

ECSEL Research and Innovation actions (RIA)



AMASS

**Architecture-driven, Multi-concern and Seamless Assurance and
Certification of Cyber-Physical Systems**

**AMASS open source platform provisioning and
website (c)
D7.7**

Work Package:	WP7 Industrial Impact and Community Building
Dissemination level:	PU = Public
Status:	Final
Date:	21st December 2018
Responsible partner:	Gaël Blondelle (ECL)
Contact information:	gael.blondelle@eclipse-foundation.org
Document reference:	AMASS_D7.7_WP7_ECL_V1.0

PROPRIETARY RIGHTS STATEMENT

This document contains information, which is proprietary to the AMASS consortium. Permission to reproduce any content for non-commercial purposes is granted, provided that this document and the AMASS project are credited as source.

Contributors

Names	Organisation
Gaël Blondelle	Eclipse Foundation Europe GmbH (ECL)
Maria Teresa Delgado	Eclipse Foundation Europe GmbH (ECL)

Reviewers

Names	Organisation
Stefano Puri (Peer Reviewer)	Intecs (INT)
Jose Luis de la Vara (Peer Reviewer)	Universidad Carlos III de Madrid (UC3)
Cristina Martinez (Quality Manager)	Tecnalia Research & Innovation (TEC)
Barbara Gallina (Technical manager)	Mälardalen University (MDH)
Alejandra Ruiz Lopez (Project manager)	Tecnalia Research & Innovation (TEC)

TABLE OF CONTENTS

Executive Summary.....	5
1. Introduction	6
2. The AMASS Open Platform.....	7
3. Stakeholders.....	10
4. Considerations about Open Source and Security	12
4.1 No direct relationship between open source and security.....	12
4.2 The specificities of open source regarding to security.....	12
4.3 Projection in the AMASS context	13
5. Consideration about Open Source and Sustainability	15
5.1 Porting EPF Composer	15
5.2 Activity of the Open Source Community.....	16
6. Updated Version of the AMASS Open Source Platform Website	17
7. Website Content Summary and Future Work.....	21
Abbreviations	22
References.....	23
Appendix A: Updating the OpenCert Website	24
A.1 Clone the current website repository	24
A.2 Add your content.....	25
Add a Blog or News entry	27
Check your modifications	27
Push the changes.....	28
Build the website.....	29
Check online	30

List of Figures

Figure 1.	AMASS Reference Tool Architecture with responsibilities of the particular work packages.....	7
Figure 2.	AMASS Platform Tools ecosystem.....	8
Figure 3.	AMASS Tool Platform roles	10
Figure 4.	Eclipse Process Framework release.....	15
Figure 5.	PolarSys CHESS and OpenCert activity and diversity	16
Figure 6.	Banner asking for Cookies consent on the OpenCert website.....	17
Figure 7.	OpenCert website home page.....	18
Figure 8.	A new page to promote screenshots	19
Figure 9.	The Community page	20
Figure 10.	The About page	20
Figure 11.	OpenCert website directory structure	25
Figure 12.	OpenCert WWW project in Eclipse	26
Figure 13.	Index.md file content	26
Figure 14.	Hugo server window.....	27
Figure 15.	Visualize website update before commit.....	28
Figure 16.	Commit a change to the repository	29
Figure 17.	Jenkins dashboard with the OpenCert “build-and-publish-website” job	30

List of Tables

Table 1.	Matrix of assets expected on the website for each visitor profile.....	11
-----------------	--	----

Executive Summary

This report is the third version of the deliverable “AMASS open source platform provisioning and website”, released as part of task T7.3 (Building and Coordination of AMASS Open-Source Community), of AMASS WP7 (Industrial Impact and Community Building). It improves and completes deliverable D7.6 “AMASS open source platform provisioning and website (b)” [4], which was delivered in March 2018.

Since its inception, the AMASS consortium has considered the use of open source distribution mechanisms and engagement in community building activities as critical considerations for the sustainability of AMASS results and the AMASS Open Platform. This report presents the improvements made to the OpenCert website since the previous version of the report. The OpenCert website is also the place where web presence of the AMASS Open Platform is managed.

This document follows deliverables D7.3 “AMASS open source platform project proposal” [1] and D7.4 “AMASS open source platform marketing and outreach plan” [2], which describe the creation of the AMASS Open Platform constituted by the OpenCert [8], CHESS [11], EPF [14] projects and additional open source components, and presents the plan to promote the platform.

This deliverable presents how the AMASS Open Platform website is designed to address the needs of different profiles of visitors, the technology selected to implement it, its contents as of December 2018, and some future work planned for it.

1. Introduction

AMASS is creating and consolidating a *de-facto* assurance and certification open tool platform, published in open source under the PolarSys Top Level Project [19], which is dedicated to Open Source tools for Systems Engineering and Embedded Systems.

This document presents the third version of deliverable “AMASS open source platform provisioning and website”, released as part of the AMASS WP7 (Industrial Impact and Community Building). Since its first version published in June 2017 [3], the website has been updated regularly to promote new open source results. This version of the deliverable adds a section to address a recurring question about how open source relates to security in the context of AMASS.

The use of open source distribution mechanisms makes it easy for users to: (a) test an open platform before adopting it, (b) customize the platform to their specific deployment constraints. But in order to be successful, an open source project must provide not only interesting features, but also good online resources and documentation. That’s why a new area called “Resources” was introduced in D7.6 [4], that lists resources available to users and adopters of the AMASS open source platform. This website area has been added to convince website visitors to adopt the AMASS platform by providing extensive documentation and material to increase their confidence in the project and reduce their perceived risk of adopting it.

The subsequent sections of this deliverable are structured as follows:

- Chapter 2 outlines the structure of the AMASS Open Platform.
- Chapter 3 presents a matrix of online assets targeted at different profiles of website visitors.
- Chapter 4 exposes considerations about security and open source, in general, and in the specific context of AMASS.
- Chapter 5 exposes considerations about open source and sustainability, based on the contributions from AMASS partners to the EPF project.
- Chapter 6 presents the updates made to the website since the previous version (described in D7.6 [4]), discusses the rationale for them where appropriate, and includes screenshots highlighting new and improved pages on the website.
- Chapter 7 summarizes the contents of the website and lists future work planned for it.
- Appendix A provides a tutorial on how to update the website, or more specifically, how to write a change and submit it to the project team as a contribution.

2. The AMASS Open Platform

As introduced in D2.4 AMASS reference architecture (c) [6] section 2.2, the ARTA (Figure 1) specification describes how the AMASS building blocks run and operate in order to perform architecture driven, multi-concern assurance, and cross-and-intra-domain/concern assurance in an interoperable environment.

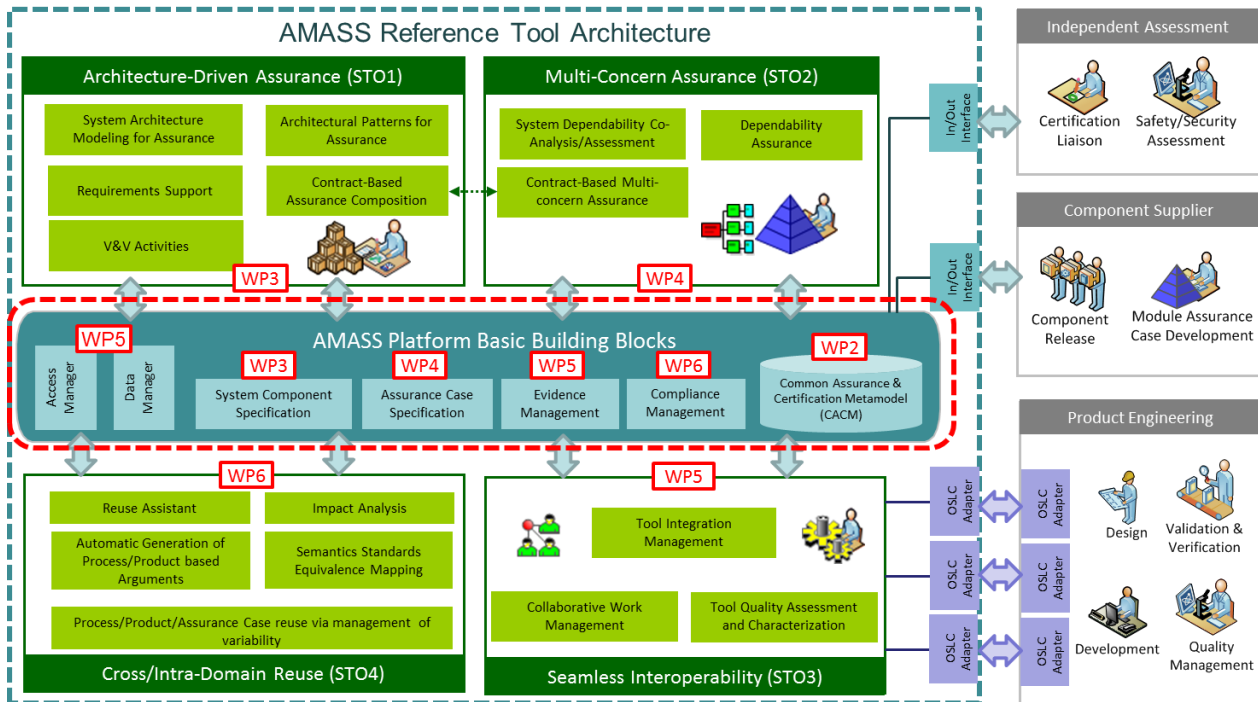


Figure 1. AMASS Reference Tool Architecture with responsibilities of the particular work packages

AMASS has defined four pillars, which correspond to the specific project Scientific and Technical Objectives (STOs) summarized below:

- **Architecture-Driven Assurance (STO1):** Explicit integration of assurance and certification activities with the CPS development activities, including specification and design. It provides support for system components composition in accordance with the domain best practices, guaranteeing that emerging behaviour does not interfere with the whole system assurance.
- **Multi-concern Assurance (STO2):** Tool-supported methodology for the development of assurance cases, co-assessment, and contract-based assurance, which addresses multiple system characteristics (mainly safety and security, but also other dependability aspects such as availability, robustness and reliability).
- **Seamless Interoperability (STO3):** Open and generically applicable approach to ensure the interoperability between the tools used in the modelling, analysis, and development of CPS, among other possible engineering activities; in particular, interoperability from an assurance and certification-specific perspective, and collaborative work among the stakeholders of the assurance and certification of CPS.
- **Cross/Intra-Domain Reuse (STO4):** Consistent assistance for intra- and cross-domain reuse, or cross-concern, based on a conceptual framework to specify and manage assurance and certification assets.

As described in D2.5 AMASS User guidance and Methodological framework [7] section 2.3, the internal and external tools integrated in and available for the AMASS Platform are shown in Figure 2, grouped along the

aforementioned STOs. The STO Seamless Interoperability is not explicitly added as it is provided by the internal platform tools.

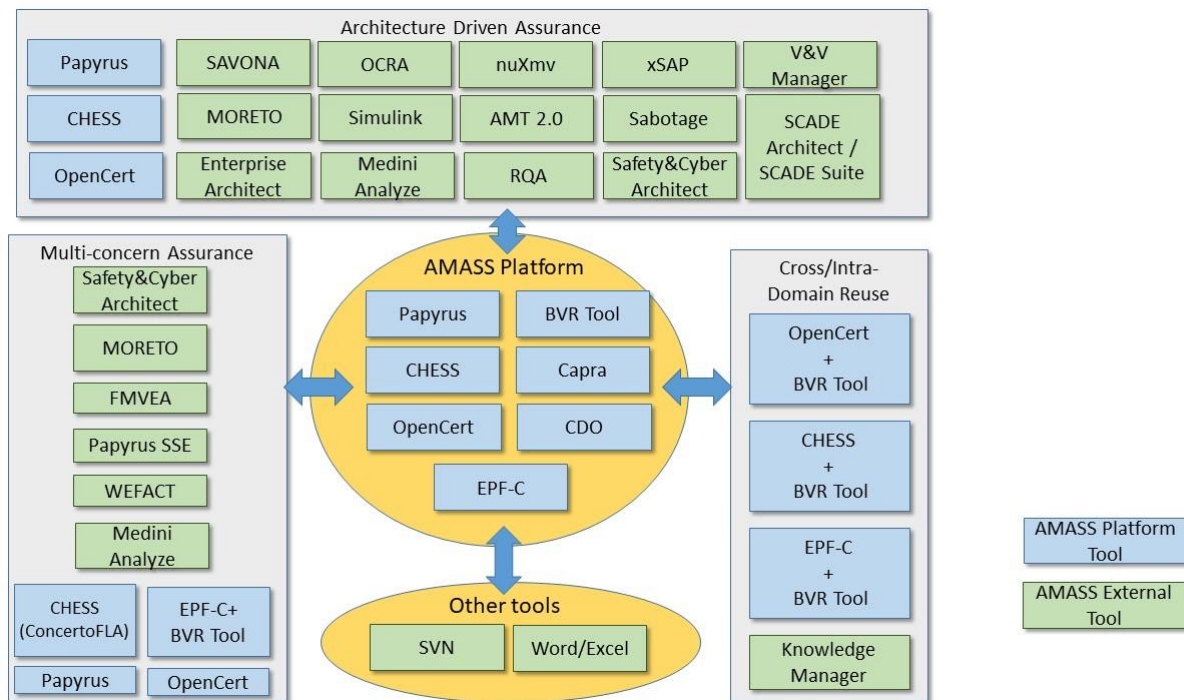


Figure 2. AMASS Platform Tools ecosystem

The internal tools are blue highlighted whereas the external ones are depicted in green. The internal tools are all available in Open Source and integrated in the AMASS Open Platform:

- **PolarSys OpenCert [8]** is the core of the AMASS Open Source platform. OpenCert, which was created by the members of the OPENCOS research project [10], supports evidence management, assurance case specification and part of the compliance management functionalities from the Basic Building Blocks. It also includes new functionalities implemented during the AMASS project.
- The **PolarSys CHES [12]** toolset, which was created by the CHES research project [11] and continued by SafeCer [13], adds support for Architecture-Driven Assurance. The CHES toolset leverages another important Eclipse project, the Papyrus [14] platform for UML (Unified Modelling Language) design and profiles, including system modelling with SysML (Systems Modelling Language).
- **EPF Composer [15]**; this pre-existing Eclipse project, created by IBM some years ago and already used in the context of SafeCer, is a key component used to describe and support the processes for Cross-domain and Intra-Domain reuse.
- **BVR Tool [16]** is a series of Eclipse plugins that implement the BVR language, a language for enabling variability management in the context of safety-critical systems engineering. The BVR Tool supports feature modelling, resolution, realization and derivation of specific family members.
- **Capra [17]** is a dedicated traceability management tool that allows the creation, management, visualisation, and analysis of trace links within Eclipse. Capra is highly configurable and is used in AMASS to allow to create and link assurance argument fragments and evidences or other traceability links.
- **CDO [18]** is both a development-time model repository and a run-time persistence framework. It is used to store all the data in AMASS.

Whereas each Eclipse project has its own lifecycle according to the Eclipse Development Process and its own website, the advantage of the AMASS Open Platform is to synchronize several different projects and contributions in consistent versions that can be downloaded from the Polarsys OpenCert Tools Platform website [9].

3. Stakeholders

Before designing the website, the matrix shown in Table 1 was created to identify the expectations of the different stakeholders visiting the website.

The profiles of AMASS users can be groups as shown in Figure 3, and described in detail in D2.5 User guidance and methodological framework [7], section 3.1.

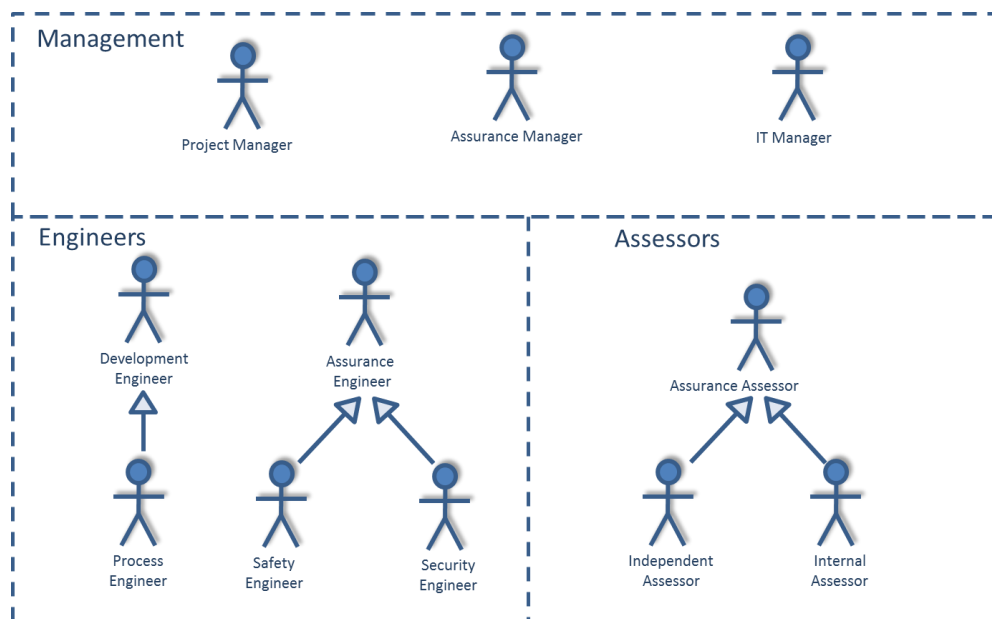


Figure 3. AMASS Tool Platform roles

Stakeholders of the Open Source community in general were identified in deliverable D7.4 AMASS open source platform marketing and outreach plan [2], section 2. Here is a short description of those stakeholders:

- Tools Architects work for large organisations, typically in the methods and tools department, to create customized tool suites for their R&D teams.
- Eclipse Modelling experts are a powerful lever to foster the usage of the AMASS Open Platform as they can promote it to their partners and customers. They are also likely to integrate the AMASS Open Platform in a larger solution.
- AMASS open platform developers participate to the development of the AMASS platform itself. As insiders they expect to find slightly different information on the website.
- Researchers may visit the website to figure out if the AMASS Open Platform is a good framework on which they can build their own research.
- Journalists usually expect to quickly understand the goal of a project, and to find news that they can transform in articles.
- Industry / Decision makers are not likely to use the platform but may visit the website to figure out why their teams want to use it, and if it is reliable. They are likely to be the most interested in use cases and success stories.
- Open Source Contributors can help fix some issues, documentation, ...

Table 1. Matrix of assets expected on the website for each visitor profile

Profiles/Asset	Download	General information	News / Press releases	Snapshots / Videos	Code	Getting Started	Use cases / Success Story	User documentation	Reference / API documentation
AMASS profiles									
Manager	X	X	X	X		X	X	X	
Engineers	X	X	X	X		X		X	X
Assessor	X	X		X		X	X	X	
Tool providers profiles									
Tools Architect	X	X	X	X	X	X	X	X	X
Eclipse Modelling expert	X	X	X	X	X	X	X	X	X
AMASS open platform developer	X	X			X	X			X
General profiles									
Researcher	X	X		X	X	X		X	X
Journalist		X	X	X			X		
Industry / Decision maker		X	X	X			X	X	
Open Source Contributor	X	X			X	X		X	X

4. Considerations about Open Source and Security

The definition of Open Source implies that source code is distributed under a license in which the copyright holder grants the users the power to access, modify and re-distribute the software to anyone and for any purpose [20][21] and thus is developed under an open collaborative model.

Throughout the years, the Open Source Software (OSS) development model has gained more and more popularity around the globe. Nowadays, open source components are the core building blocks of application software, providing developers with an ever-growing offer of off-the-shelf possibilities that they can use for assembling their products faster and more efficiently [22].

The AMASS project is no exception to this reality and the AMASS Open Platform not only leverages existing open source components, but also publishes new open source components to the community. As AMASS is addressing a wide universe of application areas while implementing an open collaboration model to develop its technology solutions, it is not surprising that the community would express its concern on the platform security aspects.

4.1 No direct relationship between open source and security

In order to address these concerns, we need to keep in mind a few things, starting with the fact that the OSS movement was not designed with security in mind: OSS is all about open collaboration and open innovation. However, there is a famous law in the Open Source community, called the Linus' law, which states the following: "given enough eyeballs, all bugs are shallow".

In this perspective, developers believe that they receive better protection — and often trust — from opening their code up to the crowd to inspect and offer fixes, than from hiding behind the high walls of a closed system. Nonetheless, since the time when Linus Torvalds wrote this law, the OSS community has evolved, and the vast majority of users are only downloading the resources of interest without actually reviewing the source code itself. This means that the number of users is far greater than the number of eyeballs reviewing the code [22]. Thus, having the source code available for scrutiny could be either a good or a bad thing, depending on the size of the community and the user perspective.

Other sources have studied the correlation between security and open source in a more structured way. For example, let's consider the work published by Dr. Guido Schryen, professor of Information Systems Research at the University of Regensburg, who performed a thorough literature review on security aspects in open source vs. closed source software [23]. He concluded that the discussion is often biased depending on the position of the author regarding open source, and also that the literature was lacking appropriate metrics, a common methodology and hard data. Software bugs that can be used by attackers to gain access to a system or network are commonly referred to as vulnerabilities¹. Schryen's work analyses and compares published vulnerabilities of a set of eight open source software and nine closed source software packages, all of which are widely deployed. Through an empirical analysis, his investigation reveals that (a) the mean time between vulnerability disclosures was lower for open source software in half of the cases, while the other cases showed no differences, (b) 14 out of 17 software packages showed a significant linear correlation between the time and the number of published vulnerabilities, and (c) no significant differences in the severity of vulnerabilities were found between open source and closed source software.

4.2 The specificities of open source regarding to security

Security wise, the main concern remains the surface of exposure of software code: all the different points where an unauthorized party could try to inject or extract data, which is of course bigger for publicly

¹ According to the U.S. MITRE corporation (<https://www.mitre.org/>)

available source code, as is the case of OSS. The openness in OSS makes it easier for both the good guys and the bad guys to find vulnerabilities in the code, since it is available for anyone to review (and to fix!).

However, more closed models, implementing a “security through obscurity” approach are not necessarily better. Security is a holistic concept, not only depending on the final product or result, but also linked to the creation and maintenance process. Based on this approach and following the “many-eyeballs” principle, open source has the potential to be better than closed source software in terms of security vulnerabilities being available for public scrutiny and accepting security fixes that could be suggested by anyone among a large community of users. But simply being open is not a guarantee of security [24]. Over the past few years a few examples have made this clear for the OSS community, as are the cases of: (a) the Heartbleed bug², which put the spotlight on OpenSSL, the security toolkit used by many of the internet's largest sites, maintained primarily by two men who had never met in person, (b) the Equifax breach⁴ that exposed sensitive data for as many as 143 million U.S. consumers, accomplished by exploiting a web application vulnerability that had been patched more than two months earlier (a House Oversight Committee report⁵ released in December 2018, found the Equifax Inc. hack preventable, citing the credit bureau's failure to “implement an adequate security program to protect this sensitive data.”)⁶, and (c) the Apache Struts 2 flaw uncovered recently, which promises to be even more critical than the Equifax Bug⁷, a critical remote code-execution vulnerability in the popular open-source framework for developing web applications in the Java programming language which exploitation could lead to full endpoint and network compromise.

On the other side, if you need to think about security breaches in proprietary solutions or closed source software, just think about the Microsoft security breaches that we were never told about, - for example when Microsoft responded quietly to a detected secret database hack in 2013⁸. Security in the 21st century has proven to have suffered enough breaches both in the open and closed software worlds⁹.

4.3 Projection in the AMASS context

To come back to AMASS, how do these considerations about security apply to the AMASS open platform? Both the project partners and early adopters agree that security is crucial in all tools, systems and platforms, and of course AMASS is no exception. However, the previous section makes clear that security and openness are two orthogonal issues. Moreover, OSS is potentially better -security wise- due to the public availability of source code that enables inspections of the tool.

AMASS is about assurance and certification and the tool can be used in several domains to improve the system efficiency and to assist in the assurance and certification process, but AMASS is not the core of a CPS, and thus the AMASS tool platform by itself is not a liability for the developer of the CPS: the AMASS open platform is instantiated in the context of a global certification and assurance process. In this context, the global certification process must address security, by taking into account that the AMASS open platform may not be fully secure and adding some additional security constraints around the open source

² <http://heartbleed.com/>

³ <https://www.buzzfeed.com/chrisstokelwalker/the-internet-is-being-protected-by-two-guys-named-st>

⁴ <https://arstechnica.com/information-technology/2017/09/massive-equifax-breach-caused-by-failure-to-patch-two-month-old-bug/>

⁵ <https://oversight.house.gov/wp-content/uploads/2018/12/Equifax-Report.pdf>

⁶ <https://blogs.wsj.com/cio/2018/12/11/the-morning-download-house-equifax-report-cites-faulty-it-structure/>

⁷ <https://threatpost.com/apache-struts-2-flaw-uncovered-more-critical-than-equifax-bug/136850/>

⁸ <https://www.reuters.com/article/us-microsoft-cyber-insight/exclusive-microsoft-responded-quietly-after-detecting-secret-database-hack-in-2013-idUSKBN1CM0DO>

⁹ <https://www.csoonline.com/article/2130877/data-breach/the-biggest-data-breaches-of-the-21st-century.html>

platform. For example, if access to the application could be hacked too easily, the decision should be taken to isolate the deployed AMASS platform on a separate network, available only on some specific premises that benefit from additional physical security.

In addition to that, in the context of the Eclipse ecosystem, some specific efforts could be put in place to reinforce the security of the AMASS open platform. Some of these aspects are already covered by the Eclipse Development Process, through the traceability of the code published in open source: The Eclipse IP process tracks the provenance of each contribution as well as the provenance of each dependency, recursively. Also, the Eclipse Development Process requires that the binaries would be signed during the release process.

To go further, the AMASS project partners could decide to use the AMASS open platform for the process assurance of the AMASS platform itself, for traceability, and could also proceed to the inspection of some parts of the code.

Moreover, the AMASS open platform is supposed to be embedded in a larger environment: a proprietary product or a specific deployment by a large organisation. In both cases, additional measures can be integrated to ensure the security of the platform.

On a technical note, the data managed by the AMASS platform are stored in a database through CDO [18]. The data in a CDO repository can be secured through some Role Based Access Control approach. When the AMASS Platform is deployed in a company in order to be used, the data and the server become part of the company infrastructure and should be protected in a similar way to other assets (databases, web applications...) of the company infrastructure. Similar access policies should be applied to the AMASS database than to other company databases (where company IP data is stored) so data privacy and consistency are consistently implemented.

Finally, the AMASS open platform should be deployed in the context of a complete assurance and certification process that integrates the tools into a global process. Such a global process should consider the security risks related to the tools in order to effectively mitigate them.

5. Consideration about Open Source and Sustainability

5.1 Porting EPF Composer

During the second half of 2017, Mälardalen University (MDH), with the help of the Eclipse Foundation, started an important contribution to the Eclipse Process Framework project and more specifically to EPF Composer. The project was under maintenance since 2013, with no new official release since then, and still running on an old version of the Eclipse platform.

Due to this too old version of the Eclipse platform, it was not possible to properly integrate EPF Composer in the AMASS Open Platform core prototype.

MDH stepped up to fix this, and with the help of the Eclipse Foundation, the AMASS partners started to collaborate with the IBM team which was maintaining EPF Composer.

After initial contacts, MDH developed a patch to port EPF to Eclipse Neon. This patch was then tested by the EPF development team before integration.

Later, as MDH designated developer, Muhammad Atif Javed, had proved his ability to contribute to the project, he was elected as a new committer of the EPF project in conformance with the principles of meritocracy promoted by the Eclipse Development Process.

Then, the EPF project team collaborated to create a new release, which is the one integrated in the AMASS Open Platform.



Name	Date
1.5.2	2018-04-28
1.5.1.6	2013-11-01

Figure 4. Eclipse Process Framework release

As presented during the EclipseCon France [31], this is a good example of how open source in general, and in particular source code hosted by a foundation with a clear development process like the Eclipse Development Process, is an advantage for sustainability. In a non-open-source context, it is likely that EPF Composer could not be integrated in the AMASS Open Platform because a software vendor would have probably decided against such a technical migration for a research project. At Eclipse, the EPF project is open and meritocratic, having given the developers the opportunity to lead the porting to a new version of the Eclipse platform, and recognizing his contribution by electing him as a committer of the project.

5.2 Activity of the Open Source Community

One important criterion to assess the sustainability of an open source project is to observe the diversity of the community of developers and the activity of those developers.

Both PolarSys CHESS and PolarSys OpenCert, which are two important components of the AMASS Open Platform, show a good level of activity, as well as a good diversity of active developers thanks to the tasks executed in the context of the AMASS project.

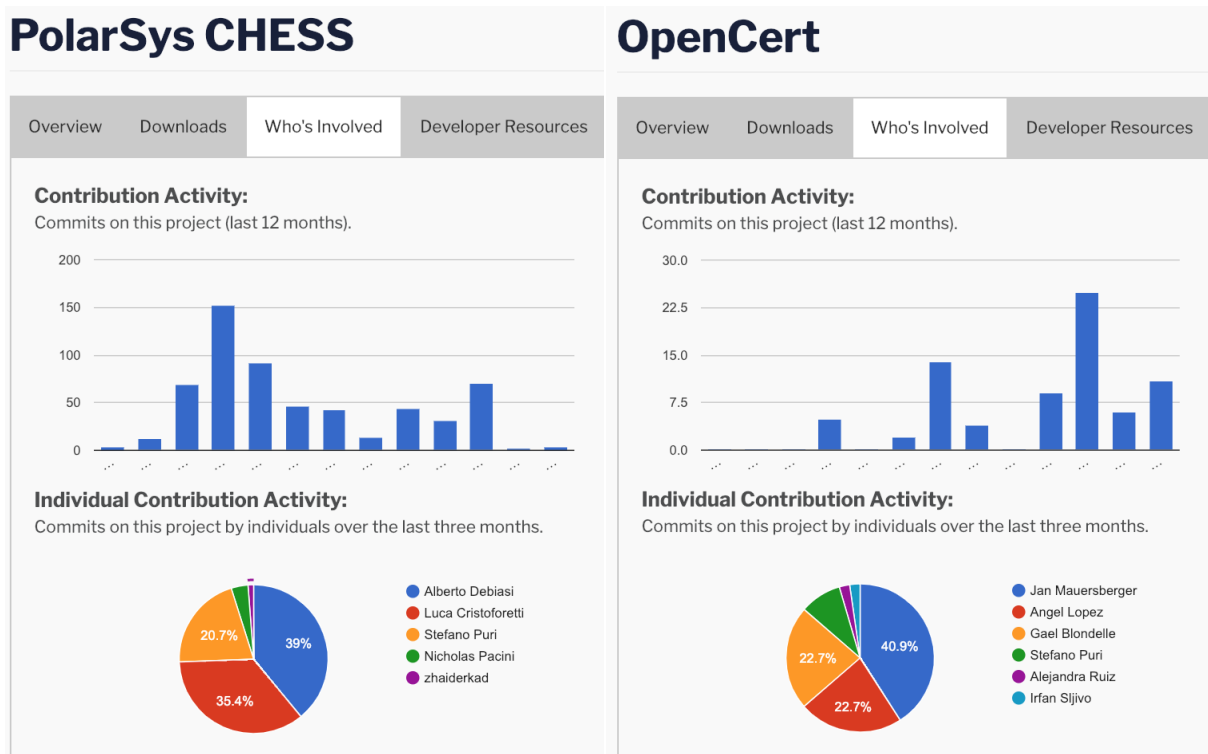


Figure 5. PolarSys CHESS and OpenCert activity and diversity

Since the initial release of the website (described in D7.5 [3]), significant effort has been made to comply with the Eclipse Development Process for the Polarsys OpenCert Tools Platform. In particular, the project partners have managed the elections of new committers for the PolarSys OpenCert and PolarSys CHESS projects and those statistics show that the project development happens in an open and transparent manner. As we publish this deliverable, the OpenCert project team is launching a new election to replace Huascar Espinoza as the OpenCert project leader due to a change in his working assignments and responsibilities. That's a good sign for an open source project when such things happen as it shows the sustainability of the project.

6. Updated Version of the AMASS Open Source Platform Website

The project team selected to use Hugo [25] to create and maintain the AMASS Open Source platform website. Hugo is a command line tool that can create static websites based on configuration files, a theme, and content written in the “Markdown” mark-up language [27].

Hugo has proven to be efficient in other contexts at Eclipse, including the Eclipse IoT Working group [26] and several other Eclipse projects, whose websites are managed with this website generator technology.

One of the main advantages of Hugo is that it is fully consistent with the Eclipse development process, and the Eclipse Foundation provides a default template that looks professional and makes it easier to create a website for a project.

For the publication of deliverable D7.6 [4], the AMASS partners implemented a more automated mechanism to update the website, which uses a Jenkins job [28] to build the website from “source files” stored in an OpenCert git repository [29]. The website source files are reviewed using Gerrit [30], which is the standard tool used by most of the Eclipse projects for code review. Thus, every partner can submit improvements to the website.

One of the new features implemented since D7.6 [4] is the support of the GDPR on the Polarsys OpenCert Tools Platform website. The website has been adapted to fully comply with the requirements to ask for user consent before using any tracking mechanism like Cookies.

Figure 6 shows the banner asking for cookie consent. The website also benefits for the general support of cookies selection as provided by the Eclipse Foundation.

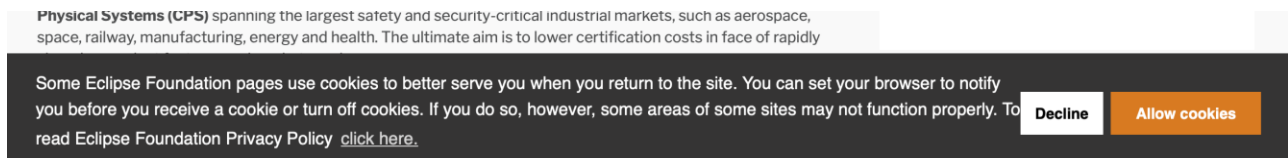
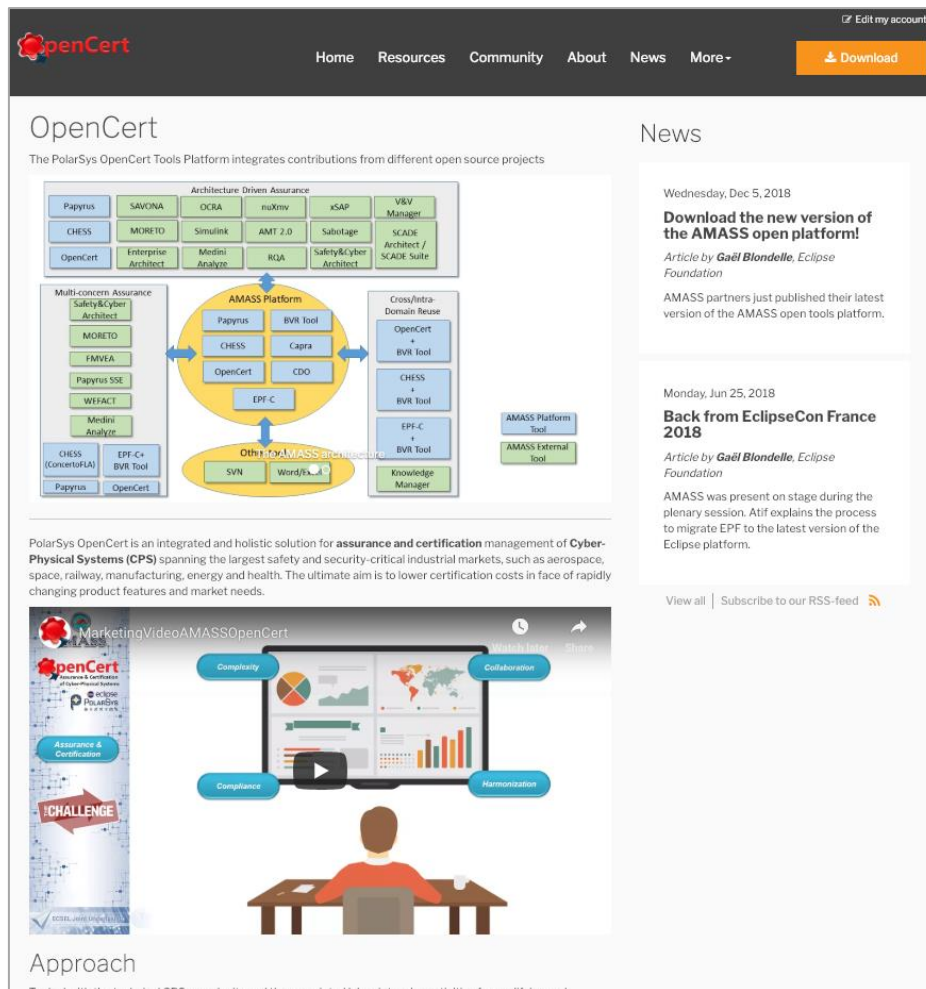


Figure 6. Banner asking for Cookies consent on the OpenCert website

This feature impacts the analytics of the project as only users who consent to the usage of cookies get tracked through Google Analytics.

The website also links to the Eclipse Foundation Website Privacy Policy [32] that describes all the actions taken by the Eclipse Foundation to ensure the privacy of website visitors.

The homepage of the Polarsys OpenCert Tools Platform website (<https://www.polarsys.org/opencert/>) has been improved and now shows a carousel that displays not only the general architecture of the AMASS platform, but also snapshots of the AMASS Open Platform (see Figure 7).



OpenCert
The PolarSys OpenCert Tools Platform integrates contributions from different open source projects

News

Wednesday, Dec 5, 2018
Download the new version of the AMASS open platform!
Article by *Gaël Blondelle*, Eclipse Foundation
AMASS partners just published their latest version of the AMASS open tools platform.

Monday, Jun 25, 2018
Back from EclipseCon France 2018
Article by *Gaël Blondelle*, Eclipse Foundation
AMASS was present on stage during the plenary session. Atif explains the process to migrate EPF to the latest version of the Eclipse platform.

View all | Subscribe to our RSS-feed

Approach
To deal with the technical CSE complexity and the associated labor-intensive activities for qualification and

Figure 7. OpenCert website home page

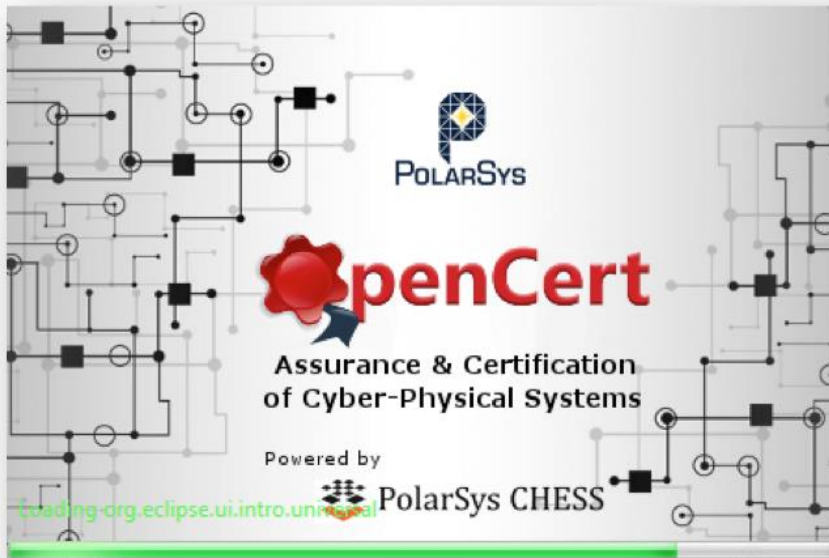
The Resources section (Figure 8) has been updated to link to the latest versions of the documentation, and to add a section with screenshots of the different tools. The presentation and the description of those screenshots will be improved in the coming period.

OpenCert / Resources / Screenshots

Screenshots

This page shows some screenshots that illustrate the AMASS Open Platform in action.

Starting the AMASS Tools Platform



The big picture

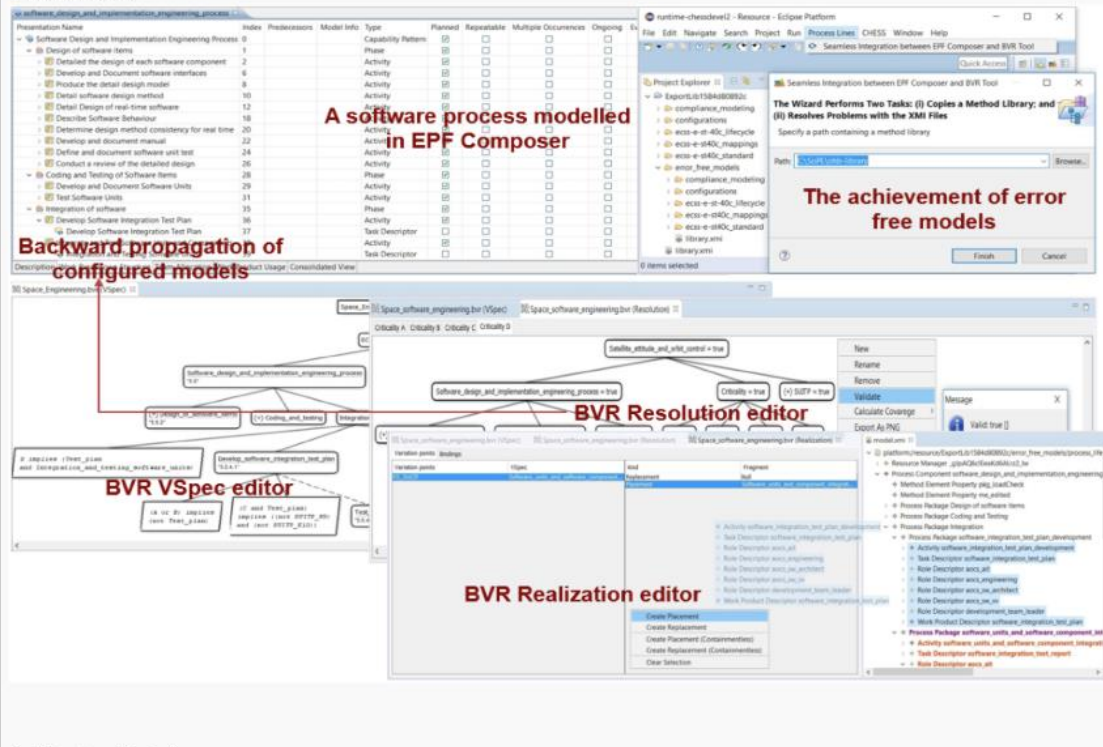


Figure 8. A new page to promote screenshots

New details have been added to the Community page (Figure 9), including links to the AMASS channels on the different social networks.

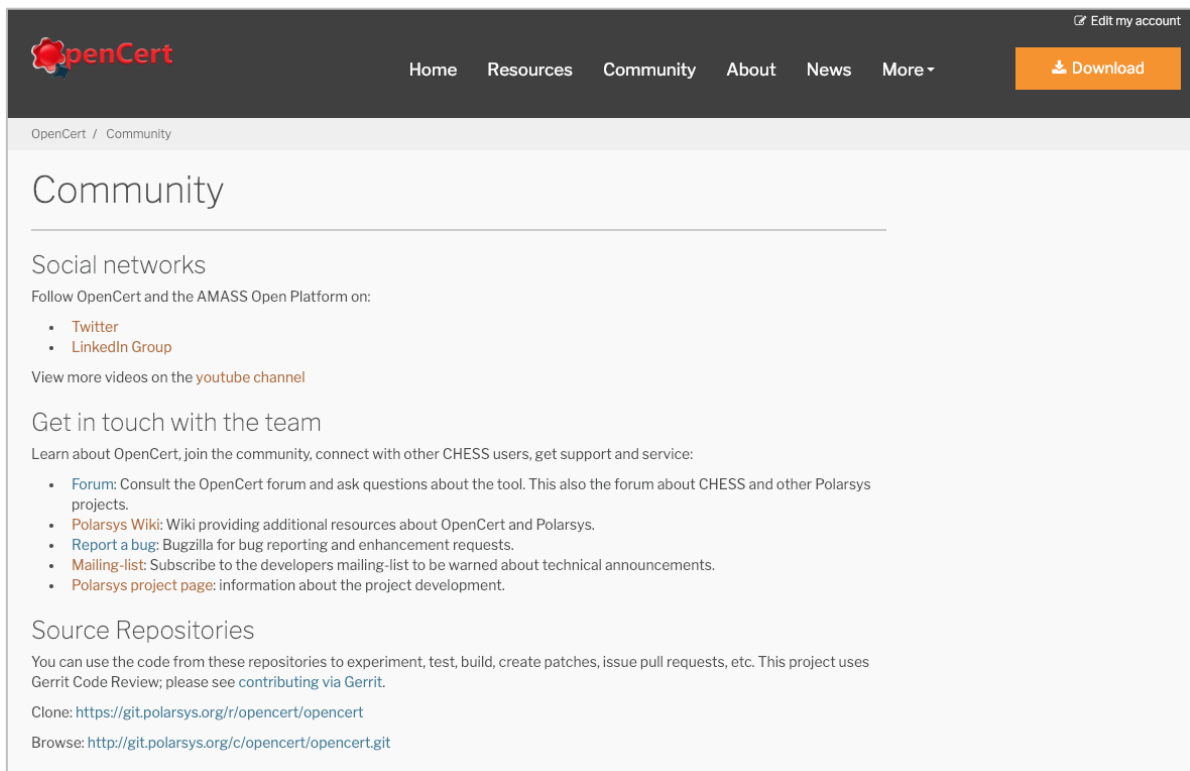


Figure 9. The Community page

The About section (Figure 10) has also been improved with the inclusion of additional Open Source components that contribute to the AMASS Open Platform.

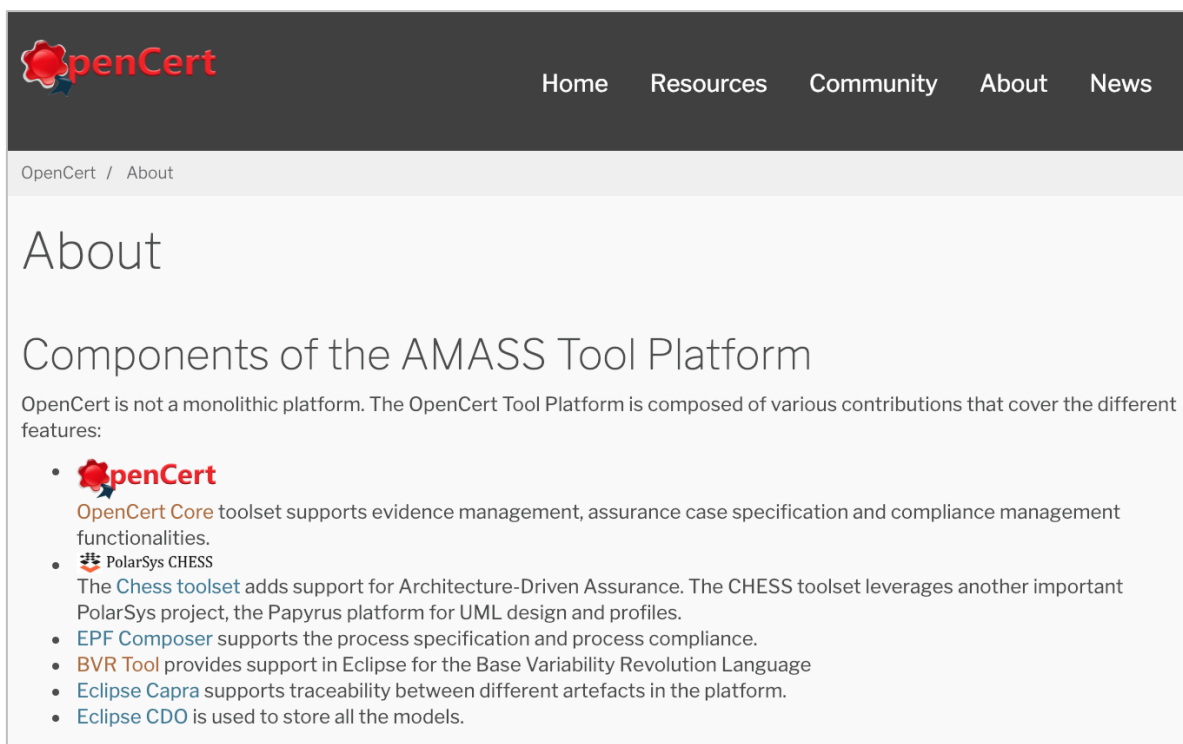


Figure 10. The About page

7. Website Content Summary and Future Work

The current version of the AMASS Open Platform website (<https://www.polarsys.org/opencert/>) provides:

- General information about the Polarsys OpenCert Tools Platform
- Downloads of the Polarsys OpenCert Tools Platform package (including OpenCert, CHESS, Eclipse Process Framework Composer and the other integrated open source tools)
- Support for news and blog posts
- Link to Source Code and downloads
- Getting Started documentation
- User's documentation
- Developer's documentation
- Training material linking to videos from the Polarsys OpenCert Tools Platform YouTube channel
- Snapshots

More content will be added to the “Resources” page and the “News” page by the end of the project to support AMASS newcomers, users and adopters in evaluating the last AMASS Open Platform prototype. In particular, we plan to publish some information about the AMASS Case studies and their usage of the AMASS Open Platform.

Abbreviations

AMASS	Architecture-driven, Multi-concern and Seamless Assurance and Certification of Cyber-Physical Systems
ARTA	AMASS Reference Tool Architecture
BVR	Base Variability Resolution
BVR-T	BVR Tool
CDO	Connected Data Objects
CHESS	Composition with Guarantees for High-integrity Embedded Software Components Assembly
CPS	Cyber-Physical System
EPF	Eclipse Process Framework
GDPR	General Data Privacy Regulation
HTML	HyperText Markup Language
IoT	Internet Of Things
IP	Intellectual Property
OpenSSL	Open Secure Sockets Layer
OSI	Open Source Initiative
OSS	Open Source Software
RSS	Really Simple Syndication
SysML	Systems Modelling Language
UML	Unified Modelling Language
WP	WorkPackage

References

- [1] [AMASS D7.3 AMASS open source platform project proposal](#), January 2017
- [2] [AMASS D7.4 AMASS open source platform marketing and outreach plan](#), April 2017
- [3] [AMASS D7.5 AMASS open source platform provisioning and website \(a\)](#), July 2017
- [4] [AMASS D7.6 AMASS open source platform provisioning and website \(b\)](#), March 2018
- [5] AMASS ECSEL project website <http://amass-ecsel.eu/>
- [6] [AMASS D2.4 Reference architecture \(c\)](#), June 2018
- [7] [AMASS D2.5 AMASS User guidance and Methodological framework](#), November 2018
- [8] OpenCert project page <http://www.polarsys.org/projects/polarsys.opencert>
- [9] Polarsys OpenCert Tools Platform website <http://www.polarsys.org/opencert/>
- [10] OPENCROSS research project <http://opencross-project.eu/>
- [11] CHESS research project <http://www.chess-project.org/>
- [12] PolarSys CHESS project <http://www.polarsys.org/chess/start.html>
- [13] SafeCer project (Certification of Software-intensive Systems with Reusable Components) http://cordis.europa.eu/project/rcn/103721_en.html and http://cordis.europa.eu/project/rcn/105610_en.html
- [14] Papyrus project <http://www.eclipse.org/papyrus/>
- [15] Eclipse Process Framework Project (EPF) <http://eclipse.org/epf/>
- [16] BVR Tool <http://www.amass-ecsel.eu/content/bvr-tool-amass>
- [17] Capra traceability framework <http://projects.eclipse.org/projects/modeling.capra>
- [18] CDO – Connected Data Objects <http://www.eclipse.org/cdo/>
- [19] PolarSys website <http://www.polarsys.org/>
- [20] St. Laurent, Andrew M. "Understanding Open Source and Free Software Licensing" (2008). O'Reilly Media. p. 4. ISBN 9780596553951
- [21] "The Open Source Definition (Annotated)." History of the OSI | Open Source Initiative, opensource.org/osd-annotated, <https://opensource.org/osd-annotated>
- [22] "The Ultimate Guide to Open Source Security." WhiteSource, resources.whitesourcesoftware.com/white-papers/the-complete-guide-on-open-source-security, <https://resources.whitesourcesoftware.com/white-papers/the-complete-guide-on-open-source-security>
- [23] Schryen, Guido, "Security of Open Source and Closed Source Software: An Empirical Comparison of Published Vulnerabilities" (2009). AMCIS 2009 Proceedings. 387, <https://aisel.aisnet.org/amcis2009/387>
- [24] Wide Open Source <https://www.securityfocus.com/news/19>
- [25] Hugo website generator <http://gohugo.io/>
- [26] Eclipse IoT Working Group <https://iot.eclipse.org/>
- [27] Markdown markup language <https://fr.wikipedia.org/wiki/Markdown>
- [28] OpenCert Jenkins job: <https://ci.polarsys.org/opencert/>
- [29] OpenCert website Git repository: <http://git.polarsys.org/c/opencert/www.git/>
- [30] OpenCert Gerrit review server: <https://git.polarsys.org/r/#/q/project:opencert/www>
- [31] M. A. Javed and B. Gallina. Get EPF Composer back to the future: A trip from Galileo to Photon after 11 years. EclipseCon, Toulouse, France, June 13-14, 2018.
- [32] Eclipse Foundation Website Privacy Policy <https://www.eclipse.org/legal/privacy.php>

Appendix A: Updating the OpenCert Website

This part is still valid. It has not been updated but is added here for convenience and completeness.

This appendix presents a tutorial that explains how to add or modify the content of the Polarsys OpenCert Tools Platform website.

To update the website, you must create content using the “Markdown” markup language and configure Hugo to build the new content. Then, you upload the generated website to the Git repository on the PolarSys forge. Finally, the generated website is pushed to the web server thanks to an automatic script.

Before editing any content for the OpenCert website, you must have the following tools installed on your computer:

- A command line terminal
- Git ([installing git](#))
- Hugo ([installing Hugo](#))
- A text editor, for “.md” files (Markdown content files) and “.toml” files (configuration files)

You may also find the following resources useful:

- [Markdown Reference Guide](#) – Markdown is used to edit content on the website.
- [Hugo Quickstart guide](#) - Hugo is the website engine used to generate static web content.
- [Gerrit documentation](#) - On the Eclipse and PolarSys forge, Gerrit is used to manage code submission and peer review. While using Gerrit, you will need to:
 - [Upload a change](#)
 - [Sign off](#) commits
 - Know your [http account / password](#)

A.1 Clone the current website repository

The website’s source files are hosted in a Git repository on the PolarSys forge. To add content, or modify content, you must first clone it on your hard drive to work locally.

```
git clone https://userid@git.polarsys.org/r/a/opencert/www opencert-www
```

This is the repository where all the Hugo source code is stored. Go to the “hugo” folder and start to use the Hugo website engine.

```
cd hugo
```

Let's have a look at the structure:

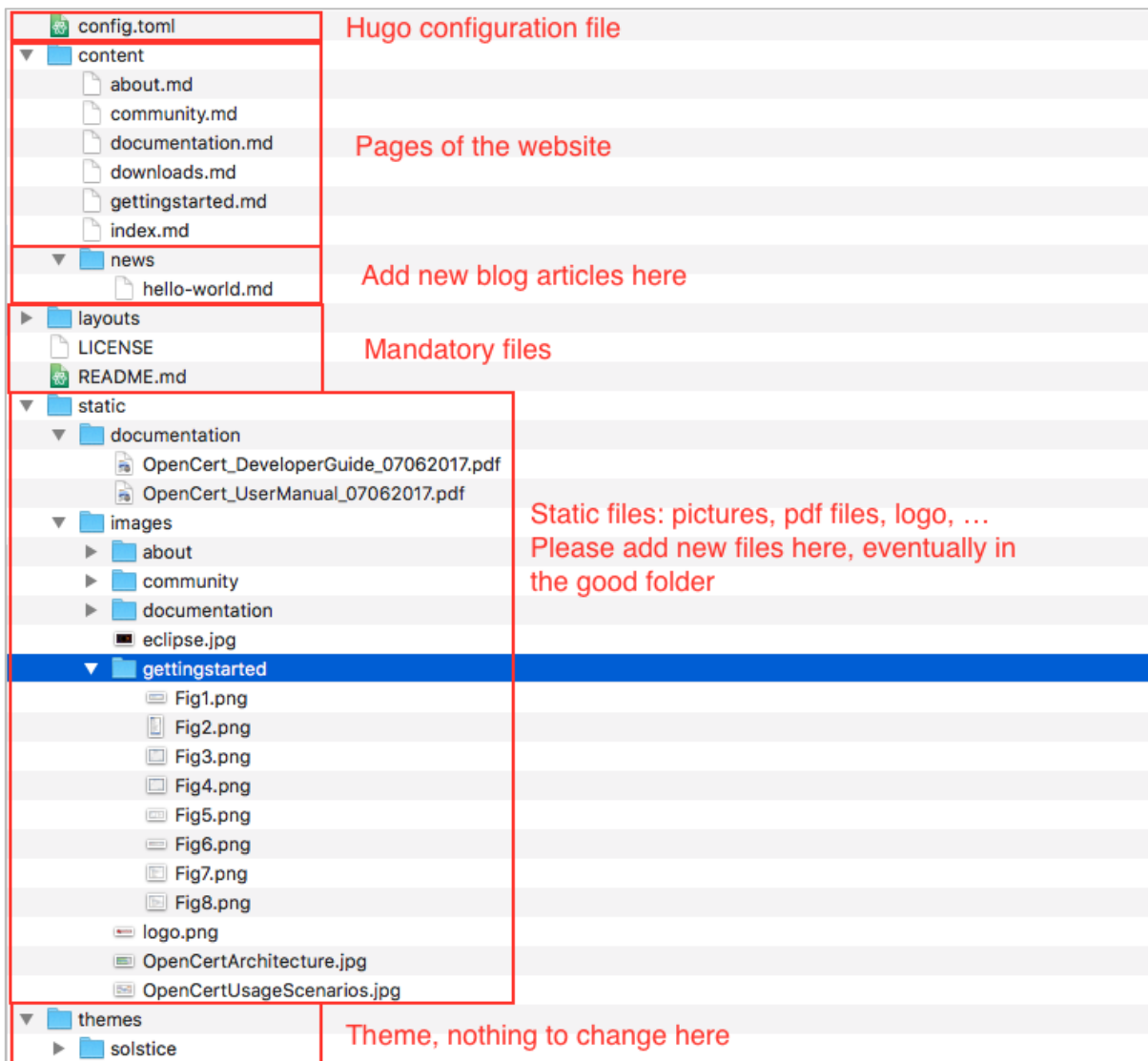


Figure 11. OpenCert website directory structure

You can also import a project in Eclipse from the repository as indicated in Figure 12.

A.2 Add your content

If you want to modify the content of a page, go to the “content” folder. Then open a file; change it using Markdown syntax and HTML.

You can do this by using the file explorer and a text editor, or a command line tool.

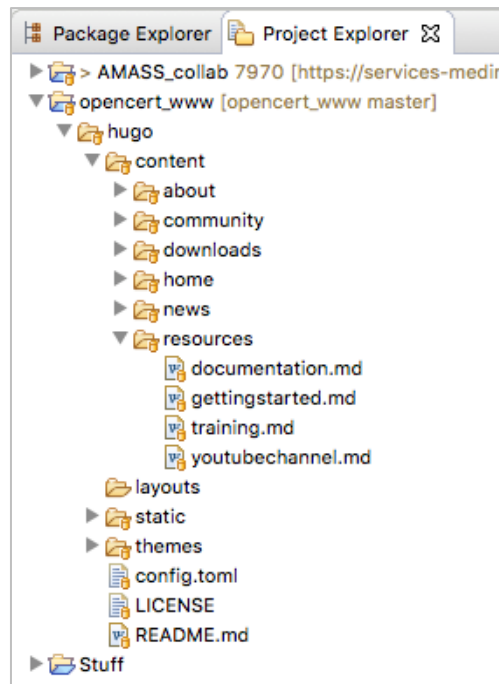


Figure 12. OpenCert WWW project in Eclipse

The “index.md” file is used to generate the “index.html” static webpage:

```

1  -----
2  date: 2017-06-07
3  title: Eclipse OpenCert
4  type: index
5  -----
6
7  -----
8  OpenCert is an integrated and holistic solution for **assurance and certification** management of
9  * **Cyber-Physical Systems (CPS)** spanning the largest safety and security-critical industrial markets, such as
10 * aerospace, space, railway, manufacturing, energy and health. The ultimate aim is to lower certification costs
11 * in face of rapidly changing product features and market needs.
12
13 <p>
14 
15 <br />
16 </p>

```

Figure 13. Index.md file content

The top of this file includes some meta data:

- *Date*
- *Title*
- *Type*

While *Date* and *Title* are available on all pages, “type: index” is only on the index page.

Markdown can be used for basic content, but it is also possible to use HTML. In this example, HTML is used to insert a picture.

Do not forget to save the file before launching the generator.

Add a Blog or News entry

If you do not want to change content, but just want to add an article to the blog, then go to the “content/news” folder. Just copy the first news, change the file name, edit the metadata (*Date* and *Title*), add some content, and save.

The new article will be added to the news feed at the right column of the Home page and the RSS feed when the website is rebuilt.

Make sure you edit the *Date* metadata: the news feed is sorted using this value rather than the file modification date.

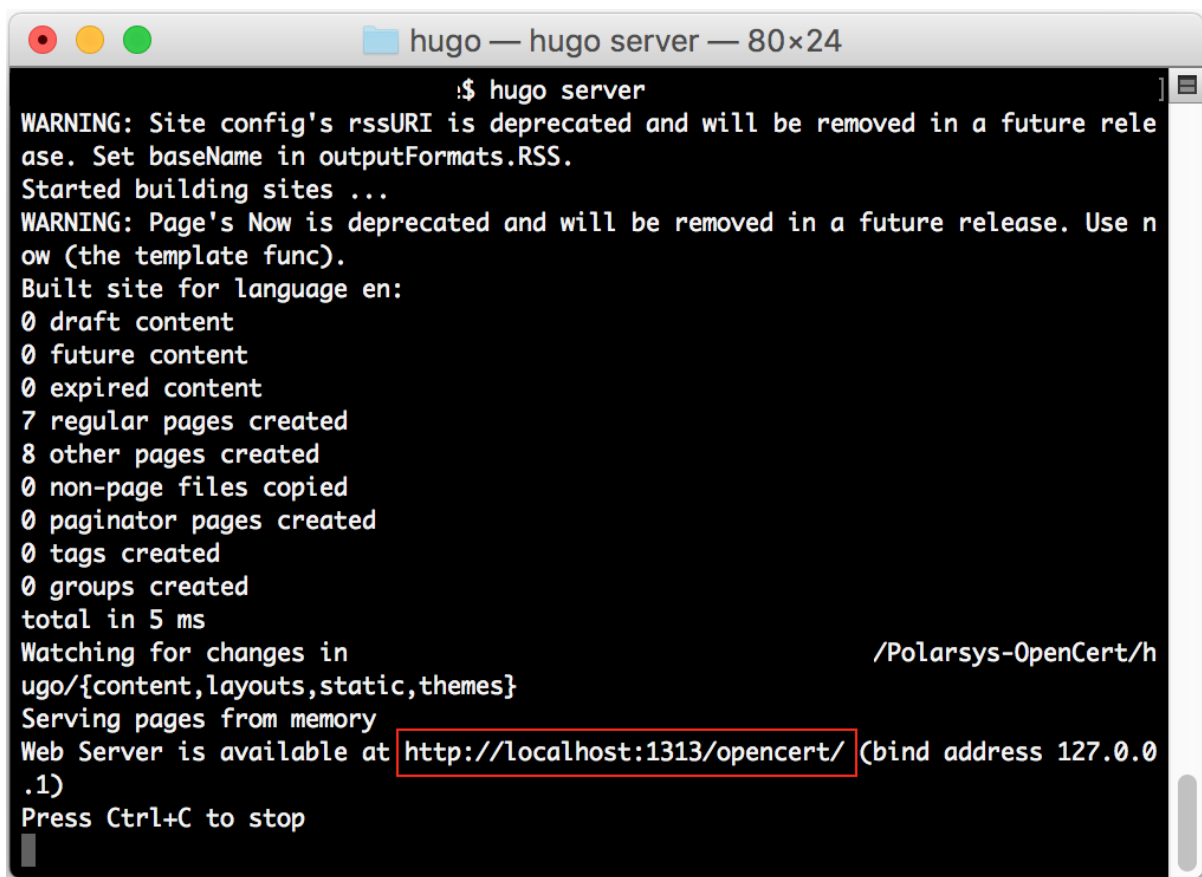
Check your modifications

Before uploading the files to the web server, you must check your modifications. Hugo provides a very convenient way to do that: it has its own local webserver.

In the terminal, be sure to be in the “hugo” folder. Then, run:

```
hugo server
```

You will get:



```
hugo server

WARNING: Site config's rssURI is deprecated and will be removed in a future release. Set baseName in outputFormats.RSS.
Started building sites ...
WARNING: Page's Now is deprecated and will be removed in a future release. Use now (the template func).
Built site for language en:
0 draft content
0 future content
0 expired content
7 regular pages created
8 other pages created
0 non-page files copied
0 paginator pages created
0 tags created
0 groups created
total in 5 ms
Watching for changes in /Polarsys-OpenCert/hugo/{content,layouts,static,themes}
Serving pages from memory
Web Server is available at http://localhost:1313/opencert/ (bind address 127.0.0.1)
Press Ctrl+C to stop
```

Figure 14. Hugo server window

This will generate a new version of the website, with the latest changes you have done on your computer. Then, it starts a local webserver.

Copy and paste the web address in your web browser.

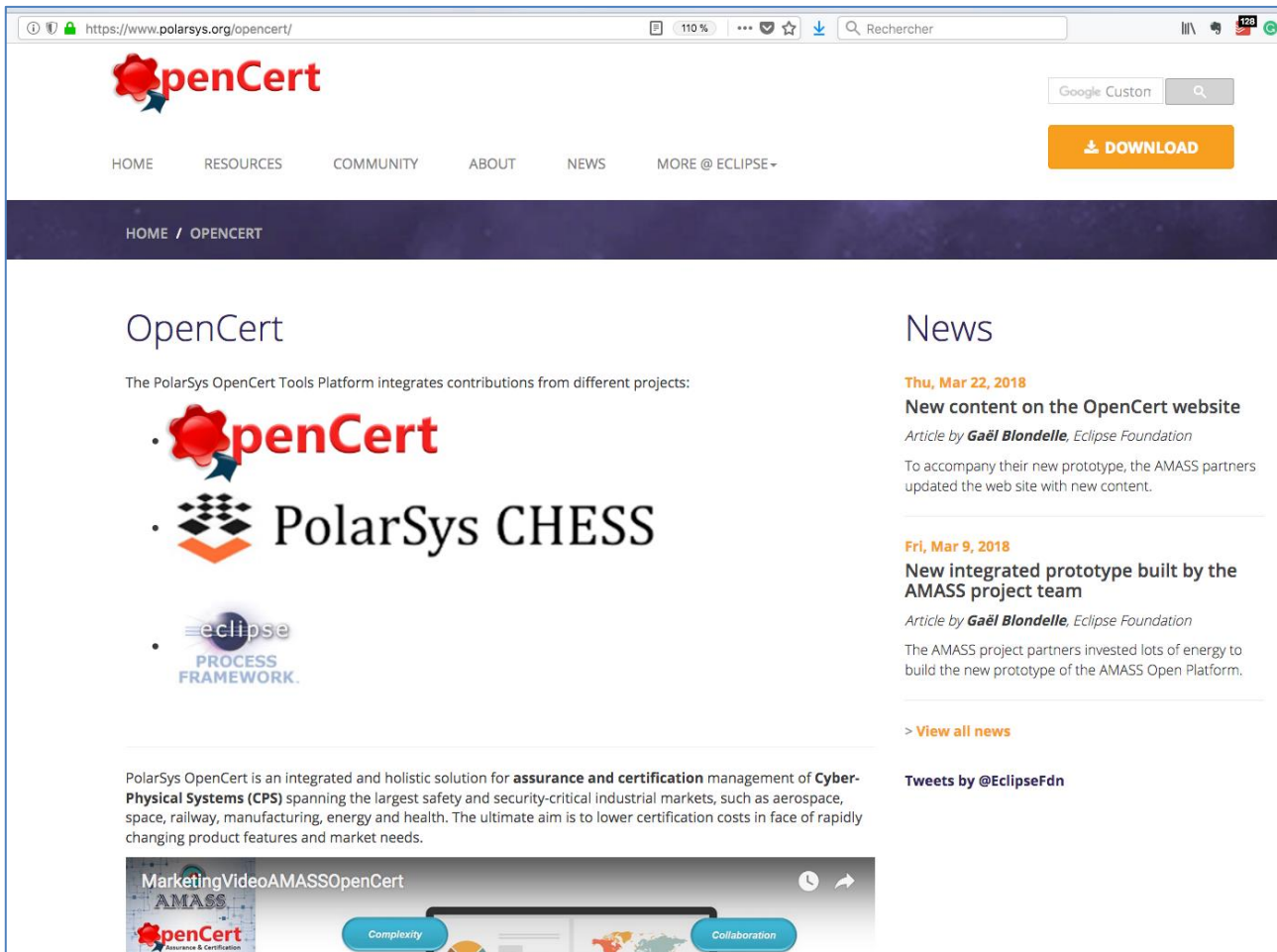


Figure 15. Visualize website update before commit

You can browse the website to check your modification. In some cases, you might have to force refresh the content in the browser, e.g. with “ctrl+r”.

To stop the web server, go back to the terminal and press “ctrl+c”.

You are ready to push the content online.

Push the changes

Go back to the website root. You must be in “/opencert_www/”, not in “/opencert_www/hugo”.

First, add the changes and new files (like news or pictures) to git:

```
git add *
```

Then, commit your changes:

```
git commit -m "add a description here" -m "Signed off by: Firstname Name <email@email.com>"
```

Please change “add a description” with a quick description of your change. And change “Firstname Name <email@email.com>” with your personal information. This must be the same than on your Gerrit profile: see <https://git.polarsys.org/r/#/settings/>

If you are not signing off your commit, it will be refused by the git server.

Check your Gerrit credentials and submit your commit for review:

```
git push origin HEAD:refs/for/master
```

Then, contact OpenCert project's committers and ask them to review your commit. They can see it in their "Incoming reviews" list at <https://git.polarsys.org/r/#/dashboard/self>.

An alternative is to use Eclipse to commit the change as shown below:

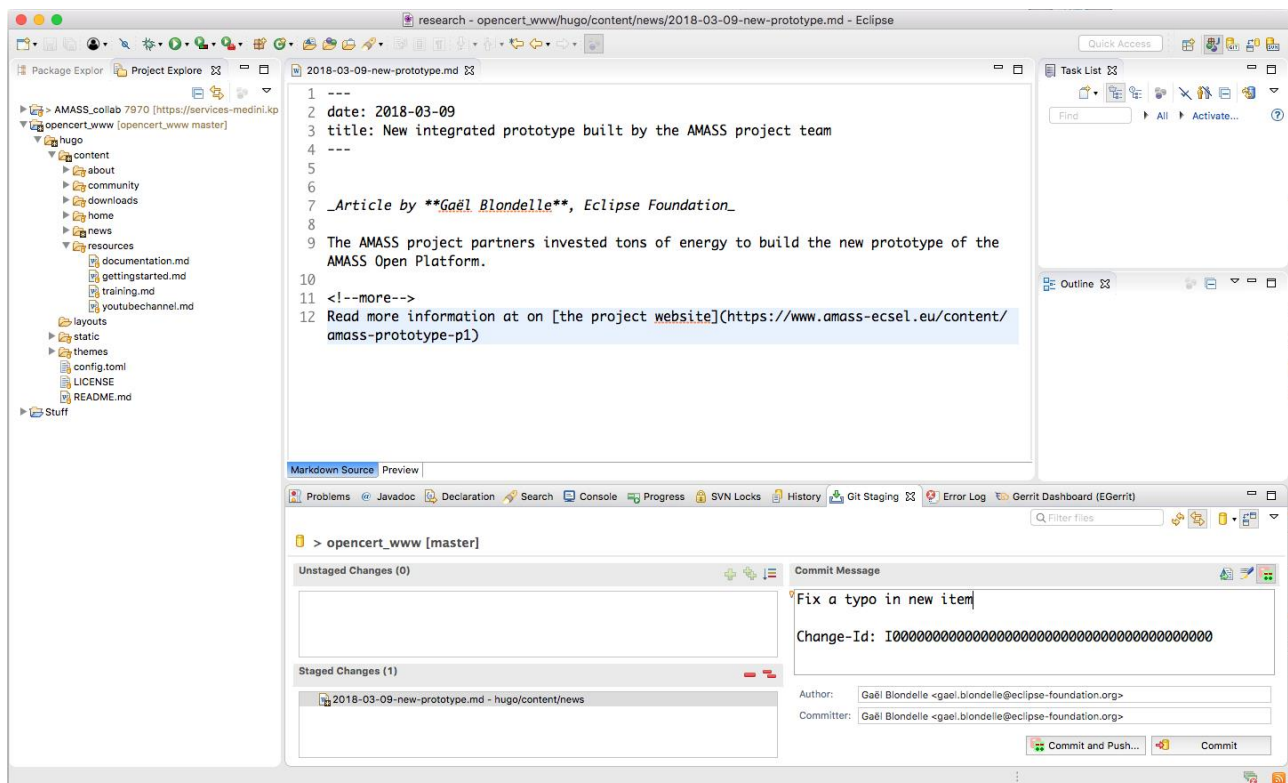
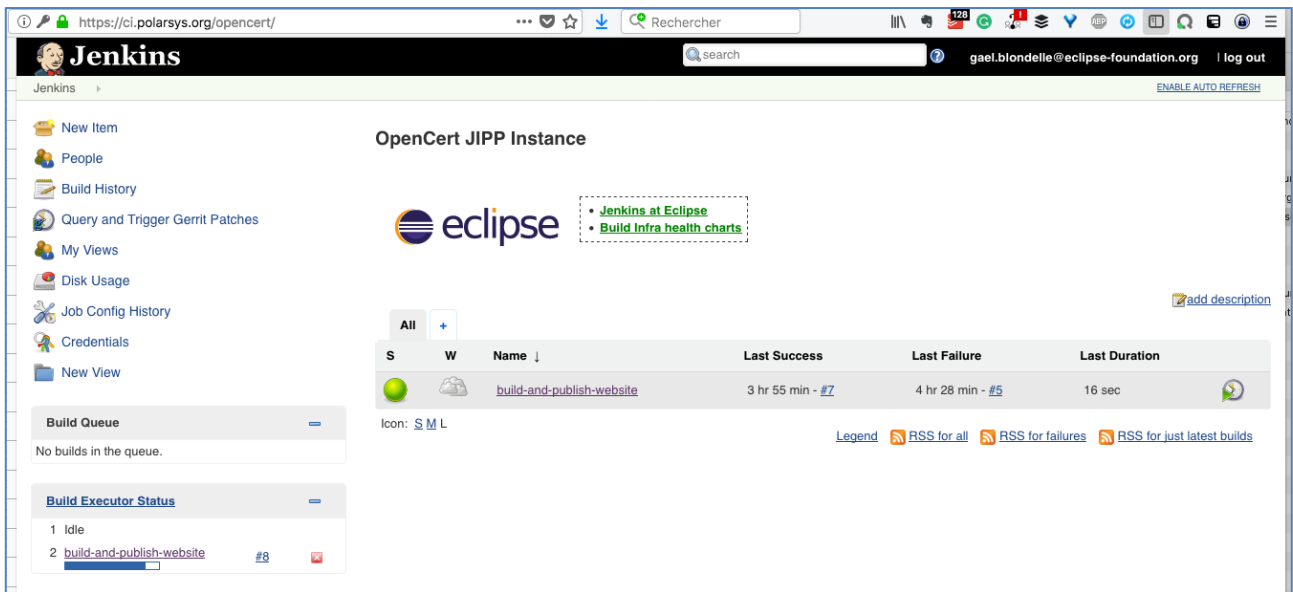


Figure 16. Commit a change to the repository

Build the website

We use a Jenkins Job to publish the website automatically from the latest source in the Git repository. The job "compiles" the web site with Hugo, then compares with the online version, and updates it if necessary.

As such, you don't need to do anything to rebuild the website.



Jenkins

OpenCert JIPP Instance

[Jenkins at Eclipse](#)
[Build Infra health charts](#)

[add description](#)

S	W	Name ↓	Last Success	Last Failure	Last Duration
		build-and-publish-website	3 hr 55 min - #7	4 hr 28 min - #5	16 sec

Icon: [S](#) [M](#) [L](#)

[Legend](#) [RSS for all](#) [RSS for failures](#) [RSS for just latest builds](#)

Build Queue

No builds in the queue.

Build Executor Status

- Idle
- [build-and-publish-website](#) #8

Figure 17. Jenkins dashboard with the OpenCert “build-and-publish-website” job

Check online

Once pushed, it takes a few minutes to see the change on the website. You may have to force the reload of the content of a page using “ctrl+r” in your browser.