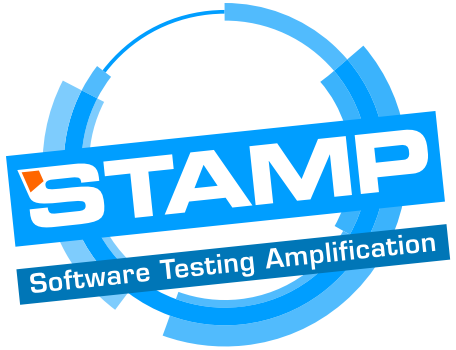
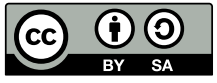


Software Testing AMPlification for the DevOps Team

WP 4 Development and Integration

Daniele Gagliardi, Ciro Formisano (ENG)

STAMP Project Final Review
Brussels, 6 February, 2020



The STAMP project received funding from European Union's Horizon 2020 research and innovation programme under grant agreement 731529.

WP4 tasks

ID	Description	M1 – M36
T4.1	Collaborative Software Engineering Platform setup and management	M1 – M36
T4.2	Stamp product architecture definition and implementation	M1 – M30
T4.3	STAMP assets integration in various software factory	M1 – M36
T4.4	Stamp assets documentation	M12 – M36

WP Leader: ENG

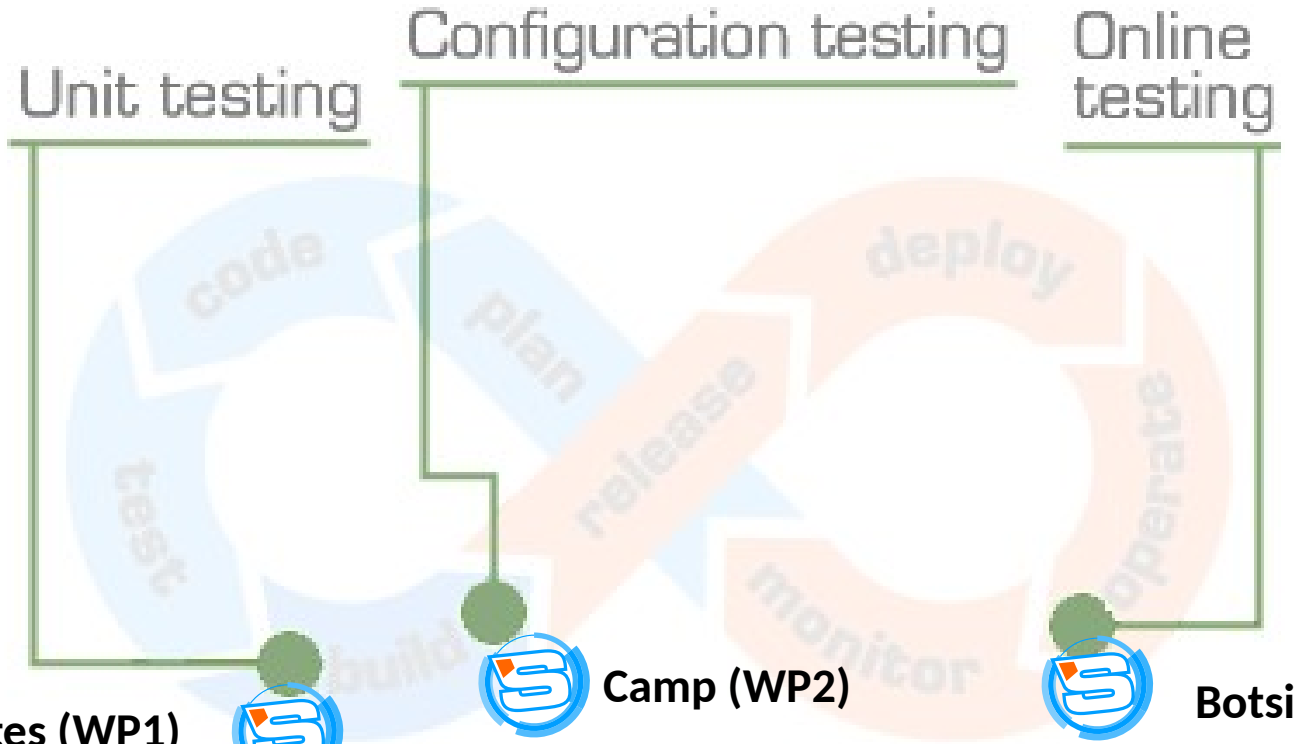
WP Participants: ATOS, ActiveEon, XWiki, INRIA, KTH

WP4 Deliverables

ID	Description	Date
D4.1	STAMP Collaborative Software Engineering Platform	M6
D4.2	First public version of the API and initial implementation of services and courseware	M14
D4.3	Second public version of the API and initial implementation of services and courseware	M24
D4.4	Final public version of the API and initial implementation of services and courseware	M26

Development & Integration

What did we do?

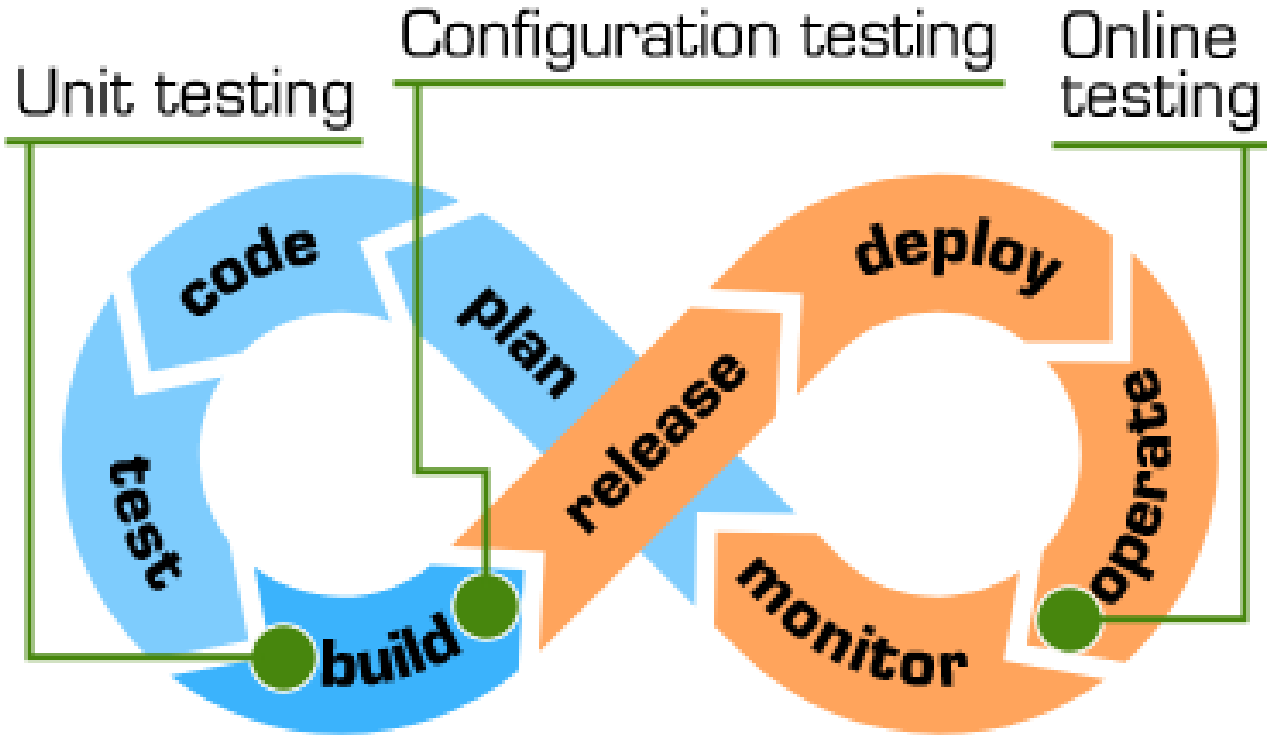


Dspot/Descartes (WP1)

Camp (WP2)

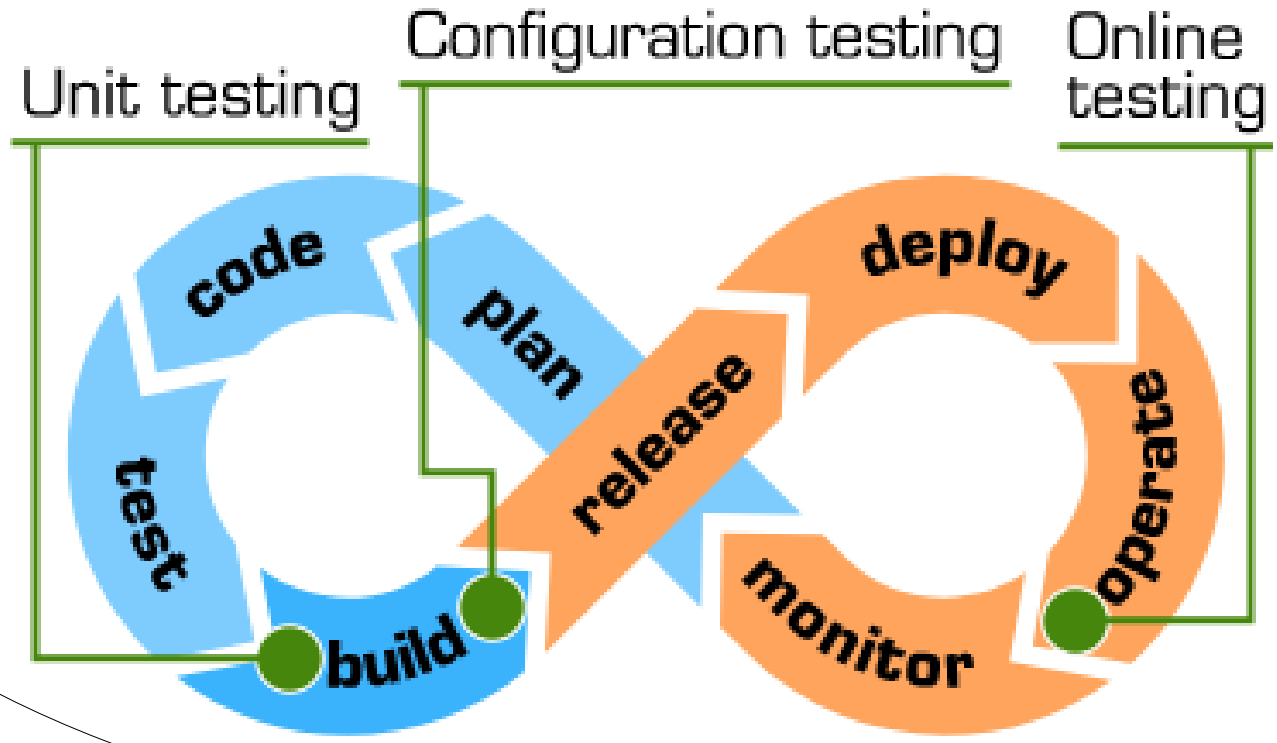
Botsing/RAMP (WP3)

What did we do?



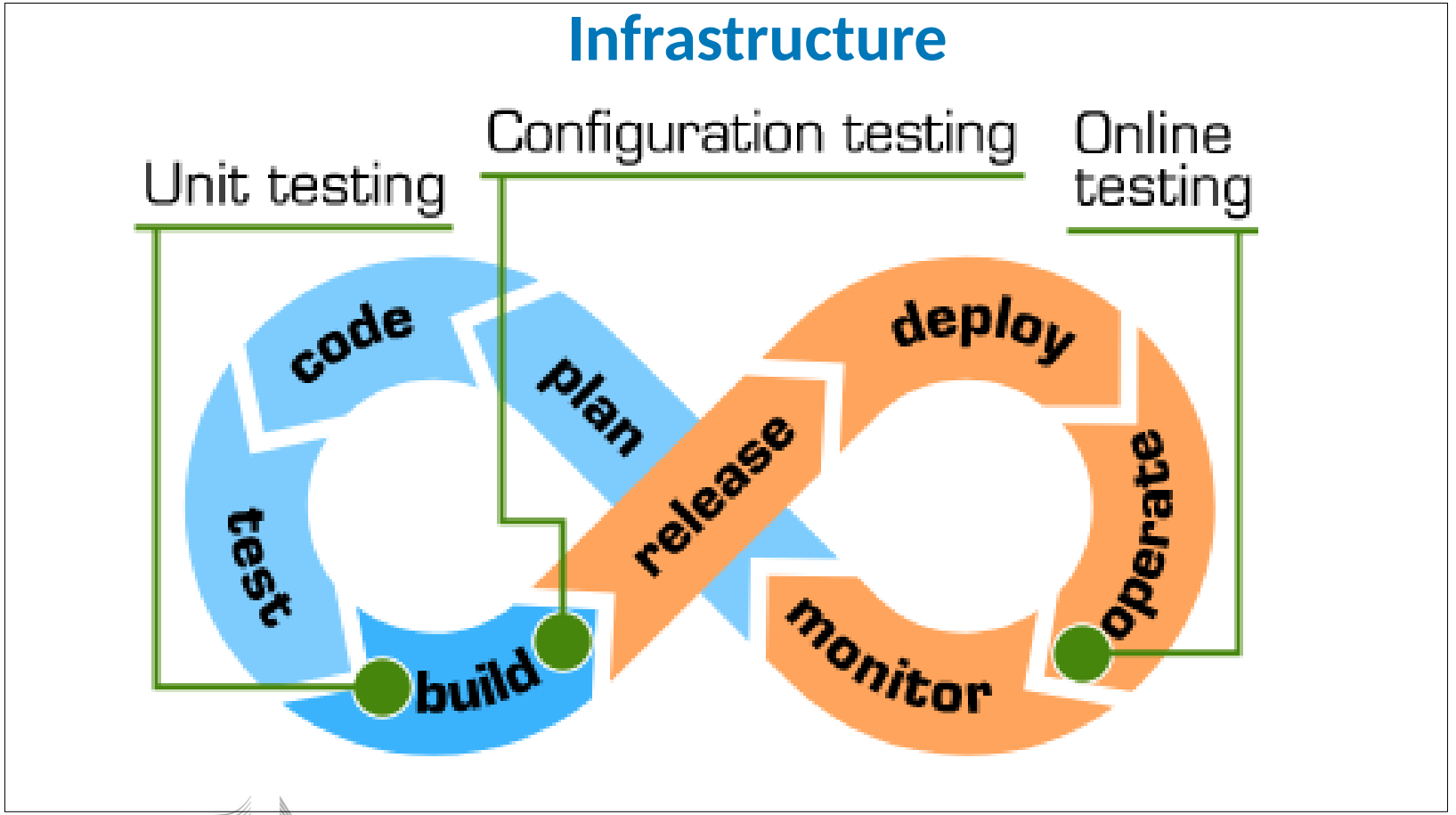
What did we do?

Scenarios

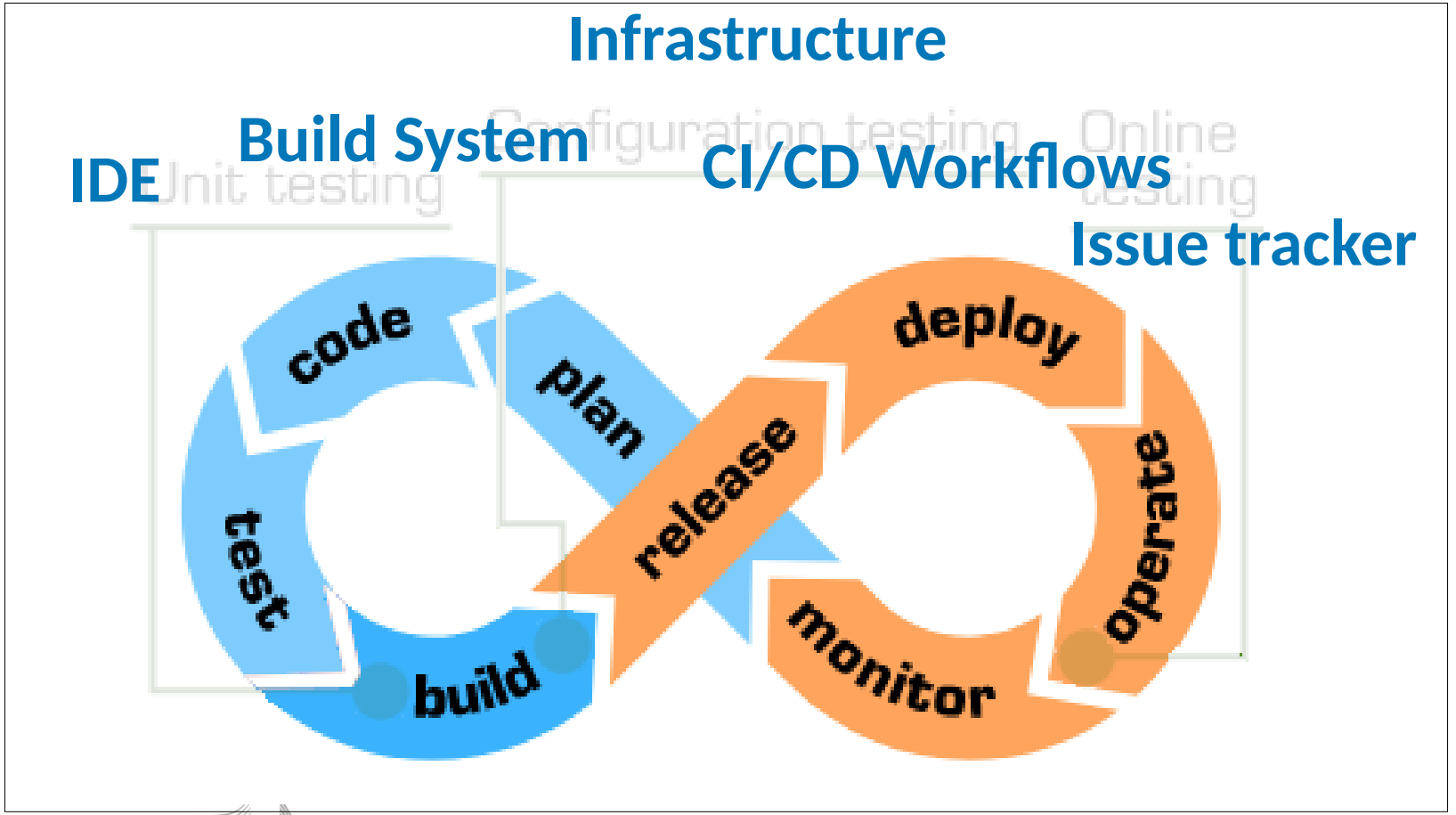


European Commission

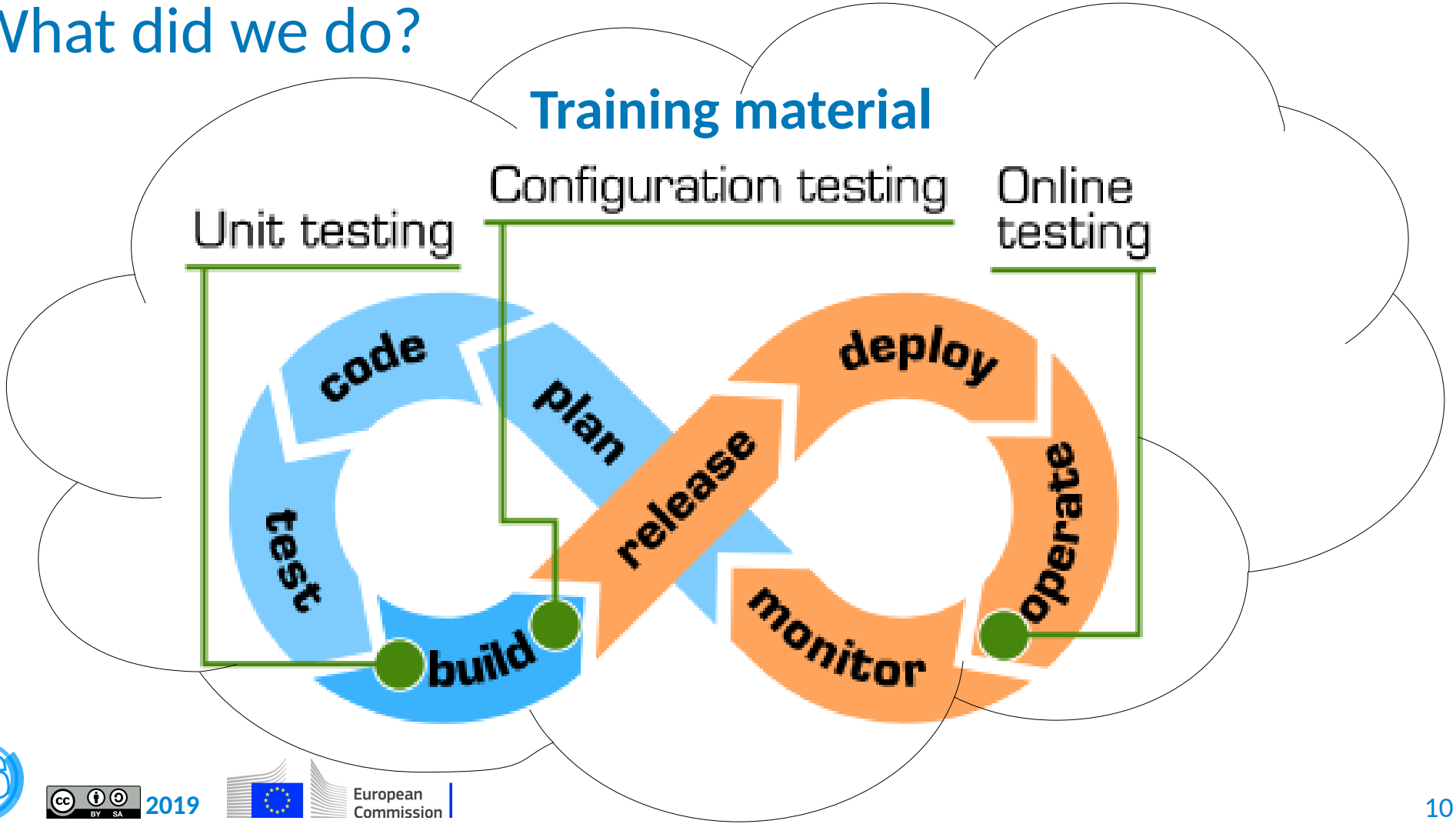
What did we do?



What did we do?



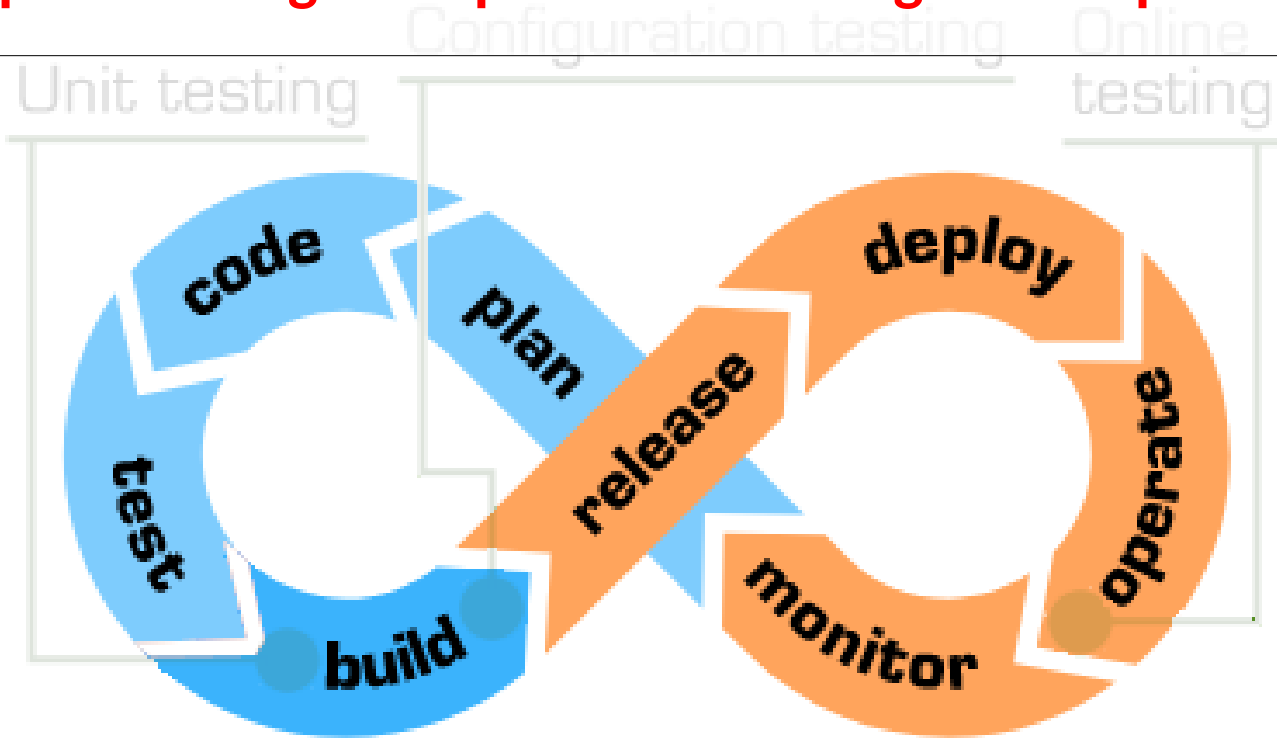
What did we do?



European Commission

What did we do?

Important target: to preserve existing infra & processes



European Commission

Scenarios

Starting point: use cases and scenarios

- Software companies interested in increasing the quality of their products
- Cover different areas, providing generic and specific requirements to obtain an highly exploitable outcome
- One of the objectives of the work on development and integration is to support provided experimentation and scenarios
- … but it shall not be limited to project use cases

Use cases: concrete requirements providers

Heterogeneous domains covered

- Enterprise wiki (XWIKI)
- Business processes management (Proactive – ActiveEon)
- Open source projects marketplace (OW2)
- Cloud and IoT (tellu)
- Smart cities solutions (Atos)

Integration requirements mainly originated by the use cases

- Wide **exploitability** due to the wide overall domain covered
- High **extensibility**

Integration

Client side: STAMP IDE

- **Requirements:**
  - test amplification feature to a usable IDE
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - Eclipse integration
  - Support for DSpot, Descartes, Botsing and RAMP
  - Enhanced wizards
  - interoperability with Jira

Client side: STAMP IDE

- **Requirements:**
  - test amplification feature to a usable IDE
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - Eclipse integration
  - Support for DSpot, Descartes, Botsing and RAMP
  - Enhanced wizards
  - interoperability with Jira

Client side: STAMP IDE

- **Requirements:**
  - test amplification feature to a usable IDE
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - https://github.com/STAMP-project/stamp-ide

Client side: STAMP IDE

- **Requirements:**
  - test amplification feature to a usable IDE
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - https://github.com/STAMP-project/stamp-ide

- **To support the following use cases**:

Server side: Build Systems

- **Requirements:**
  - Integrate development lifecycle with test amplification
  - solution based on widespread tools to preserve existing processes
- **Solution:**
  - Maven and Gradle integration
  - Avaliable for DSpot, Descartes, Botsing and RAMP
  - Integrated with Botsing Server to interoperate with Jira and GitHub

Server side: Build Systems

- **Requirements:**
  - Integrate development lifecycle with test amplification
  - solution based on widespread tools to preserve existing processes
- **Solution:**
  - Maven and Gradle integration
  - Avaliable for DSpot, Descartes, Botsing and RAMP
  - Integrated with Botsing Server to interoperate with Jira and GitHub

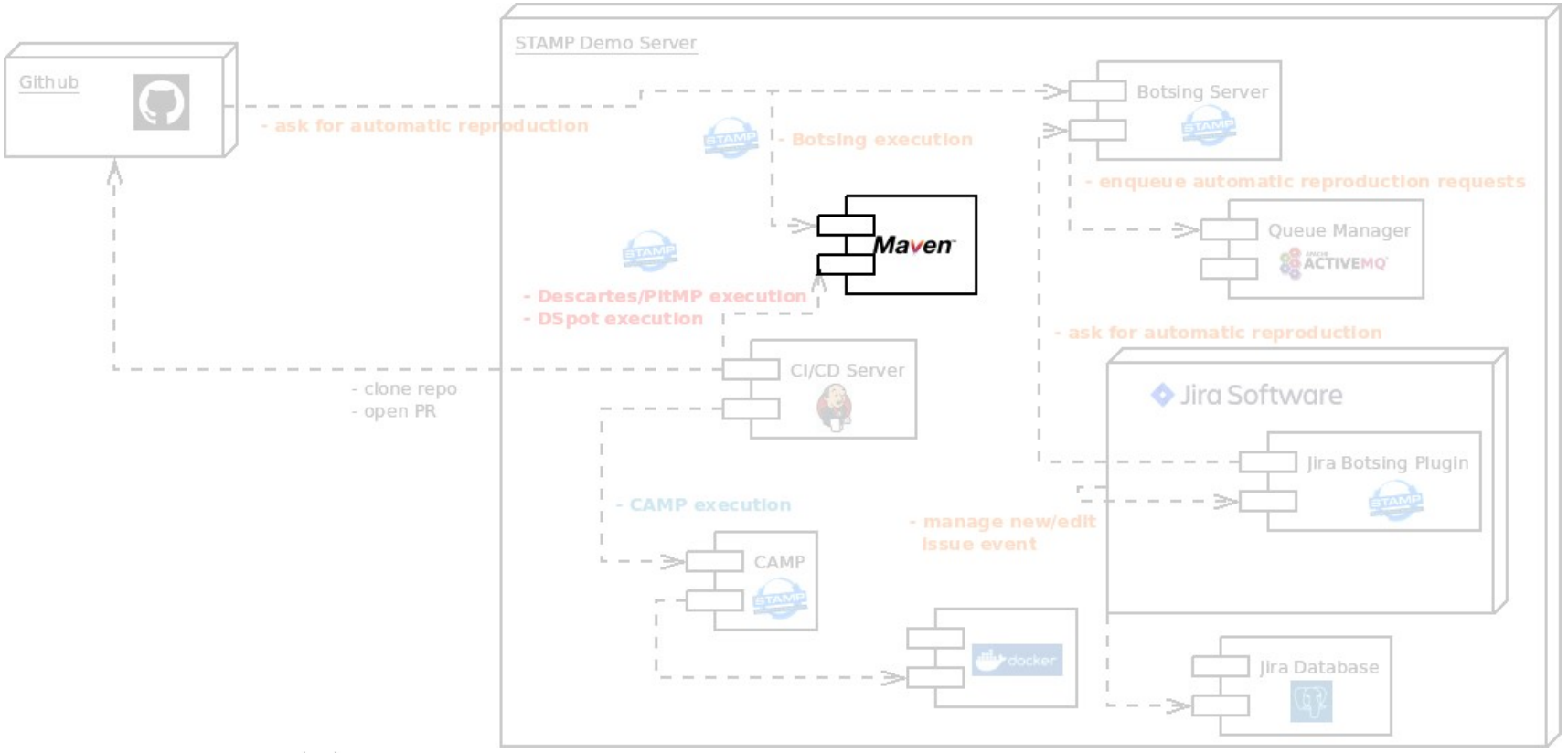
Server side: Build Systems

- **Requirements:**
  - Integrate development lifecycle with test amplification
  - solution based on widespread tools to preserve existing processes
- **Solution:**
  - https://github.com/STAMP-project/dspot/tree/master/dspot-maven
  - https://github.com/STAMP-project/pitmp-maven-plugin
  - https://github.com/STAMP-project/botsing/tree/master/botsing-maven
  - https://github.com/STAMP-project/botsing-gradle-plugin

Server side: Build Systems

- **Requirements:**
  - Integrate development lifecycle with test amplification
  - solution based on widespread tools to preserve existing processes
- **Solution:**
  - https://github.com/STAMP-project/dspot/tree/master/dspot-maven
  - https://github.com/STAMP-project/pitmp-maven-plugin
  - https://github.com/STAMP-project/botsing/tree/master/botsing-maven
  - https://github.com/STAMP-project/botsing-gradle-plugin
- **To support the following use cases:**

Towards STAMP Server



Server side: CI/CD workflows

- **Requirements:**
  - support for complete CI/CD workflow
  - solution based on a widespread tool to preserve existing processes
- **Solution:**
  - Jenkins pipeline libraries and pipelines for test & configuration amplification
  - Available for DSpot and Descartes (tests assessment and amplification)
  - Available for CAMP (functional and performance tests)

Server side: CI/CD workflows

- **Requirements:**
  - support for complete CI/CD workflow
  - solution based on a widespread tool to preserve existing processes
- **Solution:**
  - Jenkins pipeline libraries and pipelines for test & configuration amplification
  - Available for DSpot and Descartes (tests assessment and amplification)
  - Available for CAMP (functional and performance tests)

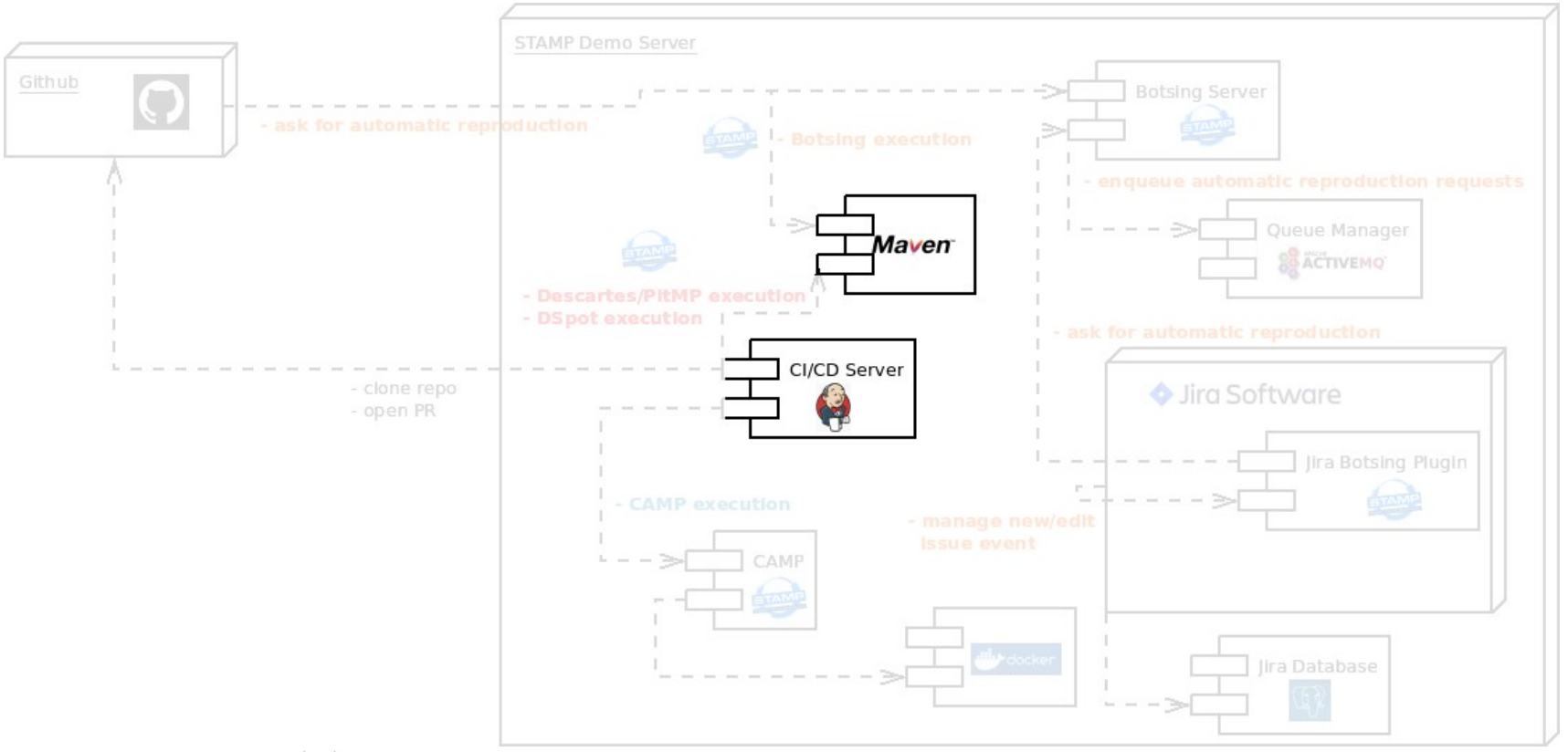
Server side: CI/CD workflows

- **Requirements:**
  - support for complete CI/CD workflow
  - solution based on a widespread tool to preserve existing processes
- **Solution:**
  - https://github.com/STAMP-project/pipeline-library
  - https://github.com/STAMP-project/joram/blob/master/Jenkinsfile
  - https://github.com/STAMP-project/lutece-demo-site-forms/blob/master/Jenkinsfile
  - https://github.com/STAMP-project/stamp-ci/
  - https://github.com/STAMP-project/e2e-STAMP-CI-demo/blob/master/code/camp-perf-test/Jenkinsfile

Server side: CI/CD workflows

- **Requirements:**
  - support for complete CI/CD workflow
  - solution based on a widespread tool to preserve existing processes
- **Solution:**
  - https://github.com/STAMP-project/pipeline-library
  - https://github.com/STAMP-project/joram/blob/master/Jenkinsfile
  - https://github.com/STAMP-project/lutece-demo-site-forms/blob/master/Jenkinsfile
  - https://github.com/STAMP-project/stamp-ci/
  - https://github.com/STAMP-project/e2e-STAMP-CI-demo/blob/master/code/camp-perf-test/Jenkinsfile
- **To support the following use cases:**

Towards STAMP Server: STAMP CI infra is taking shape



European Commission

Server side: Issue trackers

- **Requirements:**
  - Keep track of all the issues and bugs
  - Manage the CI/CD process
  - solution based on a widespread tool to preserve existing processes
- **Sulution:**
  - Botsing Maven plugin
  - Botsing Gradle plugin
  - Jira plugin
  - GitHub Issues integration

} Botsing Server

Server side: Issue trackers

- **Requirements:**
  - Keep track of all the issues and bugs
  - Manage the CI/CD process
  - solution based on a widespread tool to preserve existing processes
- **Sulution**:
  - Botsing Maven plugin
  - Botsing Gradle plugin
  - Jira plugin
  - GitHub Issues integration

Botsing Server

GitHub

Jira Software

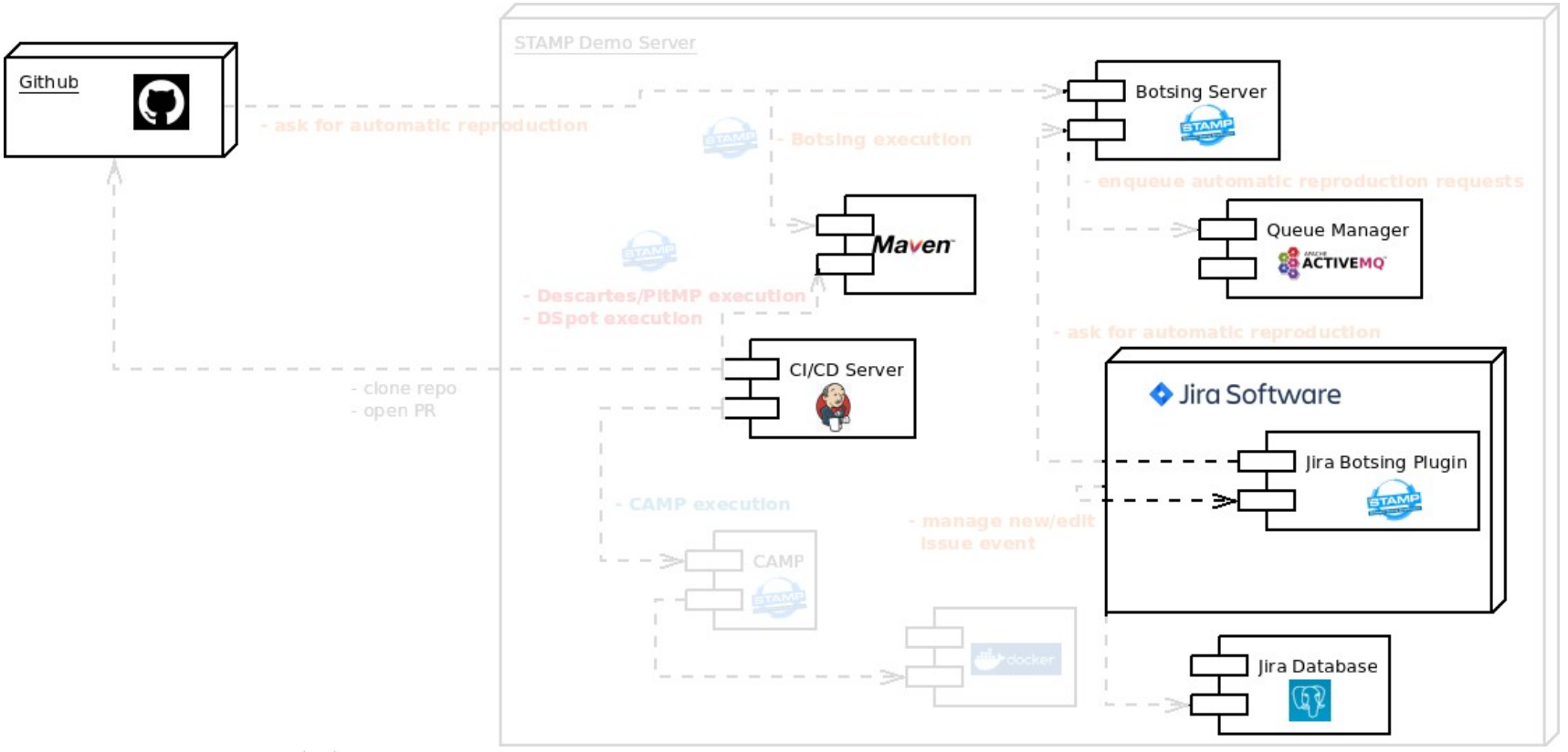
Server side: Issue trackers

- **Requirements:**
  - Keep track of all the issues and bugs
  - Manage the CI/CD process
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - https://github.com/STAMP-project/botsing/tree/master/botsing-maven
  - https://github.com/STAMP-project/botsing-gradle-plugin
  - https://github.com/STAMP-project/botsing-jira-plugin
  - https://github.com/STAMP-project/botsing-server/

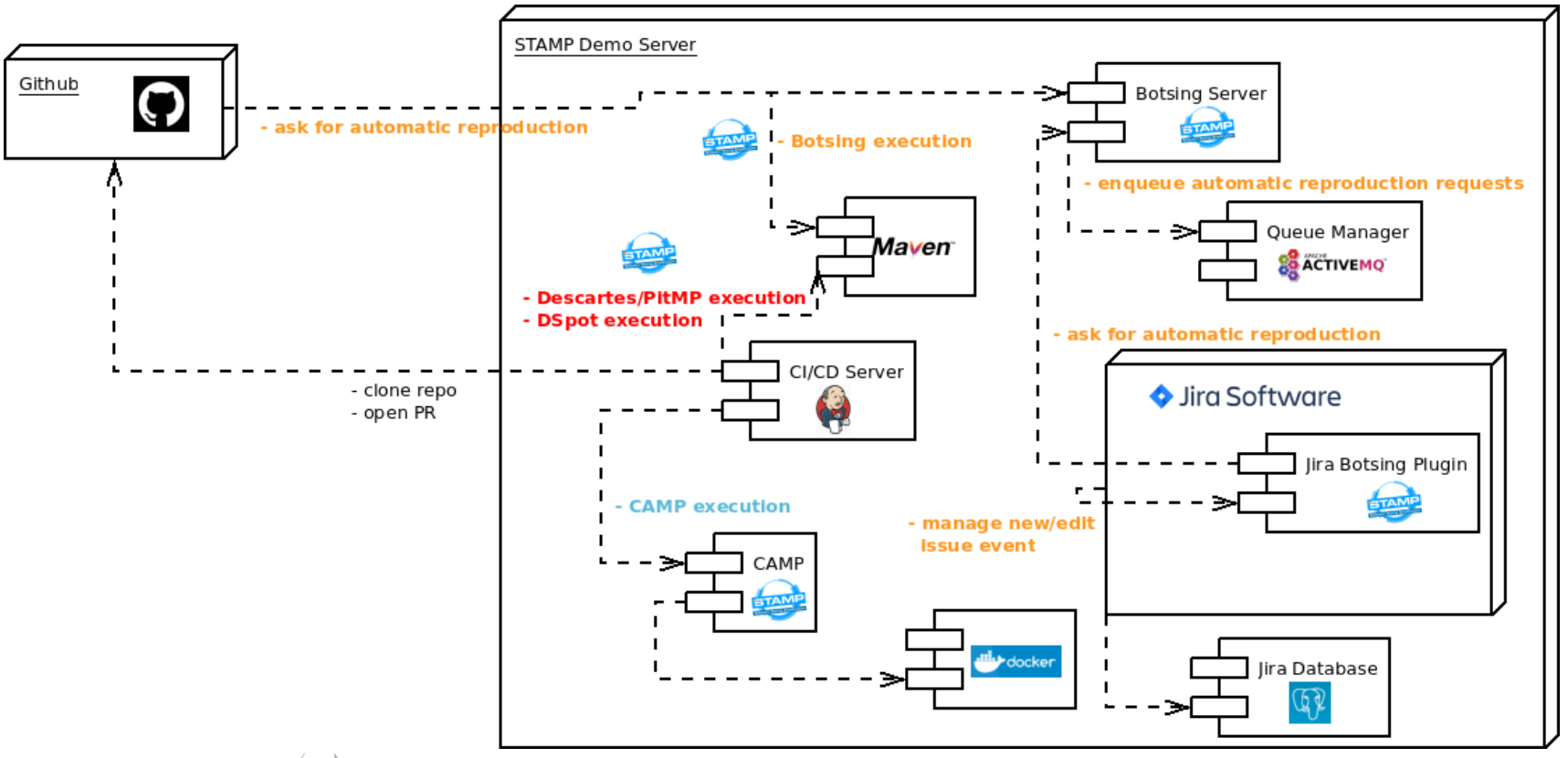
Server side: Issue trackers

- **Requirements:**
  - Keep track of all the issues and bugs
  - Manage the CI/CD process
  - solution based on a widespread tool to preserve existing processes
- **Solution**:
  - https://github.com/STAMP-project/botsing/tree/master/botsing-maven
  - https://github.com/STAMP-project/botsing-gradle-plugin
  - https://github.com/STAMP-project/botsing-jira-plugin
  - https://github.com/STAMP-project/botsing-server/
- **To support the following use cases:**

Towards STAMP Server



STAMP Server

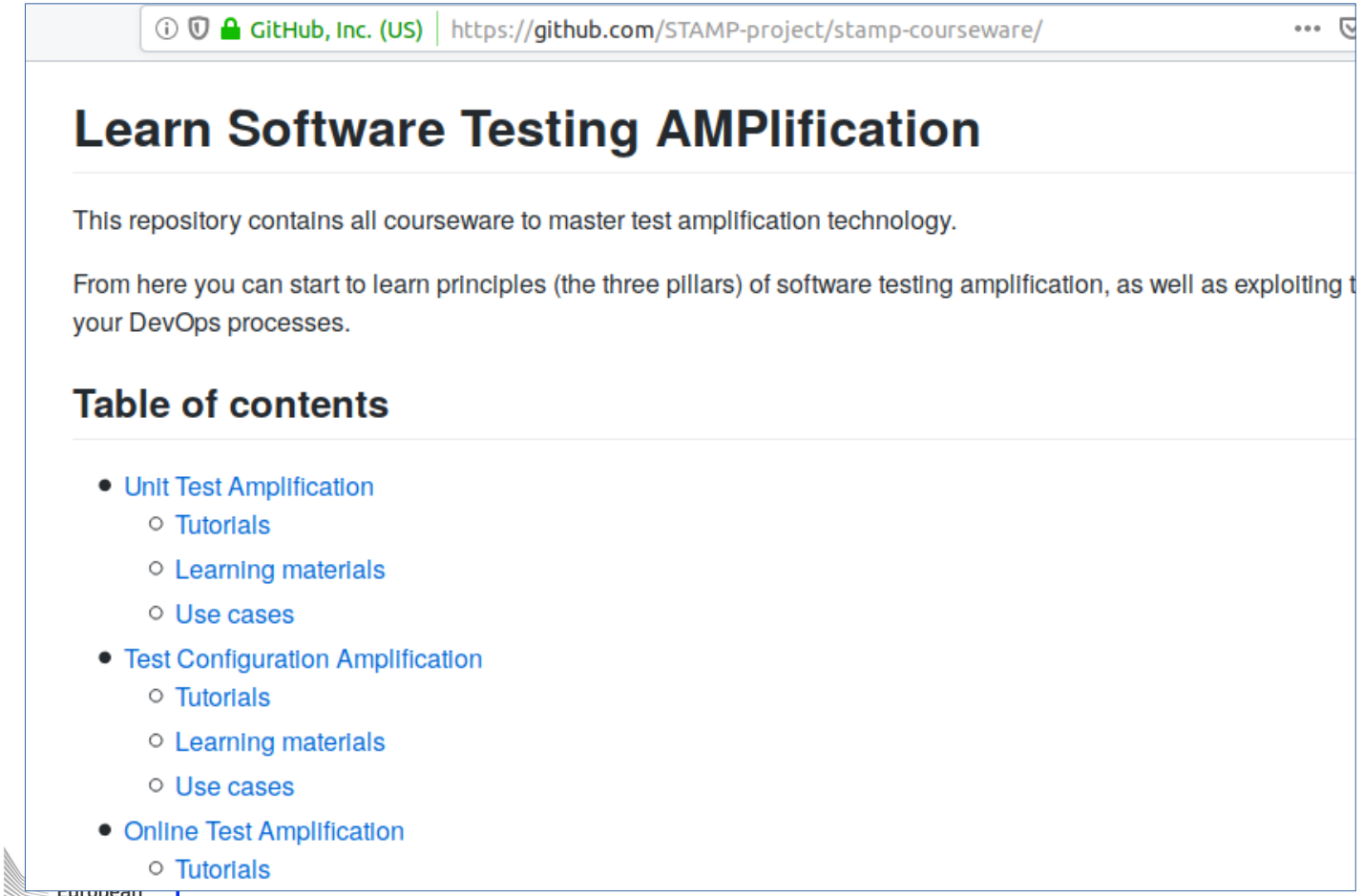


European Commission

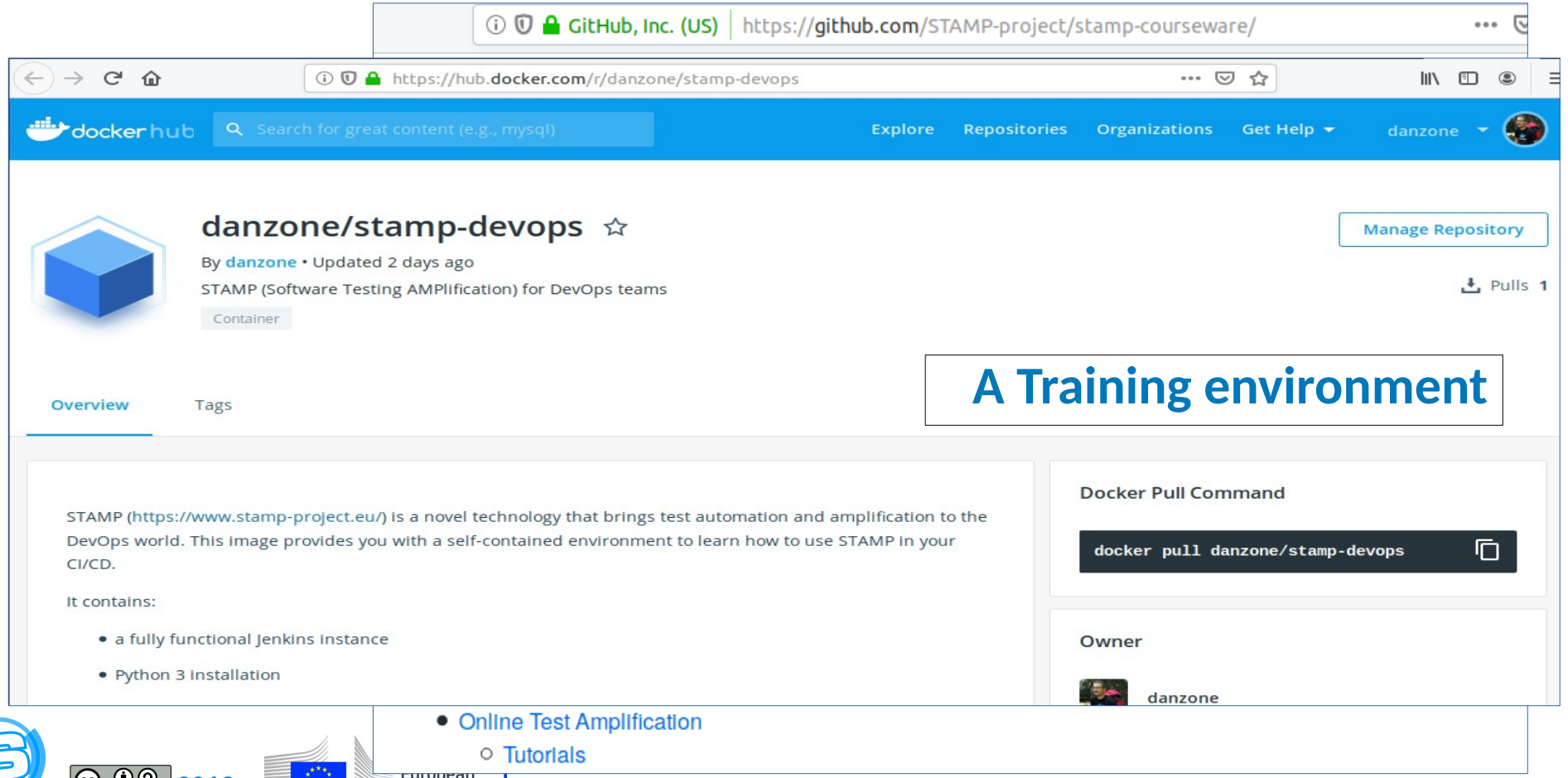
Training material

Training material

A Learning path



Training material



Training material

STAMP DevOps Docker image

This image provides you with a self-contained environment to learn how to use STAMP in your CI/CD.

It contains:

- a fully functional Jenkins instance
- Python 3 installation
- CAMP 0.6.2
- CAMP dependencies:
  - Z3 solver, version 4.7.1
  - Docker
  - Docker Compose

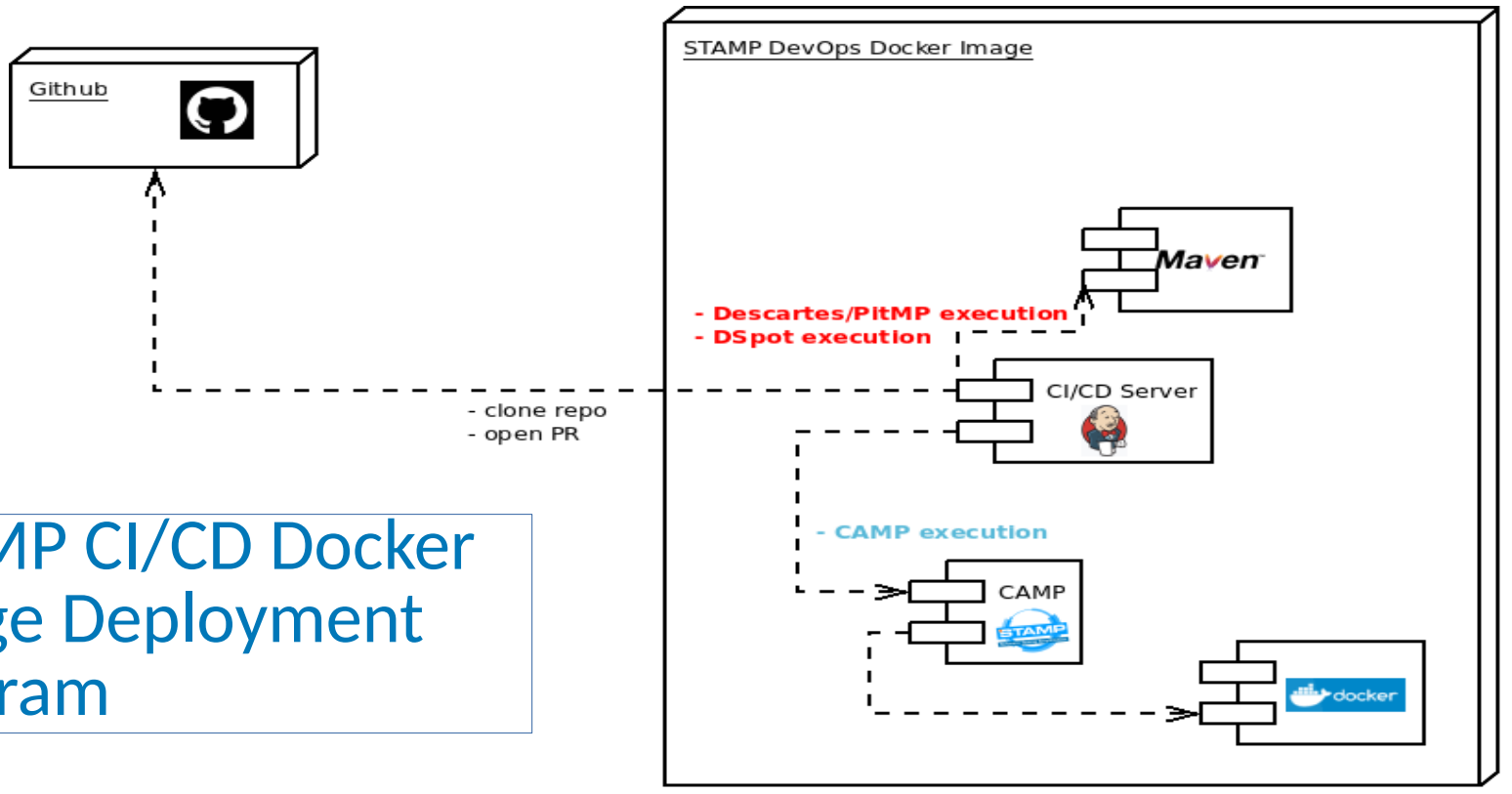
A Training environment

To run it, simply execute:

```
docker run -p 8080:8080 -p 50000:50000 -v /var/run/docker.sock:/var/run/docker.sock -v jenkins_home:/var
```

This command will download and run a Jenkins CI server in your machine, and STAMP components to perform unit test amplification and test configuration amplification within it. For more information about first steps, go here.

Training material



STAMP CI/CD Docker Image Deployment Diagram

Training material

Collaboration

- **Collaborative platform:**
  - OW2
- **DSpot and Descartes Integration:**
  - INRIA
  - KTH
  - ATOS
- **CAMP Integration:**
  - SINTEF
  - XWiki
  - ActiveEon
- **Botsing Integration:**
  - ActiveEon
  - TUD
  - OW2

WP4: Conclusions

- To start from the use cases provided two advantages
  - The integrated version of STAMP is immediately and concretely usable
  - The wide range of domains covered assures high exploitability far beyond the mere use cases
- Heterogeneous functionalities and tools successfully integrated
  - Plugins mechanism allows to interact with the most popular ALM and CI/CD tools (e.g. Jira and Jenkins)
  - A complete amplification test framework released for the whole CI/CD Process
  - The developed mechanism is potentially further extensible
- Learning material (Learning path & containerized training environment) publicly available on GitHub and DockerHub

Q&A

STAMP

Software Testing Amplification

European Commission

Thank you!

The STAMP project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 731529.