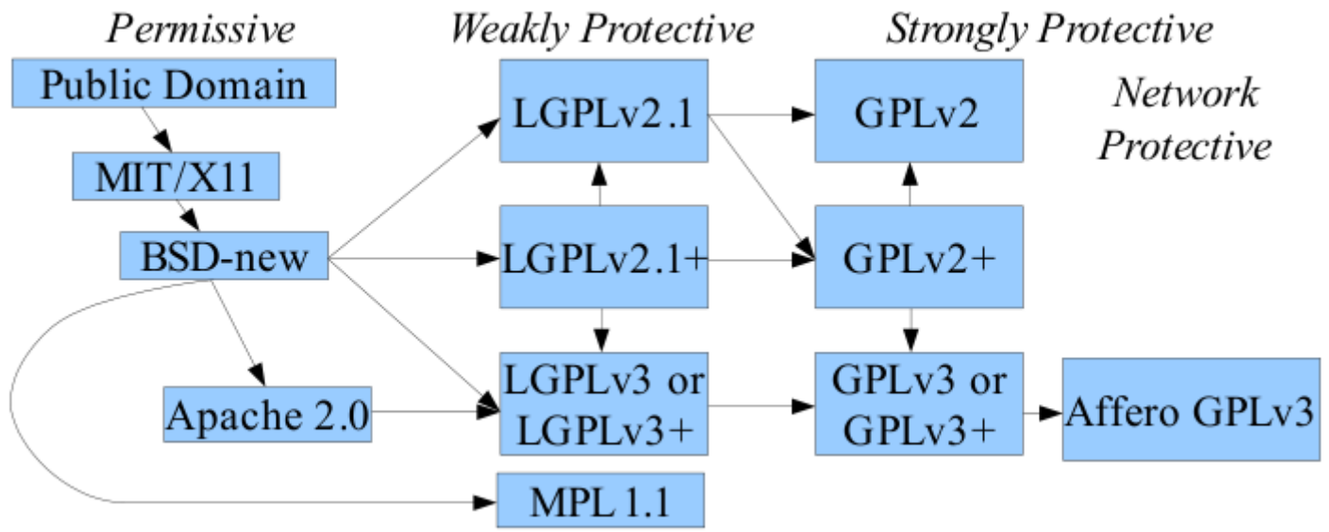
Problem statements

- Insufficient clarity about liability exposure and possible infringement of property rights make public administrations reluctant to take a collaborative approach **to developing IT solutions**.
- Issues of incompatibility between licences, which can result from public administrations writing their own licences instead of using existing ones.
- The proliferation of licences leads to the coexistence of multiple licences with similar terms but with potential incompatibilities.

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

Solutions

- It is important that public administrations use licences with the least legal friction possible, i.e. with the minimum possible restrictions in terms of sharing IT solutions. The best solution is to use open source licences to share your IT solution .
- By doing so, the existence of similar licences with potential incompatibilities can be avoided.



By David A. Wheeler
http://www.dwheeler.com/essays/floss-license-slide.html, CC BY-SA 3.0,
https://commons.wikimedia.org/w/index.php?curid=41060008

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

Solutions

Furthermore, the Sharing and Reuse Framework for IT Solutions recommends to:

- **Detect licence compatibility issues** – when merging two pieces of software code with difference licences, it is important to detect licence incompatibilities as soon as possible by using dedicated tools.
- **Use standard templates for liability agreements** – these practical instruments cover key IPR aspects to take into account when sharing IT solutions.
- **Decide and communicate as early as possible about the type of rights attribution approach** – providing developers with requirement guidelines to follow when developing and maintaining IT solutions.

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice

- Which licence should I use ?
- Which websites can support me ?
- How do I add a licence to the project ?

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: which licence should I use?

MIT License

A short and simple permissive license with conditions only requiring preservation of copyright and license notices. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions	Conditions	Limitations
Commercial Use	License and Copyright Notice	Hold Liable
Distribution		
Modification		
Private Use		

```
MIT License

Copyright (c) [year] [fullname]

Permission is hereby granted, free of charge, to any person obtaining a copy
of this software and associated documentation files (the "Software"), to deal
in the Software without restriction, including without limitation the rights
to use, copy, modify, merge, publish, distribute, sublicense, and/or sell
copies of the Software, and to permit persons to whom the Software is
furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all
copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR
IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,
OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE
SOFTWARE.
```

The **MIT licence** allows developers to share their IT solutions without restrictions on the condition that they cite the licence and copyright notice.

Examples of well known software released under this licence include:

Jenkins

jQuery write less, do more.

ATOM

GitLab

ANGULARJS by Google

node JS

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: which licence should I use?

Apache License 2.0

A permissive license whose main conditions require preservation of copyright and license notices. Contributors provide an express grant of patent rights. Licensed works, modifications, and larger works may be distributed under different terms and without source code.

Permissions	Conditions	Limitations
Commercial Use	License and Copyright Notice	Hold Liable
Distribution	State Changes	Trademark Use
Modification		
Patent Use		
Private Use		

```
                                 Apache License
                           Version 2.0, January 2004
                        http://www.apache.org/licenses/

   TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

   1. Definitions.

      "License" shall mean the terms and conditions for use, reproduction,
      and distribution as defined by Sections 1 through 9 of this document.

      "Licensor" shall mean the copyright owner or entity authorized by
      the copyright owner that is granting the License.
```

The **Apache licence** allows developers to share their IT solutions without restriction on the condition that they cite the licence and copyright notice and state the changes applied.

Examples of well known software released under this licence include:

SUBVERSION

Apache HTTP SERVER PROJECT

Apache OpenOffice™

Apache Tomcat®

android

Apache Allura™
Forge software for hosting software projects

openstack

PHABRICATOR

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: which licence should I use?

GNU General Public License v3.0

GNU GPLv3

Permissions of this strong copyleft license are conditioned on making available complete source code of licensed works and modifications, which include larger works using a licensed work, under the same license. Copyright and license notices must be preserved. Contributors provide an express grant of patent rights.

Permissions	Conditions	Limitations
Commercial use	Disclose source	Liability
Distribution	License and copyright notice	Warranty
Modification	Same license	
Patent use	State changes	
Private use		

```
                    GNU GENERAL PUBLIC LICENSE
                       Version 3, 29 June 2007

 Copyright (C) 2007 Free Software Foundation, Inc. <http://fsf.org/>
 Everyone is permitted to copy and distribute verbatim copies
 of this license document, but changing it is not allowed.

                            Preamble

  The GNU General Public License is a free, copyleft license for
software and other kinds of works.

  The licenses for most software and other practical works are designed
to take away your freedom to share and change the works.  By contrast,
```

The **GPL licence** allows developers to share their IT solutions without restriction, on the condition that they cite the licence and copyright notice. Furthermore they need to state the applied changes, keep the same licence and disclose the source.

Examples of well known software released under this licence include:

WordPress.org

REDMINE
flexible project management

MediaWiki

VLC media player
The best free media player for video and DVDs

Drupal

Notepad++

MySQL

GIMP

FusionForge

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: which licence should I use?

EUROPEAN UNION PUBLIC LICENCE v. 1.2

The Licensor hereby grants You a worldwide, royalty-free, non-exclusive, sublicensable licence to do the following, for
the duration of copyright vested in the Original Work:

— **use** the Work in any circumstance and for all usage,

— **reproduce** the Work,

— **modify** the Work, and make Derivative Works based upon the Work,

— **communicate** to the public, including the right to make available or display the Work or copies thereof to the public and perform publicly, as the case may be, the Work,

— **distribute** the Work or copies thereof,

— **lend and rent** the Work or copies thereof,

— **sublicense rights** in the Work or copies thereof.

The **EUPL licence** allows developers to share their IT solutions without restrictions under the conditions to keep the licence and copyright notice, state the changes applied, keep the same licence and disclose the source.

This licence has official value in the 23 languages of the EU.

Examples of well known software released under this licence:

CIRCABC

EUSurvey

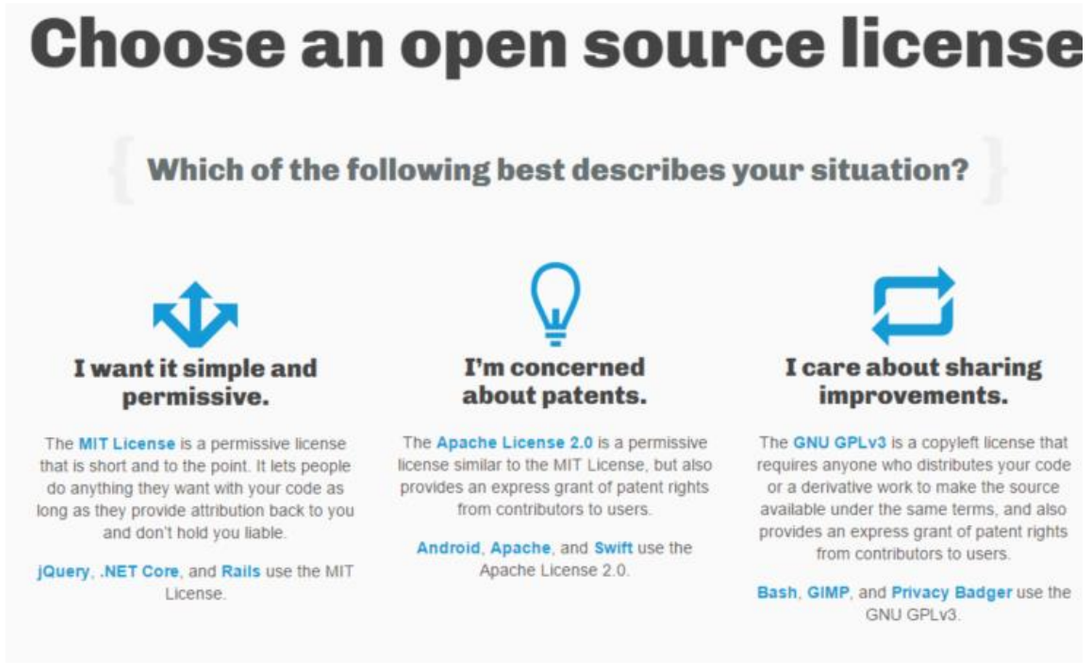
sdmx

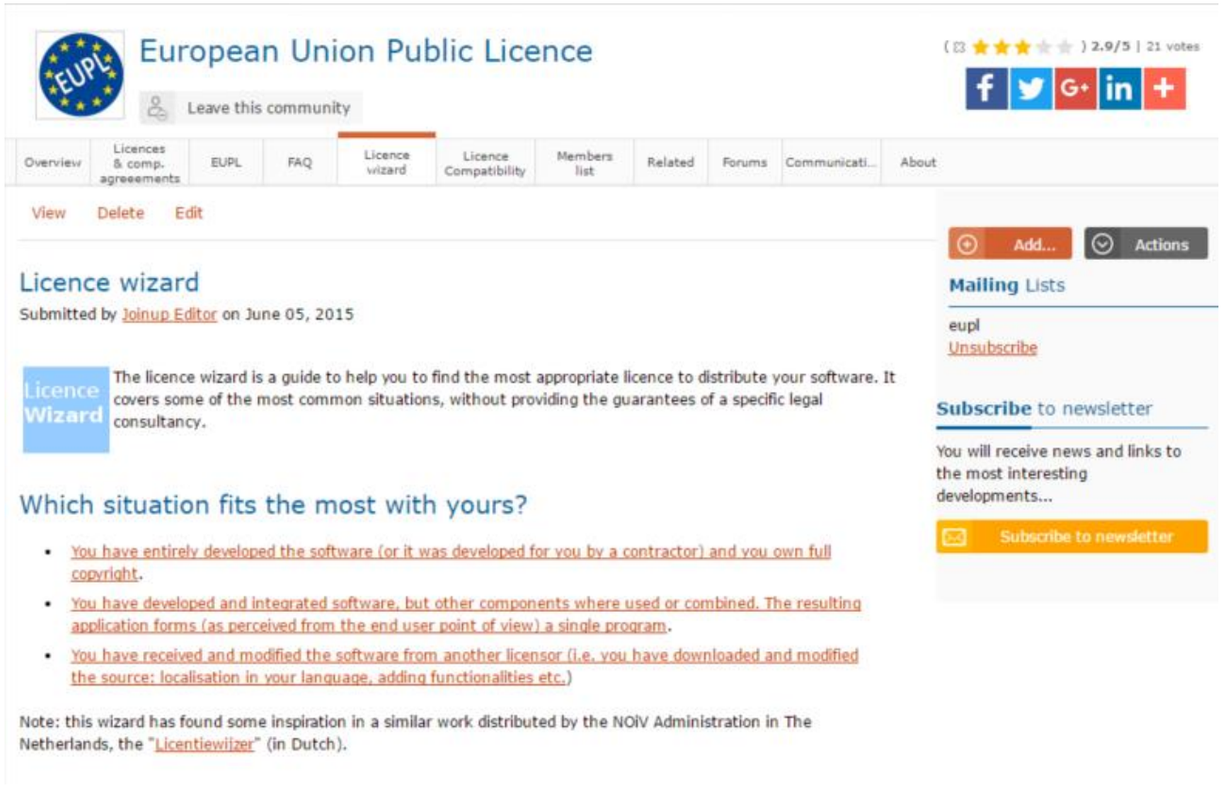
DCAT-AP Validator

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: which licence should I use?

"Choose a licence" and Joinup provide licence wizards to choose the licence that fits your project best.





4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: how do I add a licence to the project ?

1) Add a COPYING/LICENCE file at project level

2) Add a licence note (short note) at the beginning of each file:

General Public License note

one line to give the program's name and an idea of what it does.
Copyright (C) **yyyy name of author**

This program is free software; you can redistribute it and/or
modify it under the terms of the GNU General Public License
as published by the Free Software Foundation; either version 2
of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful,
but WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
GNU General Public License for more details.

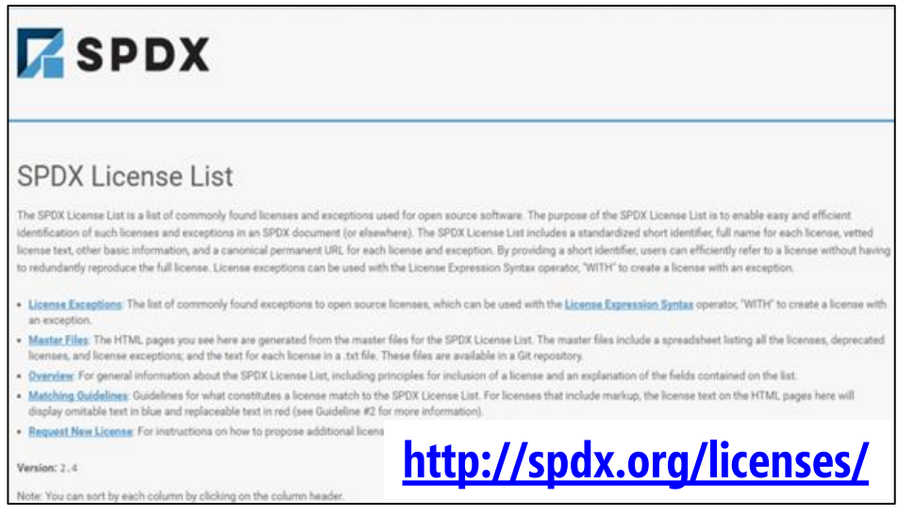
You should have received a copy of the GNU General Public License
along with this program; if not, write to the Free Software
Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

EUPL licence notice

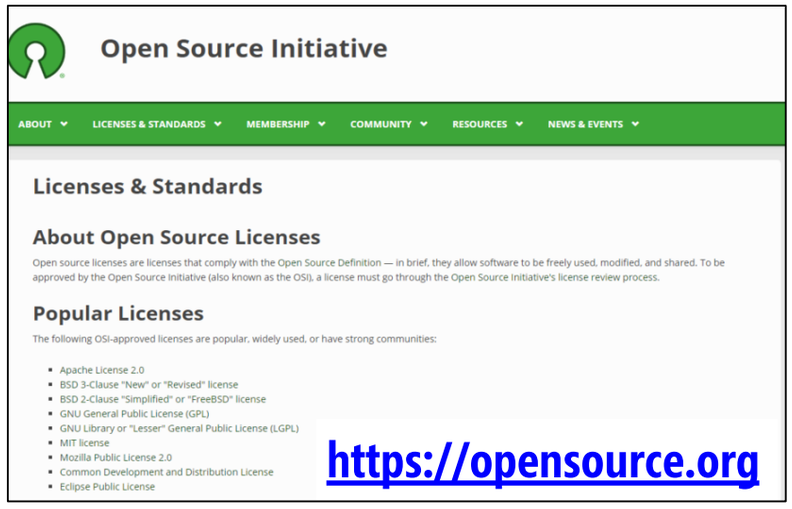
/*
* Copyright **YYYY YOUR-ORGANISATION**
*
* Licensed under the EUPL
*/

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

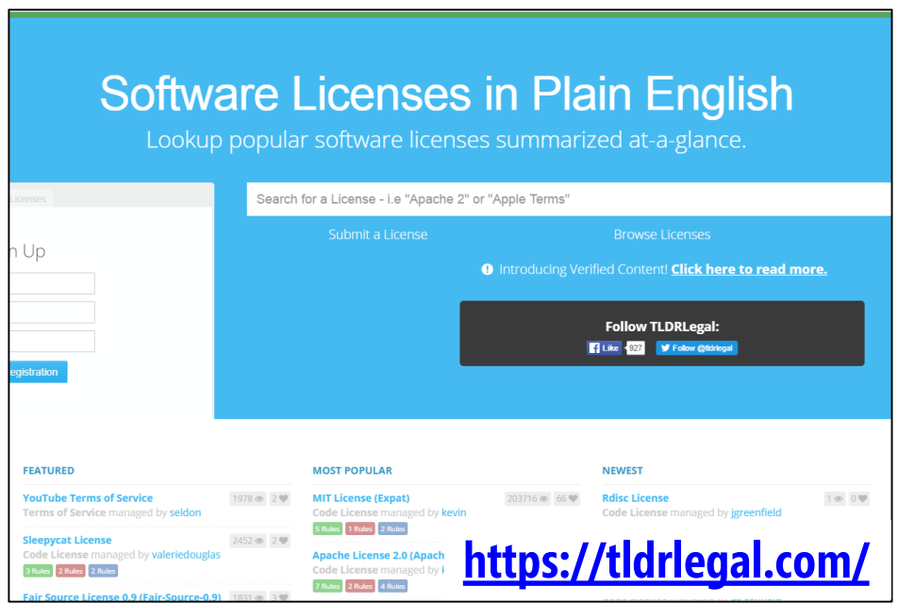
In practice: which websites can support me ?



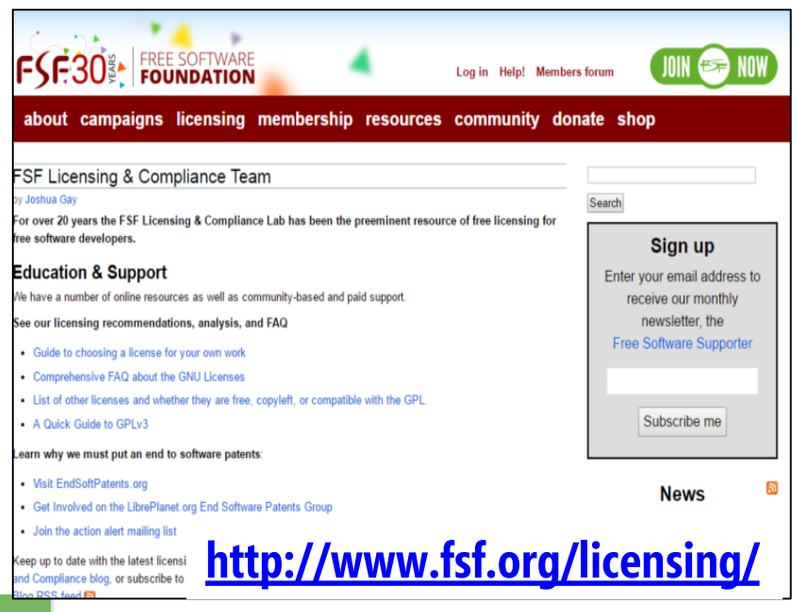
http://spdx.org/licenses/



https://opensource.org



https://tldrlegal.com/



http://www.fsf.org/licensing/

4.1 USE EXISTING OPEN SOURCE SOFTWARE LICENCES TO SHARE YOUR IT SOLUTION

In practice: EUPL examples in the EU

- Eurostat: several statistical tools are provided under EUPL.
- Estonia, Malta, Netherlands and Spain adopted the EUPL at governmental level.
- Austria: different applications concerning security such as certificates and signature management are released under EUPL.
- Germany: an extension of OpenOffice has been released under EUPL.
- Italy: a digital signature application has been released under EUPL.
- Slovenia: the secure e-Delivery system has been released under EUPL.

European Commission

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

Problem statements

- Insufficient clarity about liability exposure and possible infringement of property rights make public administrations reluctant to take a collaborative approach **to reusing existing IT solutions.**

- When reusing existing solutions, public administrations may incur licence incompatibilities due to the original project licences.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

Solutions

- It is important that public administrations have means to check under which licence an IT solution is released.
- By doing so, public administrations can have more trust in using such a solution.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

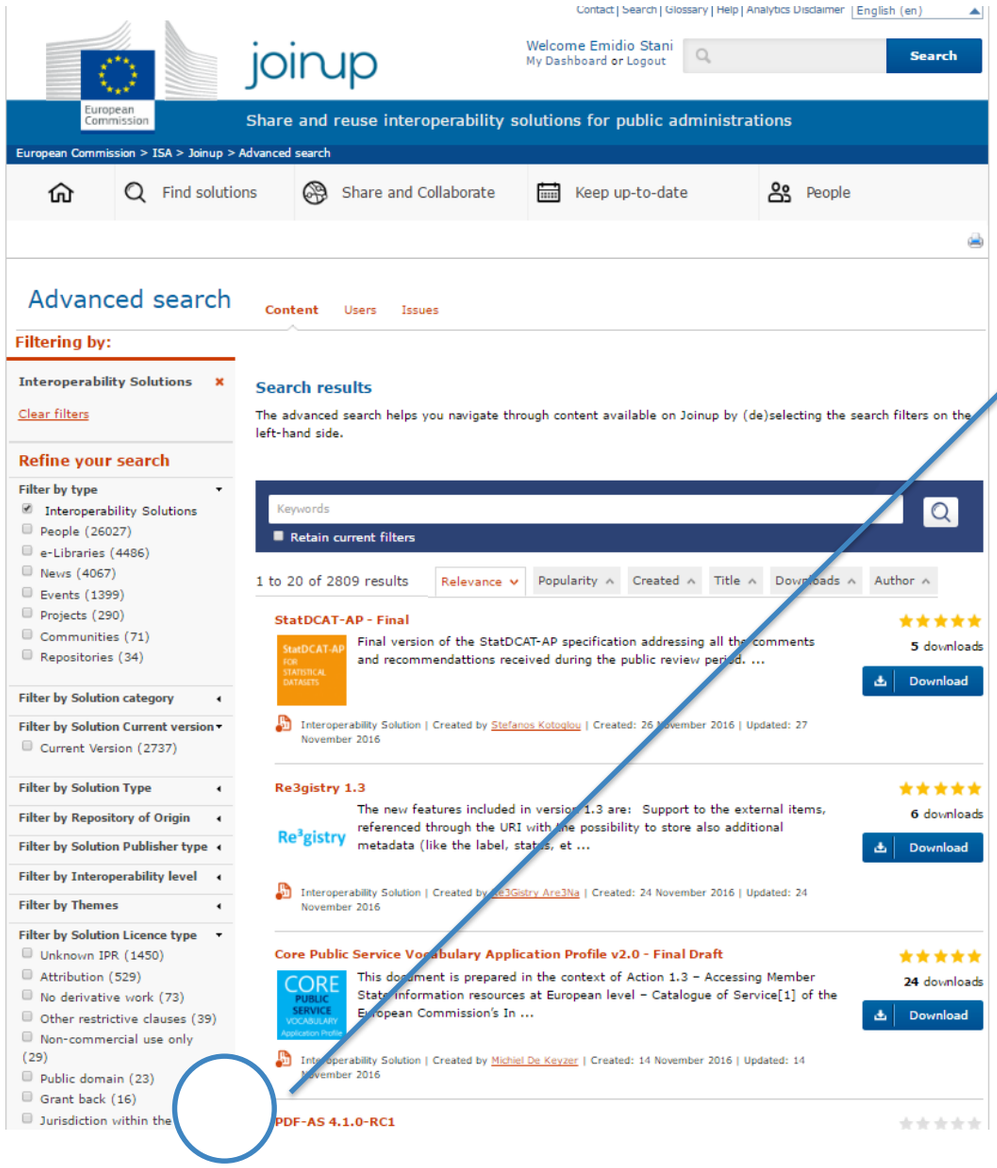
In practice

- Which websites can support me in finding IT solutions that have a specific licence ?

- How do I establish under which licence an IT solution is released ?

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

In practice: Search for existing IT solutions filtering by their licence type



Joinup allows users to search easily for solutions by licence type so they can be adopted in your organisation, thus avoiding compatibility risks.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

In practice: which tools can detect the licences of a project ?

TOOL	LICENCE	WEBSITE
Fossology	GPL 2 or LGPL 2.0	https://github.com/fossology/fossology
NPM License Checker	BSD 3	https://github.com/davglass/license-checker
Licensee	MIT	https://github.com/benbalter/licensee
License Finder	MIT	https://github.com/pivotal/LicenseFinder
Ninka	GPL 2	https://github.com/dmgerman/ninka
Jninka	AGPL 3	https://github.com/whitesource/jninka
Scancode toolkit	Apache 2	https://github.com/nexB/scancode-toolkit
License Maven plugin	LGPL 3	https://github.com/mojohaus/license-maven-plugin
Artifactory PRO	Commercial	https://www.jfrog.com/artifactory/
Nexus Pro	Commercial	http://www.sonatype.com/nexus-repository-sonatype

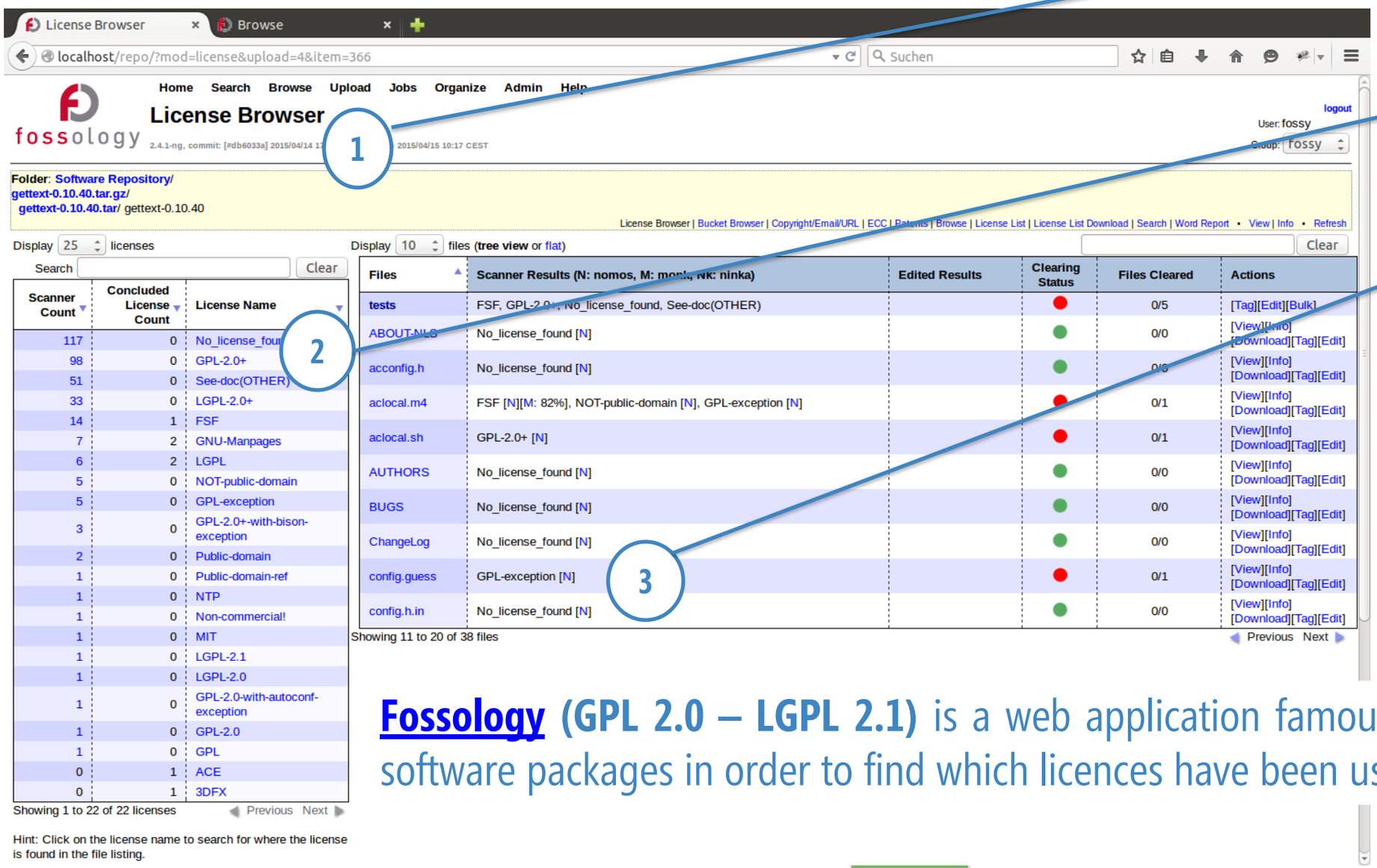
4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

In practice: which tools can detect the licences of a project ?

Upload iso, tar, rpm, jar, zip, bz2, msi, cab, etc.

Summary of the licences found

Licences found per file from different scanners (Nomos, Monk and Ninka)



Fossology (GPL 2.0 – LGPL 2.1) is a web application famous for scanning several type of software packages in order to find which licences have been used.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

In practice: which tools can detect the licences of a project ?

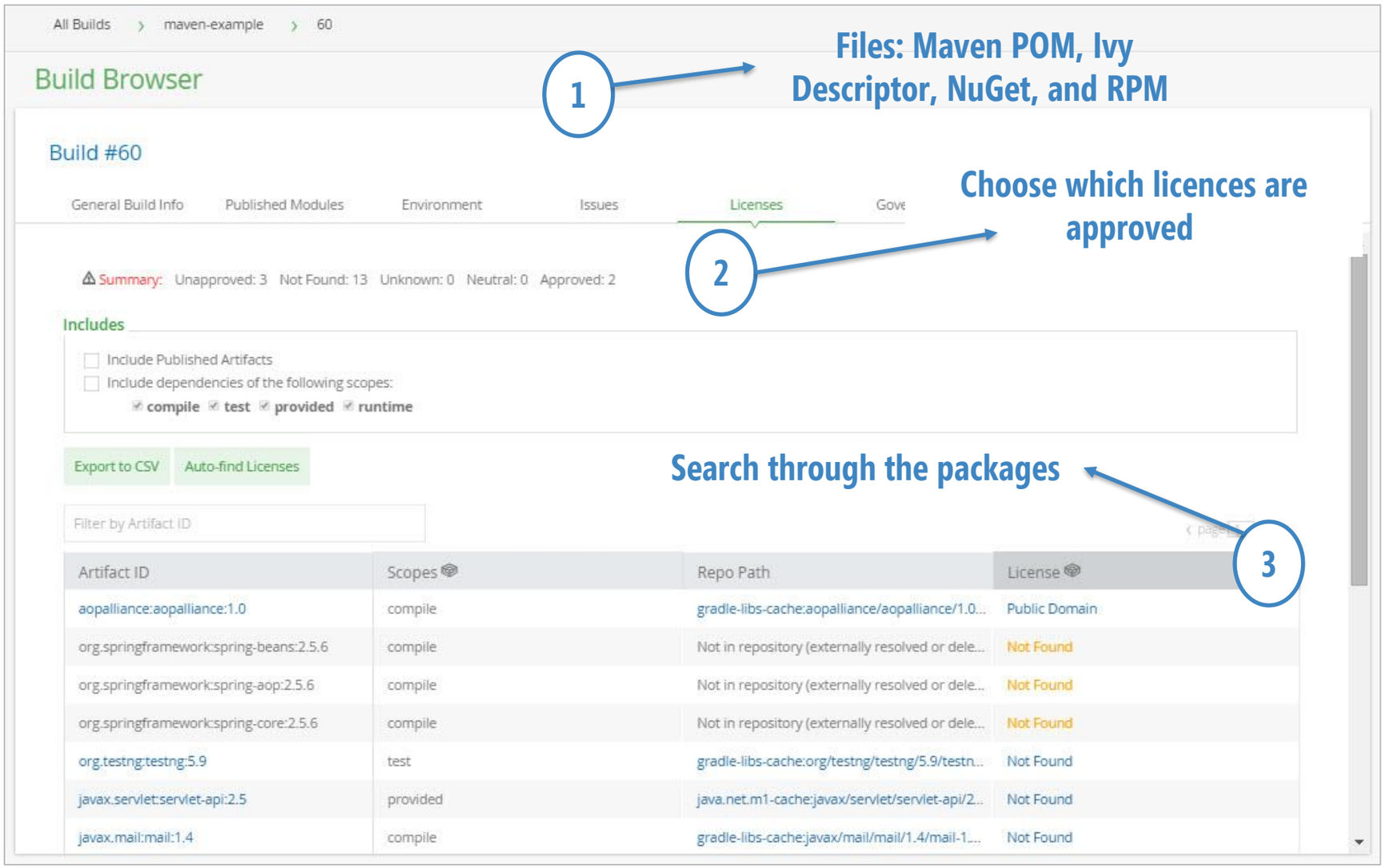
```
├─ cli@0.4.3
│  ├─ repository: http://github.com/chriso/cli
│  └─ licenses: MIT
├─ glob@3.1.14
│  ├─ repository: https://github.com/isaacs/node-glob
│  └─ licenses: UNKNOWN
├─ graceful-fs@1.1.14
│  ├─ repository: https://github.com/isaacs/node-graceful-fs
│  └─ licenses: UNKNOWN
├─ inherits@1.0.0
│  ├─ repository: https://github.com/isaacs/inherits
│  └─ licenses: UNKNOWN
├─ jshint@0.9.1
│  └─ licenses: MIT
├─ lru-cache@1.0.6
│  ├─ repository: https://github.com/isaacs/node-lru-cache
│  └─ licenses: MIT
├─ lru-cache@2.0.4
│  ├─ repository: https://github.com/isaacs/node-lru-cache
│  └─ licenses: MIT
├─ minimatch@0.0.5
│  ├─ repository: https://github.com/isaacs/minimatch
│  └─ licenses: MIT
├─ minimatch@0.2.9
│  ├─ repository: https://github.com/isaacs/minimatch
│  └─ licenses: MIT
├─ sigmund@1.0.0
│  ├─ repository: https://github.com/isaacs/sigmund
│  └─ licenses: UNKNOWN
└─ yui-lint@0.1.1
   ├─ licenses: BSD
   └─ repository: http://github.com/yui/yui-lint
```

NPM license checker

NPM allows users to automatically download software packages. The NPM license checker provides users with the possibility of retrieving all of the dependencies' licences.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

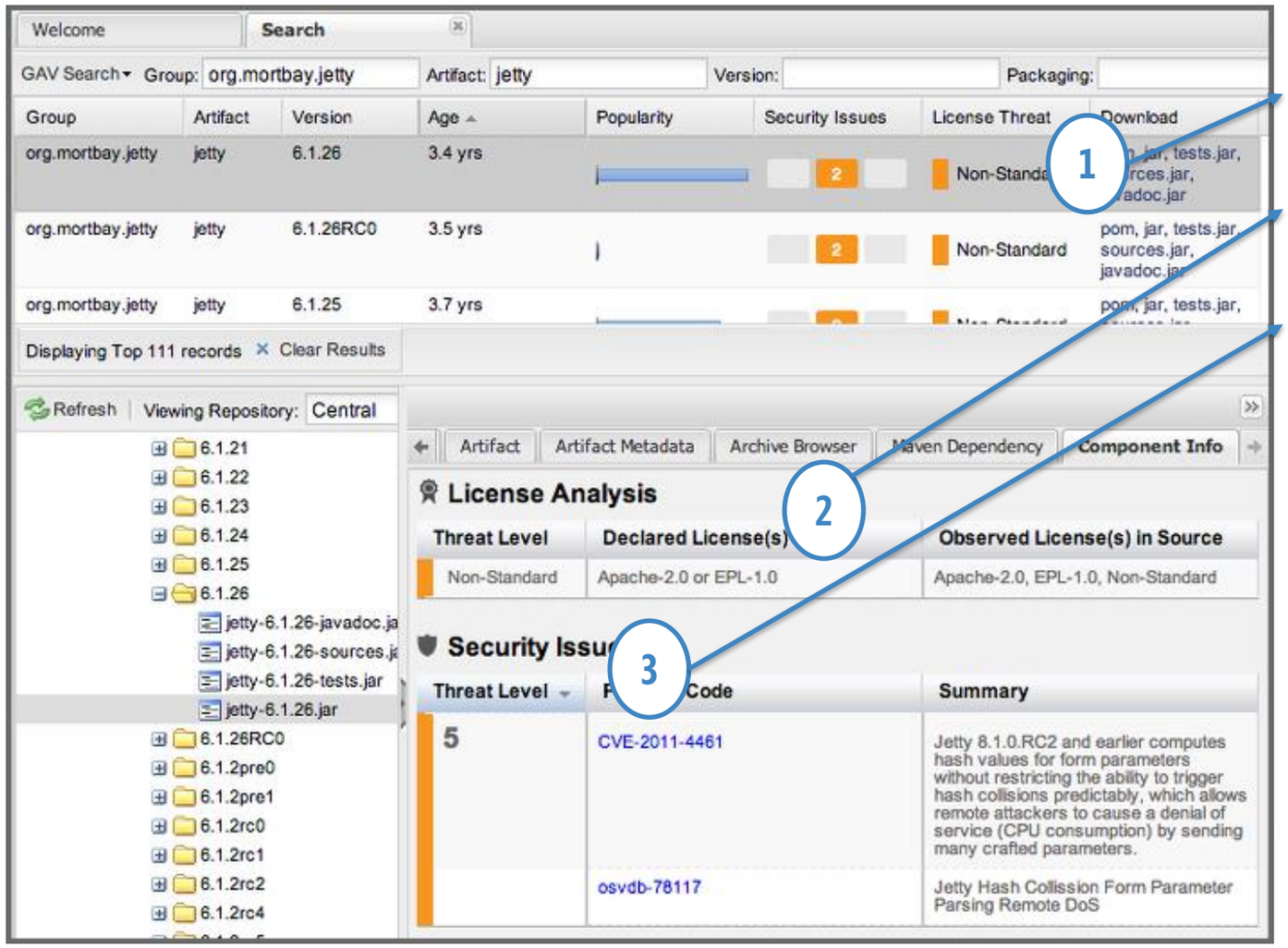
In practice: which tools can detect the licences of a project ?



The commercial version of **Artifactory** allows users to perform an automatic check of the licences used in different archives.

4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

In practice: which tools can detect the licences of a project ?



Licence classification

Licence declared in the pom.xml

Security issues

Maven projects are described by a pom.xml file, which contains a description about the licence.

The commercial version of **Nexus** allows users to check security and licence issues before releasing software.

European Commission

5. ENSURE THAT YOUR IT SOLUTION IS EXTENSIBLE

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

Problem statements

- Public administrations need to consider the technical characteristics of an IT solution when taking the decision whether to reuse it.
- The internal architecture of an IT solution can impact its reusability.
- Monolithically built systems are very difficult to modify even to accommodate small differences in business needs.
- Outdated, inflexible technologies make it difficult to scale up IT systems.

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

Solutions

- Develop IT solutions that are able to evolve beyond their current functionalities so that the internal structure is hardly affected, if at all.
- Allowing third-parties to extend an IT solution has the intrinsic effect of spreading the IT solution, i.e. increasing its reusability. This implies that the IT solutions needs to be flexible enough to accommodate extensions.

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

In practice: manage dependencies automatically

There are several tools available to easily manage dependencies for different programming languages:

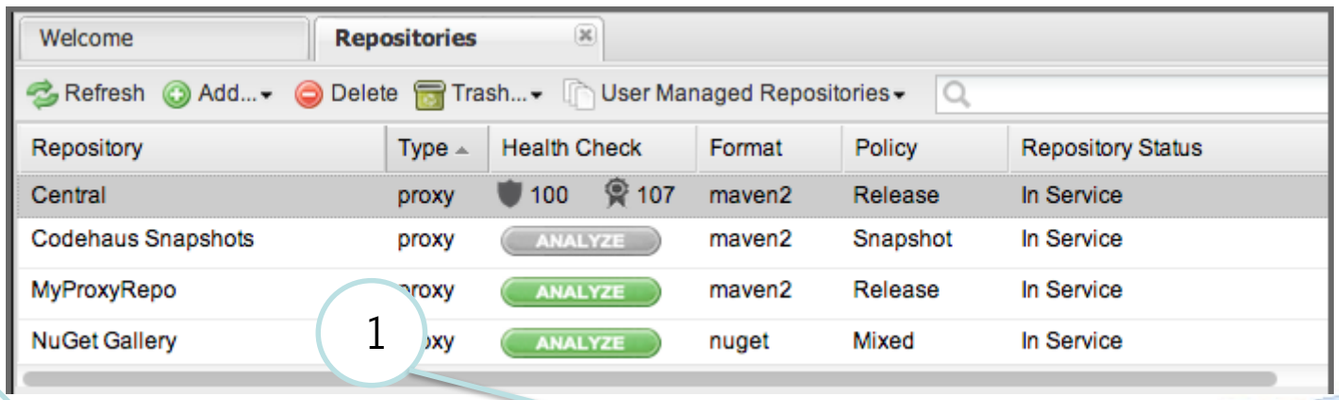
- Maven (Java)
- NPM (node.js)
- Packagist (PHP)
- Pypi (Python)
- Rubygems (Ruby)
- Bower (HTML,CSS, Javascript)

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

In practice: manage dependencies automatically

Maven repository manager by Artifactory or Nexus can act as a dedicated proxy server for public repositories. Developers have to indicate the Maven repository used in their pom.xml file.

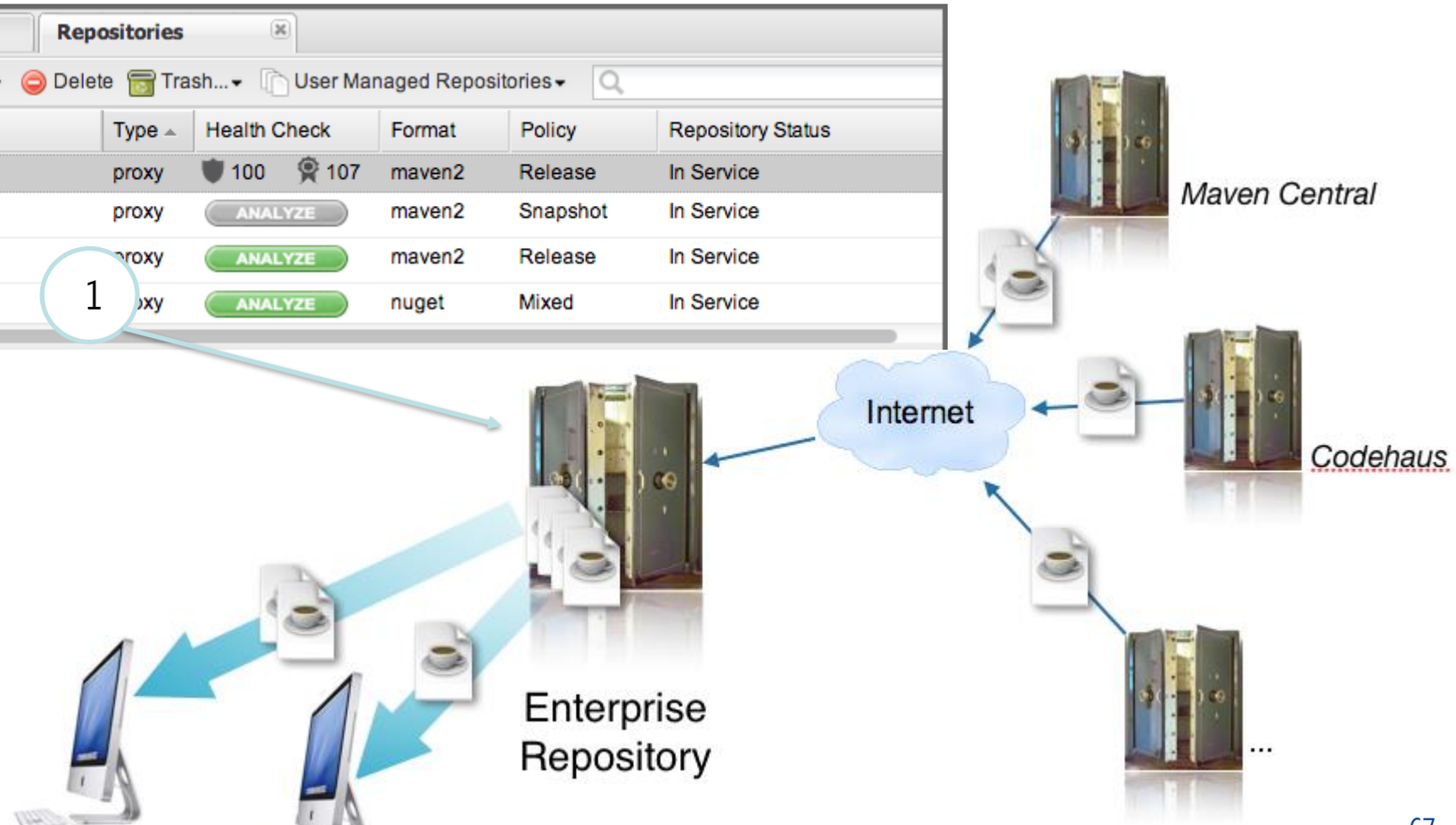
NEXUS CONFIGURATION PANEL



POM.XML

```
<repositories>
  <repository>
  <id>my-entrerpise-repo</id>
  <url>http://myserver/repo</url>
</repository>
</repositories>
<dependencies>
  <dependency>
    <groupId>junit</groupId>
    <artifactId>junit</artifactId>
    <version>4.10</version>
  </dependency>

  <dependency>
    <groupId>log4j</groupId>
    <artifactId>log4j</artifactId>
    <version>1.2.17</version>
  </dependency>
</dependencies>
```



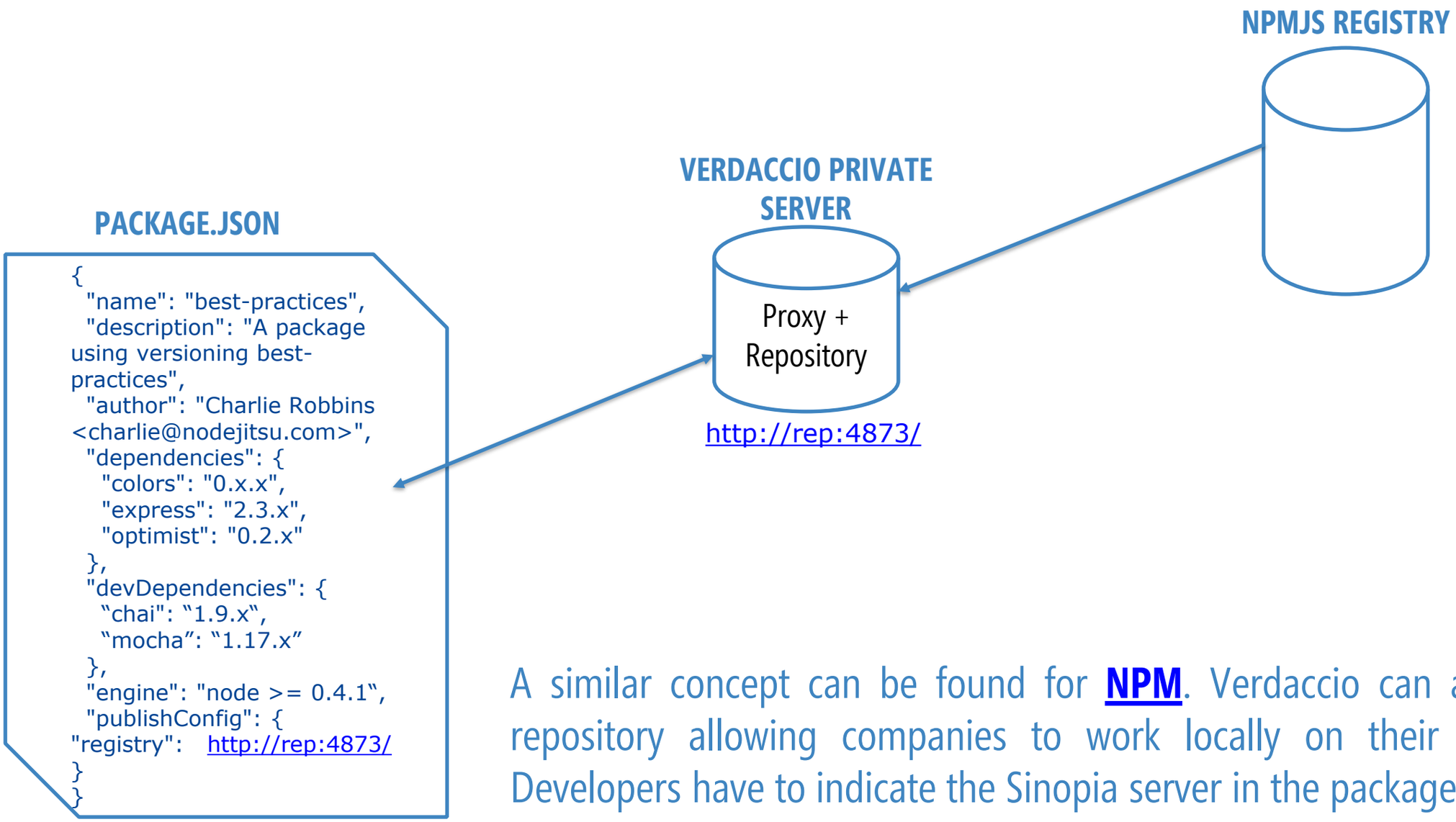
5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

In practice: manage dependencies automatically - examples in the EU

- eSignature and eDelivery from CEF programme use Maven.
- Open e-TrustEx for secure exchange of digital documents use Maven.
- The Publications Office with the eParticipation portal and RDFEdit use Maven.
- The Europass tool is based on Maven.
- Eurostat: statistical software are based on Maven.
- Austria: the MOCCA application for citizen card and PDF-AS/PDF-Over for digital signature use Maven.
- Belgium: the MAGDA platform uses Maven.
- Spain: the GIS library SEXTANTE and OpenFWPA use Maven.

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

In practice: manage dependencies automatically



A similar concept can be found for **NPM**. Verdaccio can act as proxy and repository allowing companies to work locally on their private packages. Developers have to indicate the Sinopia server in the package.json file.

5.1 DESIGN YOUR IT SOLUTION TO BE EXTENSIBLE & MODULAR

In practice: extend solutions via hooks

Hooks are functions exposed by an IT solution. They are used at specific times to alter the behaviour or data of an application.

This is the case for the **CMS Drupal (used by Joinup)**.

hook_file_validate	modules/system/system.api.php	Check that files meet a given criteria.
hook_filter_format_disable	modules/filter/filter.api.php	Perform actions when a text format has been disabled.
hook_filter_format_insert	modules/filter/filter.api.php	Perform actions when a new text format has been created.
hook_filter_format_update	modules/filter/filter.api.php	Perform actions when a text format has been updated.
hook_filter_info	modules/filter/filter.api.php	Define content filters.
hook_filter_info_alter	modules/filter/filter.api.php	Perform alterations on filter definitions.
hook_flush_caches	modules/system/system.api.php	Add a list of cache tables to be cleared.
hook_form	modules/node/node.api.php	Display a node editing form.
hook_forms	modules/system/system.api.php	Map form_ids to form builder functions.
hook_form_alter	modules/system/system.api.php	Perform alterations before a form is rendered.
hook_form_BASE_FORM_ID_alter	modules/system/system.api.php	Provide a form-specific alteration for shared ('base') forms.
hook_form_FORM_ID_alter	modules/system/system.api.php	Provide a form-specific alteration instead of the global hook_form_alter().
hook_help	modules/system/system.api.php	Provide online user help.

```
/**
 * Implements hook_help().
 *
 * Displays help and module information.
 *
 * @param path
 *   Which path of the site we're using to display help
 * @param arg
 *   Array that holds the current path as returned from arg() function
 */
function current_posts_help($path, $arg) {
  switch ($path) {
    case "admin/help#current_posts":
      return '' . t("Displays links to nodes created on this date") . '';
      break;
  }
}
```

Drupal has 30.000 modules (extensions) as a result of the high number of hooks ready to be used.

5.2 DESIGN YOUR IT SOLUTION TO BE SCALABLE

Problem statements

- Public administrations may have similar needs. However this does not imply that their needs are identical.

- In addition to similarities, there are differences and dependencies imposed by existing applications, which organisations may be unwilling or unable to abandon.

- As electronic public services become more common, IT solutions may need to be scaled up. The existence of inflexible technologies can limit the scale-up of IT solutions.

5.2 DESIGN YOUR IT SOLUTION TO BE SCALABLE

Solutions

Public administrations should ensure the **scalability** of their solution by:

- Designing a modular architecture
- Analysing the building blocks that best fit the objectives of the IT solution
- Monitoring performance during and after the implementation of the IT solution

Performance influences the maximum number of visitors using your web application!

5.2 DESIGN YOUR IT SOLUTION TO BE SCALABLE

Solutions

The architecture of IT solutions can be designed to be scalable by taking advantage of different concepts such as:

- **Multi-threading**, which depends on the nature of the chosen programming language for the IT solution, such as Java or C#.
- **Caching**, which depends on whether the IT solution has a caching mechanism (e.g. Drupal) or it uses a caching framework (Xcache).
- **Clustering framework**, which depends on whether the IT solution is deployed on a server which supports clustering (e.g. Tomcat).
- **Clustered file system**, which depends on whether the IT solution can take advantage of the underlying file system (e.g. HDFS).

5.2 DESIGN YOUR IT SOLUTION TO BE SCALABLE

In practice

To monitor the performance with analytics **Piwik** (GPL 3 license) can be used.

Recommendations to test the performance:

- Jmeter (Apache license);
- Gatling (Apache license);
- Grinder (BSD license);
- Funkload (GPL license);
- Web page test (BSD license).

5.2 DESIGN YOUR IT SOLUTION TO BE SCALABLE

In practice: examples in the EU

- Joinup uses Piwik to web site monitor performances and it is hosted as AWS machine.
- Open e-PRIOR and OpenFWPA use Jmeter for performance test.
- Ireland: Police force uses SugarCRM for its scalability.
- Netherlands: OpenStack as cloud platform for the datacentre.
- Spain: Hadoop and Hbase for distributed file system.
- UK: Government portal runs on AWS machine and uses Memcached.

5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

Problem statements

- Public administrations need to share more and more information across borders and sectors.
- Improved information exchange, therefore, can facilitate better services to the citizens.
- Information exchange is easier when interoperability exists and when there are few or no obstacles to communication across borders and sectors.
- Reusing software can imply a lot of effort, especially if the solution needs to be adapted to the organisation's needs.

5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

Solutions

- Certain functionalities could be offered as services to all public administrations or accessed through an API.
- APIs make it possible for developers of a solution to expose functionalities of the solution, while protecting the rest of the application.

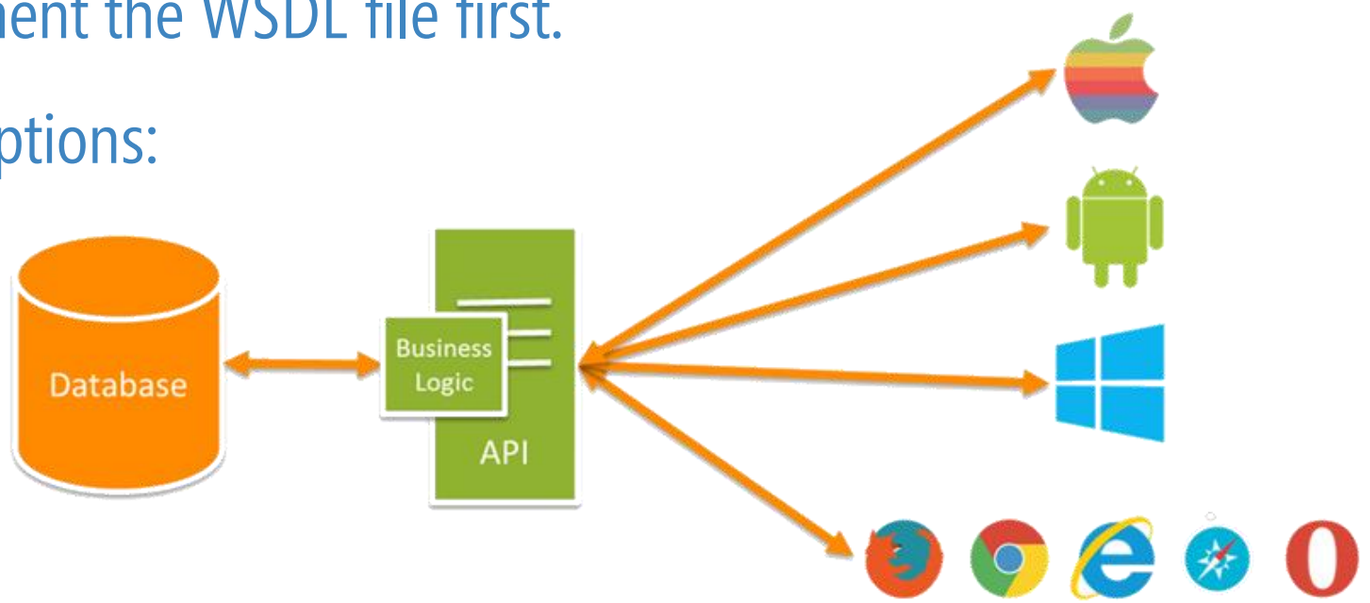
5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

In practice

When creating an API model, contracts should be developed first:

1. You can reuse existing models and controlled vocabularies such as **Eurovoc**;
2. A contract helps you to focus on the data exchanged. Client and server code will be generated from the contract.

- For **SOAP API,** you need to implement the WSDL file first.
- For **REST API,** there are different options:
  - SWAGGER (now OAI);
  - RAML;
  - Blueprint.



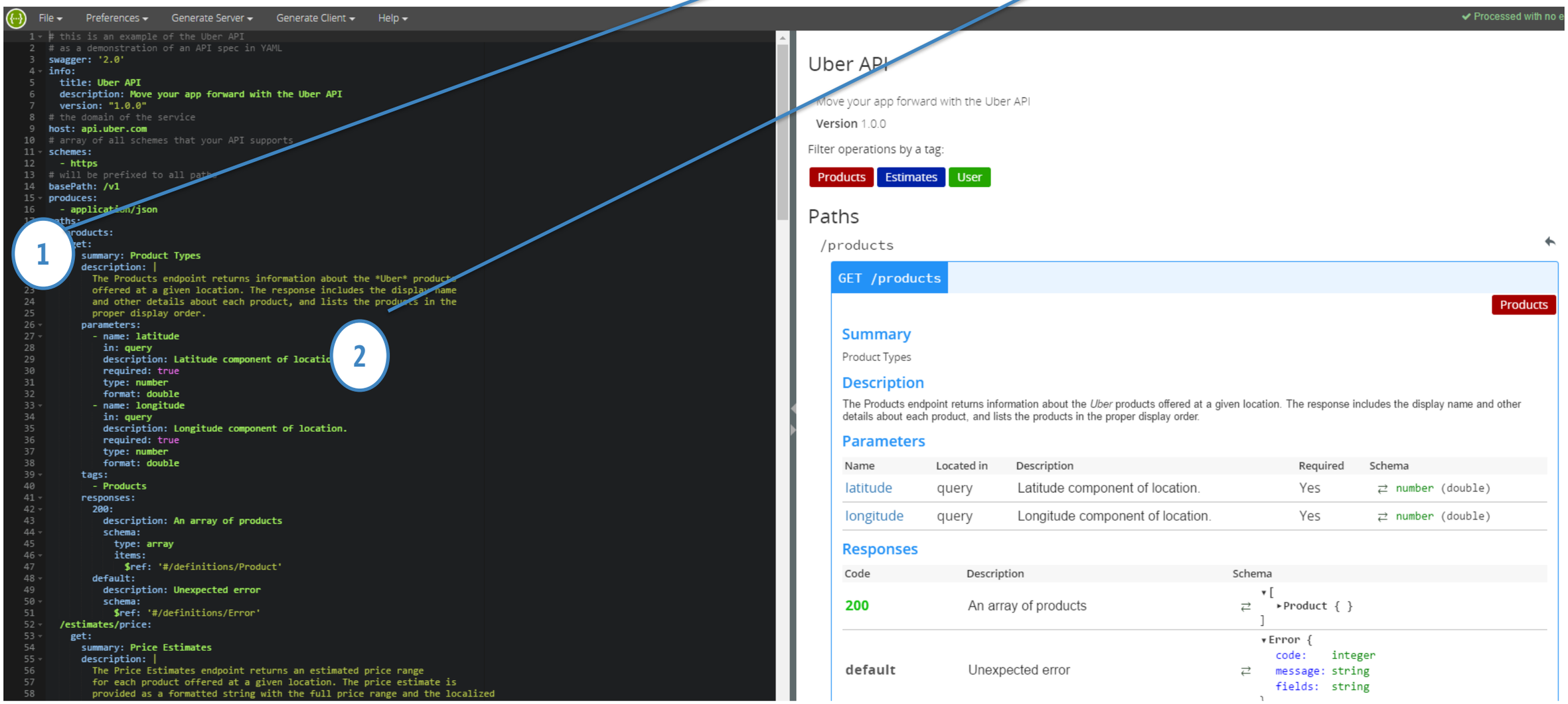
European Commission

5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

In practice

Path + protocol

Parameters



The **swagger editor** allows to describe the interface and provide a preview of the generated documentation.

5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

In practice: examples in the EU

- Austria: MOA ZS API based on SOAP with XML syntax
- Estonia: X-road API are based on REST with JSON syntax
- Finland: Finnish Service Catalogue based on REST with JSON syntax
- France: SIRENE API based on REST with JSON syntax
- Germany: Xrepository API based on SOAP with XML syntax
- Italy: FatturaPA, API based on REST with XML syntax
- UK: Waste Management API based on REST with JSON syntax

European Commission

6. MAKE YOUR SOLUTION INTEROPERABLE

6.1 FOLLOW BASIC INTERNATIONALISATION PRINCIPLES

Problem statements

- Public administration leaders may not always see a business need for multilingualism from a user's perspective.

- However, for an IT solution to be reused across borders, it should be internationalised. Internationalisation in this context refers to designing software so that it can be adapted to other languages or regions without needing major engineering changes.

- A lack of multilingual support in the structure of an IT solution can make reuse impossible. As a result, the IT solution will not be able to build a community around itself, thus endangering its sustainability.

6.1 FOLLOW BASIC INTERNATIONALISATION PRINCIPLES

Solutions

- Localisation means adapting an IT solution to meet linguistic, cultural and other requirements of a specific market.

- The effort required for localisation should be limited. This is achieved by designing software with internationalisation principles in mind; making it possible for others to adapt the IT solution to various languages and regions without the need for major changes.

- Subtle differences like date formats, for example, need to be taken into account as they differ between languages.

6.1 FOLLOW BASIC INTERNATIONALISATION PRINCIPLES

In practice

There are plugins available to help parse and format numbers, currencies, and dates according to culture, language or region. The **javascript Globalize plugin** is a JavaScript library that covers information for more than 350 languages, ranging from obvious differences (currency) to subtle ones (grouping digits by 3 or by 2 in large numbers).

English = Wednesday, June 09, 2010
Arabic = 26/1431/جمادى الثانية
Bulgarian = 09 юни 2010 г.
Catalan = dimecres, 9 / juny / 2010
Chinese (Simplified) = 2010年6月9日
Czech = 9. června 2010
Danish = 9. juni 2010
German = Mittwoch, 9. Juni 2010
Greek = Τετάρτη, 9 Ιουνίου 2010

From Scott Gu's blog

6.2 PROVIDE DOCUMENTATION IN MULTIPLE LANGUAGES

Problem statements

- The EU is a multilingual environment by definition, with 24 official languages spoken by its population.
- To facilitate the reuse of IT solutions, their documentation should be easy to understand by IT professionals across the EU.
- Translating technical documentation can be a heavy task, even for one additional language.

6.2 PROVIDE DOCUMENTATION IN MULTIPLE LANGUAGES

Solutions

- To reduce language barriers, public administrations should document IT solutions in one or more additional languages. To make this process easier and more efficient, machine translation is a possible solution.

- To facilitate machine translation, it is a good practice to use common file formats such as Po, XLIFF or YML.

6.2 PROVIDE DOCUMENTATION IN MULTIPLE LANGUAGES

In practice

To increase collaboration, developers should use software to support translation such as:

- Pootle (GPL 3)
- Weblate (GPL 3)

Furthermore developers can rely on services that can help to support the translation:

- Zanata (LGPL)
- Transifex
- Poeditor

6.2 PROVIDE DOCUMENTATION IN MULTIPLE LANGUAGES

In practice: examples in the EU

- In France: the OpenMairie framework provides a catalogue of solutions for municipalities and is provided in French and English.
- In Spain: Junta de Andalucia provides Gecos, with IT management infrastructure, comment and documentation available in Spanish and English.

European Commission

7. PROVIDE MAINTENANCE AND SUPPORT

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

Problem statements

- Potential reusers look at the availability of maintenance and support for an IT solution.

- A lack of maintenance and support structures have a considerable impact on the reusability of a solution and the trust that it inspires.

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

Solutions

- Plan for a proper organisation of maintenance and support services.

- If support is provided by a third party, potential reusers should verify if and under what conditions they are entitled to support.

- Using a website to handle the Frequently Asked Questions can improve end user support activity.

- Rely on existing software to provide support.

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice

An optimal way to define maintenance and support structures are **Service Level Agreements**.

SLAs include support and maintenance operations such as:

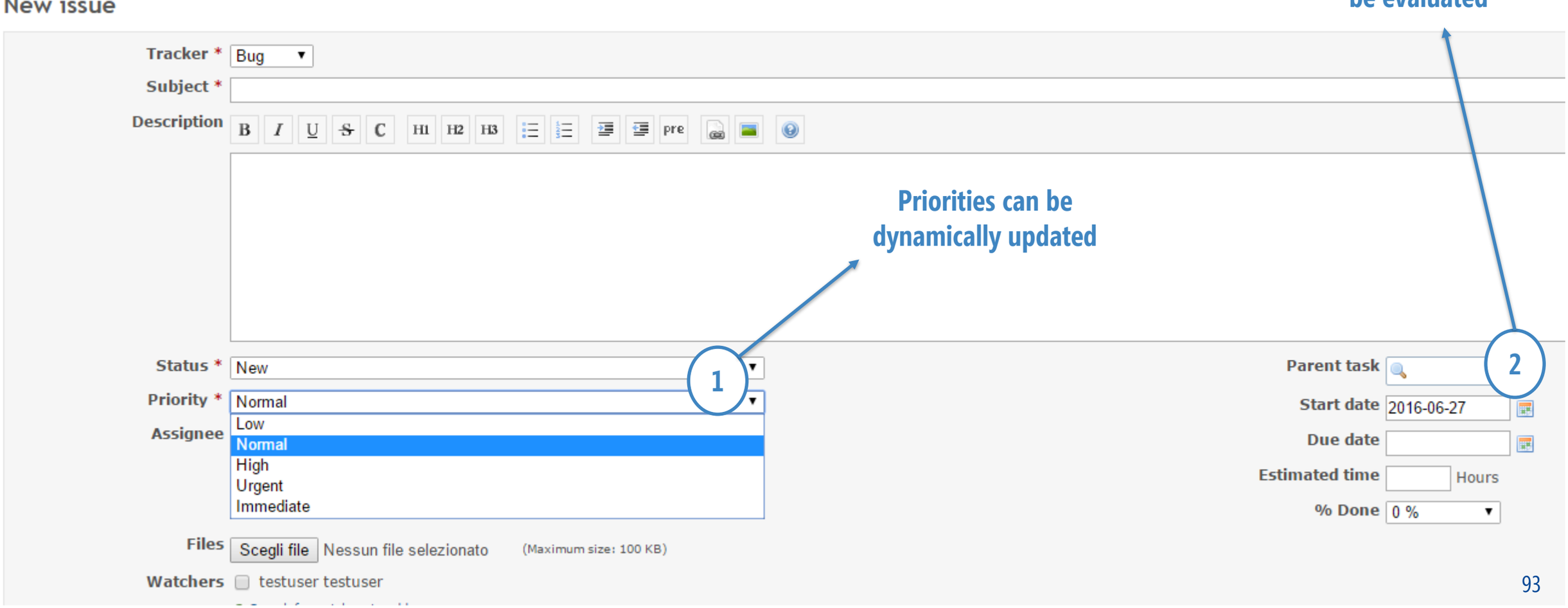
- Hours of operation
- Help desk contact availability
- Priority assignment criteria
- Definition of Response Time depending on the Severity Level
- Definition of Resolution Time to solve issues
- Maintenance schedules
- Notification mechanisms

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: Maintenance and support services

Dealing with priorities and time in **Redmine**:

Response time and resolution progress can be evaluated



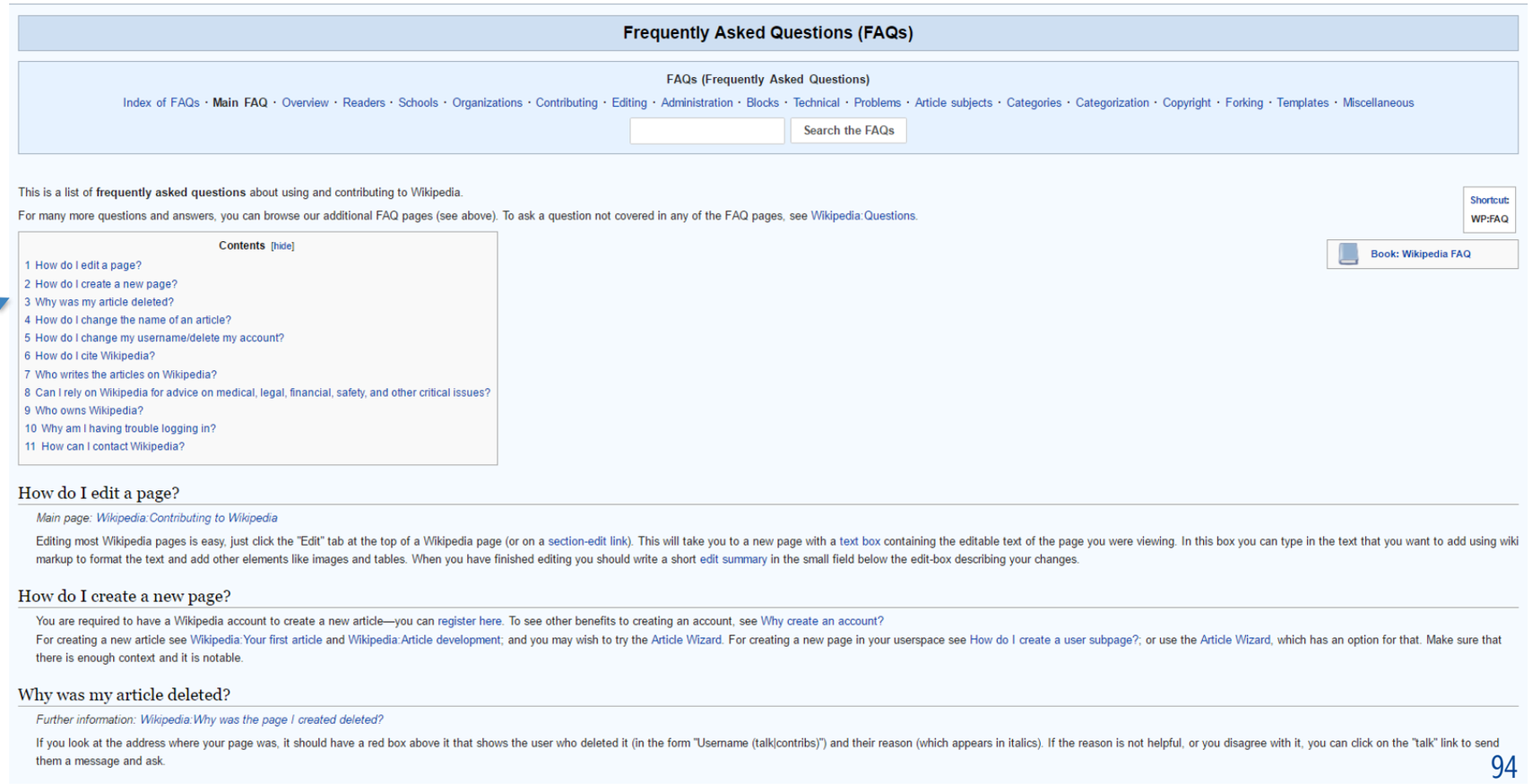
Priorities can be dynamically updated

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: frequently asked questions

For the FAQ, you can use a static site for this or a wiki such as **GitHub wiki, Redmine Wiki or Mediawiki** (GPL 2) as presented below.

Wikis such as Mediawiki allow the automatic creation of a Table of Contents, which is helpful for users to find the answer to their question.



Frequently Asked Questions (FAQs)

FAQs (Frequently Asked Questions)

Index of FAQs · **Main FAQ** · Overview · Readers · Schools · Organizations · Contributing · Editing · Administration · Blocks · Technical · Problems · Article subjects · Categories · Categorization · Copyright · Forking · Templates · Miscellaneous

Search the FAQs

This is a list of **frequently asked questions** about using and contributing to Wikipedia.

For many more questions and answers, you can browse our additional FAQ pages (see above). To ask a question not covered in any of the FAQ pages, see Wikipedia:Questions.

Shortcut: WP:FAQ

Book: Wikipedia FAQ

Contents [hide]

1 How do I edit a page?
2 How do I create a new page?
3 Why was my article deleted?
4 How do I change the name of an article?
5 How do I change my username/delete my account?
6 How do I cite Wikipedia?
7 Who writes the articles on Wikipedia?
8 Can I rely on Wikipedia for advice on medical, legal, financial, safety, and other critical issues?
9 Who owns Wikipedia?
10 Why am I having trouble logging in?
11 How can I contact Wikipedia?

How do I edit a page?

Main page: Wikipedia:Contributing to Wikipedia

Editing most Wikipedia pages is easy, just click the "Edit" tab at the top of a Wikipedia page (or on a section-edit link). This will take you to a new page with a text box containing the editable text of the page you were viewing. In this box you can type in the text that you want to add using wiki markup to format the text and add other elements like images and tables. When you have finished editing you should write a short edit summary in the small field below the edit-box describing your changes.

How do I create a new page?

You are required to have a Wikipedia account to create a new article—you can register here. To see other benefits to creating an account, see Why create an account?

For creating a new article see Wikipedia:Your first article and Wikipedia:Article development; and you may wish to try the Article Wizard. For creating a new page in your userspace see How do I create a user subpage?; or use the Article Wizard, which has an option for that. Make sure that there is enough context and it is notable.

Why was my article deleted?

Further information: Wikipedia:Why was the page I created deleted?

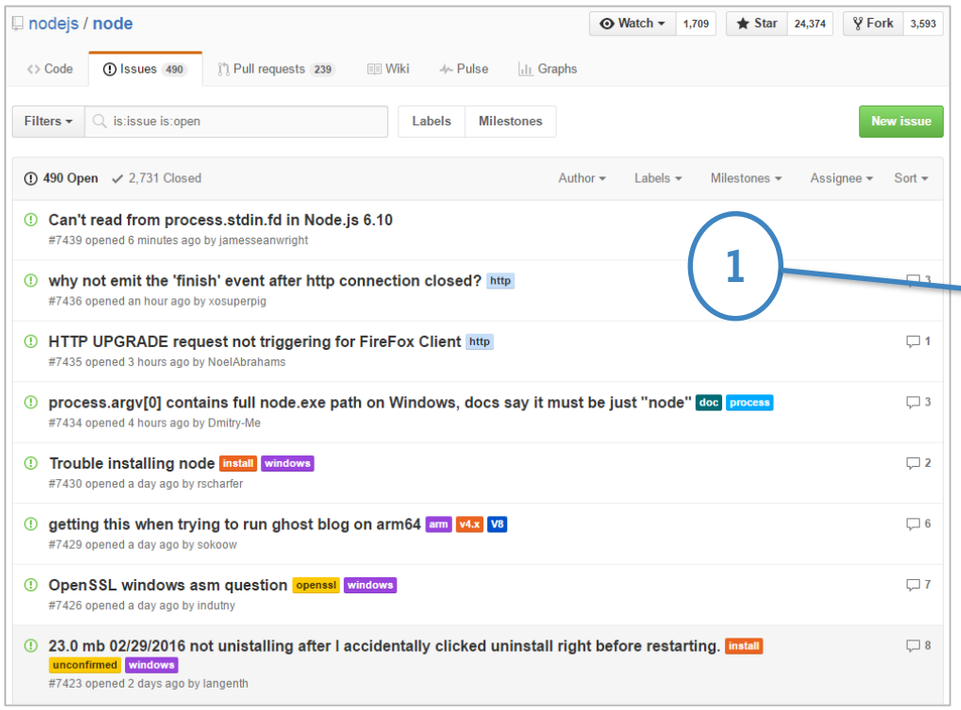
If you look at the address where your page was, it should have a red box above it that shows the user who deleted it (in the form "Username (talk|contribs)") and their reason (which appears in italics). If the reason is not helpful, or you disagree with it, you can click on the "talk" link to send them a message and ask.

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

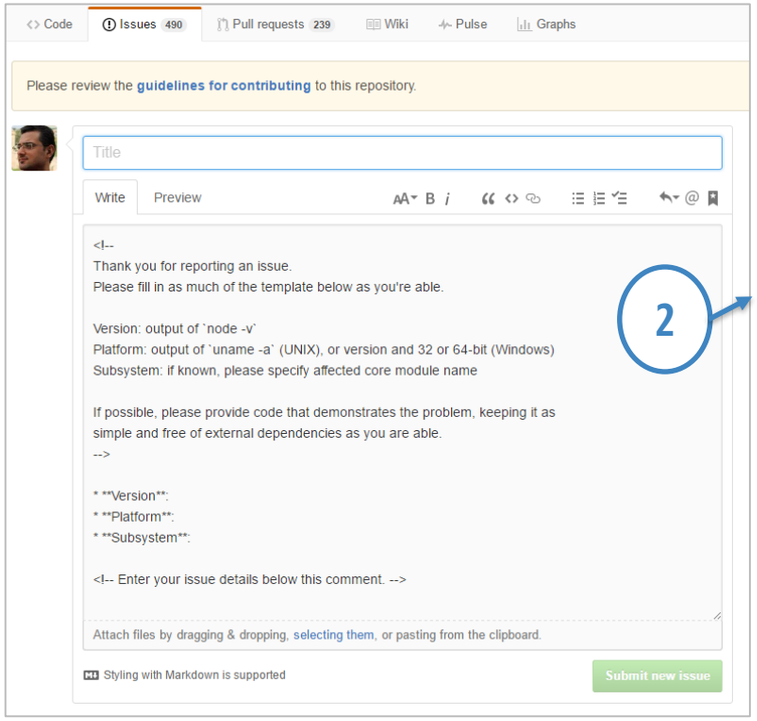
In practice: issue tracking

Public administrations are encouraged to provide user support through existing issue tracking systems. All repository tools mentioned before provide this.

GitHub: categorisation based on labels and issue templating



Colourful labels to categorise issues

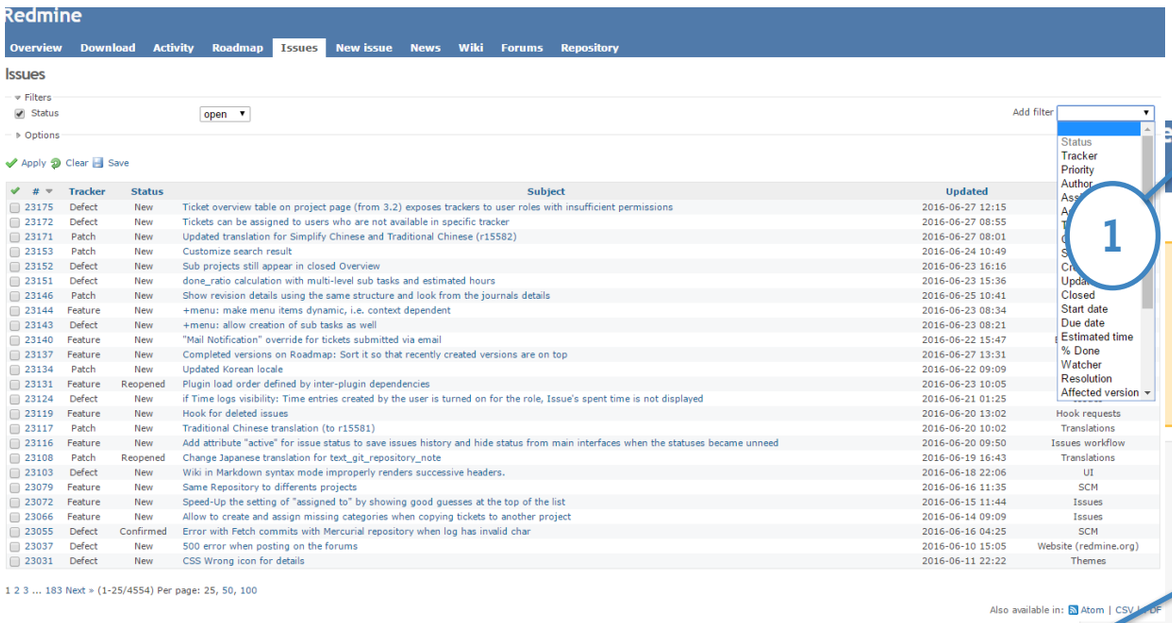


Templates help developers to understand issues better.

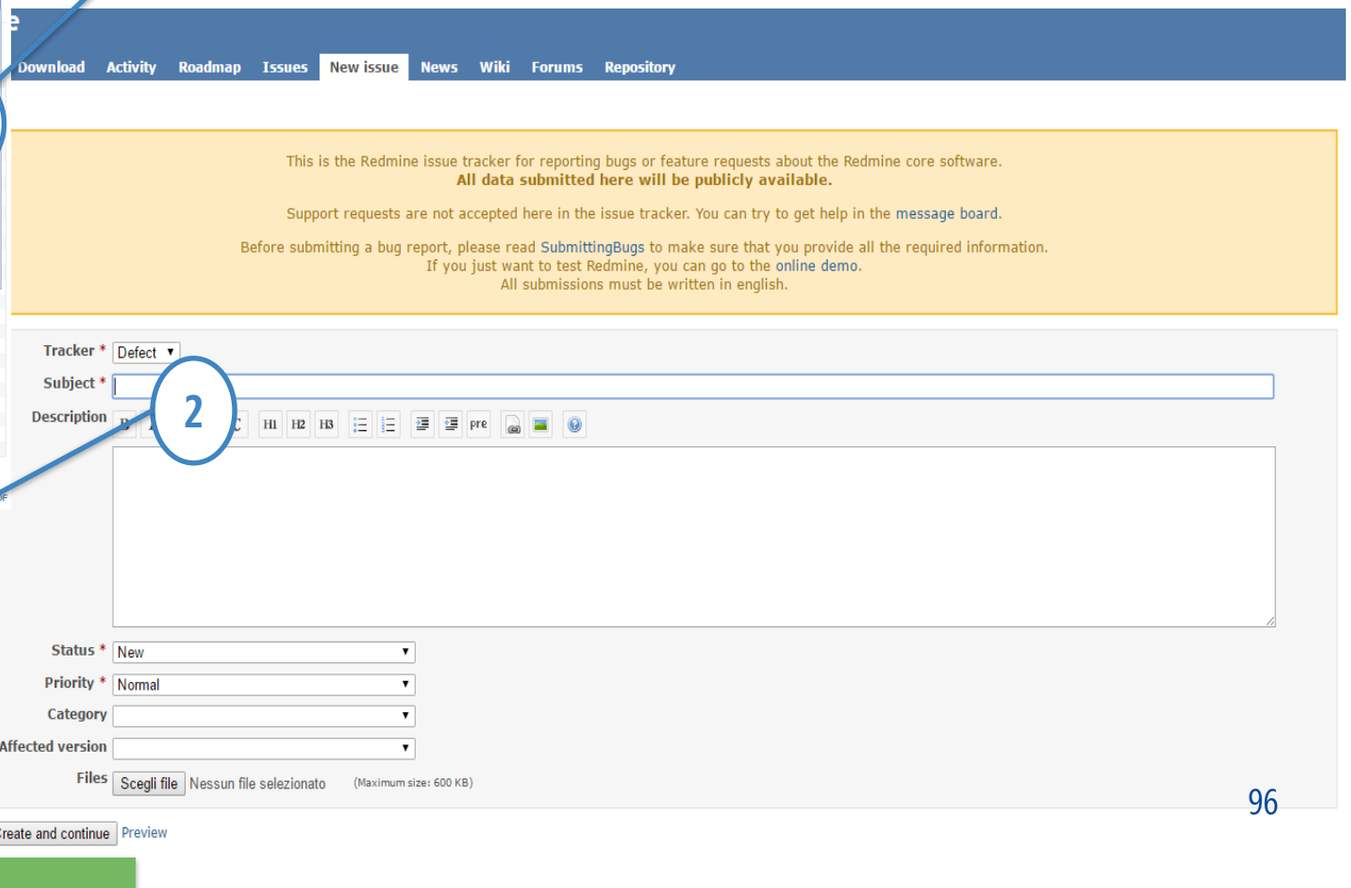
7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: issue tracking

Redmine: more issue field customisation



Filtering on the categories such as type of issue, start date, due date

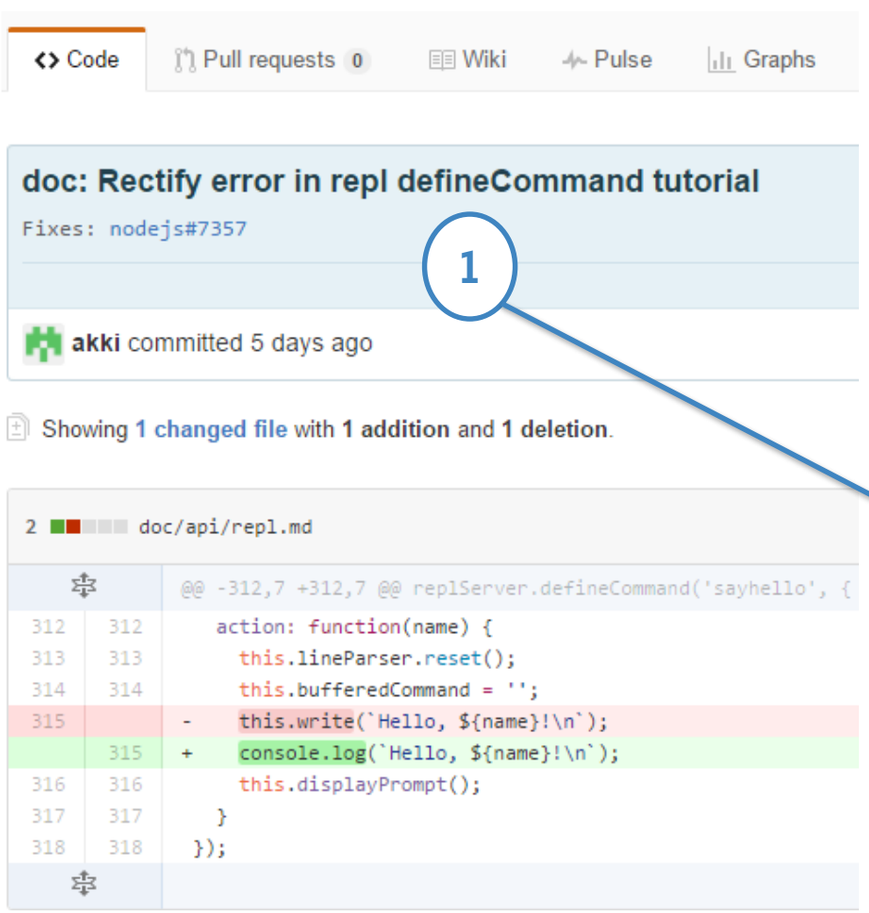


When a new issue is created, Redmine mandates the completion of some required fields such as tracker, subject, status and priority.

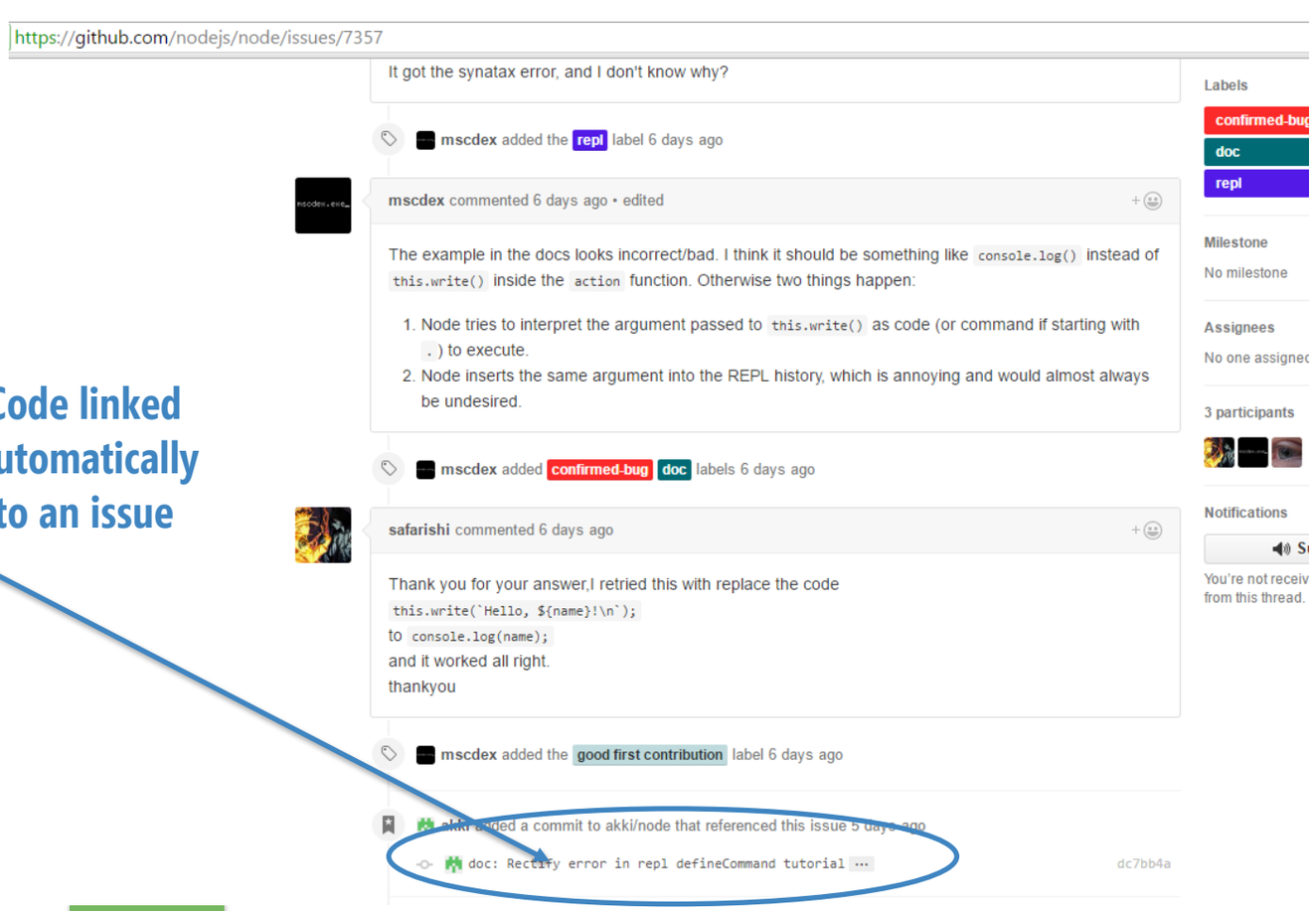
7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: issue tracking

Issue tracking solutions are able to link code to issues so it is possible to check whether an issue has been correctly implemented. Hereby an example in **GitHub**:



Code linked Automatically to an issue



7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: issue tracking

Example with **Redmine**:

Defect #23054

Clearing time entry custom fields while bulk editing results in values set to __none__

Added by Felix Schäfer 14 days ago. Updated 9 days ago.

Status:	Closed	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:	Jean-Philippe Lang	**% Done:**	0%
Category:	Time tracking		
Target version:	3.2.4		
Resolution:	Fixed	**Affected version:**	

23054-clear_time_entry_custom_fields.patch (886 Bytes) Felix Schäfer, 2016-06-13 11:48

History

Updated by Felix Schäfer 14 days ago #1

- **File** 23054-clear_time_entry_custom_fields.patch added

Updated by Jan from Planio www.plan.io 14 days ago #2

- **Target version** set to *Candidate for next minor release*

Updated by Jean-Philippe Lang 12 days ago #3

- **Target version** changed from *Candidate for next minor release* to *3.2.4*

Updated by Jean-Philippe Lang 9 days ago #4

- **Category** set to *Time tracking*
- **Status** changed from *New* to *Closed*
- **Assignee** set to *Jean-Philippe Lang*
- **Resolution** set to *Fixed*

Revision 15533

Added by Jean-Philippe Lang 12 days ago

Adds a test for #23054.

Related issues

- Defect #23054: Clearing time entry custom fields while bulk editing results in values set to __none__

Files

View differences

trunk
test
functional
timelog_controller_test.rb (diff)

1

Code linked Automatically to an issue

Associated revisions

Revision 15532
Added by Jean-Philippe Lang 12 days ago

Clearing time entry custom fields while bulk editing results in values set to *none* (#23054).

Patch by Felix Schäfer.

Revision 15533
Added by Jean-Philippe Lang 12 days ago

Adds a test for #23054.

Revision 15534
Added by Jean-Philippe Lang 12 days ago

Code cleanup (#23054).

Revision 15537
Added by Jean-Philippe Lang 9 days ago

Merged r15532 to r15534 (#23054).

Revision 15538
Added by Jean-Philippe Lang 9 days ago

Merged r15532 and r15533 (#23054).

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: examples in the EU

- Belgium: IMIO, IT service provider support for 75% of all municipalities in the Walloon region
- France: Adullact provides HelpDesk support
- Netherlands: TYPO3 association provides support through conferences and workshops
- Norway: FriKomPort, a training portal, provides support for municipalities
- Spain: gvHidra uses Redmine for issue tracking

8. CONTACT

CONTACT

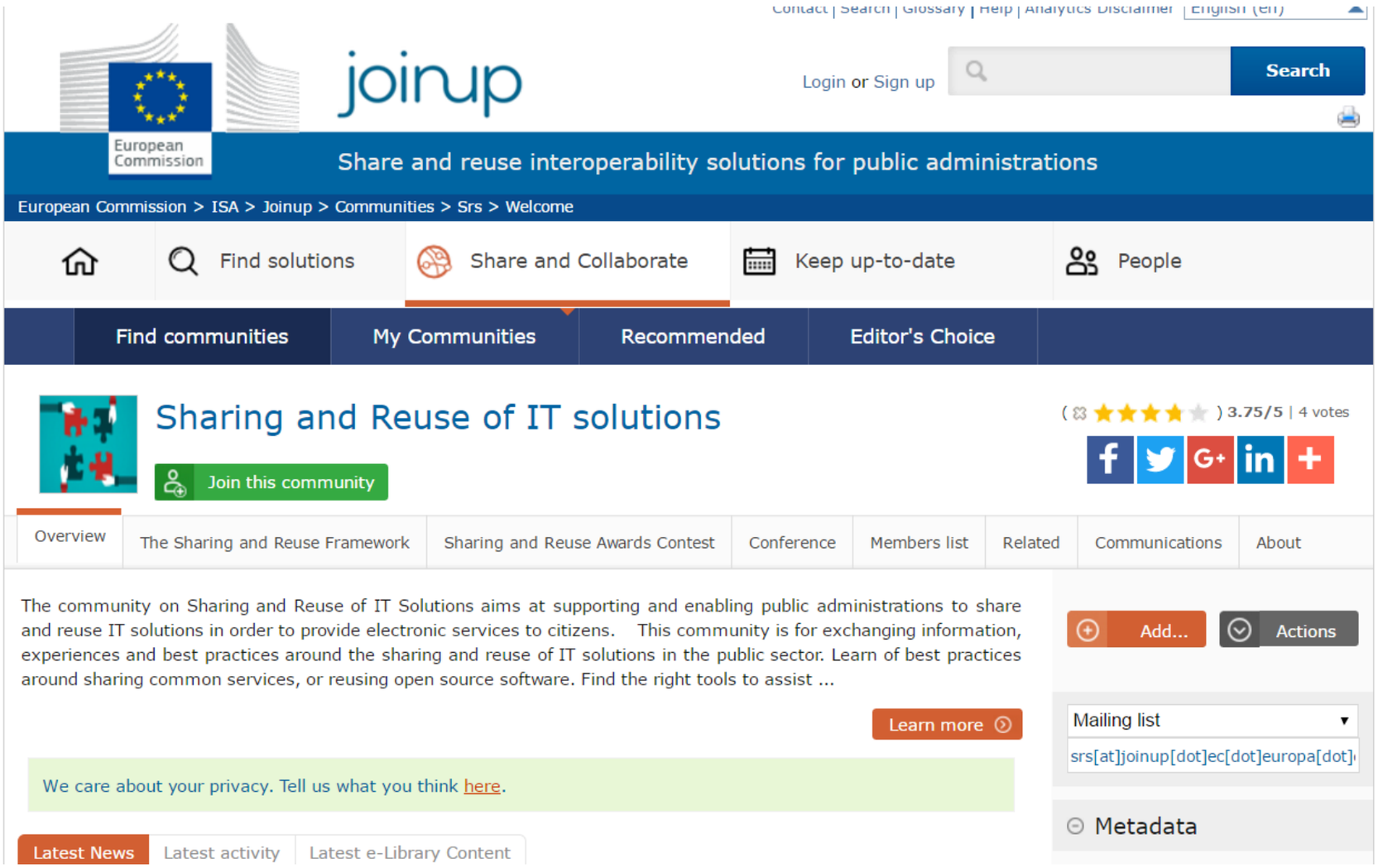
Find news about sharing and reusing IT solutions

Download the Sharing and Reuse Framework for IT Solutions

Take part in other related activities on the Joinup communities!

Join our Sharing and Reuse of IT Solutions Community and contact us:

Directorate D — Digital Services
Unit DIGIT.D.2 — Interoperability
E-mail: ISA2@ec.europa.eu
European Commission
B-1049 Brussels



European Commission