



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**.

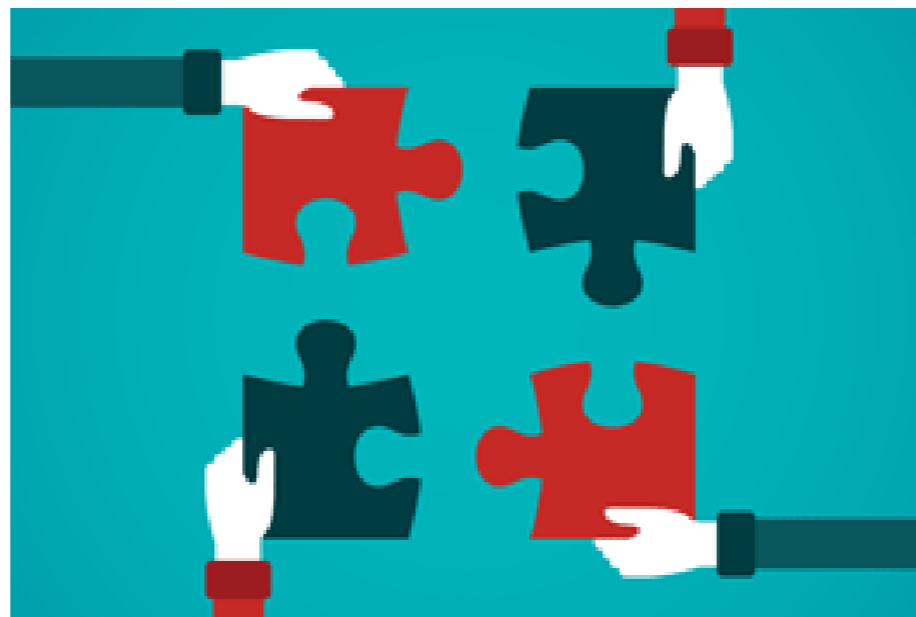


2. USE COMMON REGISTRIES TO SHARE AND REUSE IT SOLUTIONS



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.





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

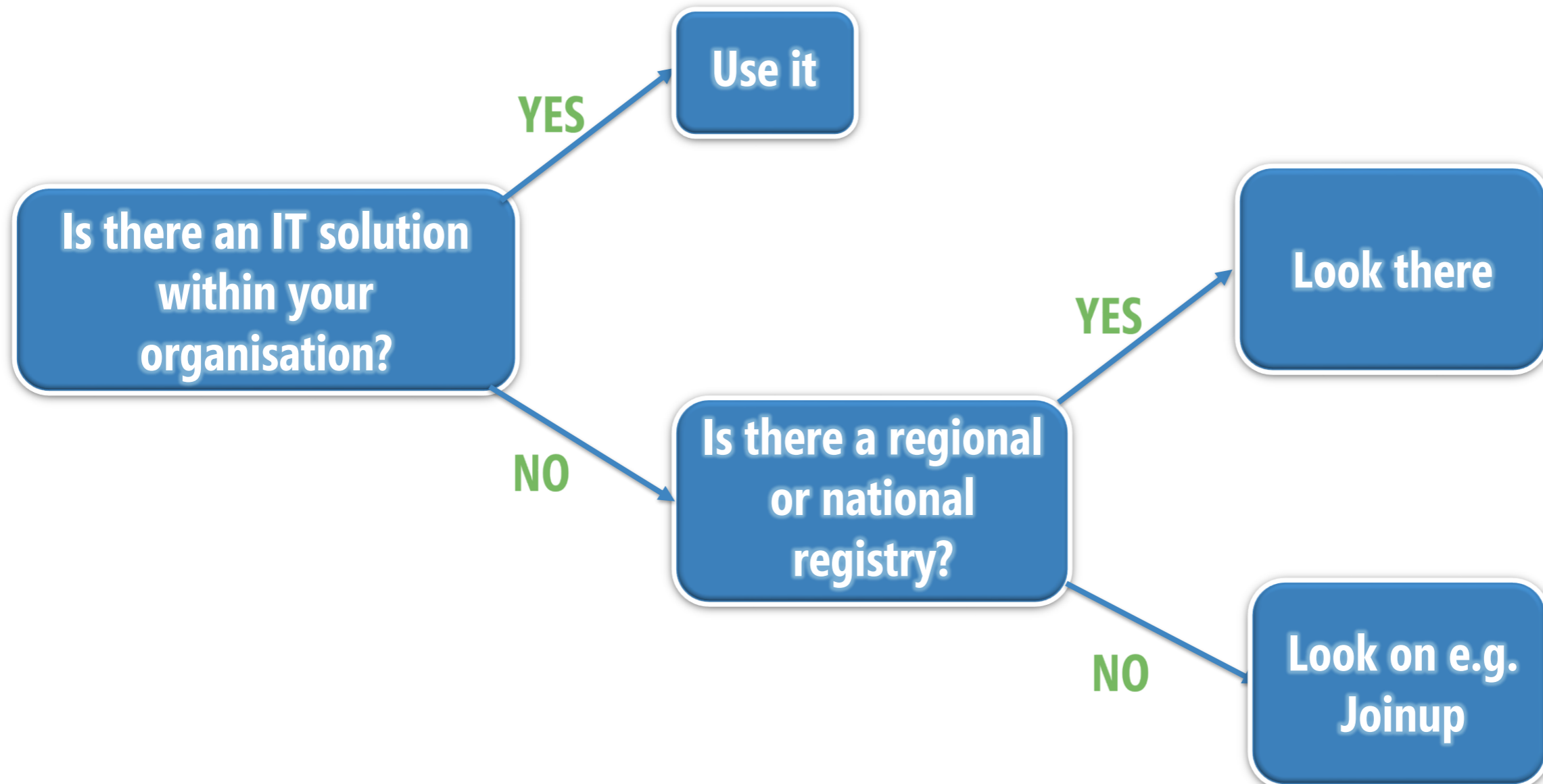
- A common registry is a system devoted to the proper description, structuring and publishing of IT solutions, together with the relevant documentation and process description in an integrated schema.
- Public administrations should consult common registries with a view to **reuse** existing IT solutions 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](#))

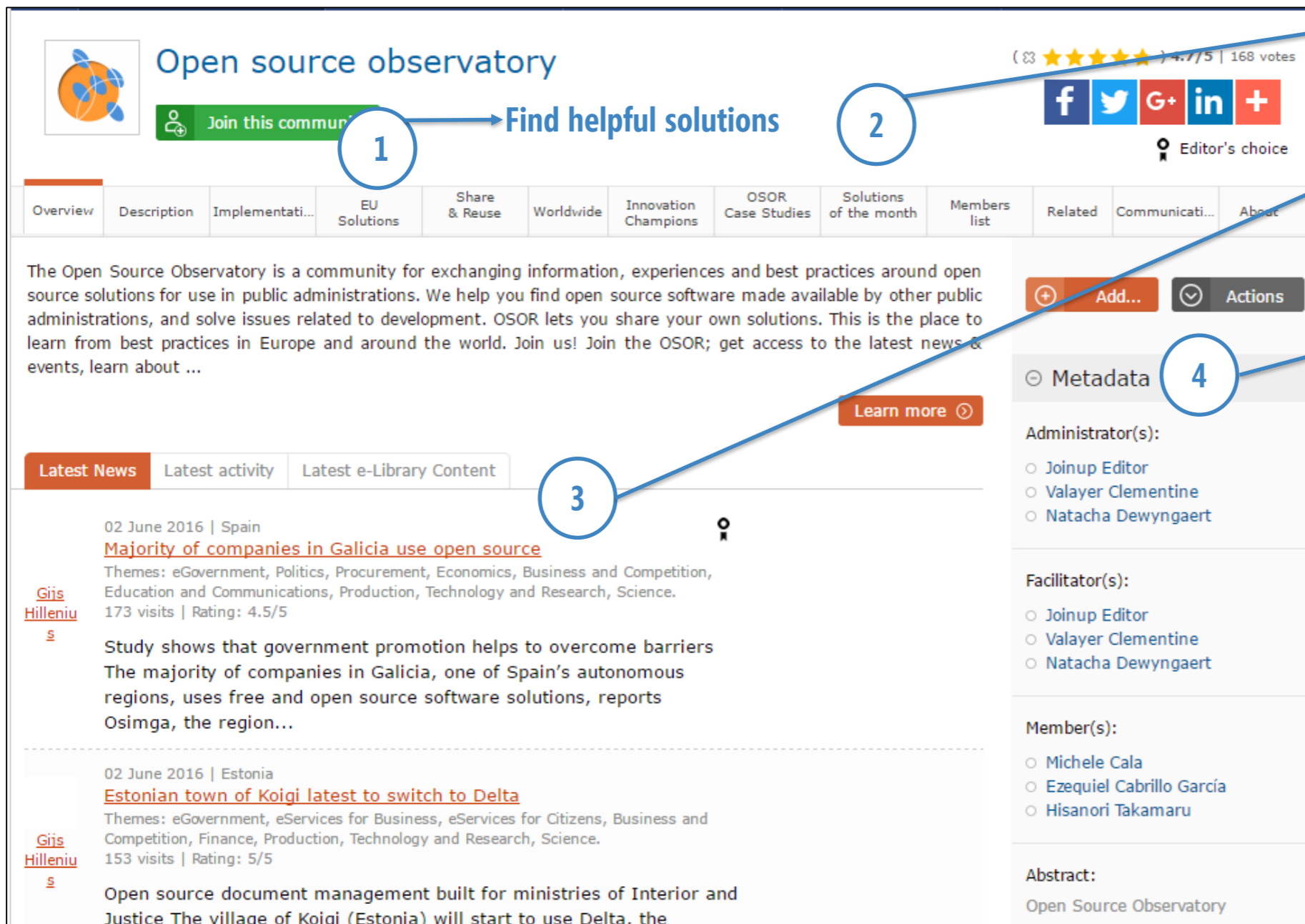


France (13)

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



The screenshot shows the OSOR page on Joinup. It features a header with the site logo and name, a navigation menu, a main content area with a description and news items, and a right-hand sidebar with metadata and contact information. Four numbered annotations are present: 1 points to the 'Join this community' button; 2 points to the star rating and social media icons; 3 points to the 'Latest News' tab; and 4 points to the 'Metadata' section in the sidebar.

Look through selected solutions

Check recent activities

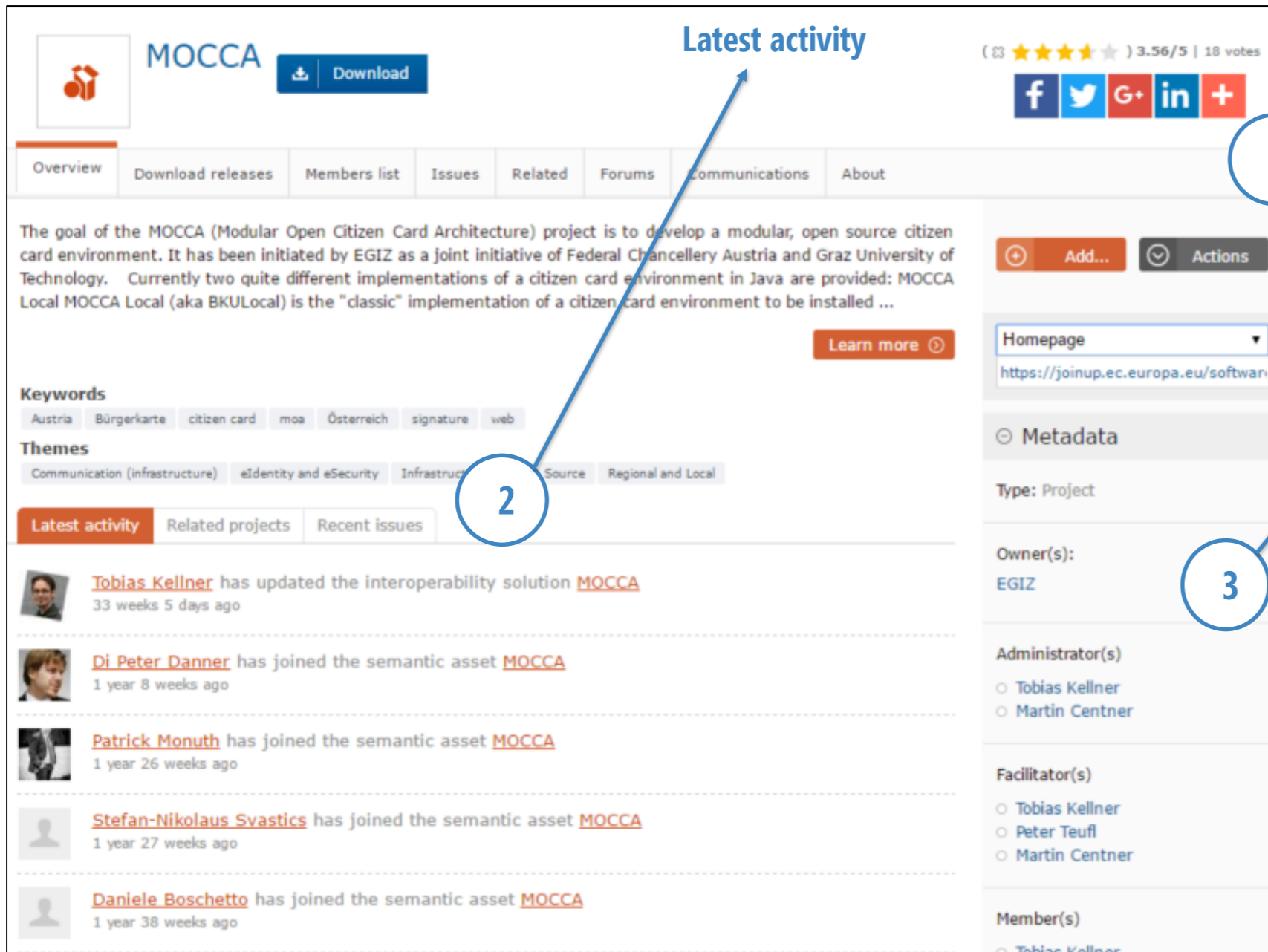
Contact

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)



The screenshot displays the MOCCA project page on the Joinup platform. The page includes a header with the MOCCA logo, a 'Download' button, and a 'Latest activity' section. The main content area features a description of the project, keywords, and themes. The 'Latest activity' section shows a list of recent updates and joins. The right sidebar contains a 'Metadata' section with details about the project's owner, administrators, and facilitators.

Latest activity

(3.56/5 | 18 votes)

Overview | Download releases | Members list | Issues | Related | Forums | Communications | About

The goal of the MOCCA (Modular Open Citizen Card Architecture) project is to develop a modular, open source citizen card environment. It has been initiated by EGIZ as a joint initiative of Federal Chancellery Austria and Graz University of Technology. Currently two quite different implementations of a citizen card environment in Java are provided: MOCCA Local MOCCA Local (aka BKULocal) is the "classic" implementation of a citizen card environment to be installed ...

Learn more

Keywords

Austria | Bürgerkarte | citizen card | moa | Österreich | signature | web

Themes

Communication (infrastructure) | eIdentity and eSecurity | Infrastructure | Source | Regional and Local

Latest activity | Related projects | Recent issues

Latest activity

- Tobias Kellner** has updated the interoperability solution **MOCCA** 33 weeks 5 days ago
- Di Peter Danner** has joined the semantic asset **MOCCA** 1 year 8 weeks ago
- Patrick Monuth** has joined the semantic asset **MOCCA** 1 year 26 weeks ago
- Stefan-Nikolaus Svastics** has joined the semantic asset **MOCCA** 1 year 27 weeks ago
- Daniele Boschetto** has joined the semantic asset **MOCCA** 1 year 38 weeks ago

Metadata

Type: Project

Owner(s): EGIZ

Administrator(s)

- Tobias Kellner
- Martin Centner

Facilitator(s)

- Tobias Kellner
- Peter Teufl
- Martin Centner

Member(s)

- Tobias Kellner

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

1

Contact

3

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

Advanced search

Content Users Issues

Filtering by:

Interoperability Solutions ✕
Projects ✕

[Clear filters](#)

Refine your search

Filter by type

- Interoperability Solutions
- Projects
- People (24655)
- e-Libraries (4436)
- News (3905)
- Events (1374)
- Communities (71)
- Repositories (34)

Filter by Solution category

Filter by Solution Current version

- Current Version (2720)

Filter by Solution Type

Filter by Repository of Origin

- W3C Standards and Technical Reports (330)
- Spain - Center for Technology Transfer (240)
- The Junta de Andalucía (156)
- XRepository (126)
- European Committee for

We care about your privacy. Tell us what you think [here](#).

Search results

The advanced search helps you navigate through content available on Joinup by (de)selecting the search filters on the left-hand side.

Keywords

Retain current filters

1 to 20 of 3075 results

Relevance ▾

Popularity ▲

Created ▲

Title ▲

Downloads ▲

Author ▲

EUSurvey 1.3.2

EUSurvey 1.3.2 has been published. You will find more information in the EUSurvey 1.3.2 OSS release note, available in the communications section of this project. ...

★★★★★

12 downloads

[Download](#)

Interoperability Solution | Created by [Margot Fassian](#) | Created: 08 August 2016 | Updated: 08 August 2016

Re3gistry 1.2

The new features included in version 1.2 are: New authentication method added - Apache SHIRO Addition of the new webapp. nthe web application reads the data from the JSON format produced ...

★★★★★

23 downloads

[Download](#)

Interoperability Solution | Created by [Re3Gistry_Are3Na](#) | Created: 18 July 2016 | Updated: 19 July 2016

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

1

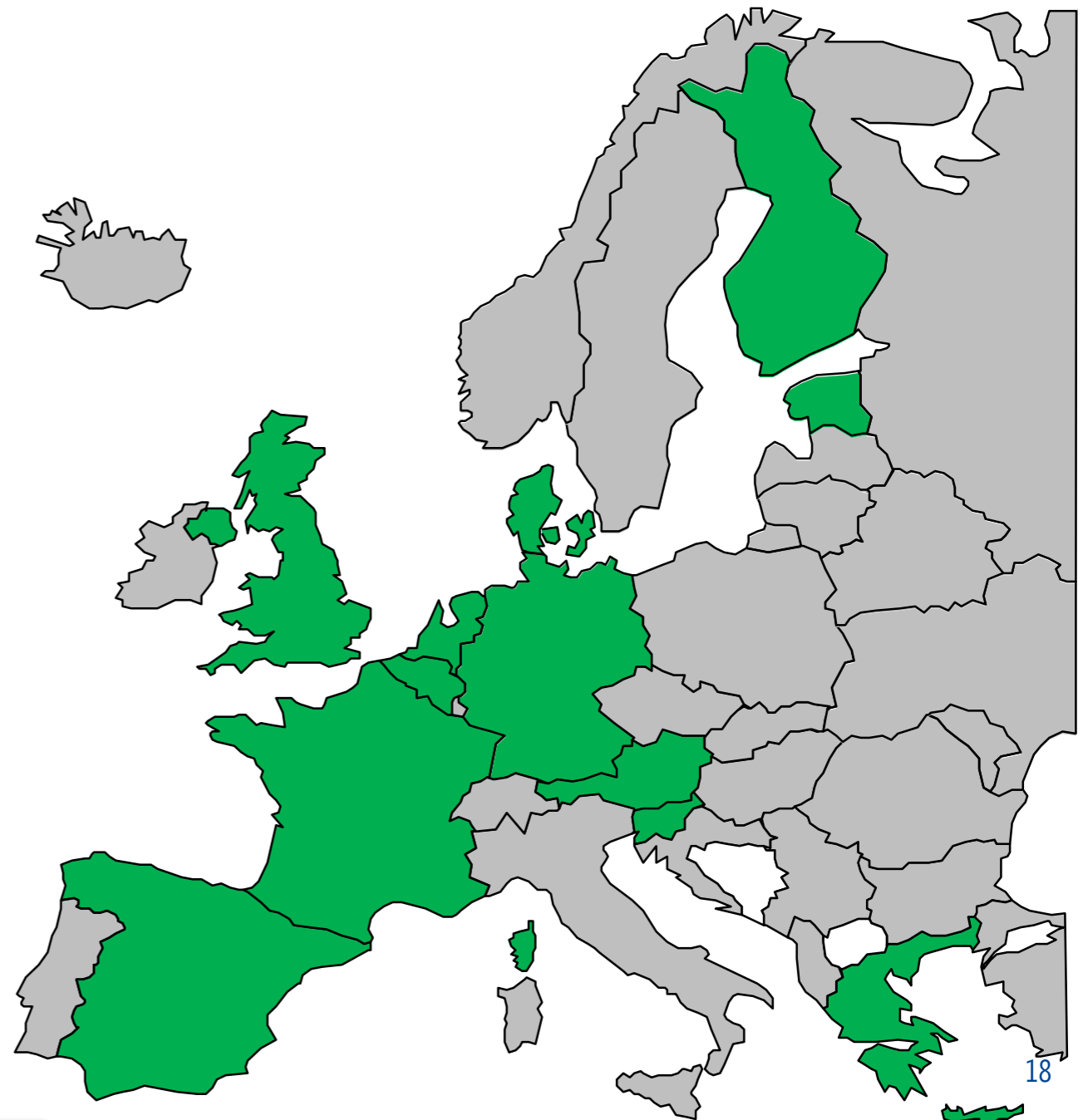
2

3

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

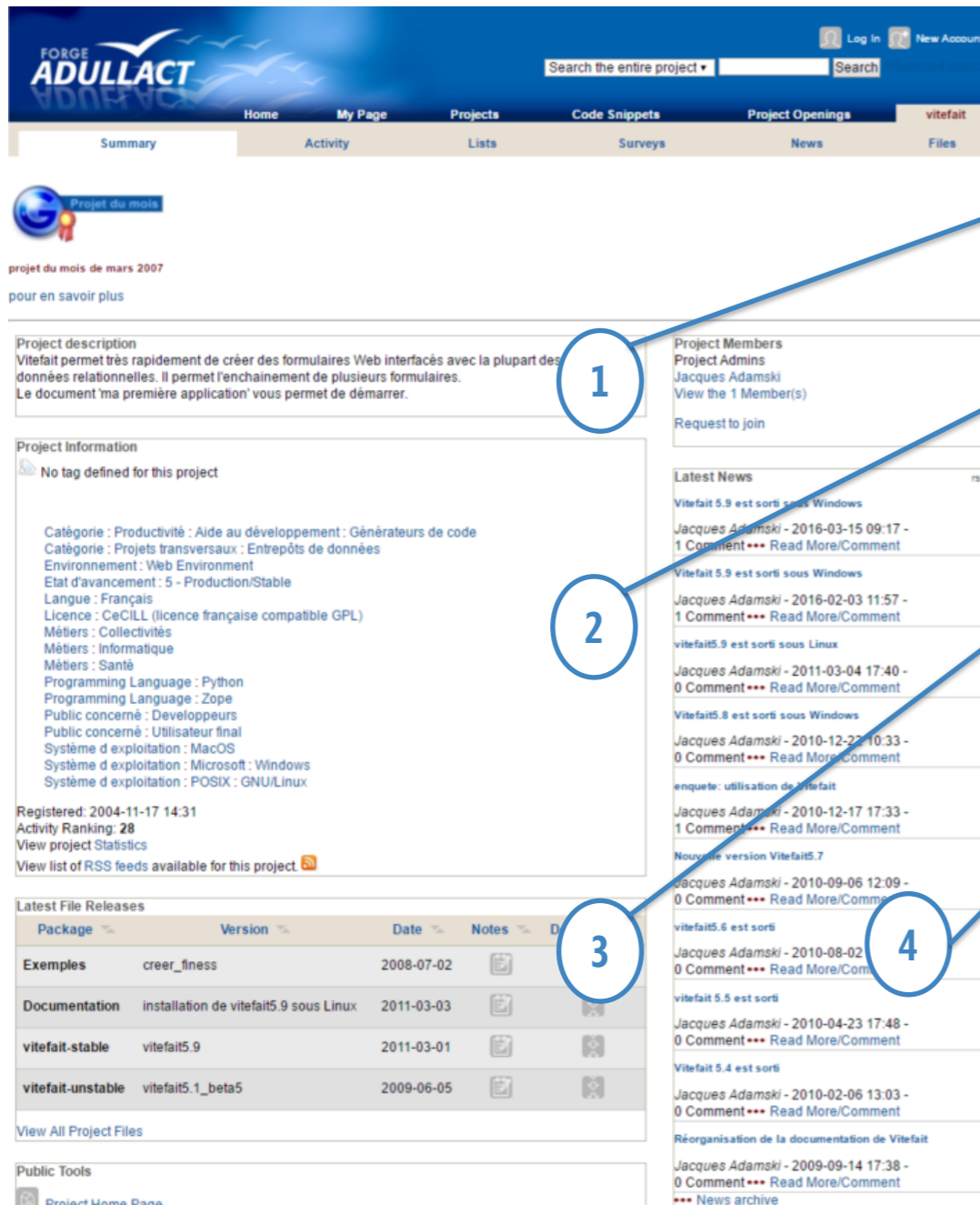
In practice: Joinup Federation

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



2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: National Registry Example: Adullact (France)



The screenshot shows the Adullact project page for 'Vitefait'. Annotations 1, 2, 3, and 4 point to specific sections of the page:

- 1**: Project description
- 2**: Project Information
- 3**: Latest File Releases table
- 4**: Latest News

Package	Version	Date	Notes	D
Exemples	creer_finess	2008-07-02		
Documentation	installation de vitefait5.9 sous Linux	2011-03-03		
vitefait-stable	vitefait5.9	2011-03-01		
vitefait-unstable	vitefait5.1_beta5	2009-06-05		

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)



Castellano | Català | Euskara | Galego | Valencià | English

Escuchar Identificarse Registrarse

Actualidad Estrategias Soluciones - CTT Observatorio - OBSAE Documentación Organización

Estás en: Inicio > Soluciones - CTT > Framework Abierto para la Administración Pública (openFWPA) > General

Soluciones - CTT

- ¿Qué es el CTT?
- Actualidad CTT
- Soluciones por área técnica
- Soluciones por área funcional
- Soluciones por área orgánica
- Soluciones por modo de uso
- Buscador de soluciones
- Buscador federado
- Forja CTT
- Registro de una solución
- Comunidades

Centro de Transferencia de Tecnología

Framework Abierto para la Administración Pública (openFWPA)

General Info. Adicional Área Descarga

Descripción | Noticias | Suscribirse

- Nombre Abreviado: openfwpa
- Resumen: openFWPA es un framework J2EE de desarrollo para administración electrónica desarrollado por el Principado de Asturias que permite crear potentes aplicaciones de eAdministración, unificando y simplificando su diseño, implementación y mantenimiento.
- Destinatarios: Ciudadano, Empresa, Cualquier Administración Pública
- Organismos Responsables: PRINCIPADO DE ASTURIAS
- Modo de Uso: Producto instalable
- Contacto: OPENFWPA@asturias.org
- Tipo de Solución: Aplicación de Fuentes Abiertas
- Estado de la Solución: Producción
- Área orgánica: Autonómico
- Área técnica: Tramitación electrónica, Soporte a la tramitación electrónica
- Área funcional: Administración

Responsable: GOBIERNO DEL PRINCIPADO DE ASTURIAS www.asturias.es

Enlaces Relacionados: Descarga del código fuente

Destacados CTT: Soporte CTT

- Cambios en infraestructura y servicios del CTT
- Nueva Guía Funcional para Oficinas de Registro SIR
- Actualización de la Sede-e de la UNED a la versión 3.0 de ACCEDA
- Competición europea de soluciones TIC reutilizables en la administración pública
- Aragón firma el Convenio de colaboración con el MINHAP para el suministro de ORVE para el acceso al Registro Electrónico Común y al SIR
- Publicación de la nueva versión distribuible de ARCHIVE

Downloads area

Project organisation

Categories

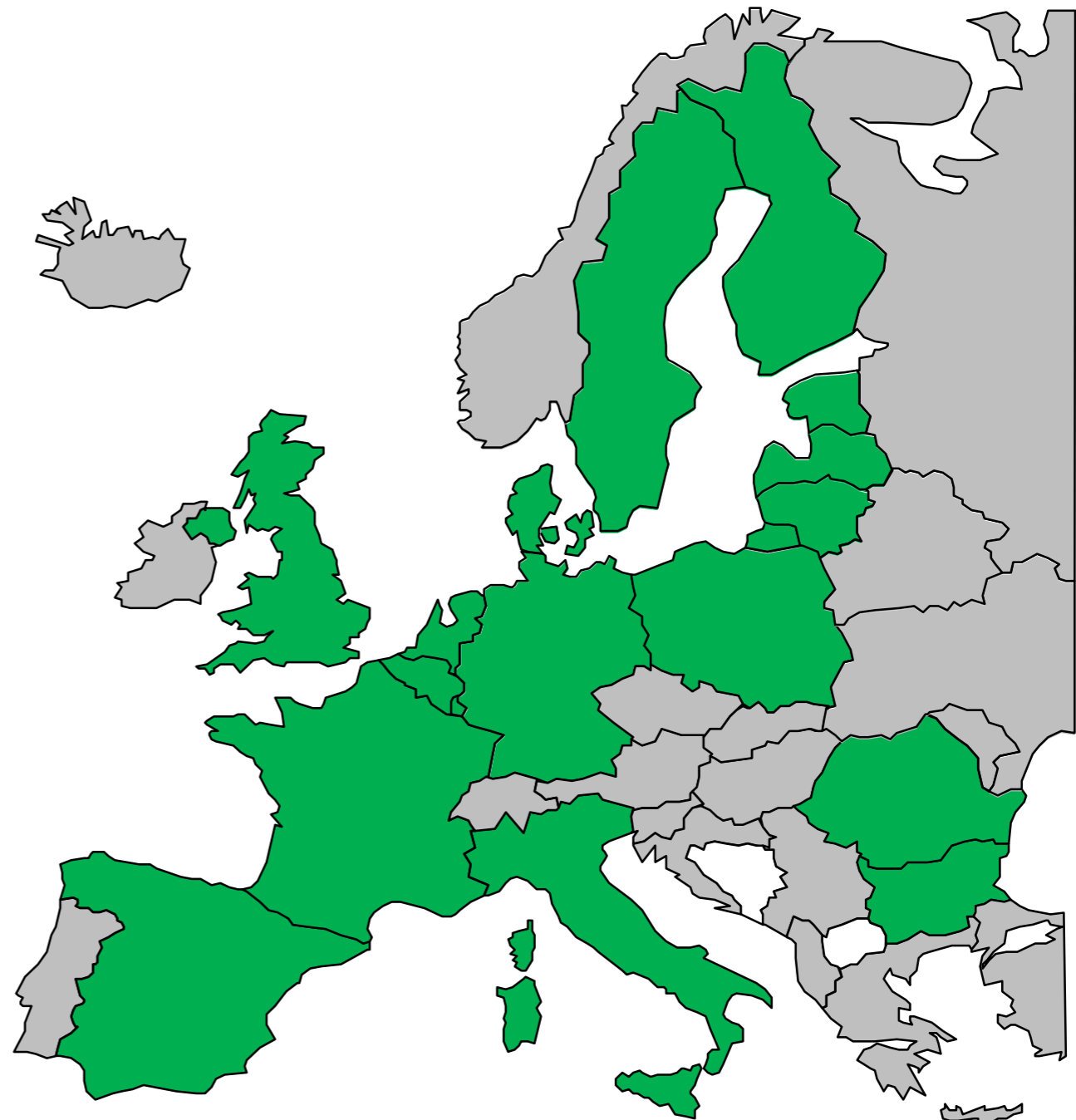
The Centro de Transferencia de Tecnología (Technology Transfer Centre) provides similar project information to Joinup, including:

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

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: GitHub Public Repositories

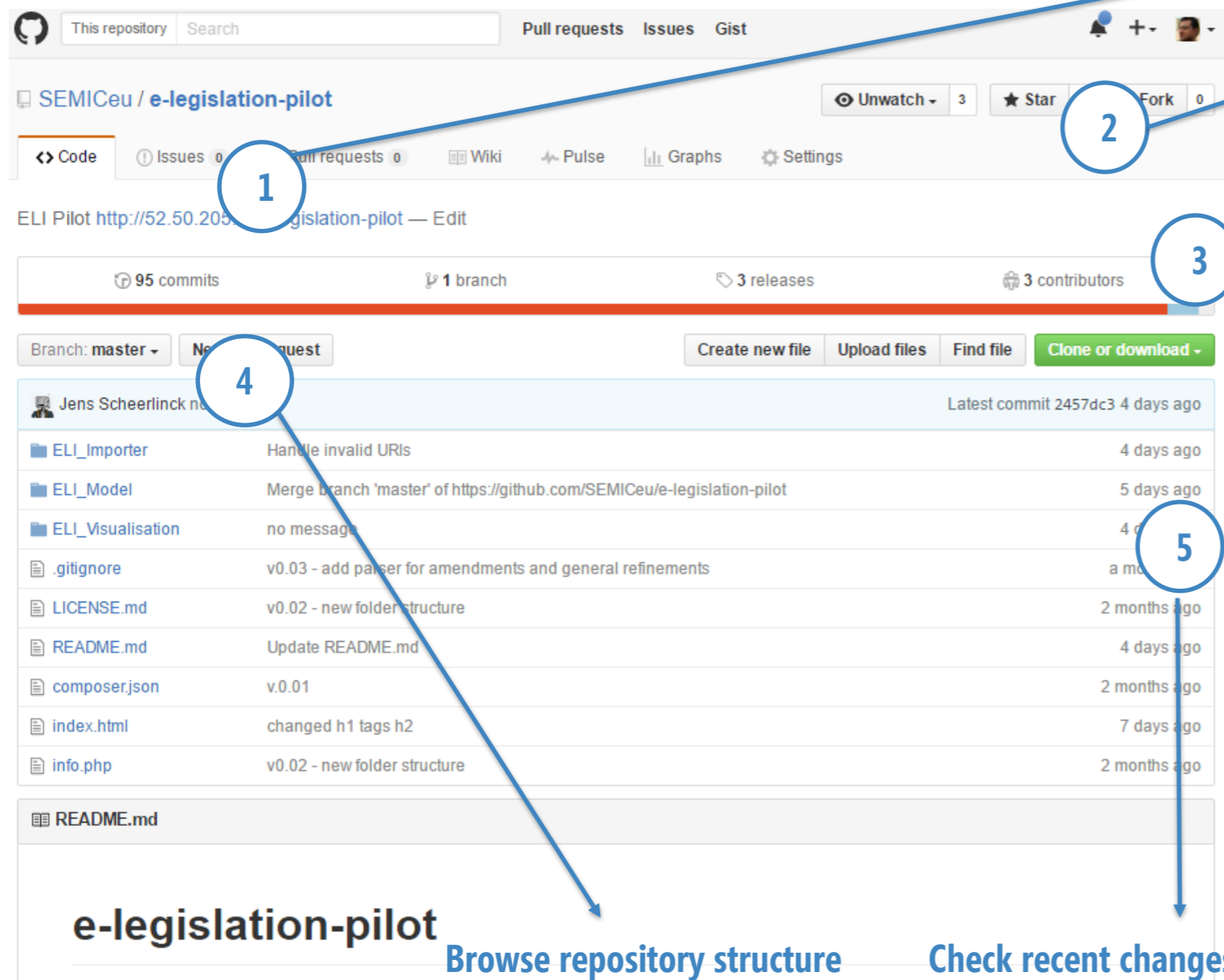
Country	Repositories
Belgium	5
Bulgaria	2
Denmark	1
Estonia	6
Finland	13
France	13
Germany	4
Italy	5
Latvia	1
Lithuania	2
Luxembourg	2
Netherlands	13
Poland	2
Romania	3
Spain	9
Sweden	12
U.K.	79



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

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: GitHub (repository at the centre)



SEMICeu / e-legislation-pilot

95 commits 1 branch 3 releases 3 contributors

File	Commit Message	Time
ELI_Importer	Handle invalid URIs	4 days ago
ELI_Model	Merge branch 'master' of https://github.com/SEMICeu/e-legislation-pilot	5 days ago
ELI_Visualisation	no message	4 days ago
.gitignore	v0.03 - add parser for amendments and general refinements	a month ago
LICENSE.md	v0.02 - new folder structure	2 months ago
README.md	Update README.md	4 days ago
composer.json	v.0.01	2 months ago
index.html	changed h1 tags h2	7 days ago
info.php	v0.02 - new folder structure	2 months ago

e-legislation-pilot

Browse repository structure

Check recent changes quickly

See people's interactions

See people's interests

Check project activities

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

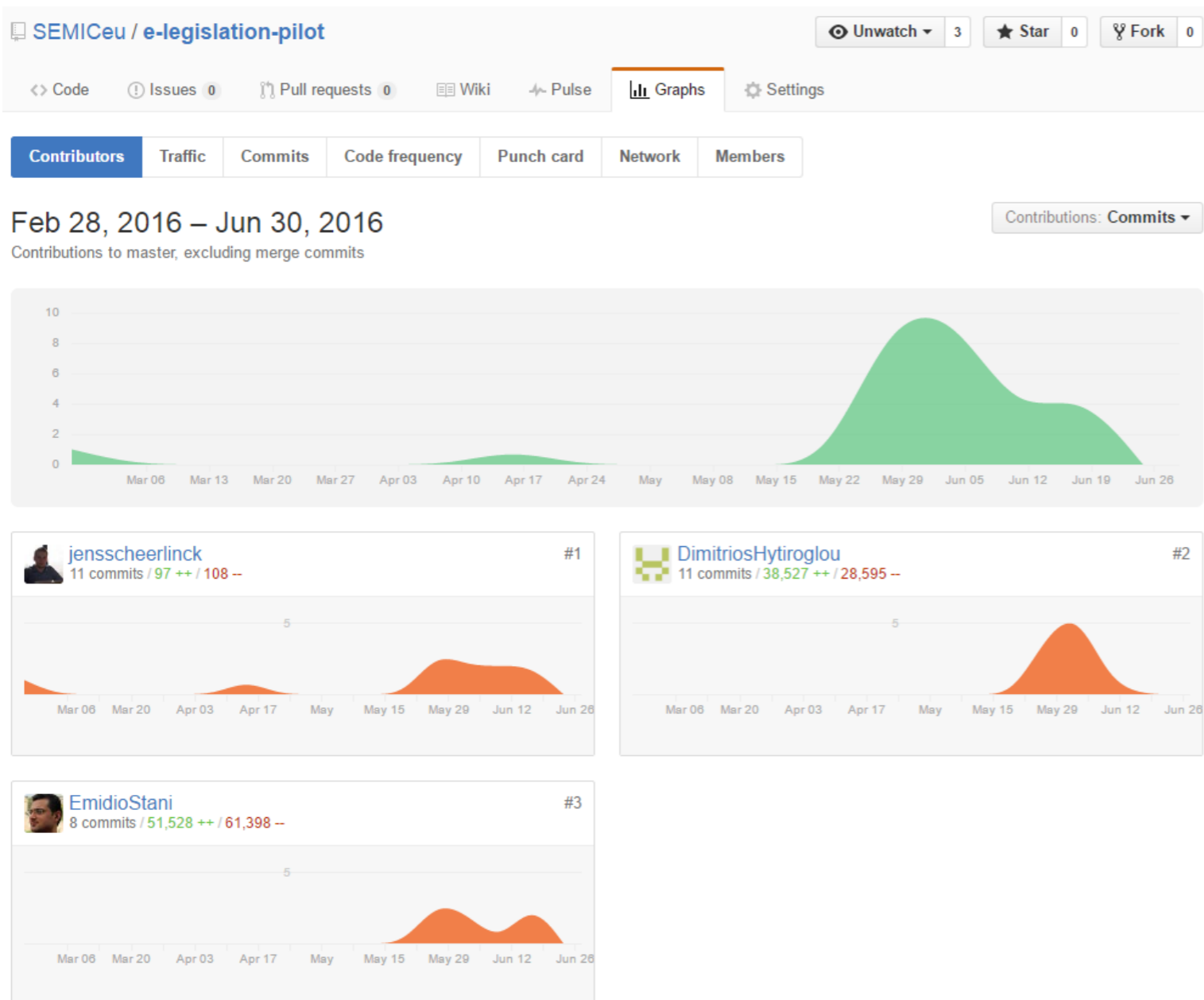
- Minimalistic project description
- No classification
- Minimalistic wiki

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)

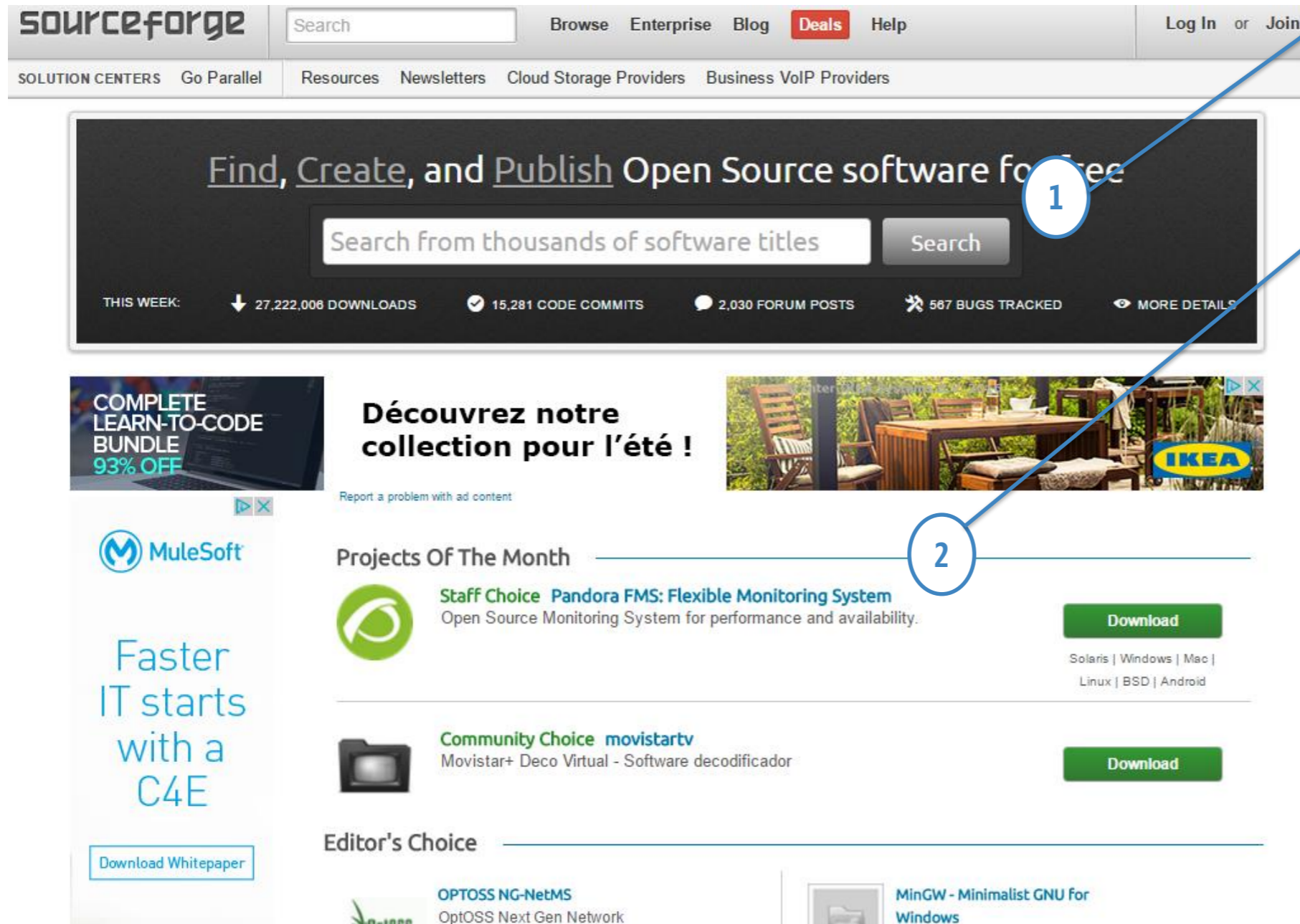


[GitHub](#) also provides a lot of statistics around a software:

- 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



The screenshot shows the Sourceforge website interface. At the top, there is a navigation bar with the Sourceforge logo, a search bar, and links for Browse, Enterprise, Blog, Deals, and Help. Below this is a secondary navigation bar with links for SOLUTION CENTERS, Go Parallel, Resources, Newsletters, Cloud Storage Providers, and Business VoIP Providers. The main content area features a large banner with the text "Find, Create, and Publish Open Source software for free" and a search bar. Below the banner, there are statistics for "THIS WEEK": 27,222,006 DOWNLOADS, 15,281 CODE COMMITS, 2,030 FORUM POSTS, and 567 BUGS TRACKED. The page is divided into several sections: "Découvrez notre collection pour l'été !" with an IKEA advertisement, "Projects Of The Month" featuring "Pandora FMS: Flexible Monitoring System" and "movistartv", and "Editor's Choice" featuring "OPTOSS NG-NetMS" and "MinGW - Minimalist GNU for Windows". Annotations include a blue circle with the number "1" pointing to the search bar in the banner, and a blue circle with the number "2" pointing to the "Projects Of The Month" section.

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).

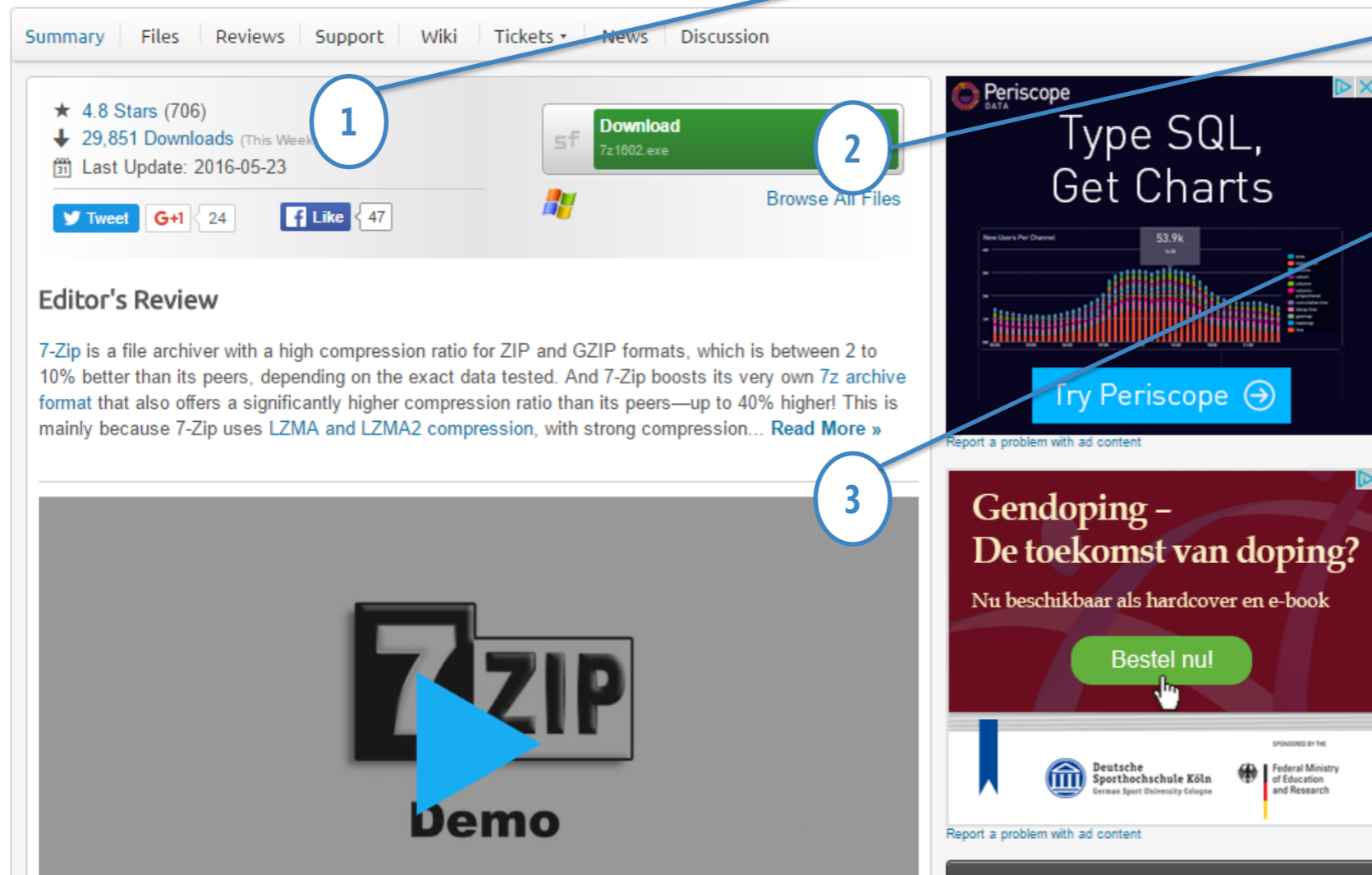
The Sourceforge business model is based on advertising.

2.1 USE COMMON REGISTRIES TO REUSE IT SOLUTIONS

In practice: Sourceforge Project

Home / Browse / System Administration / Storage / Archiving / Compression / 7-Zip

 **7-Zip**
A free file archiver for extremely high compression
Brought to you by: ipavlov



Summary | Files | Reviews | Support | Wiki | Tickets | News | Discussion


★ 4.8 Stars (706)
↓ 29,851 Downloads (This Week)
📅 Last Update: 2016-05-23

[Download](#)
7z1802.exe

[Browse All Files](#)

Editor's Review

7-Zip is a file archiver with a high compression ratio for ZIP and GZIP formats, which is between 2 to 10% better than its peers, depending on the exact data tested. And 7-Zip boosts its very own 7z archive format that also offers a significantly higher compression ratio than its peers—up to 40% higher! This is mainly because 7-Zip uses LZMA and LZMA2 compression, with strong compression... [Read More »](#)



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.



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

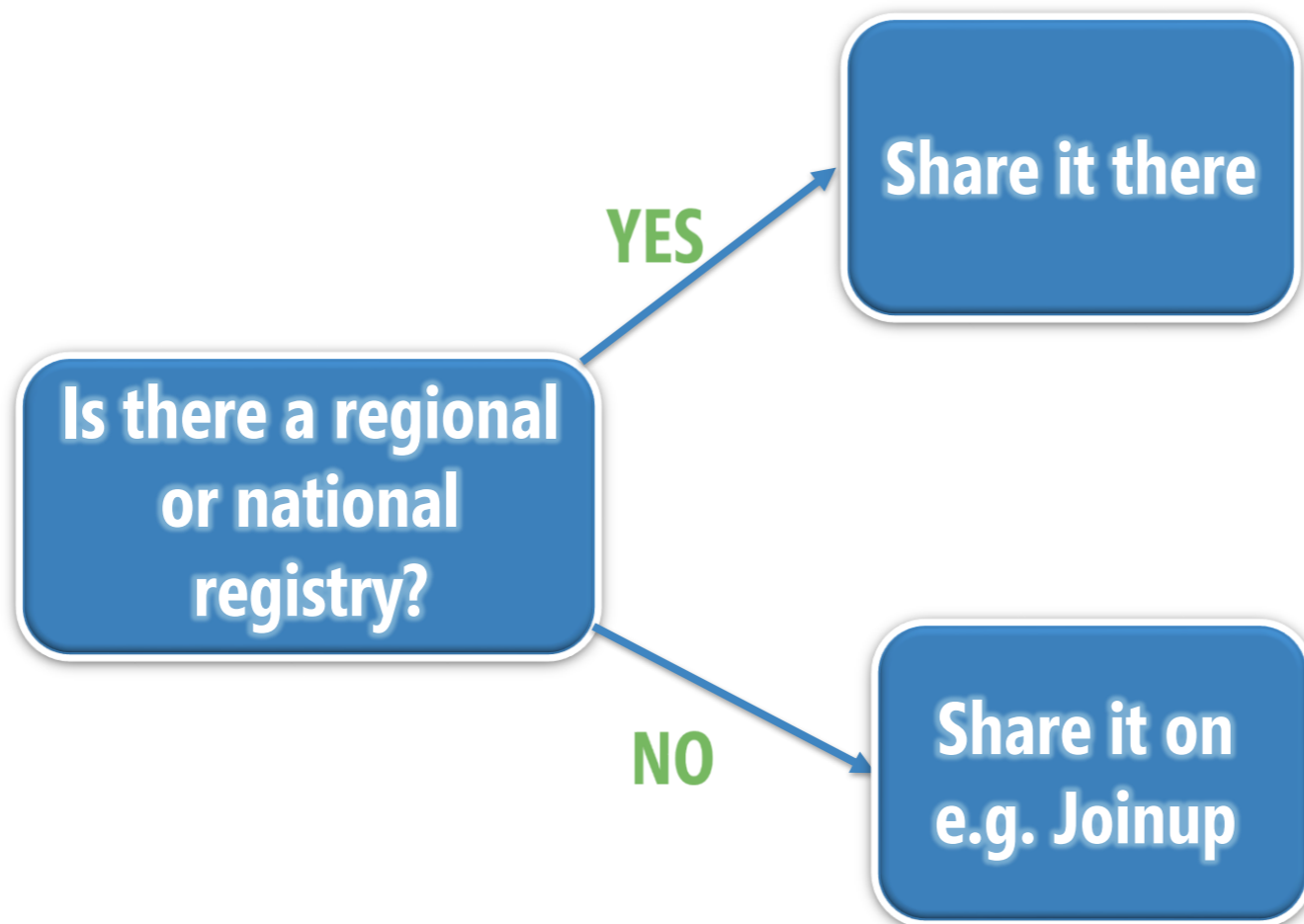
- A common registry is a system devoted to the proper description, structuring and publishing of IT solutions, together with the relevant documentation and process description in an integrated schema.
- Public administrations should publish their IT solutions on common registries instead of limiting 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.

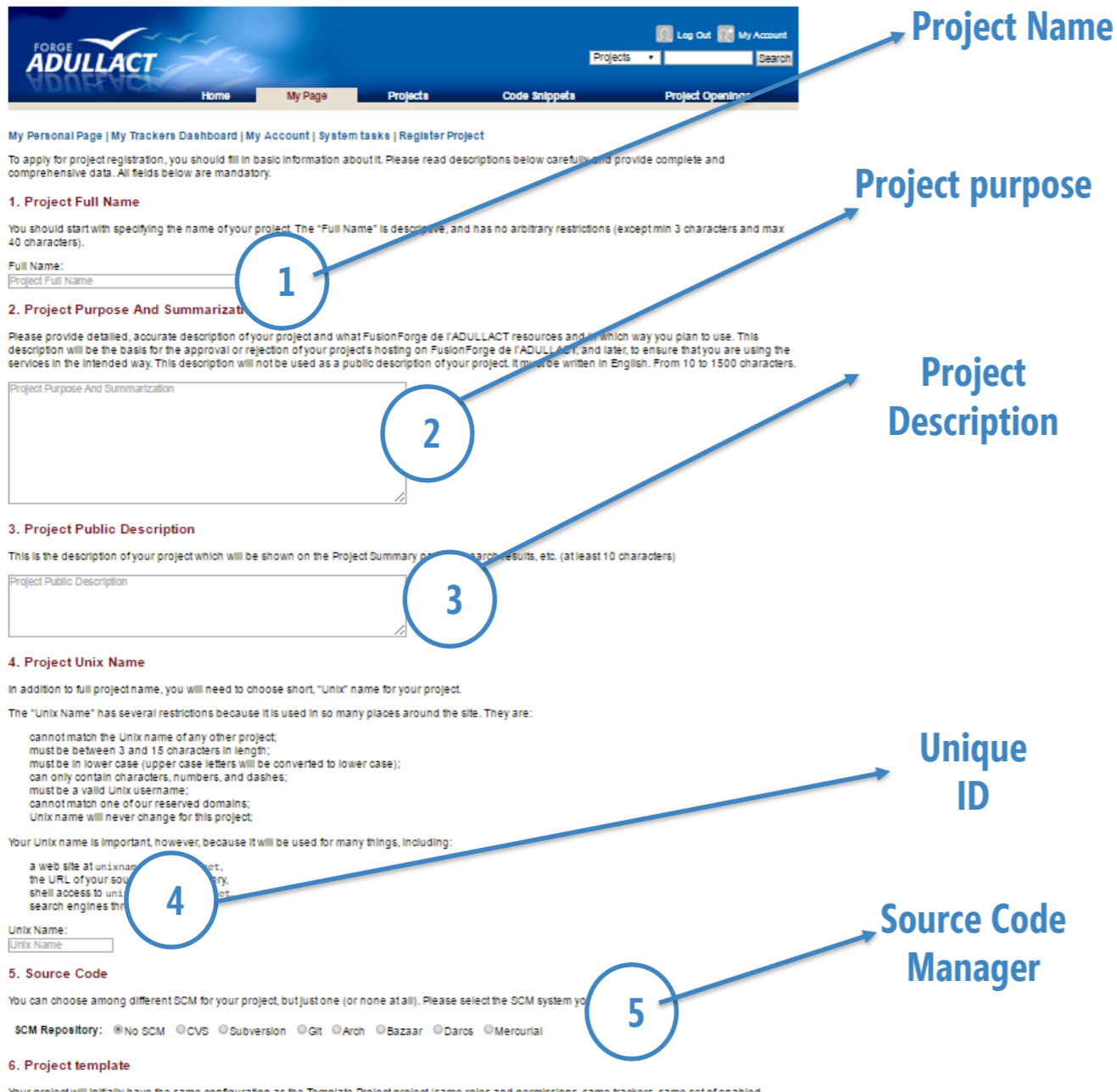


Example of a EU registry

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

2.2 USE COMMON REGISTRIES TO SHARE IT SOLUTIONS

In practice: National Registry Example: Adullact (France)



The screenshot shows the 'FORGE ADULLACT' project registration form. It is annotated with five numbered circles and arrows pointing to labels:

- 1** points to the 'Full Name' input field, labeled **Project Name**.
- 2** points to the 'Project Purpose And Summarization' text area, labeled **Project purpose**.
- 3** points to the 'Project Public Description' text area, labeled **Project Description**.
- 4** points to the 'Unix Name' input field, labeled **Unique ID**.
- 5** points to the 'SCM Repository' radio button options, labeled **Source Code Manager**.

The form includes the following sections:

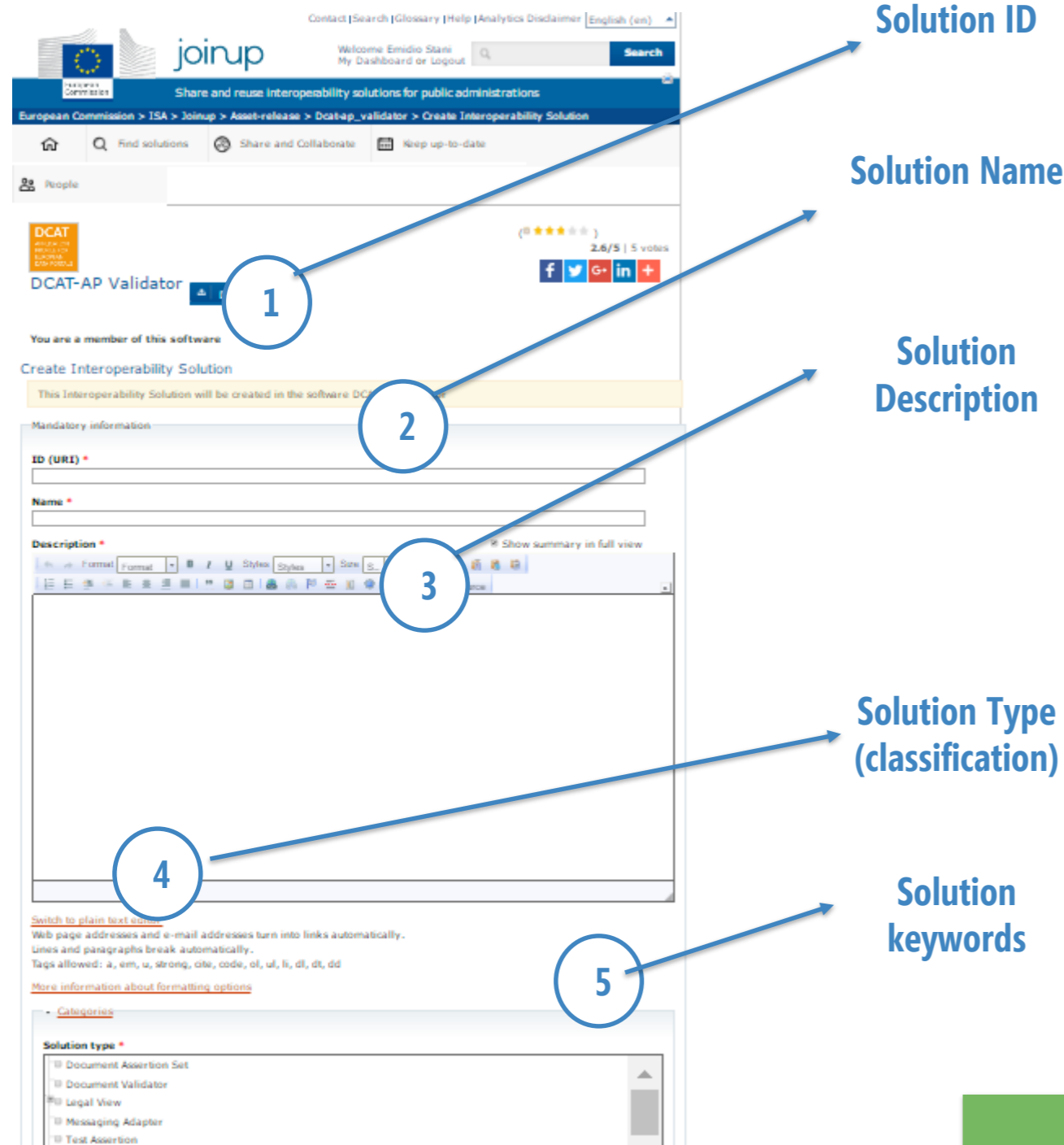
- 1. Project Full Name**: You should start with specifying the name of your project. The "Full Name" is descriptive, and has no arbitrary restrictions (except min 3 characters and max 40 characters).
- 2. Project Purpose And Summarization**: Please provide detailed, accurate description of your project and what FusionForge de l'ADULLACT resources and in which way you plan to use. This description will be the basis for the approval or rejection of your project's hosting on FusionForge de l'ADULLACT, and later, to ensure that you are using the services in the intended way. This description will not be used as a public description of your project. It must be written in English. From 10 to 1500 characters.
- 3. Project Public Description**: This is the description of your project which will be shown on the Project Summary page, search results, etc. (at least 10 characters)
- 4. Project Unix Name**: In addition to full project name, you will need to choose short, "Unix" name for your project. The "Unix Name" has several restrictions because it is used in so many places around the site. They are:
 - cannot match the Unix name of any other project;
 - must be between 3 and 15 characters in length;
 - must be in lower case (upper case letters will be converted to lower case);
 - can only contain characters, numbers, and dashes;
 - must be a valid Unix username;
 - cannot match one of our reserved domains;
 - Unix name will never change for this project.
 Your Unix name is important, however, because it will be used for many things, including:
 - a Web site at unixname.net,
 - the URL of your source code repository,
 - shell access to unixname.net,
 - search engines through unixname.net.
- 5. Source Code**: You can choose among different SCM for your project, but just one (or none at all). Please select the SCM system you want to use.
- 6. Project template**: Your project will initially have the same configuration as the Template Project (same roles and permissions, same trackers, same cat of artifacts)

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



The screenshot shows the 'Create Interoperability Solution' form in the Joinup platform. The form is titled 'DCAT-AP Validator' and includes a 'Mandatory information' section with the following fields:

- 1**: A blue circle highlights the 'DCAT-AP Validator' title, which is linked to the 'Solution ID' label.
- 2**: A blue circle highlights the 'ID (URI)' field, which is linked to the 'Solution Name' label.
- 3**: A blue circle highlights the 'Description' field, which is linked to the 'Solution Description' label.
- 4**: A blue circle highlights the 'Solution type' dropdown menu, which is linked to the 'Solution Type (classification)' label.
- 5**: A blue circle highlights the 'Keywords' field, which is linked to the 'Solution keywords' label.

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.



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

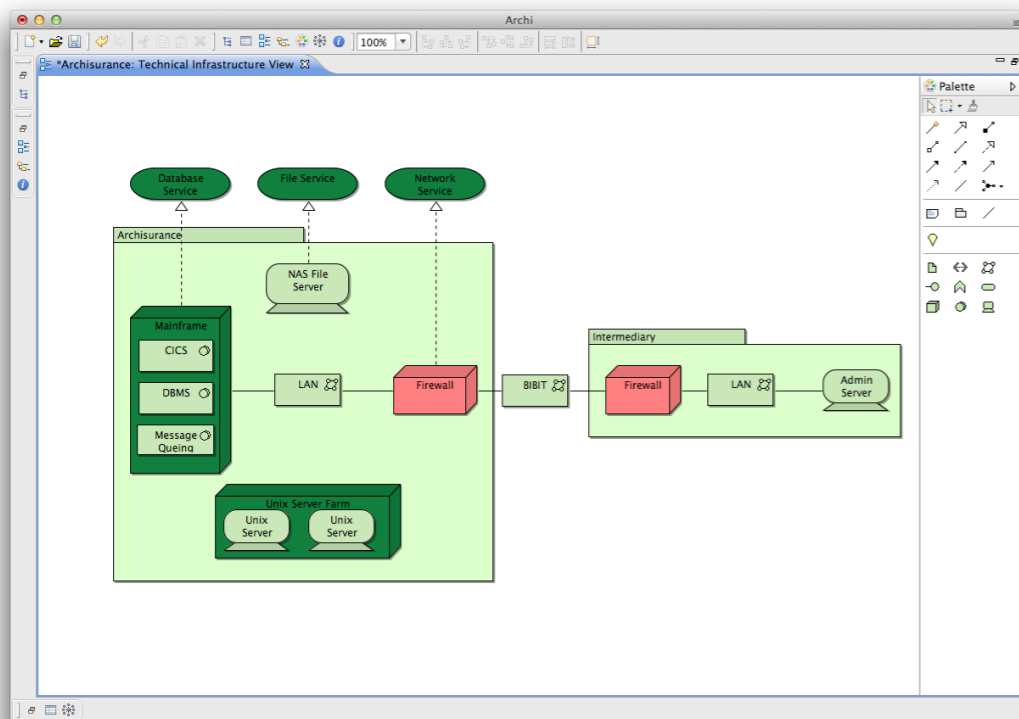
- By describing the architecture of an IT solution, the use of **UML** diagrams makes it easy to understand the structure of its components.
- Representing a data model in **XML** format will support easy transformation to other formats.

3. USE STANDARD WAYS TO DESCRIBE YOUR IT SOLUTIONS

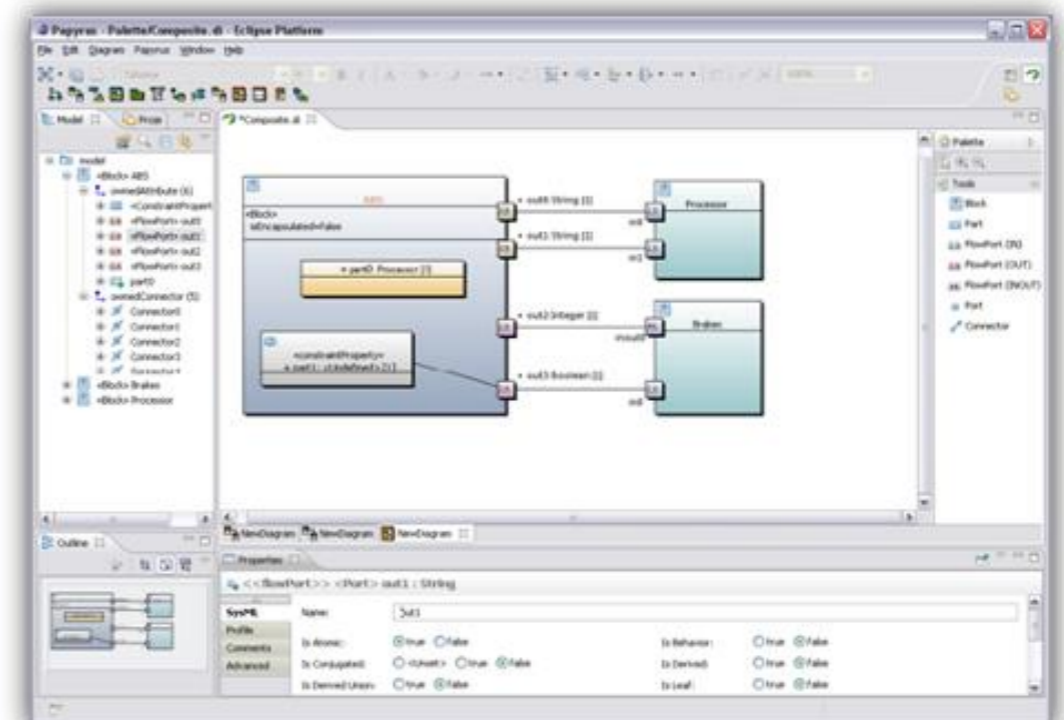
In practice

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.

ADMS
ASSET
DESCRIPTION
METADATA
SCHEMA

- 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.

DCAT-AP
FOR
DATA PORTALS
IN EUROPE

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





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

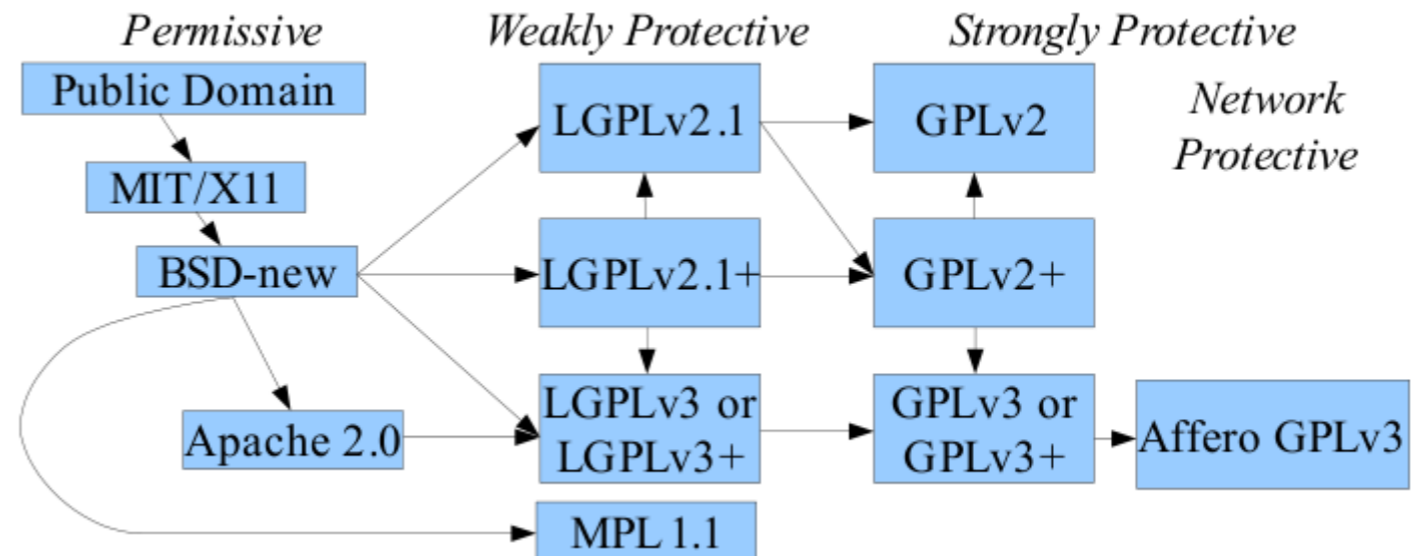
- Insufficient clarity about liability exposure and possible infringement of property rights make public administrations reluctant to take a collaborative approach **to developing IT solutions**.
- Issues of incompatibility between licences, which can result from public administrations writing their 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.



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 different 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

- Commercial Use
- Distribution
- Modification
- Private Use

Conditions

- License and Copyright Notice

Limitations

- Hold Liable

MIT License

Copyright (c) [year] [fullname]

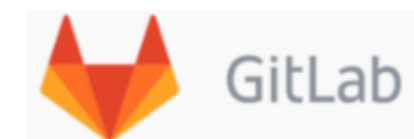
Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or 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:



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

- Commercial Use
- Distribution
- Modification
- Patent Use
- Private Use

Conditions

- License and Copyright Notice
- State Changes

Limitations

- Hold Liable
- Trademark Use

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

TERMS AND CONDITIONS FOR USE, REPRODUCTION, AND DISTRIBUTION

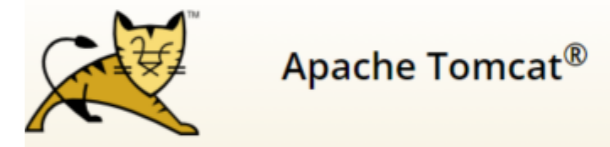
1. Definitions.

"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:



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

- Commercial use
- Distribution
- Modification
- Patent use
- Private use

Conditions

- Disclose source
- License and copyright notice
- Same license
- State changes

Limitations

- Liability
- Warranty

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:



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

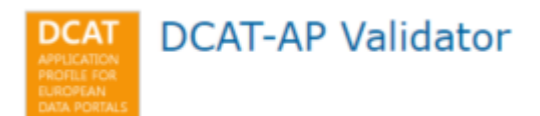
The Licensor hereby grants You a worldwide, royalty-free, non-exclusive, sublicensable licence to do the following, for the duration of copyright vested in the Original Work:

- **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:



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.

Choose an open source license

Which of the following best describes your situation?



I want it simple and permissive.

The **MIT License** is a permissive license that is short and to the point. It lets people do anything they want with your code as long as they provide attribution back to you and don't hold you liable.

jQuery, **.NET Core**, and **Rails** use the MIT License.



I'm concerned about patents.

The **Apache License 2.0** is a permissive license similar to the MIT License, but also provides an express grant of patent rights from contributors to users.

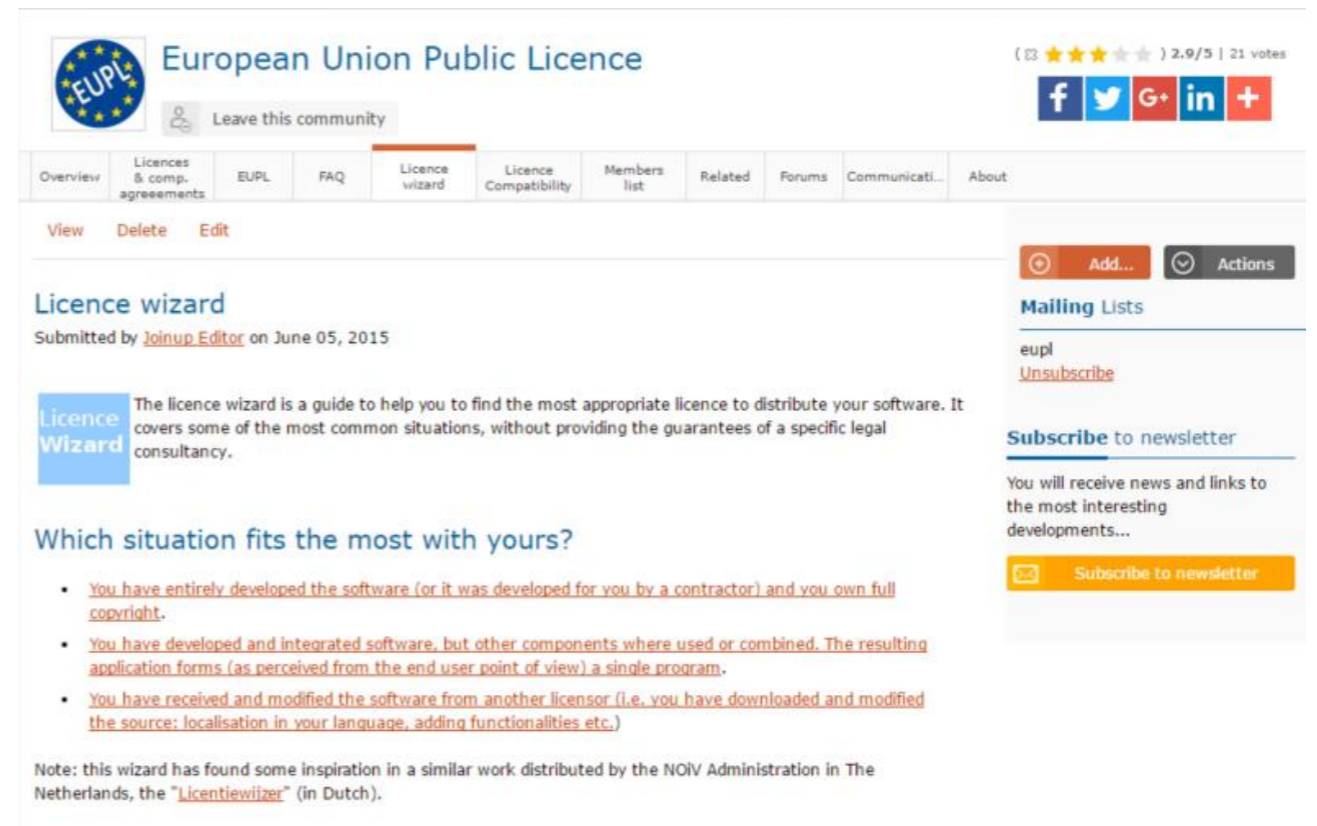
Android, **Apache**, and **Swift** use the Apache License 2.0.



I care about sharing improvements.

The **GNU GPLv3** is a copyleft license that requires anyone who distributes your code or a derivative work to make the source available under the same terms, and also provides an express grant of patent rights from contributors to users.

Bash, **GIMP**, and **Privacy Badger** use the GNU GPLv3.



The screenshot shows the EUPL website interface. At the top, there's the EUPL logo and the text 'European Union Public Licence'. Below that, there are social media icons (Facebook, Twitter, Google+, LinkedIn) and a 'Leave this community' button. A navigation menu includes 'Overview', 'Licences & comp. agreements', 'EUPL', 'FAQ', 'Licence wizard', 'Licence Compatibility', 'Members list', 'Related', 'Forums', 'Communicati...', and 'About'. The 'Licence wizard' section is active, showing a 'Submitted by Joinup Editor on June 05, 2015' and a description: 'The licence wizard is a guide to help you to find the most appropriate licence to distribute your software. It covers some of the most common situations, without providing the guarantees of a specific legal consultancy.' Below this, there's a question 'Which situation fits the most with yours?' followed by three bullet points describing different software development scenarios. At the bottom, there's a note: 'Note: this wizard has found some inspiration in a similar work distributed by the NOIV Administration in The Netherlands, the "Licentiewizer" (in Dutch).' On the right side, there are 'Add...' and 'Actions' buttons, a 'Mailing Lists' section with 'eupl' and 'Unsubscribe' links, and a 'Subscribe to newsletter' button.

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.

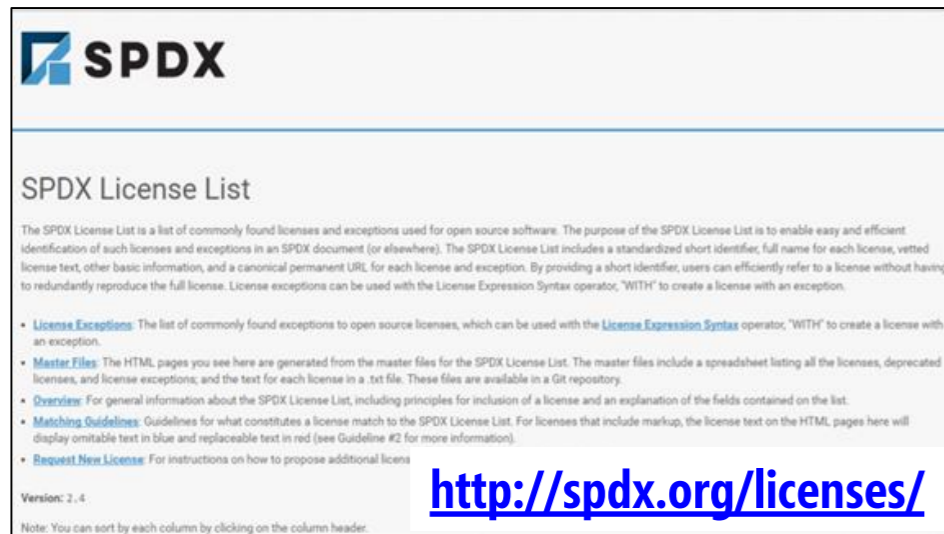
You should have received a copy of the GNU General Public License along with this program; if not, write to the Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301, USA.

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 ?



SPDX

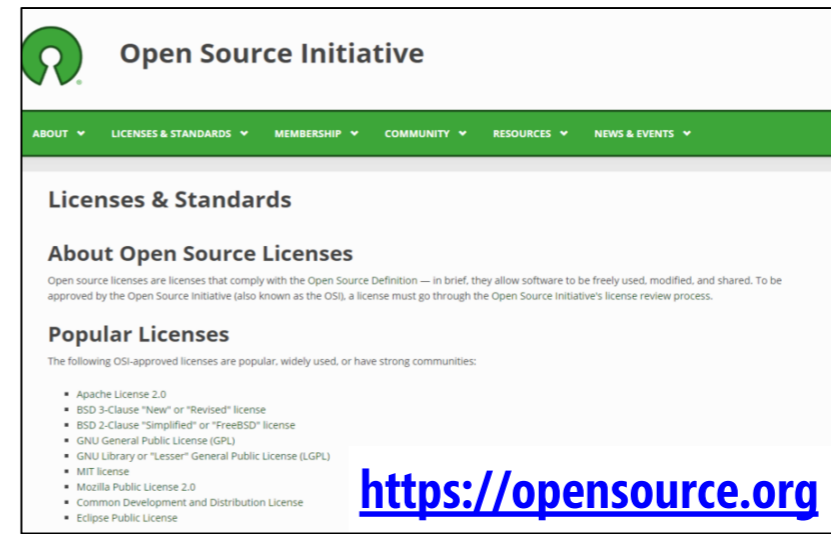
SPDX License List

The SPDX License List is a list of commonly found licenses and exceptions used for open source software. The purpose of the SPDX License List is to enable easy and efficient identification of such licenses and exceptions in an SPDX document (or elsewhere). The SPDX License List includes a standardized short identifier, full name for each license, vetted license text, other basic information, and a canonical permanent URL for each license and exception. By providing a short identifier, users can efficiently refer to a license without having to redundantly reproduce the full license. License exceptions can be used with the License Expression Syntax operator, 'WITH' to create a license with an exception.

- License Exceptions:** The list of commonly found exceptions to open source licenses, which can be used with the [License Expression Syntax](#) operator, 'WITH' to create a license with an exception.
- Master Files:** The HTML pages you see here are generated from the master files for the SPDX License List. The master files include a spreadsheet listing all the licenses, deprecated licenses, and license exceptions; and the text for each license in a .txt file. These files are available in a Git repository.
- Overviews:** For general information about the SPDX License List, including principles for inclusion of a license and an explanation of the fields contained on the list.
- Matching Guidelines:** Guidelines for what constitutes a license match to the SPDX License List. For licenses that include markup, the license text on the HTML pages here will display omissible text in blue and replaceable text in red (see Guideline #2 for more information).
- Request New License:** For instructions on how to propose additional licenses

Version: 2.4

<http://spdx.org/licenses/>



Open Source Initiative

ABOUT | LICENSES & STANDARDS | MEMBERSHIP | COMMUNITY | RESOURCES | NEWS & EVENTS

Licenses & Standards

About Open Source Licenses

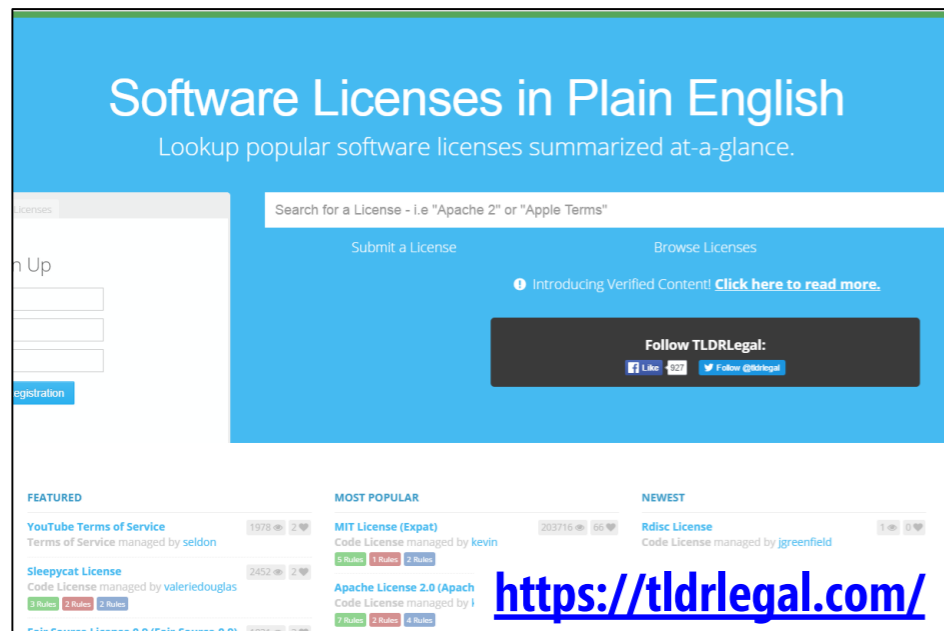
Open source licenses are licenses that comply with the Open Source Definition — in brief, they allow software to be freely used, modified, and shared. To be approved by the Open Source Initiative (also known as the OSI), a license must go through the Open Source Initiative's license review process.

Popular Licenses

The following OSI-approved licenses are popular, widely used, or have strong communities:

- Apache License 2.0
- BSD 3-Clause "New" or "Revised" license
- BSD 2-Clause "Simplified" or "FreeBSD" license
- GNU General Public License (GPL)
- GNU Library or "Lesser" General Public License (LGPL)
- MIT license
- Mozilla Public License 2.0
- Common Development and Distribution License
- Eclipse Public License

<https://opensource.org>



Software Licenses in Plain English

Lookup popular software licenses summarized at-a-glance.

Search for a License - i.e "Apache 2" or "Apple Terms"

Submit a License | Browse Licenses

Introducing Verified Content! [Click here to read more.](#)

Follow TLDRLegal:

FEATURED

- YouTube Terms of Service** (1978 likes) Terms of Service managed by seldon
- Sleepycat License** (2452 likes) Code License managed by valeriedouglas
- Fair Source License 0.9 (Fair-Source-0.9)** (1951 likes)

MOST POPULAR

- MIT License (Expat)** (203716 likes) Code License managed by kevin
- Apache License 2.0 (Apach)** (7 Rules, 2 Rules, 8 Rules) Code License managed by I

NEWEST

- Rdisc License** (1 like) Code License managed by jgreenfield

<https://tldrlegal.com/>



FSF 30 YEARS FREE SOFTWARE FOUNDATION Log in Help! Members forum JOIN NOW

about campaigns licensing membership resources community donate shop

FSF Licensing & Compliance Team

by Joshua Gay

For over 20 years the FSF Licensing & Compliance Lab has been the preeminent resource of free licensing for free software developers.

Education & Support

We have a number of online resources as well as community-based and paid support.

See our licensing recommendations, analysis, and FAQ

- Guide to choosing a license for your own work
- Comprehensive FAQ about the GNU Licenses
- List of other licenses and whether they are free, copyleft, or compatible with the GPL
- A Quick Guide to GPLv3

Learn why we must put an end to software patents:

- Visit EndSoftPatents.org
- Get Involved on the LibrePlanet.org End Software Patents Group
- Join the action alert mailing list

Keep up to date with the latest licensing and Compliance blog, or subscribe to the FSF feed

Sign up: Enter your email address to receive our monthly newsletter, the Free Software Supporter. [Subscribe me](#)

News

<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.





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.

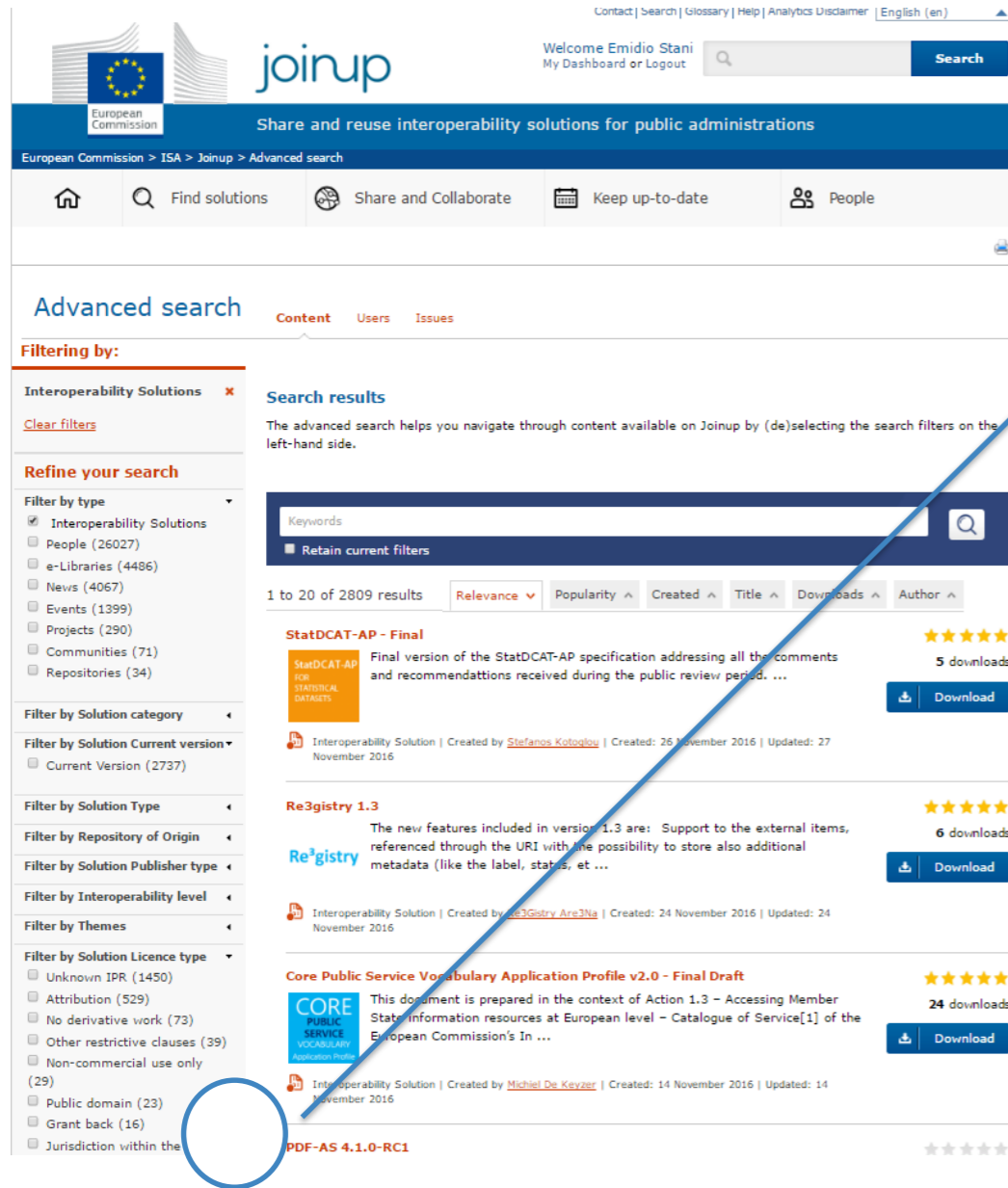
4.2 USE EXISTING OPEN SOURCE SOFTWARE LICENCES FOR REUSE

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



The screenshot shows the Joinup website interface. At the top, there is a navigation bar with the Joinup logo and the text "Share and reuse interoperability solutions for public administrations". Below this is a search bar and a "Search" button. The main content area is titled "Advanced search" and includes a "Filtering by:" section on the left. This section has several filters, including "Interoperability Solutions", "Refine your search", "Filter by type", "Filter by Solution category", "Filter by Solution Current version", "Filter by Solution Type", "Filter by Repository of Origin", "Filter by Solution Publisher type", "Filter by Interoperability level", "Filter by Themes", and "Filter by Solution Licence type". The "Filter by Solution Licence type" filter is circled in blue. The main search results area shows a list of solutions, including "StatDCAT-AP - Final", "Re3gistry 1.3", "Core Public Service Vocabulary Application Profile v2.0 - Final Draft", and "PDF-AS 4.1.0-RC1". Each result includes a title, a brief description, a star rating, and a "Download" button.

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

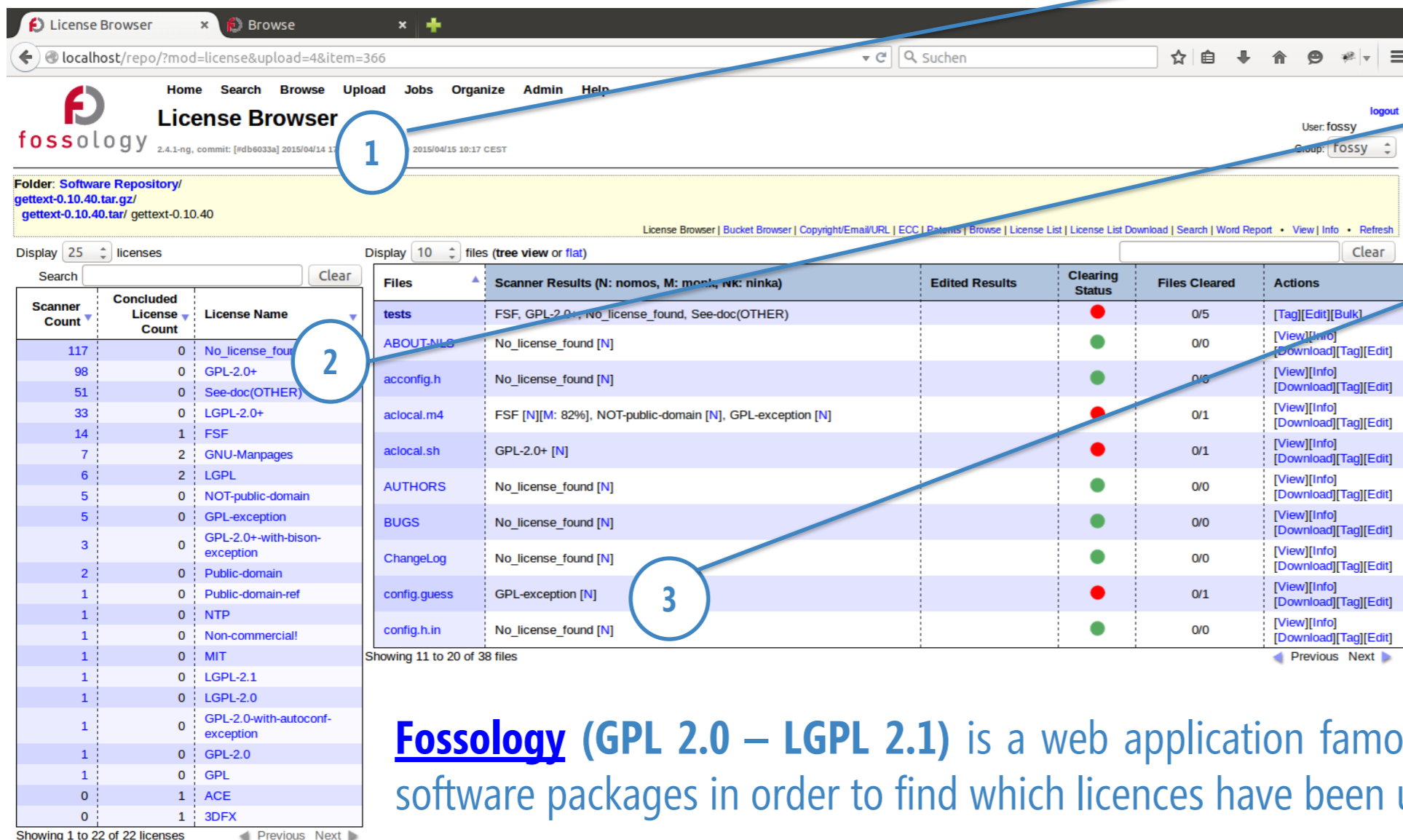
In practice: which tools can detect the licences of a project ?

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.



1

2

3

Scanner Count	Concluded License Count	License Name
117	0	No_license_found
98	0	GPL-2.0+
51	0	See-doc(OTHER)
33	0	LGPL-2.0+
14	1	FSF
7	2	GNU-Manpages
6	2	LGPL
5	0	NOT-public-domain
5	0	GPL-exception
3	0	GPL-2.0+-with-bison-exception
2	0	Public-domain
1	0	Public-domain-ref
1	0	NTP
1	0	Non-commercial!
1	0	MIT
1	0	LGPL-2.1
1	0	LGPL-2.0
1	0	GPL-2.0-with-autoconf-exception
1	0	GPL-2.0
1	0	GPL
0	1	ACE
0	1	3DFX

Files	Scanner Results (N: nomos, M: monk, nk: ninka)	Edited Results	Clearing Status	Files Cleared	Actions
tests	FSF, GPL-2.0+, No_license_found, See-doc(OTHER)		●	0/5	[Tag][Edit][Bulk]
ABOUT-NLS	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]
acconfig.h	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]
aclocal.m4	FSF [N][M: 82%], NOT-public-domain [N], GPL-exception [N]		●	0/1	[View][Info][Download][Tag][Edit]
aclocal.sh	GPL-2.0+ [N]		●	0/1	[View][Info][Download][Tag][Edit]
AUTHORS	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]
BUGS	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]
ChangeLog	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]
config.guess	GPL-exception [N]		●	0/1	[View][Info][Download][Tag][Edit]
config.h.in	No_license_found [N]		●	0/0	[View][Info][Download][Tag][Edit]

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 ?

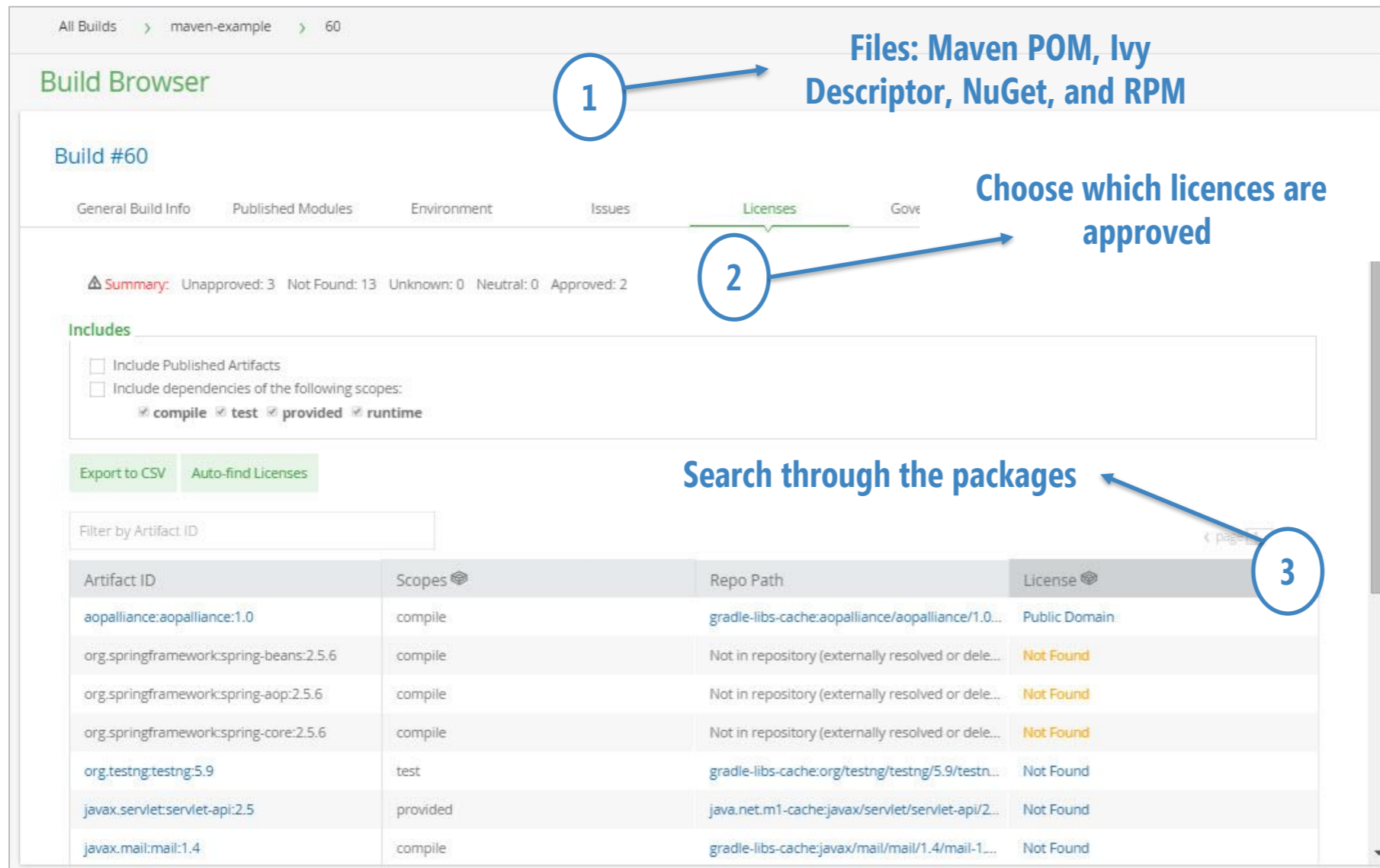
```
└─ cli@0.4.3
  └─ repository: http://github.com/chriso/cli
  └─ licenses: MIT
└─ glob@3.1.14
  └─ repository: https://github.com/isaacs/node-glob
  └─ licenses: UNKNOWN
└─ graceful-fs@1.1.14
  └─ repository: https://github.com/isaacs/node-graceful-fs
  └─ licenses: UNKNOWN
└─ inherits@1.0.0
  └─ 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 ?



Files: Maven POM, Ivy Descriptor, NuGet, and RPM

Build #60

General Build Info | Published Modules | Environment | Issues | Licenses | Gove

Summary: Unapproved: 3 | Not Found: 13 | Unknown: 0 | Neutral: 0 | Approved: 2

Includes

Include Published Artifacts

Include dependencies of the following scopes:

compile test provided runtime

Export to CSV | Auto-find Licenses

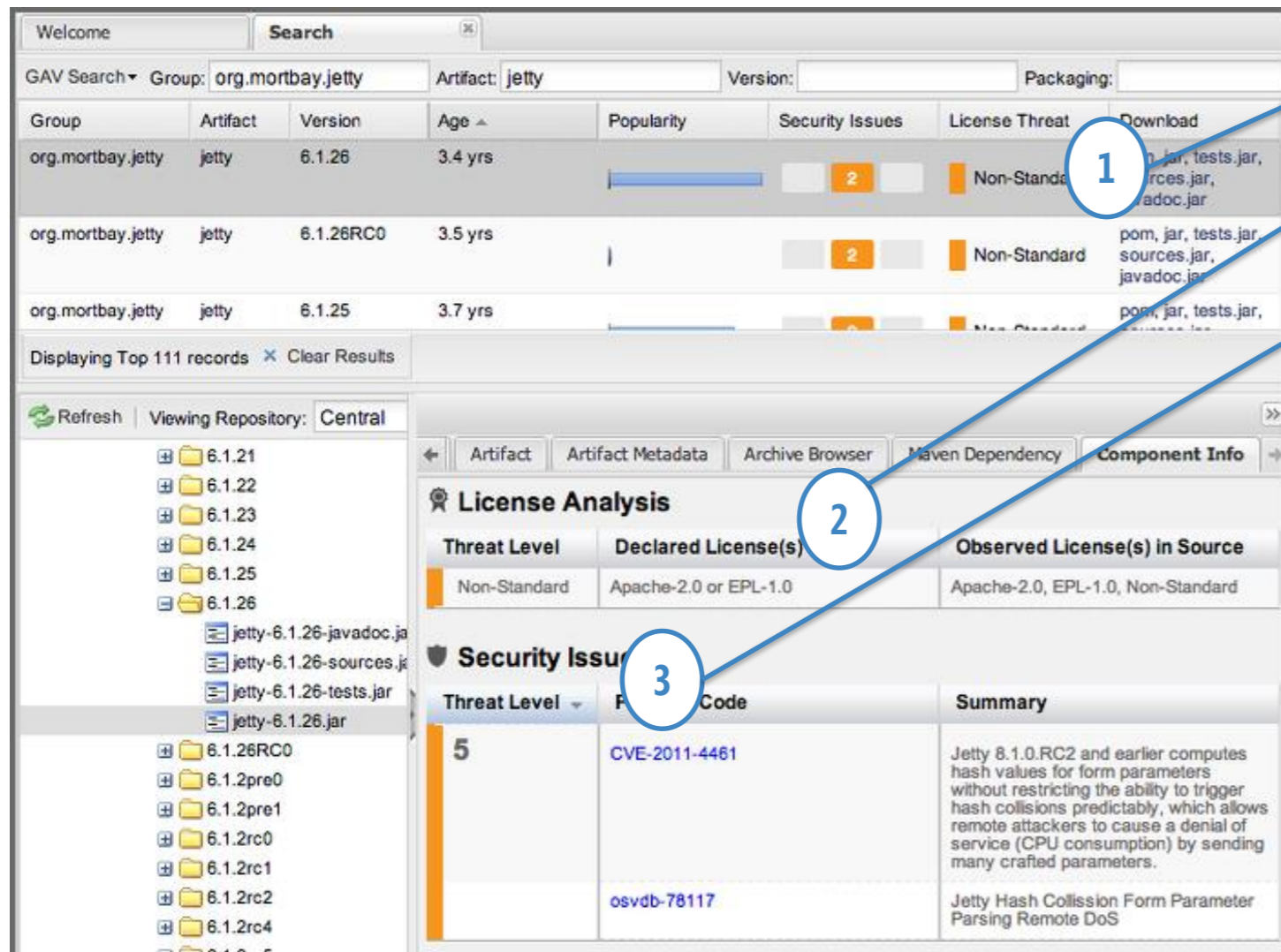
Filter by Artifact ID

Artifact ID	Scopes	Repo Path	License
aopalliance:aopalliance:1.0	compile	gradle-libs-cache:aopalliance/aopalliance/1.0...	Public Domain
org.springframework:spring-beans:2.5.6	compile	Not in repository (externally resolved or dele...	Not Found
org.springframework:spring-aop:2.5.6	compile	Not in repository (externally resolved or dele...	Not Found
org.springframework:spring-core:2.5.6	compile	Not in repository (externally resolved or dele...	Not Found
org.testng:testng:5.9	test	gradle-libs-cache:org/testng/testng/5.9/testn...	Not Found
javax.servlet:servlet-api:2.5	provided	java.net.m1-cache:javax/servlet/servlet-api/2...	Not Found
javax.mail:mail:1.4	compile	gradle-libs-cache:javax/mail/mail/1.4/mail-1...	Not Found

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 ?



Group	Artifact	Version	Age	Popularity	Security Issues	License Threat	Download
org.mortbay.jetty	jetty	6.1.26	3.4 yrs	[Progress Bar]	2	Non-Standard	jetty-6.1.26.jar, tests.jar, sources.jar, javadoc.jar
org.mortbay.jetty	jetty	6.1.26RC0	3.5 yrs	[Progress Bar]	2	Non-Standard	pom, jar, tests.jar, sources.jar, javadoc.jar
org.mortbay.jetty	jetty	6.1.25	3.7 yrs	[Progress Bar]	2	Non-Standard	pom, jar, tests.jar, sources.jar, javadoc.jar

Threat Level	Declared License(s)	Observed License(s) in Source
Non-Standard	Apache-2.0 or EPL-1.0	Apache-2.0, EPL-1.0, Non-Standard

Threat Level	CVE Code	Summary
5	CVE-2011-4461	Jetty 8.1.0.RC2 and earlier computes hash values for form parameters without restricting the ability to trigger hash collisions predictably, which allows remote attackers to cause a denial of service (CPU consumption) by sending many crafted parameters.
	osvdb-78117	Jetty Hash Collision Form Parameter Parsing Remote DoS

1 Licence classification

2 Licence declared in the pom.xml

3 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.



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

Repository	Type	Health Check	Format	Policy	Repository Status
Central	proxy	100 107	maven2	Release	In Service
Codehaus Snapshots	proxy	ANALYZE	maven2	Snapshot	In Service
MyProxyRepo	proxy	ANALYZE	maven2	Release	In Service
NuGet Gallery	proxy	ANALYZE	nuget	Mixed	In Service

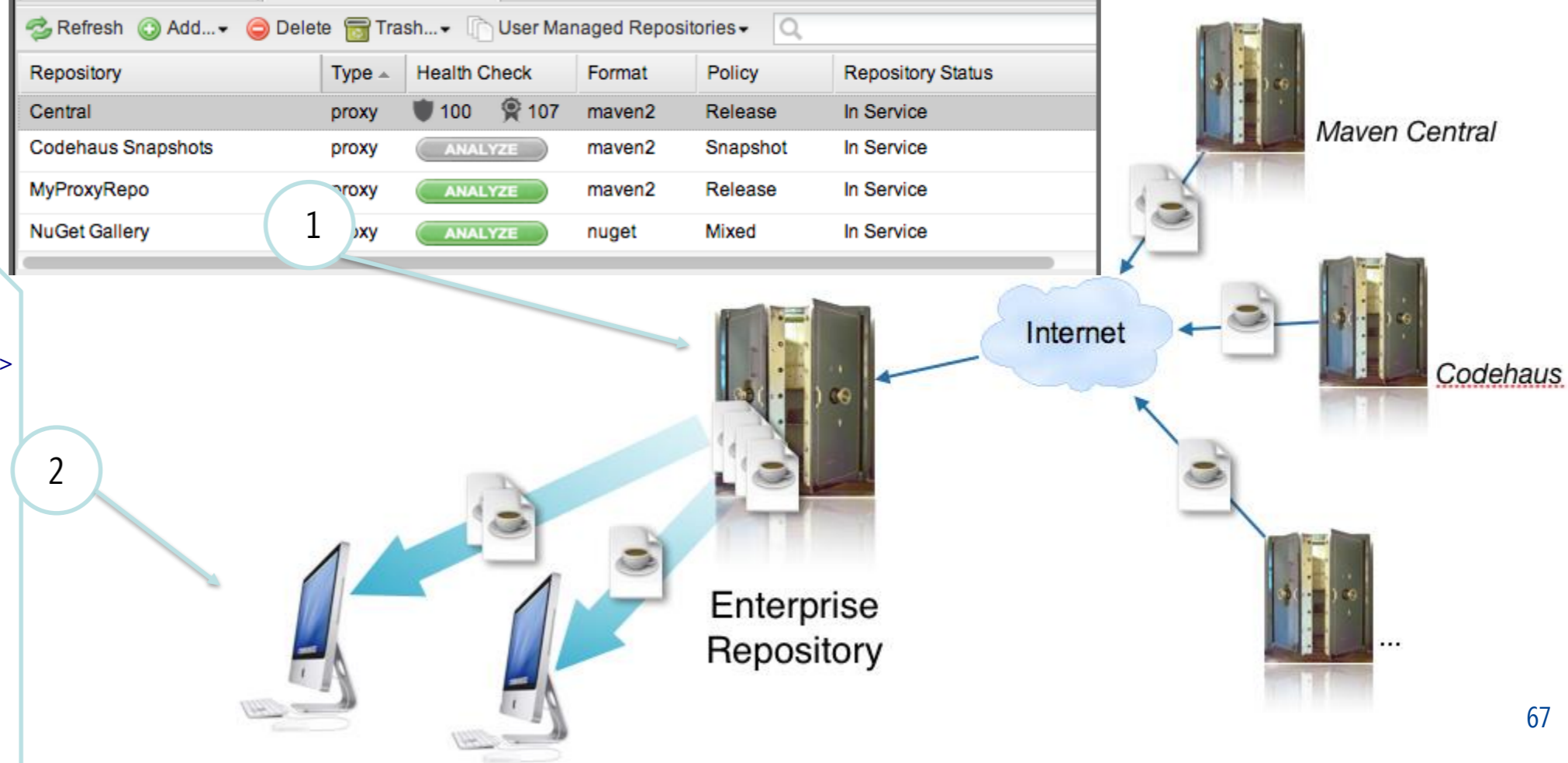
POM.XML

```

<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

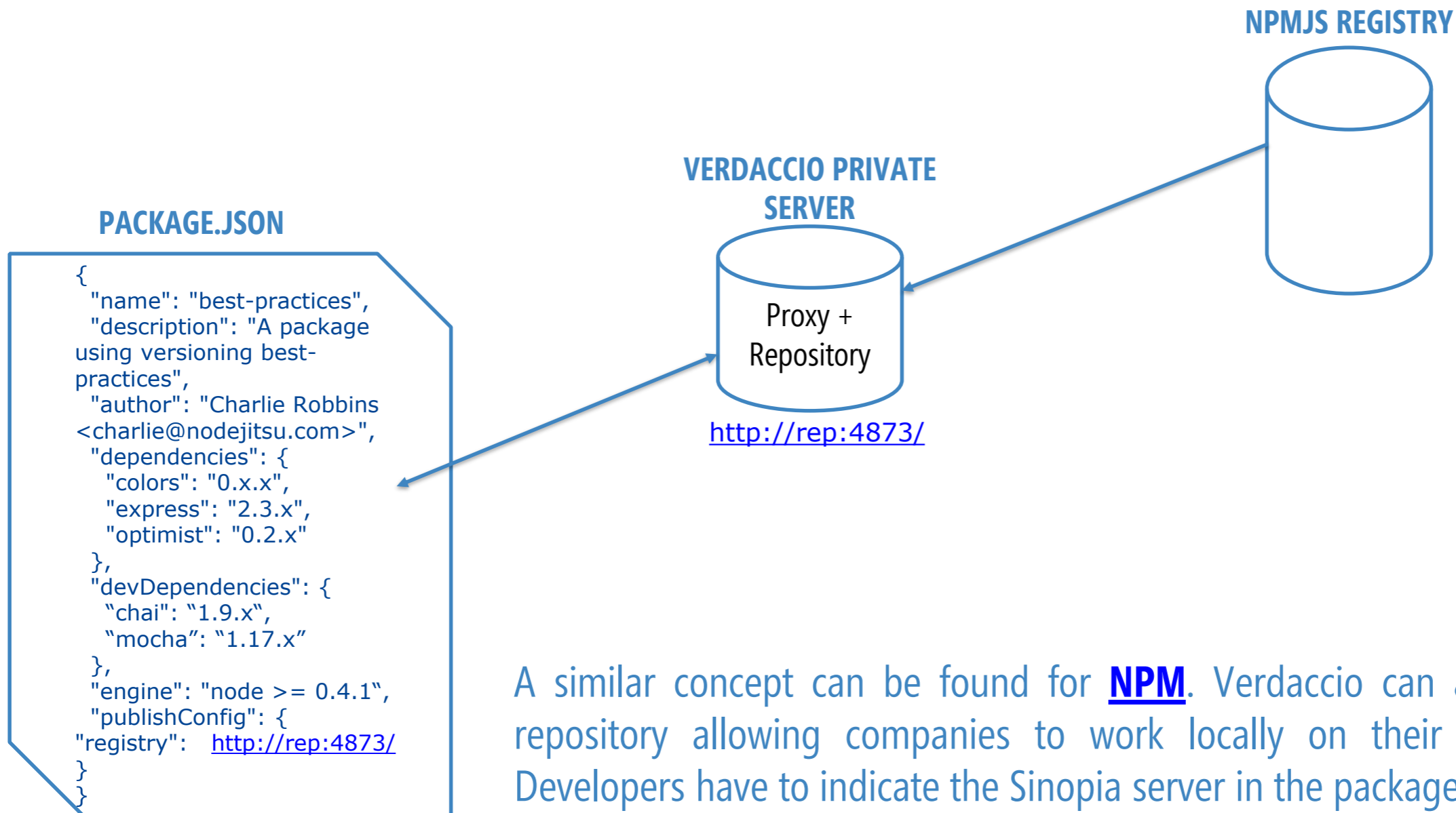
In practice: manage dependencies automatically - examples in the EU

- [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

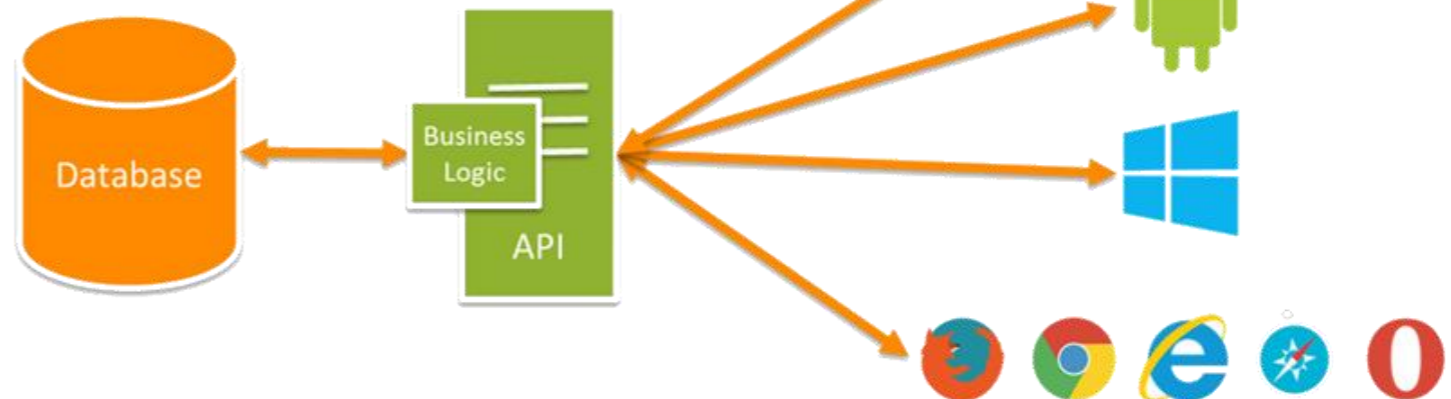
- Certain functionalities could be offered as services to all public administrations or accessed through an API.
- APIs make it possible for developers of a solution to expose functionalities of the solution, while protecting the rest of the application.

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.

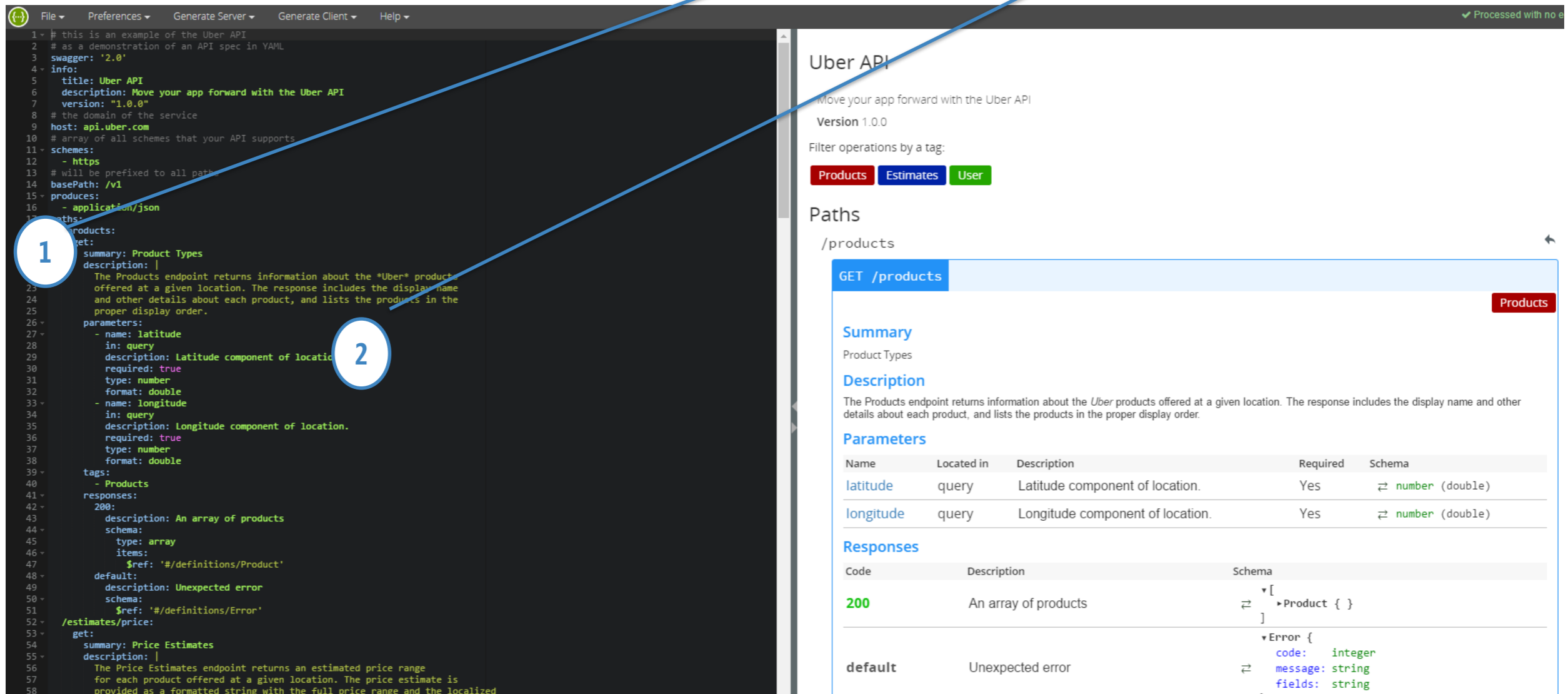


5.3 ORGANISE REUSE AS A SERVICE BY OFFERING SOLUTIONS THROUGH AN API

In practice

Path + protocol

Parameters



1

```
1 # this is an example of the Uber API
2 # as a demonstration of an API spec in YAML
3 swagger: '2.0'
4 info:
5   title: Uber API
6   description: Move your app forward with the Uber API
7   version: "1.0.0"
8 # the domain of the service
9 host: api.uber.com
10 # array of all schemes that your API supports
11 schemes:
12 - https
13 # will be prefixed to all paths
14 basePath: /v1
15 produces:
16 - application/json
17 paths:
18   products:
19     get:
20       summary: Product Types
21       description: |
22         The Products endpoint returns information about the *Uber* products
23         offered at a given location. The response includes the display name
24         and other details about each product, and lists the products in the
25         proper display order.
26       parameters:
27         - name: latitude
28           in: query
29           description: Latitude component of location.
30           required: true
31           type: number
32           format: double
33         - name: longitude
34           in: query
35           description: Longitude component of location.
36           required: true
37           type: number
38           format: double
39       tags:
40         - Products
41       responses:
42         200:
43           description: An array of products
44           schema:
45             type: array
46             items:
47               $ref: '#/definitions/Product'
48         default:
49           description: Unexpected error
50           schema:
51             $ref: '#/definitions/Error'
52   /estimates/price:
53     get:
54       summary: Price Estimates
55       description: |
56         The Price Estimates endpoint returns an estimated price range
57         for each product offered at a given location. The price estimate is
58         provided as a formatted string with the full price range and the localized
```

2

Uber API

Move your app forward with the Uber API

Version 1.0.0

Filter operations by a tag:

Products Estimates User

Paths

/products

GET /products

Products

Summary

Product Types

Description

The Products endpoint returns information about the *Uber* products offered at a given location. The response includes the display name and other details about each product, and lists the products in the proper display order.

Parameters

Name	Located in	Description	Required	Schema
latitude	query	Latitude component of location.	Yes	↔ number (double)
longitude	query	Longitude component of location.	Yes	↔ number (double)

Responses

Code	Description	Schema
200	An array of products	↔ [Product { }]
default	Unexpected error	↔ Error { code: integer, message: string, fields: string }

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





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.





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:

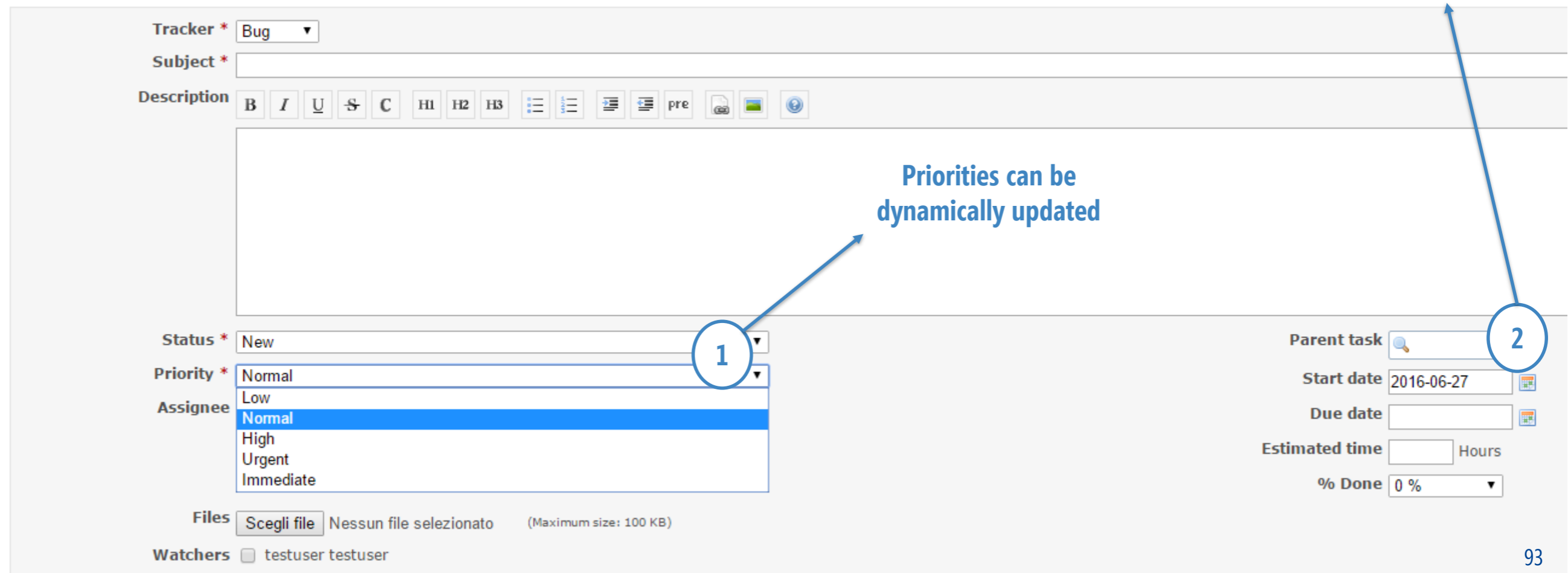
- Hours of operation
- Help desk contact availability
- Priority assignment criteria
- Definition of Response Time depending on the Severity Level
- Definition of Resolution Time to solve issues
- Maintenance schedules
- Notification mechanisms

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](#):

New issue



The screenshot shows the 'New issue' form in Redmine. The form includes fields for Tracker (Bug), Subject, Description (with a rich text editor toolbar), Status (New), Priority (Normal), Assignee (Normal), Parent task, Start date (2016-06-27), Due date, Estimated time, and % Done (0%). Annotations include a blue circle with the number '1' pointing to the Priority dropdown menu, and a blue circle with the number '2' pointing to the Parent task field. Arrows from these circles point to text annotations: 'Priorities can be dynamically updated' and 'Response time and resolution progress can be evaluated'.

Tracker * Bug

Subject *

Description

Status * New

Priority * Normal

Assignee

Parent task

Start date 2016-06-27

Due date

Estimated time Hours

% Done 0 %

Files Scegli file Nessun file selezionato (Maximum size: 100 KB)

Watchers testuser testuser

Priorities can be dynamically updated

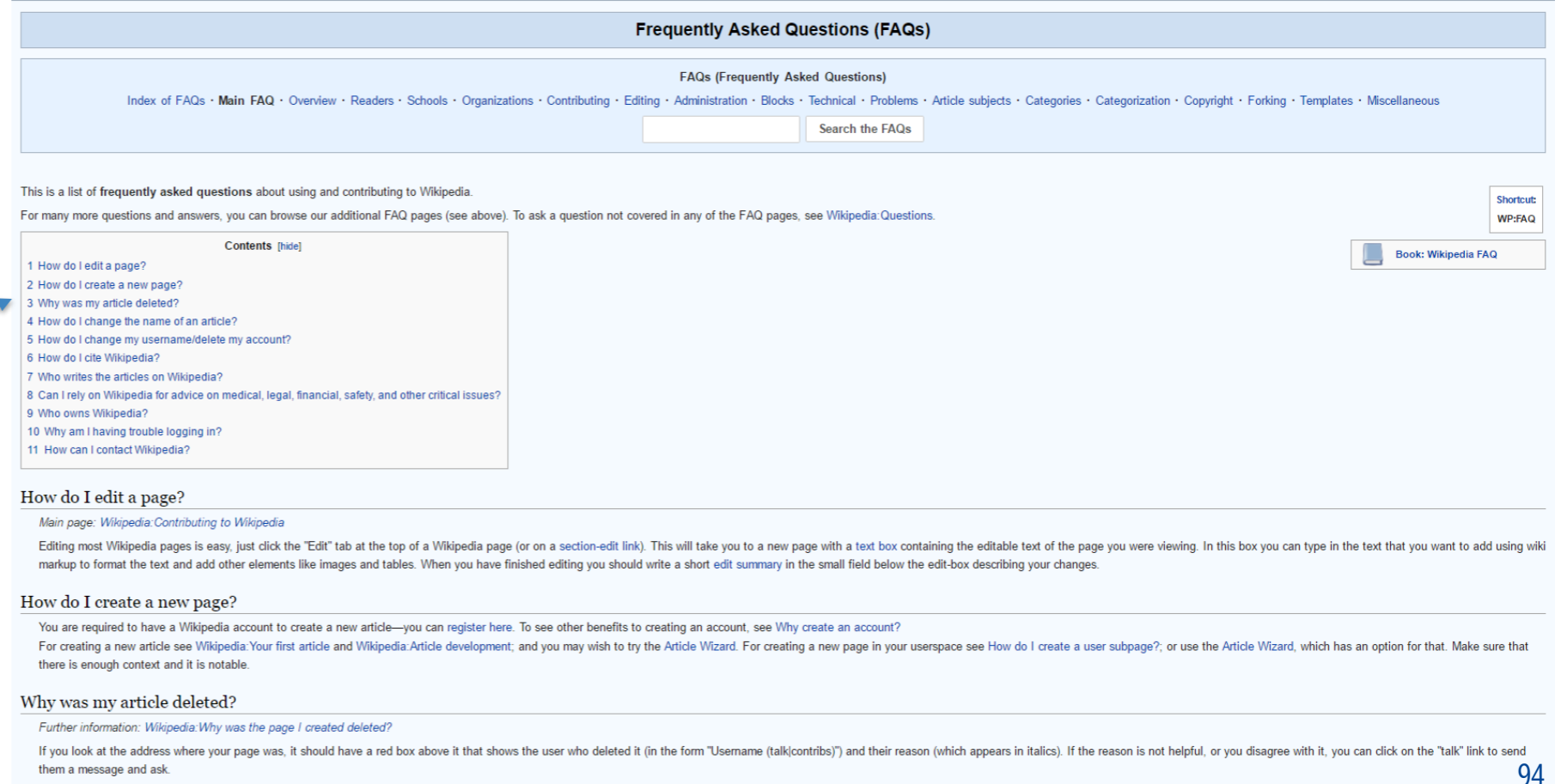
Response time and resolution progress can be evaluated

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](#)

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](#).

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.

Shortcut: WP:FAQ

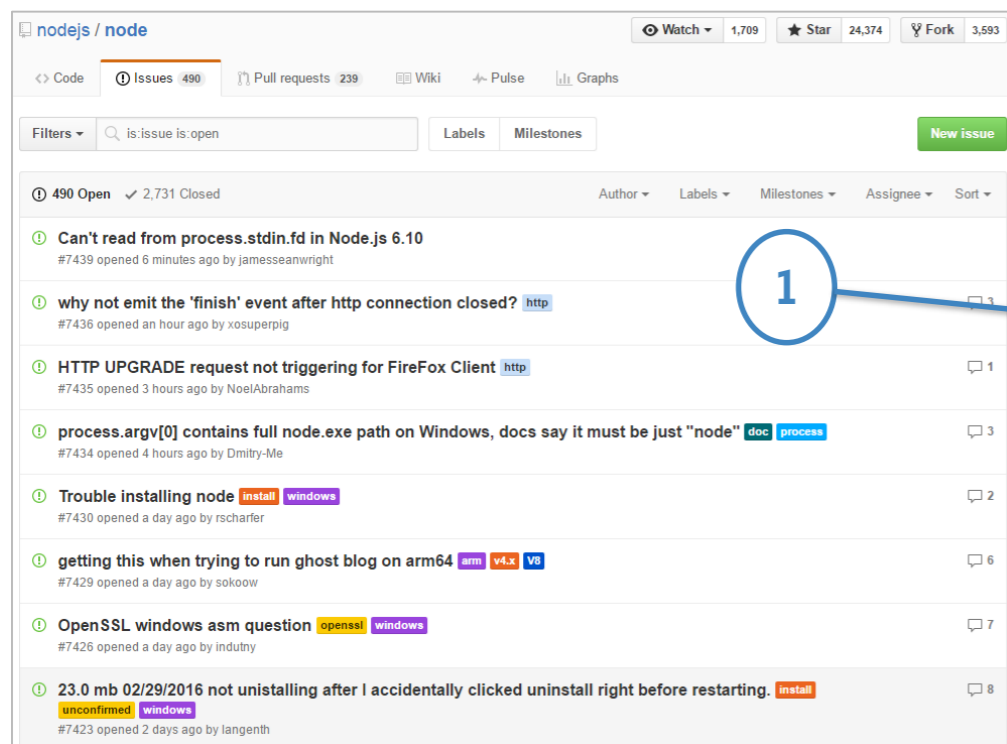
Book: Wikipedia FAQ

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

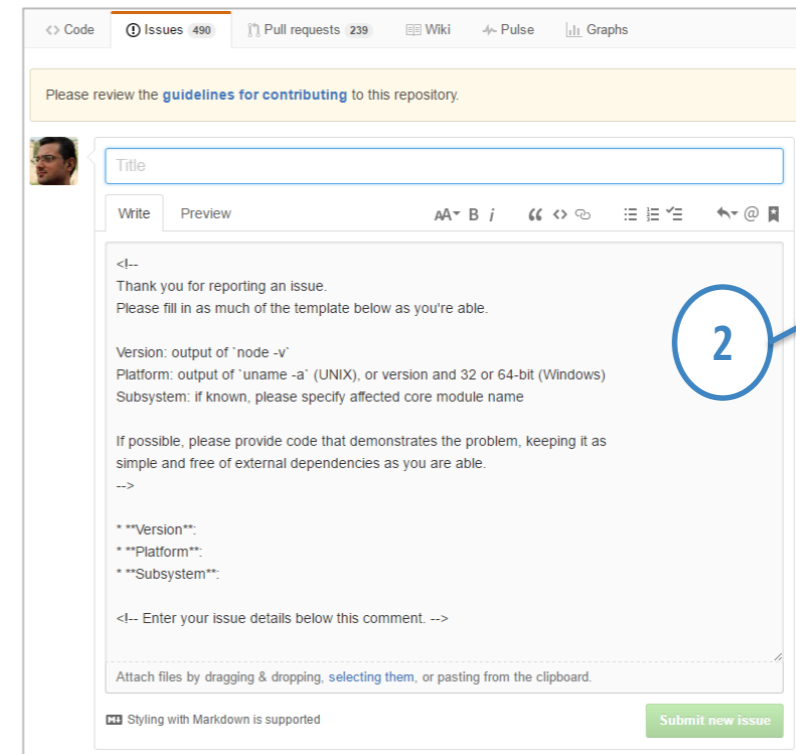
In practice: issue tracking

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



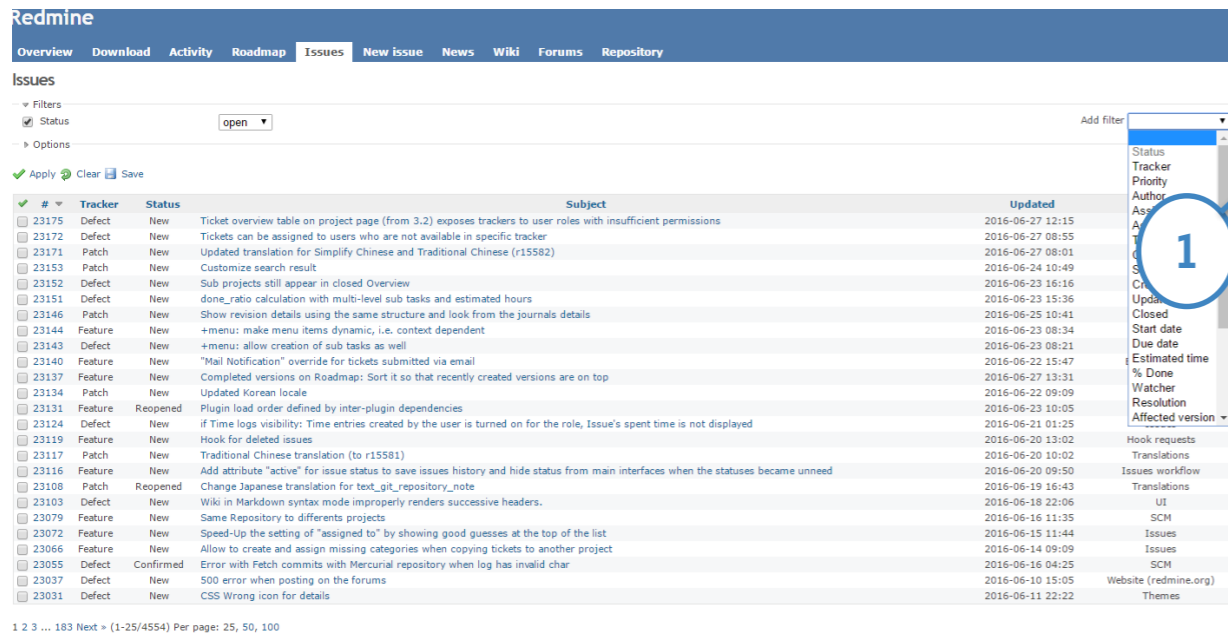
Templates help developers to understand issues better.

7.1 PLAN ADEQUATE LEVELS OF MAINTENANCE AND SUPPORT FOR YOUR SOLUTION

In practice: issue tracking

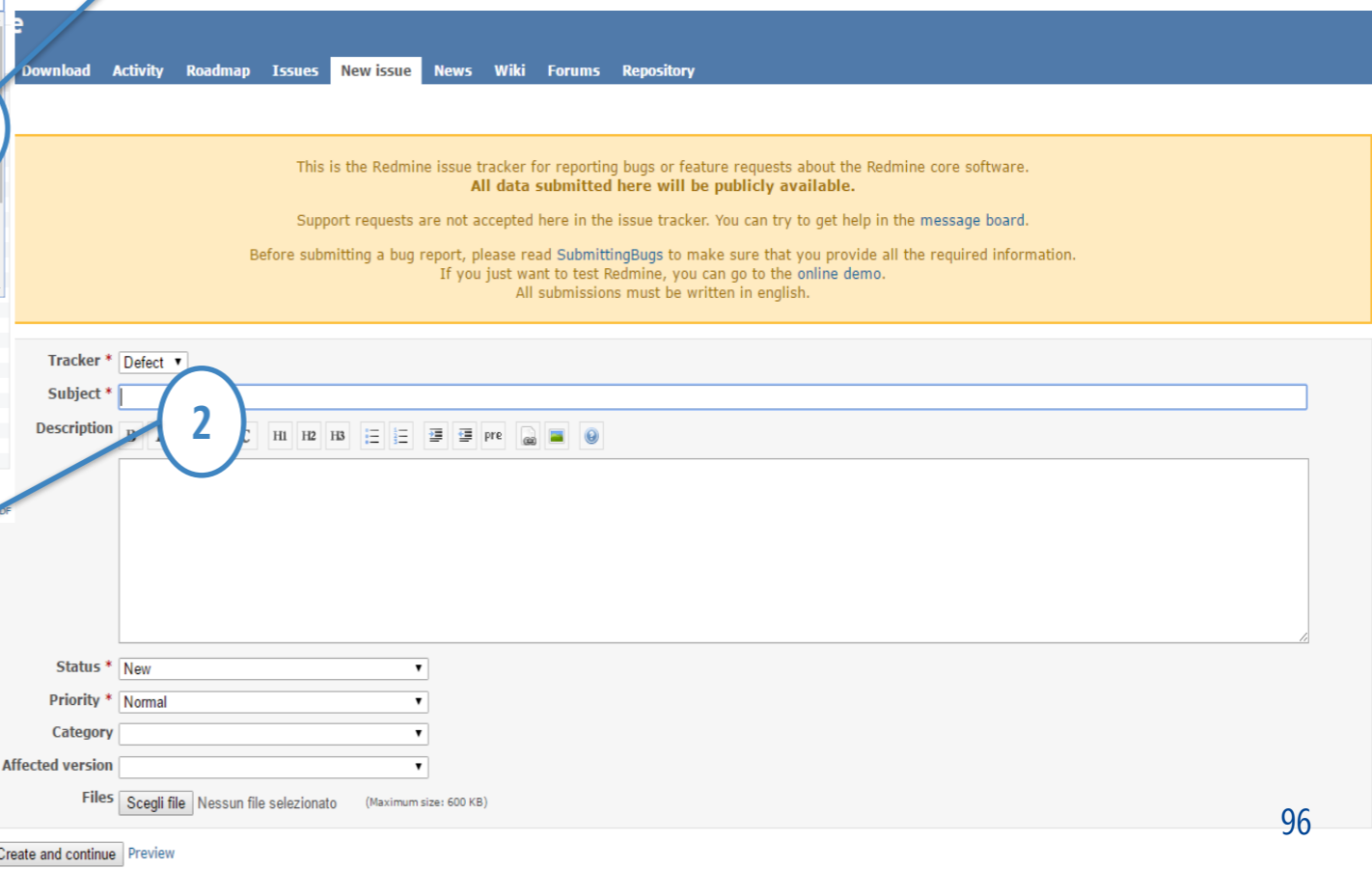
Redmine: more issue field customisation

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



Redmine Issues list showing columns: #, Tracker, Status, Subject, Updated. A filter dropdown is open, showing options like Status, Tracker, Priority, Author, etc.

#	Tracker	Status	Subject	Updated
23175	Defect	New	Ticket overview table on project page (from 3.2) exposes trackers to user roles with insufficient permissions	2016-06-27 12:15
23172	Defect	New	Tickets can be assigned to users who are not available in specific tracker	2016-06-27 08:55
23171	Patch	New	Updated translation for Simplify Chinese and Traditional Chinese (r15582)	2016-06-27 08:01
23153	Patch	New	Customize search result	2016-06-24 10:49
23152	Defect	New	Sub_projects still appear in closed Overview	2016-06-23 16:16
23151	Defect	New	done_ratio calculation with multi-level sub tasks and estimated hours	2016-06-23 15:36
23146	Patch	New	Show revision details using the same structure and look from the journals details	2016-06-25 10:41
23144	Feature	New	+menu: make menu items dynamic, i.e. context dependent	2016-06-23 08:34
23143	Defect	New	+menu: allow creation of sub tasks as well	2016-06-23 08:21
23140	Feature	New	"Mail Notification" override for tickets submitted via email	2016-06-22 15:47
23137	Feature	New	Completed versions on Roadmap: Sort it so that recently created versions are on top	2016-06-27 13:31
23134	Patch	New	Updated Korean locale	2016-06-22 09:09
23131	Feature	Reopened	Plugin load order defined by inter-plugin dependencies	2016-06-23 10:05
23124	Defect	New	if Time logs visibility: Time entries created by the user is turned on for the role, Issue's spent time is not displayed	2016-06-21 01:25
23119	Feature	New	Hook for deleted issues	2016-06-20 13:02
23117	Patch	New	Traditional Chinese translation (to r15581)	2016-06-20 10:02
23116	Feature	New	Add attribute "active" for issue status to save issues history and hide status from main interfaces when the statuses became unneed	2016-06-20 09:50
23108	Patch	Reopened	Change Japanese translation for text_git_repository_note	2016-06-19 16:43
23103	Defect	New	Wiki in Markdown syntax mode improperly renders successive headers.	2016-06-18 22:06
23079	Feature	New	Same Repository to differents projects	2016-06-16 11:35
23072	Feature	New	Speed-Up the setting of "assigned to" by showing good guesses at the top of the list	2016-06-15 11:44
23066	Feature	New	Allow to create and assign missing categories when copying tickets to another project	2016-06-14 09:09
23055	Defect	Confirmed	Error with Fetch commits with Mercurial repository when log has invalid char	2016-06-16 04:25
23037	Defect	New	500 error when posting on the forums	2016-06-10 15:05
23031	Defect	New	CSS Wrong icon for details	2016-06-11 22:22



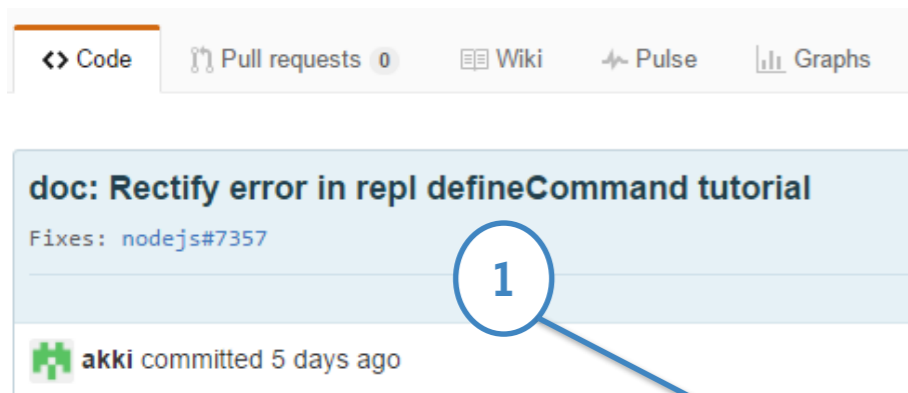
New Issue form with fields: Tracker (Defect), Subject, Description, Status (New), Priority (Normal), Category, Affected version, Files. Includes a 'Create' button.

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 Pull requests 0 Wiki Pulse Graphs

doc: Rectify error in repl defineCommand tutorial
Fixes: [nodejs#7357](#)

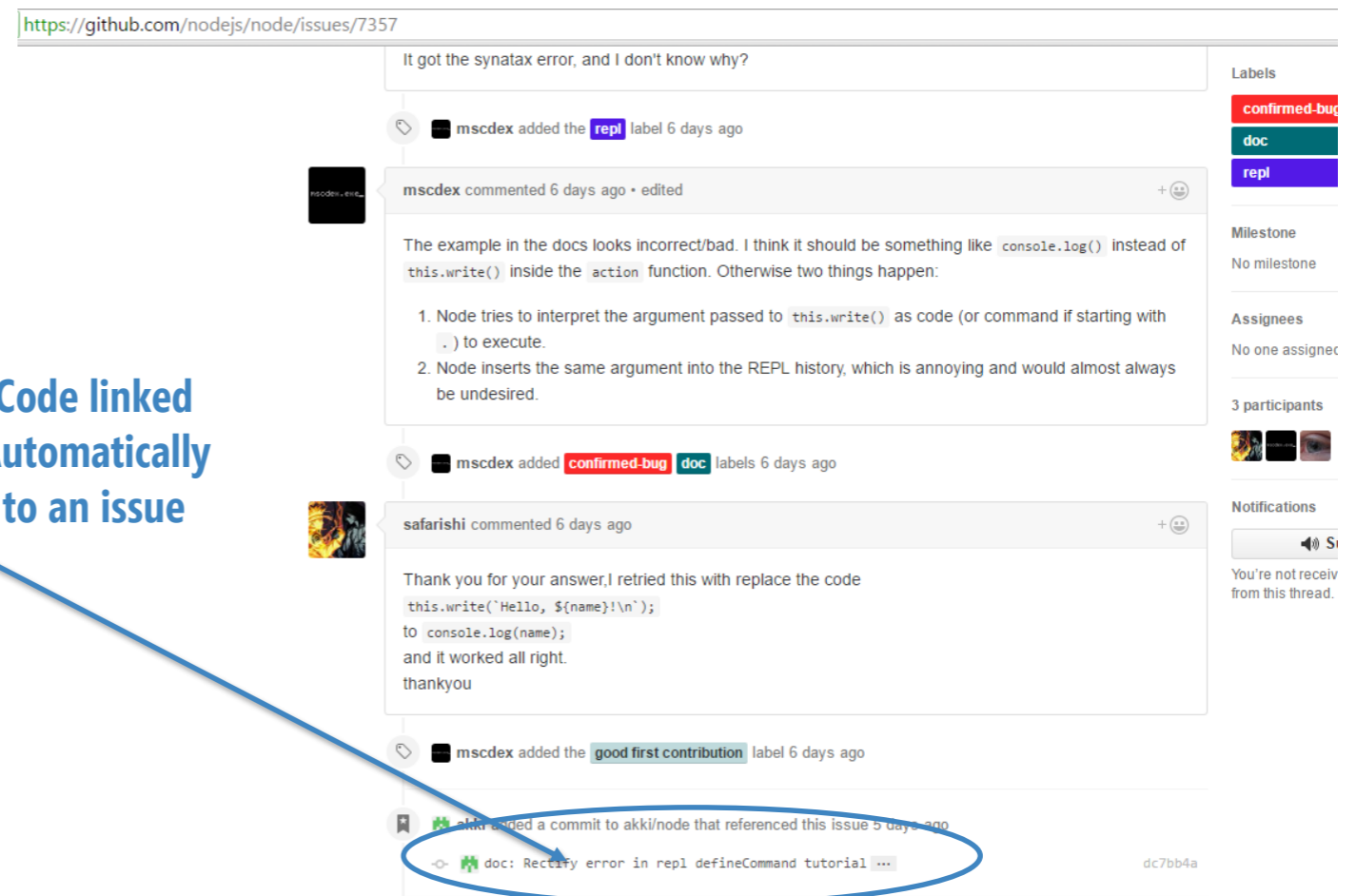
1

akki committed 5 days ago

Showing 1 changed file with 1 addition and 1 deletion.

```
2 doc/api/repl.md
@@ -312,7 +312,7 @@ replServer.defineCommand('sayhello', {
312 312     action: function(name) {
313 313         this.lineParser.reset();
314 314         this.bufferedCommand = '';
315 315         - this.write(`Hello, ${name}!\n`);
316 316         + console.log(`Hello, ${name}!\n`);
317 317         this.displayPrompt();
318 318     }
    });
```

Code linked
Automatically
to an issue



<https://github.com/nodejs/node/issues/7357>

It got the syntax error, and I don't know why?

mscdex added the **repl** label 6 days ago

mscdex commented 6 days ago • edited

The example in the docs looks incorrect/bad. I think it should be something like `console.log()` instead of `this.write()` inside the `action` function. Otherwise two things happen:

1. Node tries to interpret the argument passed to `this.write()` as code (or command if starting with `.`) to execute.
2. Node inserts the same argument into the REPL history, which is annoying and would almost always be undesired.

mscdex added **confirmed-bug** **doc** labels 6 days ago

safarishi commented 6 days ago

Thank you for your answer, I retried this with replace the code `this.write('Hello, ${name}!\n');` to `console.log(name);` and it worked all right. thank you

mscdex added the **good first contribution** label 6 days ago

akki assigned a commit to akki/node that referenced this issue 5 days ago

doc: Rectify error in repl defineCommand tutorial ...

Labels: confirmed-bug, doc, repl

Milestone: No milestone

Assignees: No one assigned

3 participants

Notifications: You're not receiving notifications from this thread.


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:	<div style="width: 0%;"></div> 0%
Category:	Time tracking	Affected version:	
Target version:	3.2.4		
Resolution:	Fixed		

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

History

-  Updated by Felix Schäfer 14 days ago
 - File 23054-clear_time_entry_custom_fields.patch added
-  Updated by Jan from Planio www.plan.io 14 days ago
 - Target version set to Candidate for next minor release
-  Updated by Jean-Philippe Lang 12 days ago
 - Target version changed from Candidate for next minor release to 3.2.4
-  Updated by Jean-Philippe Lang 9 days ago
 - Category set to Time tracking
 - Status changed from New to Closed
 - Assignee set to Jean-Philippe Lang
 - Resolution set to Fixed

Associated revisions

- #1 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.
- #2 Revision 15533
 Added by Jean-Philippe Lang 12 days ago
 Adds a test for #23054.
- #3 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).

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

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

The screenshot shows the Joinup website interface. At the top, there is a navigation bar with the European Commission logo and the 'joinup' brand name. Below this is a search bar and a 'Login or Sign up' button. The main header area contains the text 'Share and reuse interoperability solutions for public administrations'. A breadcrumb trail indicates the current location: 'European Commission > ISA > Joinup > Communities > Srs > Welcome'. Below the breadcrumb is a navigation menu with options like 'Find solutions', 'Share and Collaborate', 'Keep up-to-date', and 'People'. The main content area features a community card for 'Sharing and Reuse of IT solutions' with a 'Join this community' button, a rating of 3.75/5, and social media sharing options. A secondary navigation bar includes 'Find communities', 'My Communities', 'Recommended', and 'Editor's Choice'. Below the community card is a tabbed interface with 'Overview' selected, showing a description of the community's purpose. At the bottom, there are tabs for 'Latest News', 'Latest activity', and 'Latest e-Library Content'. A sidebar on the right contains an 'Add...' button, an 'Actions' button, a 'Mailing list' dropdown menu with the email address 'srs[at]joinup[dot]ec[dot]europa[dot]eu', and a 'Metadata' section.

Спасибо Gracías اشكر
Grazie
Obrigado Спасибо Dank U
Eυχαριστώ Danke
Dziękuję Eυχαριστώ
Merci
Thank You
Ngiyabonga
Dank U
Diolch
Thank You
Ngiyabonga
Obrigado
Dank U
תודה
Terima Kasih
Diolch
Grazie
Tack
Merci
Eυχαριστώ
Tack
Eυχαριστώ