Grant Agreement N°: 956671
Topic: ICT-56-2020

The European IoT Hub

Growing a sustainable and comprehensive ecosystem for Next Generation Internet of Things

D3.3: Open-Source Ecosystem Catalogue

Version 1

Revision: v.1.0

Work package	WP 3
Task	Task 3.2
Due date	31/03/2022
Submission date	31/03/2022
Deliverable lead	INTRASOFT International
Version	1.0

Abstract

This deliverable illustrates the EU-IoT coordination and support activities that aim at supporting the IoT community in contribute to and in using Open Source IoT software. The main outcome of the activities was a mapping of the IoT open source ecosystem and the creation of a directory of over 100 OSS projects. The latter have been analysed in terms of their technical characteristics and popularity in the community. Relevant information has been published in a white paper and disseminated to the community.

Keywords: IoT, Open Source, Catalogue, Ecosystem, Eclipse, Linux Foundation, Apache, OSS projects, Github, OSS projects analysis

Document Revision History

Version	Date	Description of change	List of contributor(s)
V0.1	23/02/22	ToC	Netcompany-Intrasoft
V0.12	24/02/22	Further Analysis and Break Down of Contents	Netcompany-Intrasoft
V0.14	03/03/22	Completion of Introductory Section (Section 1)	Netcompany-Intrasoft
V0.15	08/03/22	Open source strategy and Ecosystem contents elaboration (Section 2)	Netcompany-Intrasoft
V0.16	10/03/22	Analysis of OSS projects (Section 3)	Netcompany-Intrasoft
V0.17	15/03/22	Information on the OSS Catalogue (Section 4) – First round of conclusions	Netcompany-Intrasoft
V0.20	18/03/22	Version for first internal review	Netcompany-Intrasoft
V0.21	21/03/22	First version of Annexes	Netcompany-Intrasoft
V0.30	25/03/22	Version for Quality Review	Netcompany-Intrasoft
V0.90	30/03/22	Version Addressing Review Comments from Martel, Fortiss	Netcompany-Intrasoft
V1.0	30/03/22	PreFinal Version for Final Checks and Submission	Netcompany-Intrasoft

Project co-funded by the European Commission under H2020		
Nature of the deliverable:	Report on Training Activities Version 1	
Dissemination Level		
PU	Public, fully open, e.g., web	√
CI	Classified, information as referred to in Commission Decision 2001/844/EC	
CO	Confidential to EU-IoT project and Commission Services	

EXECUTIVE SUMMARY

One of the main objectives of the EU-IoT project is to provide coordination, orientation and support to the IoT (Internet of Things) community in terms of the integration, use and development of IoT related Open Source Software (OSS). In this direction, the project has performed a mapping of the IoT open source ecosystem to facilitate IoT developers, deployers and users in their efforts to use and leverage open source projects in their activities. This deliverable describes the outcomes of this mapping, as well as its role in the overall coordination and support strategy of EU-IoT.

The project's IoT OSS information collection and mapping focused on the following main entities:

- **IoT Open Source Projects**, which is the main vehicle for using open source software in IoT development, deployment and use activities.
- **IoT OSS Software Stakeholders**, including IoT developers, contributors, open source communities and policy makers.
- **Standards and Standardization Initiatives,** notably initiatives that are closely related to OSS such as standards used in OSS projects and OSS-based standards reference implementations.
- **OSS related European Initiatives and Projects (including EU-funded projects),** notably projects that produced open source software or created OSS communities.

EU-IoT has paid emphasis on the analysis of IoT OSS projects, as they are the vehicles for contributing to OSS and for using OSS software. Specifically, EU-IoT has created a catalogue of over 100 projects from the most prominent OSS communities, including projects from the Eclipse, the Apache and the Linux foundation communities, but also prominent EU projects. The catalogue is searchable, extensible, and publicly available at: https://www.ngiot.eu/archive-for-open-source-projects/. It is provided as a resource to the IoT ecosystem and its establishment has already attracted attention, including submissions of two new OSS projects from the IoT community. Furthermore, the project has thoroughly analysed the projects of the catalogue in terms of their technology focus, the standards to which they are related, their programming language(s), as well as their scale and community characteristics. The results of the analysis have been documented in a whitepaper, which has been disseminated to the IoT community and already received considerable attention.

EU-IoT will continue to implement coordination and support measures that facilitate the IoT stakeholders' engagement with the open source software and the open source ecosystem. This continuation will build upon the initial catalogue of projects that was developed as part of this deliverable. For instance, the catalogue will be extended with more projects and the open source developments of H2020 IoT projects (notably projects of the ICT-56 community and their open calls) will be analysed. Moreover, existing resources will be complemented with a more thorough analysis of projects that support standards-based implementation and use. Overall, by the end of the project, EU-IoT will have produced a set of interesting and useful resources for the IoT community, which will continue to be relevant following the conclusion of EU-IoT.

TABLE OF CONTENTS

LIST OF FIGURES

ABBREVIATIONS

AI	Artificial Intelligence
API	Application Programming Interface
AMQP	Advanced Message Queuing Protocol
AR	Augmented Reality
BSD	Berkeley Source Distribution
CB	Coordination Board
CC	Creative Commons
CESSDA	Consortium of European Social Science Data Archives
CoAP	Constrained Application Protocol
CSA	Coordination and Support Action
DDS	Data Distribution Service
DL	Deep Learning
DMP	Data Management Plan
FAIR	Findable, Accessible, Interoperable, Reusable
GPL	General Public License
HTML	HyperText Markup Language
HTTP	HyperText Transfer Protocol
IEC	International Electrotechnical Commission
IoT	Internet of Things
IIoT	Industrial Internet of Things
IP	Internet Protocol
JSON	Java Script Object Notation
LGPL	Lesser General Public License
M2M	Machine-to-Machine
MQTT	Message Queuing Telemetry Transport
ML	Machine Learning
NAT	Network Address Translations
NFC	Near Field Communications
ORDP	Open Research Data Pilot
OS-IoT	Open Source – Internet of Things
OSS	Open Source Software
OT	Operational Technology
PLC	Programmable Logic Controller
REST	Representational State Transfer

ROS	Robotic Operating System
RTOS	Real-Time Operating System
SCADA	Supervisory Control and Data Acquisition
SDK	Software Development Kit
SDO	Standards Development Organizations
SoC	System-on-Chip
TCP	Transmission Control Protocol
XR	Extended Reality

1 INTRODUCTION

1.1 Project brief and Deliverable Scope

EU-IoT is a Coordination and Support Action (CSA) that supports the European Internet of Things (IoT) community in various activities, including shaping of research and innovation agendas, provision of guidelines for IoT technology development, IoT life-long learning and skills development, IoT standardization, as well as open source projects development and use. The project's activities are relevant to the entire IoT ecosystem in Europe and beyond, yet they are primarily addressing the needs of EU-funded research and innovation projects, notably the needs of the projects that have been funded in the scope of the H2020 ICT-56 call.

EU-IoT open source support activities aim at facilitating IoT stakeholders, including EU funded projects, to use, extend and fully leverage open source software in their IoT developments. Open source software is a very important pillar of IoT developments for a number of factors:

- Most IoT projects comprise some open source components or modules. This holds not only for the open source projects, but also for products and projects that are ultimately offered with a proprietary license.
- Development and release of IoT software as open source remains an excellent approach to maximizing the impact of a project, a product or a service. This is because effective open source developments are taken up by the open source community, which contributes to their improvement and wider use.
- Open source IoT projects provide an excellent vehicle for standardization, as many reference implementations and IoT standards implementations are released with an open source license to maximize their adoption and use. Hence, open source developments are closely related to standards-development processes.

The above reasons have led the EU-IoT CSA to prescribe and implement distinct coordination and support activities for the IoT open source community. Specifically, the project is collecting, curating and presenting to the IoT community information about the status of the global IoT ecosystem, including information about various IoT projects. This information is destined to facilitate the IoT community in its effort to use, implement and fully leverage IoT developments.

1.2 Purpose, Scope and Methodology

The present document is the first deliverable on the project's coordination and support activities for the open source community. The purpose of the deliverable is to present the open source community support strategy of EU-IoT, along with the concrete actions that have been implemented in-line with this strategy during the first 18 months of the project's lifetime. Specifically, EU-IoT has implemented the following activities:

- **IoT OSS Projects Information Collection**: It has collected information about over 100 open source IoT projects, the majority of which are part of the world's most prominent OSS (Open Source Software) communities i.e., the Eclipse Foundation, the Linux Foundation, and the Apache community. Moreover, projects with a special interest for the European Open Source community (e.g. the FIWARE community) have been also considered.
- **IoT OSS Projects Information Curation**: The project has processed and analysed the collected information. The analysis has strived to synthesize the current landscape of IoT OSS projects in terms of their thematic and technical focus, the standards that they support and their popularity in the IoT community.

- **IoT OSS Projects Information Structuring and Presentation**: The project's findings have been made available in the form of a free whitepaper for the IoT community. Most importantly, the information has served as a basis for creating an on-line catalogue of IoT projects, which enables the IoT community to search and discover information about IoT projects. The catalogue will be expanded to include new project entries.
- **IoT OSS Projects Information Dissemination:** In conjunction with the EU-IoT dissemination and communication activities, the project has broadly disseminated IoT OSS projects information to the IoT community. Emphasis has been paid on disseminating the catalogue of the project and on encouraging its use by IoT stakeholders.

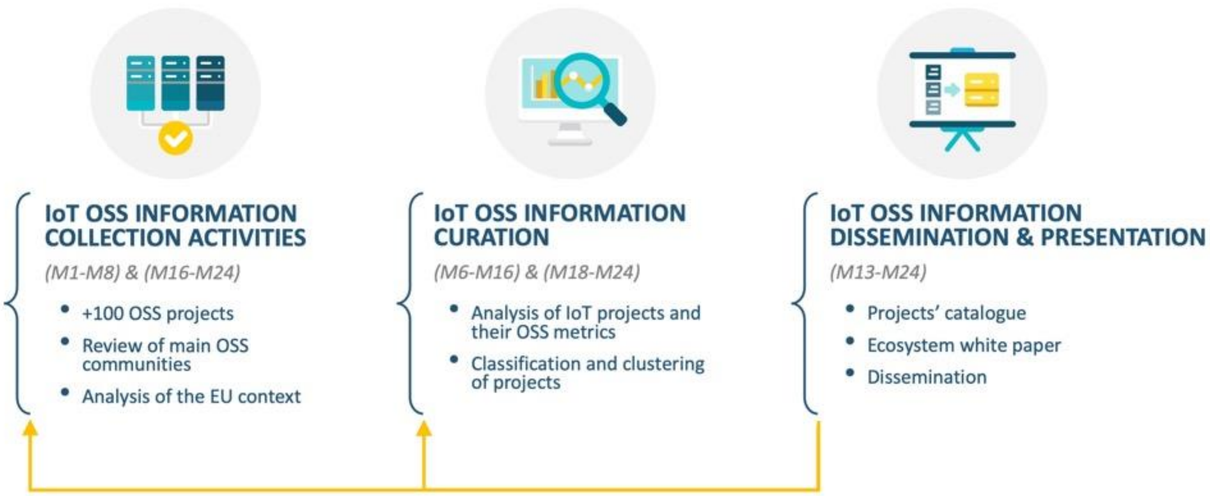


Figure 1: Phased Methodology for the EU-IoT OSS Support activities

The above listed activities map to different implementation phases as illustrated in Figure 1. Specifically:

- **Phase 1**: Information Collection activities were carried out during the first months of the project's lifetime (i.e., M1-M8), where information about IoT projects and communities was collected leveraging desk research. Following the dissemination of the open source findings and services of the project, EU-IoT is collecting additional information (M16-M24), which will serve as a basis for updating and enhancing the IoT OSS landscape analysis of the project.
- Phase 2: Information Analysis and Curation activities commenced after the first semester of the project, once a critical mass of IoT OSS projects and communities' information became available. Information analysis activities will continue till the end of the project in order to update past findings and process new information as it becomes available.
- Phase 3: Information dissemination and presentation activities commenced at the end of the first year of the project's lifetime. They will continue till the end of the project (M13-M30) in order to engage the community with the EU-IoT projects findings.

The following paragraphs detail the EU-IoT open source coordination and support activities and illustrate how they contribute to the implementation of the EU-IoT open-source strategy. It should be noted that the title of the deliverable indicates its relevance to mapping and presenting the status of the IoT ecosystem. EU-IoT has analysed different dimensions of the ecosystem, including IoT OSS communities, IoT OSS projects and the IoT OSS contributions from European projects. However, most of the emphasis has been put on the analysis of IoT projects as they represent the main vehicle for support to open-source developments and standardization.

1.3 Structure of the document

Following this introductory section, the deliverable is structured as follows:

- Section 2 illustrates the open source strategy of the project, including its intention to provide information on the landscape of IoT OSS projects. It also provides an overview of the contents of the ecosystem catalogue in terms of OSS projects, OSS communities, related EU projects and standards-related OSS initiatives.
- Section 3 presents the EU-IoT analysis of OSS projects, including how relevant information was curated and presented to the OSS community.
- Section 4 provides information on the publicly accessible on-line catalogue of IoT OSS projects that the project has developed.
- Section 5 concludes the deliverable and provides information on future activities.

2 EU-IOT OPEN SOURCE STRATEGY: ANALYSING THE ECOSYSTEM

2.1 Overview

As the title of the deliverable indicates, the main EU-IoT objective in terms of OSS was to develop a catalogue of IoT resources that should reflect the status of the IoT OSS ecosystem. This catalogue of resources is aimed at centralizing access to IoT OSS information and at easing the community in its efforts to discover and use IoT OSS components. The catalogue includes all essential information that a user of IoT OSS needs to know, including information about the most prominent OSS projects and communities.

2.2 Ecosystem Catalogue Structure and Contents: Main Entities

The mapping of the IoT open source ecosystem is structured around four main entities (communities, EU initiatives, OSS projects, standards) as presented in the following paragraphs.

2.2.1 OSS Communities

The main OSS communities that engage in the development, maintenance and use of open source software for IoT include prominent OSS communities like the Eclipse Foundation, the Linux Foundation and the Apache community, as well as smaller communities that engage in more specific, yet IoT relevant developments. Specifically:

- The Eclipse IoT Community is the IoT open source community of the Eclipse Foundation. It includes over 45 projects covering different areas of IoT development.
- The Linux foundation hosts communities dedicated to IoT developments. Among the over 375 projects of the foundation, there are some popular state of the art IoT projects such as the EdgeX Foundry, an IoT Plug n' Play platform for edge computing.
- The Apache Software foundation also includes various IoT projects within its portfolio of over 350 projects. Prominent examples are IoTDB and the Camel project that is used for IoT integration.
- The FIWARE foundation provides a rich set of open source components for building next generation internet applications, including components for IoT development such as the FIWARE Context Broker.
- Prominent open source companies like RedHat and Canonical have also developed vibrant open source communities around their IoT and cloud related open source projects and solutions (e.g., Ubuntu core).

These communities are joined by various contributors, including enterprises. For example, Bosch Software Innovations is one of the main contributors to the Eclipse IoT Working Group up to date, while being a user of open source components as part of the Bosch IoT Suite. As another example, Eurotech's main IoT product, the Everyware Software Framework (ESF) is based on Eclipse Kura project. The latter has been contributed to Eclipse by Eurotech. Beyond Bosch and Eurotech, there are other companies all over the world that develop and release enterprise scale OSS. For instance, the Dataart New York City based firm has developed and launched DeviceHive, a prominent open source cloud-based IoT development platform and M2M framework.

2.2.2 Related EU Initiatives and Projects

EU-IoT has identified EU initiatives and organizations with an active and important role in the development of the European and global IoT Open Source Ecosystem. European organisations have a very prominent position in the global IoT open source ecosystem. Specifically, the Eclipse IoT working group is largely influenced by European developers and their organisations. Furthermore, the FIWARE community supports a significant number of IoT developments and deployments, including many applications and use cases developed in the scope of European Research and Development (R&D) projects. For example, several IoT Large Scale Pilot (LSP) projects like H2020 IoF2020 make use of OSS projects.

The OW2 community is also a very prominent open source community in the European software ecosystem. It has a technology focus on infrastructure software and includes various projects that support the creation of the future internet. For example, it includes projects on service-oriented middleware, cloud services orchestration and other technologies that support the development and deployment of scalable IoT applications. Within OW2, there is the OW2 Future Internet Software and Services Initiative (FISSi), which focuses on scalable technologies for the future internet. Although OW2 does not include many projects with an exclusive IoT focus, the FIiSSi initiative supports relevant developments in the cloud/edge technologies continuum.

There are also European Standards Development Organizations (SDOs) like the European Telecommunications Standards Institute (ETSI) that engage with IoT related projects like the Open Source Mano (OSM). The OSM reflects the evolution of ETSI's Network Functions Virtualization (NFV) standardisation efforts. This is also an example, where the open source community boosts the development and validation of open standards.

2.2.3 OSS Projects

The main vehicle for stakeholders to engage in OSS developments is usually some open source project. For example, IoT developers tend to (re)use some OSS components in their own projects. In some cases, they also extend existing OSS projects according to the rules of the project and the community. Hence, IoT projects provide the most common point of engagement with the IoT OSS community. Therefore, the EU-IoT ecosystem catalogue comprises a catalogue of IoT projects with relevant metadata about them, notably data about the purpose and scope of the project, as well as metadata about the popularity of the project within the OSS community. Specifically, we collected and analysed over 120 projects, including some of the most popular projects of the IoT community. The EU-IoT project offers access to these projects through an extensible on-line catalogue. This enables the IoT community to add more projects.

2.2.4 EU funded Projects

Emphasis has been paid in the identification of EU projects that have produced or essentially contributed to IoT OSS development. This analysis has considered projects beyond the ones of the above-outlined prominent communities. In this context, an initial analysis of the open source developments of the ICT-56 projects has been also carried out.

EU projects have a good track record of open source contributions. Many EU projects of the FP7 and H2020 frameworks have provided contributions to the analysed projects, like for instance some of the Eclipse foundation ones. Also, the FIWARE community was initiated within the EC Future Internet Private Public Partnership (FI-PPP) context. FIWARE provides several open source components that are widely used, including IoT components and platforms such as the FIWARE Context Broker that simplifies the development of IoT applications. Furthermore, various projects established their IoT OSS from scratch and created their own communities. Prominent examples include:

- The FP7 OpenIoT project, which established the OpenIoT open source IoT middleware platform. It is one of the analysed projects that has still an active community and frequent mentions.

- The FP7 WebInOS (Secure WebOS Application Environment) also established an open source project with an active community for several years.

Following tables illustrate the open source developments of the ICT-56 project that are closely collaborating with EU-IoT.

2.2.4.1 ASSIST-IoT *(https://assist-iot.eu/)*

Project Summary	The project designs, implements and validates an open, decentralised reference architecture, associated enablers, services and tools, to assist human-centric IoT applications in multiple verticals. ASSIST-IoT will deliver, in a realistic, measurable, and replicable way, a unified innovative multi-plane (semi-)autonomous edge-to-cloud-continuum architecture for the future IoT deployments. ASSIST-IoT proposes to be hugely based on OSS technologies, relying on the most recent trends on microservice, containerisation and orchestration, supplemented by cross-cutting digital enablers. The architecture supports continuous integration and long-term sustainability of domain-agnostic, interoperable, self-* capable, intelligent, distributed, scalable, secure, and trustworthy IoT ecosystems.
Planned OSS Developments	ASSIST-IoT is working with open source hardware (Linux-based) and software (containerisation, lightweight K8s), and aims to produce outcomes as open source by default. The proposed license to be used is a type of Open Source Software License, Apache 2.0, GNU GPL or others, that enables distributing of the results to the community. Some results will be patentable, yet it is expected that open source by design will be the strategy for most outputs.

2.2.4.2 VEDLIOT *(https://vedliot.eu/)*

Project Summary	VEDLIoT builds a distributed Very Efficient Deep Learning IoT platform. Instead of traditional algorithms, artificial intelligence (AI) and deep learning (DL) are used to handle large complexity in IoT applications. Due to the distributed approach, VEDLIoT allows dividing the application into smaller and more efficient components and work together in large collaborative systems in the Internet of Things (IoT), enabling AI-based algorithms that are distributed over IoT devices from edge to cloud.
Planned OSS Developments	Most of the technology will be open source, although for some of the applications that the partners are working on in the automotive sector some of the technology is proprietary.

2.2.4.3 IoT-NGIN *(https://iot-ngin.eu/)*

Project Summary	IoT-NGIN introduces novel research and innovation concepts, acting as the “IoT Engine” which will fuel the Next Generation of IoT as a part of the European Next Generation Internet. IoT-NGIN uncovers a patterns-based meta-architecture that encompasses evolving, legacy, and future IoT architectures. The project also optimizes IoT/M2M and 5G/MCM

	communications, including using secure-by-design micro-services to extend the edge cloud paradigm. Moreover, it enables user and self-aware, autonomous IoT systems through privacy-preserving federated ML and ambient intelligence, with AR support for humans.
Planned OSS Developments	The project developments will be released as open source, almost entirely, the source code being available under the projects gitlab.com repository (https://gitlab.com/h2020-iot-ngin).

2.2.4.4 **INGENIOUS** *(https://ingenious-iot.eu/)*

Project Summary	INGENIOUS designs and evaluates the Next-Generation IoT (NG-IoT) solution, with emphasis on 5G and the development of Edge and Cloud computing extensions for IoT, as well as providing smart networking and data management solutions with Artificial Intelligence and Machine Learning (AI/ML). The project embraces the 5G Infrastructure Association (5G IA) and Alliance for Internet of Things Innovation (AIOTI) vision for empowering smart manufacturing and smart mobility verticals.
Planned OSS Developments	Part of the project's developments deals with safe and secure communication between parts of the computer that separates process cores. This part will comprise open source parts, to complement proprietary modules running on FPGAs (Field-Programmable Gate Arrays).

2.2.4.5 **IntellIoT** *(https://intelliot.eu/)*

Project Summary	IntellIoT focuses on the development of integrated, distributed, human-centered and trustworthy IoT frameworks applicable to agriculture, healthcare and manufacturing. Enabling technologies such as 5G, cybersecurity, distributed technology, Augmented Reality, and tactile internet, IntellIoT also champions end-user trust, adequate security and privacy by design.
Planned OSS Developments	IntellIoT's contributions to open-source projects will focus on its 5G platform which is based on the OpenAirInterface (https://www.openairinterface.org/) and Mosaic5G (http://mosaic-5g.io/). OpenAirInterface and Mosaic5G are open-source solutions for 5G and are backed by large international communities of academic and industrial stakeholders. The results on advanced network management for the NG IoT solutions e.g., open-source extension of OAI/Mosaic5G will be made publicly accessible via GitLab. The project will contribute components of its hypermedia-based Multi-Agent System, which might be integrated with the IoT Standards resources of the open-source community. This will include libraries and tooling for researchers and practitioners to design, deploy, and manage IoT/edge infrastructures. Additionally, the project will deliver open-source AI algorithms focusing on the use cases that are investigated under the project.

2.2.4.6 TERMINET *(https://terminet-h2020.eu/)*

Project Summary	The vision of TERMINET is to provide a flexible, open, and decentralised next generation IoT reference architecture based on cutting-edge technologies such as software-defined networking, multiple-access edge computing, and virtualisation for new real-time capable solutions. This goal will be achieved by enabling secure and privacy-preserving IoT services, user-aware solutions, semi-autonomous devices, and self-aware mechanisms, frameworks, and schemes, supported by distributed AI and new intelligent IoT devices within a virtualized edge-platform-cloud environment.
Planned OSS Developments	In TERMINET's integrated platform, open-source software tools will be utilized. However, either the hardware (i.e., smart glasses) or other tools required will be proprietary. The open calls of the project will look to on-board new, state-of-the-art open source tools and to integrate them in the project's developments.

2.2.5 Standards-Related OSS

EU-IoT has also identified the link of OSS projects to standardization. Specifically, it has collected and lists the standards that are implemented and/or supported by the OSS projects of the catalogue. In several cases OSS projects have been used to support reference implementation of standards, as well as broader standards development. Specifically, a variety of OSS projects have focused on the implementation of different parts of the ETSI OneM2M standard. These projects range from general platforms like Eclipse OM2M that enables the implementation of horizontal OneM2M compliant M2M systems (i.e., servers, gateways, devices) and OpenMTC that provides an integration middleware based on the oneM2M standard, to projects focused on more specific aspects like oneM2MTester which is an OneM2M conformance testing tool.

The link of OSS projects to standards will be further elaborated and presented as part of other deliverables (e.g., D3.5/D3.6) of the EU-IoT project.

2.3 Mapping Principles

The collection and analysis of the information about the open source ecosystem was driven by the following principles:

- **Completeness**: The information about the ecosystem is collected in ways that ensure very good coverage of the ecosystem. This includes putting emphasis in the collection of information about the most prominent OSS projects and OSS initiatives with direct links to the IoT ecosystem.
- **Extensibility**: The project enables the community to expand the collected information through submitting information about OSS projects and initiatives that might have not been included in the initial analysis of the project.
- **Community Engagement**: Through the dissemination and communication activities of the project, EU-IoT strives to achieve the engagement of the IoT ecosystem with the IoT OSS results of the project. This includes for example engaging in the use and in the extension of the catalogue. Community engagement activities will be intensified during the last year of the project's lifetime.
- **Linking to other EU-IoT Activities**: The IoT OSS activities of the EU-IoT projects are

closely linked to other activities of the project. Specifically, they provide content for the development of the training and road mapping activities of the project. At the same time, they will be linked to the standardization activities through identifying open source initiatives that boost standardization. As already outlined, OSS activities are also closely related to the dissemination and communication activities of the project, which act as multipliers of the outreach and of the overall impact of the OSS ecosystem catalogue(s).

2.4 Mapping Information Presentation

Information associated with the ecosystem mapping was included in an on-line catalogue of open source projects and documented in relevant whitepaper. Furthermore, code analysis tools were used for analysing the various OSS projects as outlined in the following paragraphs.

3 OPEN SOURCE PROJECTS ANALYSIS

3.1 Methodology

As part of the EU-IoT ecosystem catalogue development we analysed over 120 projects. Although not exhaustive, we consider our list of analysed projects representative of the status of the global IoT ecosystem. This is because the analysed projects include:

- **IoT projects from the most prominent IoT ecosystems worldwide**, including projects hosted by Apache, the Linux foundation, and the Eclipse foundation.
- **IoT projects that are reported in on-line reviews, surveys, and tutorials**. Such surveys are published frequently and tend to include lists of popular and high-impact projects.
- **Projects that are directly related to IoT systems and technologies**. Our list does not include projects that relate to supporting infrastructures (e.g., cloud computing infrastructures, DevOps infrastructures) that are widely used as part of IoT applications. Rather, we focus on projects and libraries that are exclusively or primarily used for developing IoT modules and services.
- **Projects which address a broad range of IoT topics and technologies**. We strived to include representative projects from every area of IoT systems development, including IoT devices and open source hardware, edge/cloud computing projects for IoT, IoT development environments, IoT analytics, as well as IoT data visualization projects.

Our methodology for analysing the various projects involved the collection and analysis of a set of characteristics for every project. These include:

- **Type**: The (general) type of the IoT project i.e., whether it concerns software, hardware or middleware.
- IoT technology area: This denotes a more detailed characterisation of the technology area of the project, such as whether it concerns IoT devices, edge/cloud computing, Big Data/IoT Analytics etc.
- IoT standards: This lists the standards that relate to the project, including standards supported by the project. In several cases, there are projects that provide reference implementations of standards.
- Primary programming language: This concerns the main programming language(s) of the project. The term "main" is used for cases of projects that comprise code in several languages. In these cases, our analysis has focused on the most dominant languages.
- License: This denotes the open source license of the project, which is an important piece of information for stakeholders wishing to adopt, use and integrate an OSS project within their developments.
- Community information: This includes a variety of attributes that indicate the popularity of the project in the OSS community, as well as the recency of the activity and engagement around the project. Specifically, wherever possible, we collected the following information:

    - **Days Since Last Push**: This is the number of days since the last time there was a push of code in the project i.e., it indicates the most recent activities from OSS contributors on the project.
    - **Days Since Created**: This is the number of days since the project was created. It is a measure on the "age" of the project.
    - **Github Stars**: The number of stars that the project has on GitHub. It is an indicator of the popularity of the project.
    - **Number of Commits**: Indicates the total number of commits of the project. It is a measure of the activity around the project.

- **Number of Watchers**: Provides the number of GitHub users that have selected to "watch" the project and get updates about its status. Any user can select to be a "watcher" of a public repository and to receive notifications about events (e.g., commits, pull requests, issues). A watcher is usually interested in the activity of the project and potentially interested in contributing to it [Dabbish12]. As such watching is considered as a passive type of project membership [Sheoran14].
- **Number of Forks**: This is another indicator of the community activity around the project. A fork occurs when a completely independent line of development based on the source code basis of the project starts. This is usually done either by the community of the project or by a third-party independent from the project's community.

3.2 Tools

To collect community information, we relied on a static React application for displaying statistics from github repositories like Star History, Fork History and more[1]. The application produces graphical representations of the repositories' information (Figure 2), yet it can be also used as a command line application to offer instant and faster extraction of statistics.

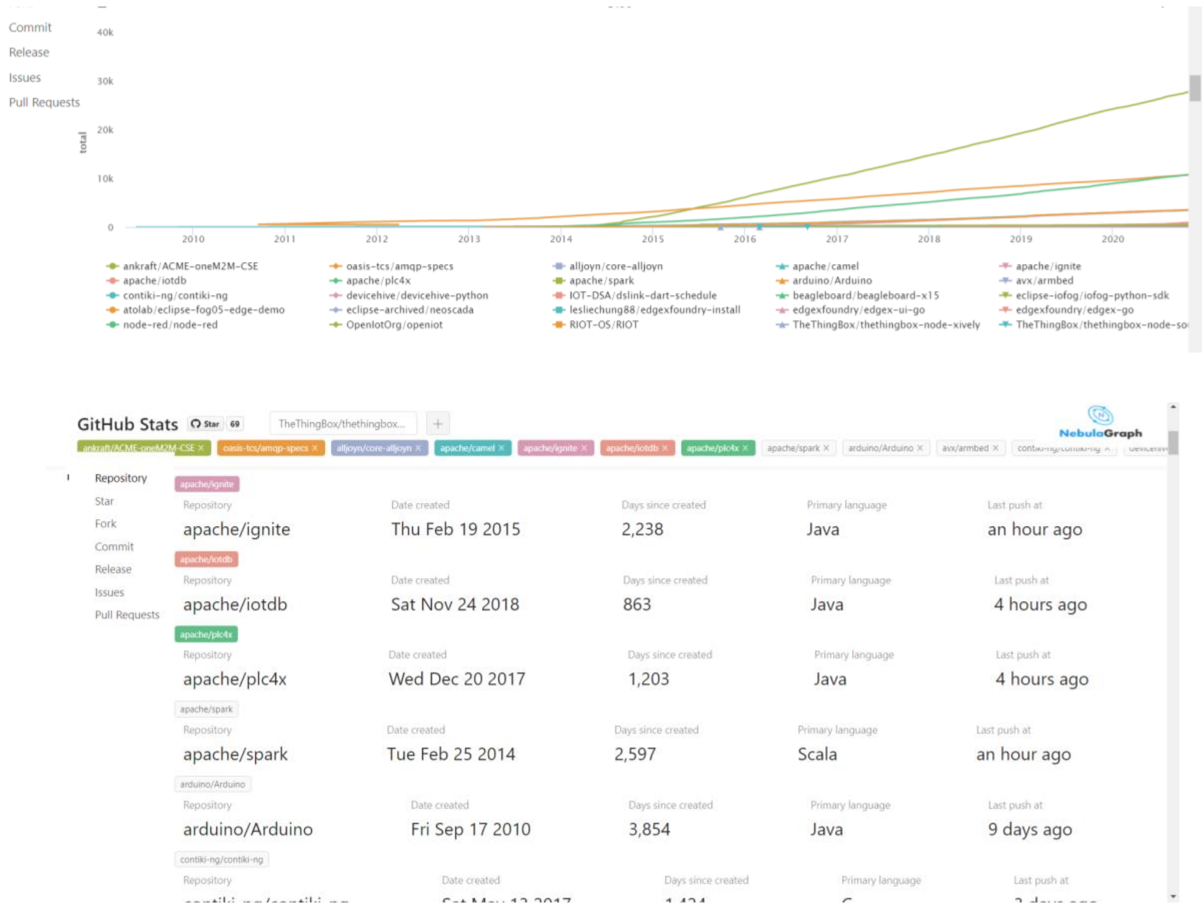


Figure 2: Snapshot of Graphs produced by the Github repositories analysis/statistics applications

Even though our analysis covered over 120 IoT projects, the above-listed information was not always available. For instance, there were some OSS projects that were not in the GitHub and some others for which the community information was not available. Therefore, the total number

[1] vesoft-inc.github.io/github-statistics/

of projects in the representations and charts presented in the relevant EU-IoT whitepaper is typically below the total number of projects analysed.

3.3 Areas of Analysis

The projects were analysed in terms of different characteristics, including their:

- **IoT Technological Area**: The various projects were classified and analysed according to the technology focus. They covered a variety of technology areas including edge computing, cloud computing, IoT device/hardware, embedded IoT, IoT connectivity, data/bigdata, industrial IoT areas and IoT security.
- **Programming Language(s)**: The projects were classified based on the programming language they were written. Several projects included code in more than one programming language. The most popular languages for IoT development in these projects are Java, Python, JavaScript and C/C++. Other, less used, but still notable mentions include Go, Perl, Rust, Objective C, SQL, and various more specialized package specific languages like HiveQL.
- **Supported or Related Standards**: The analysis also considered the main standards that are associated with the analysed projects (see for example Figure 3). Most of the analysed projects are not directly linked to some IoT standard or protocol. However, several projects were affiliated to standards like the Message Queuing Telemetry Transport (MQTT), the Constrained Application Protocol (CoAP) the oneM2M, the Data Distribution Service (DDS) and the AMQP (Advanced Message Queuing Protocol). Other notable mentions concern industrial standards and protocols such as the OPC-UA standard for industrial connectivity and IEC 61499 for distributed control.
- **Open Source License**: We explored the different open source licenses that accompanied each one of the projects, such as the Apache license, the Eclipse licenses, the Berkeley Source Distribution (BSD) license, the General Public License (GPL), the Creative Commons (CC) and the LGPL (Lesser General Public License).
- **Community Size and Characteristics**: The projects were also analysed in terms of their community figures, including their popularity. This was based on the above-mentioned GitHub statistical analysis tools.

Furthermore, the analysis considered some correlations between the above listed parameters and characteristics. For instance, the correlation between the commits and forks of the various projects was studied, which was proven to be moderate to strong as projects with significant numbers of commits are likely to attract potential third-party contributors or to be extended by members of their community in new directions (see for example Figure 4).

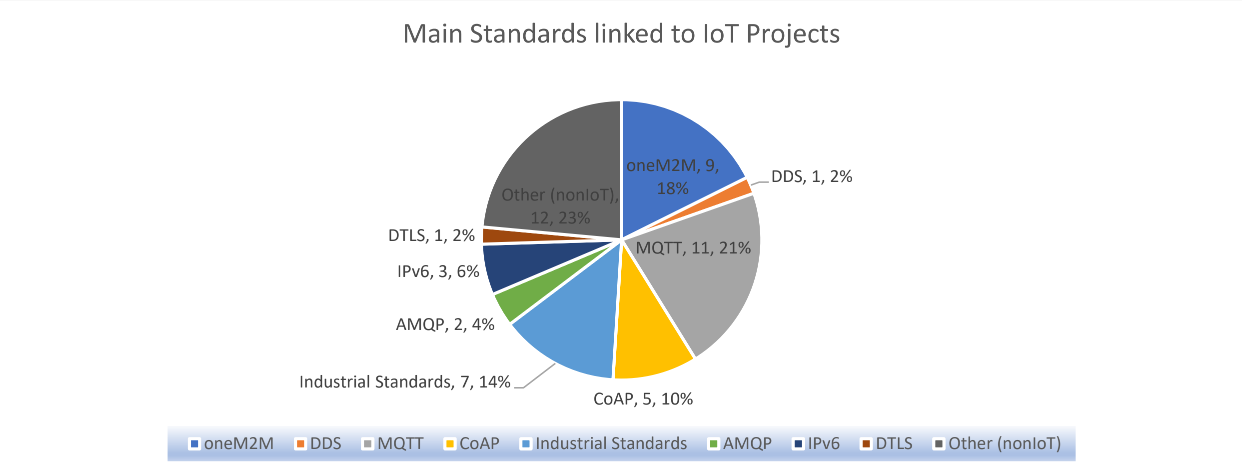


Figure 3: Analysis Example: Standards Relevance for analysed projects

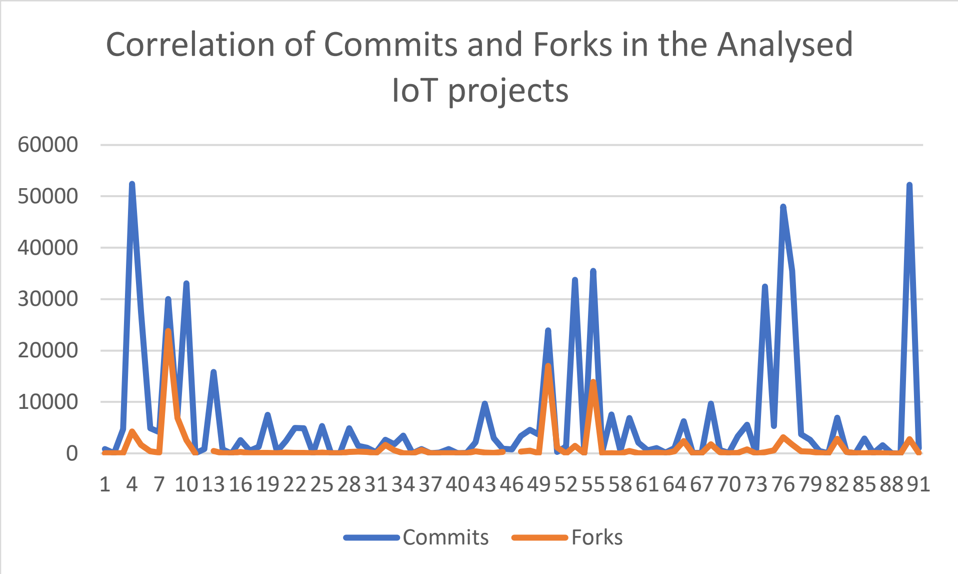


Figure 4: Analysis Example: Correlation of the Github Commits and Forks for 91 of the analysed projects

3.4 Whitepaper

The findings of our analysis are detailed in a Whitepaper that was released at the end of November 2021. The cover and the table of contents of the Whitepaper are illustrated in Figure 5. The whitepaper is available with the DOI:10.13140/RG.2.2.15421.38886 at ResearchGate, but also at the EU-IoT space in Zenodo. Furthermore, it can be also downloaded from the NGIOT/EU-IoT web site.

Figure 5: EU-IoT Whitepaper on the IoT OSS Ecosystem: Cover and Table of Contents

The post has been disseminated through various channels in conjunction with the EU-IoT dissemination and communication activities in WP5. It is noteworthy that some of the social media posts (e.g., LinkedIn Post 1 in January 2022 and LinkedIn Post 2 in December 2021) received significant attention reaching or exceeding 8000 organic views as illustrated in Figure 6 and Figure 7.

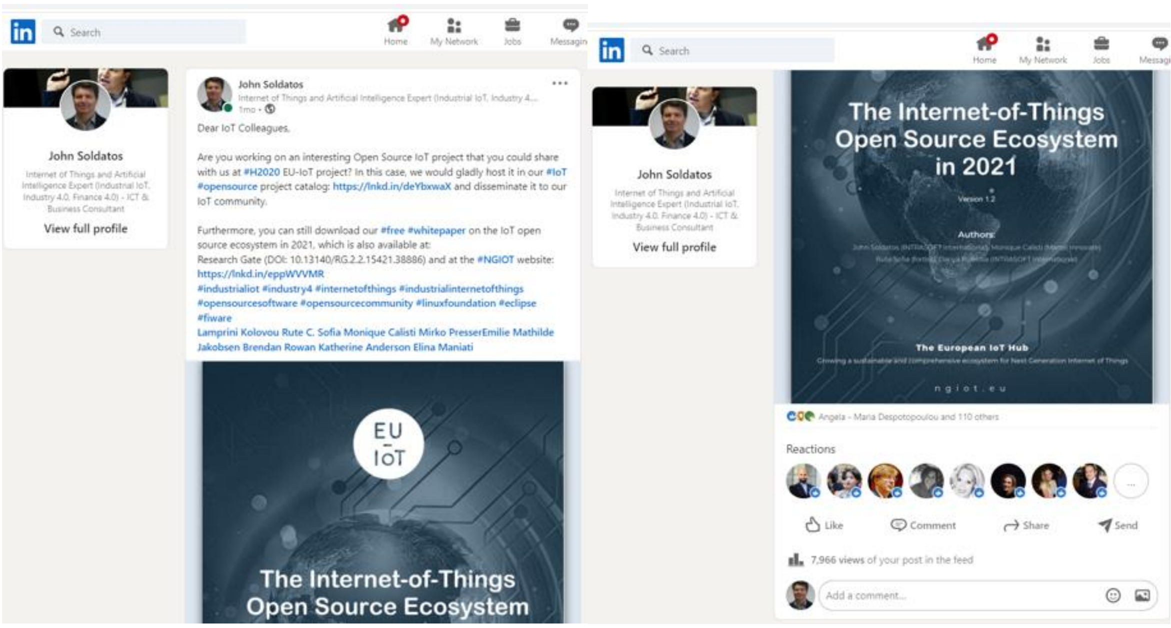


Figure 6: LinkedIn Post about the Whitepaper (7900+ views) (January 2022)

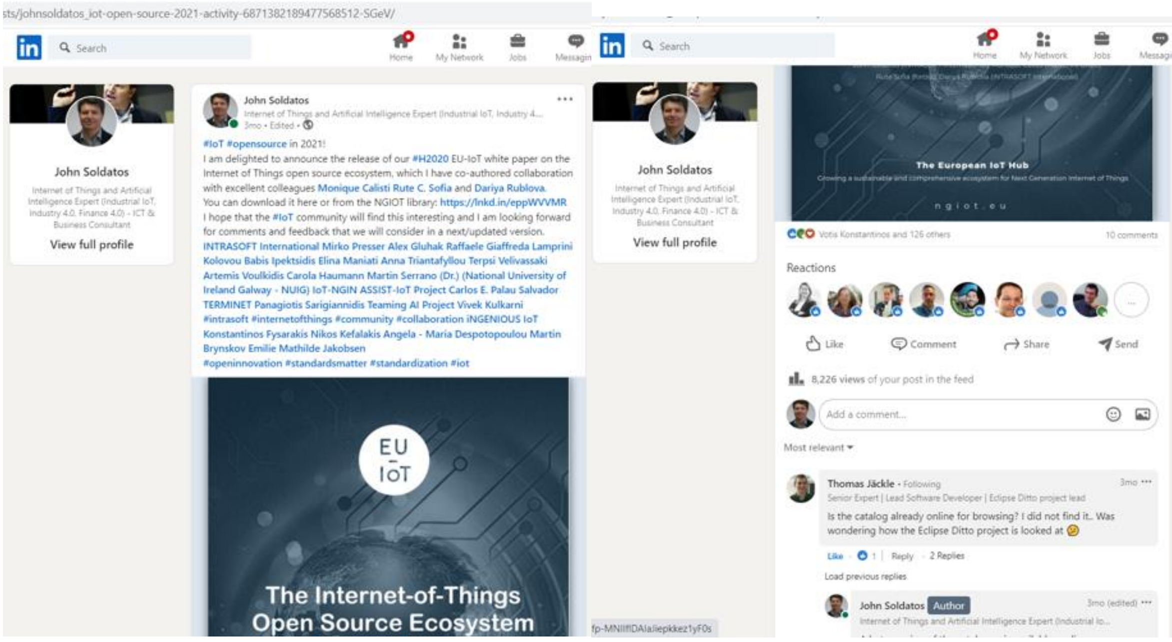


Figure 7: LinkedIn Post about the Whitepaper (8200+ views) (December 2021)

4 ON-LINE OPEN SOURCE PROJECTS CATALOGUE

The EU-IoT Open source projects catalogue is available on-line at NGIOT/EU-IoT web site: https://www.ngiot.eu/archive-for-open-source-projects/. It is searchable i.e., allows its users to search projects based on different criteria, including the IoT technology area of the project (e.g., cloud computing, edge computing), the type of project (e.g., development tool, middleware, open source hardware related), the application area/sector (e.g., energy, manufacturing), as well as the standards that it supports or implements (e.g., CoAP, MQTT). A snapshot of the on-line catalogue is illustrated in Figure 8.

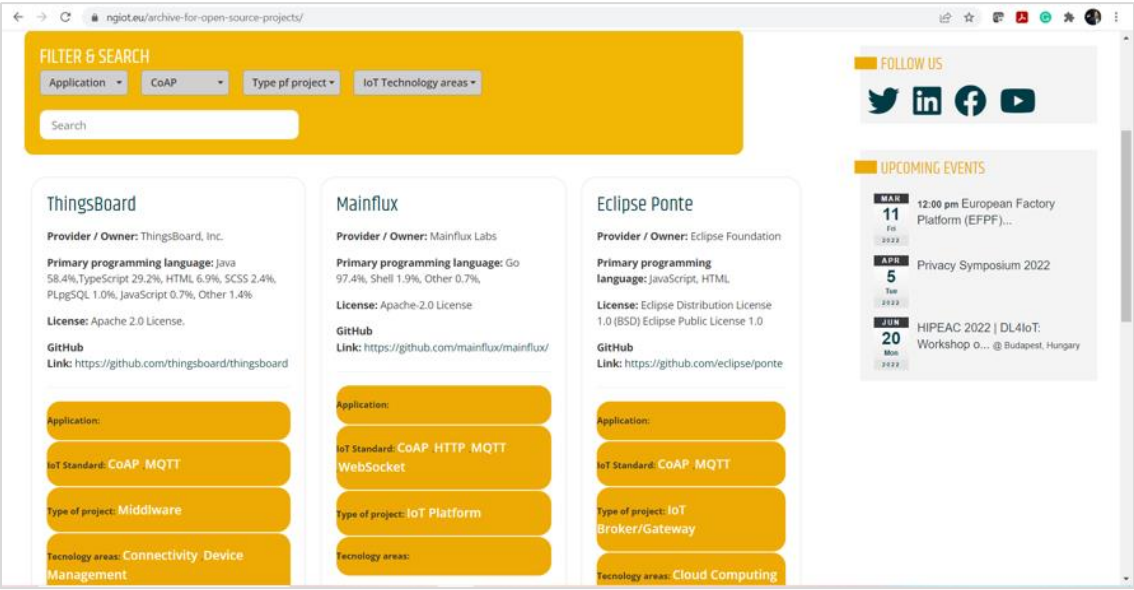


Figure 8: Snapshot of the IoT OSS projects catalogue

EU-IoT provides to the community a facility for extending the catalogue with new project entries. In this direction, a form is provided as depicted in Figure 9.

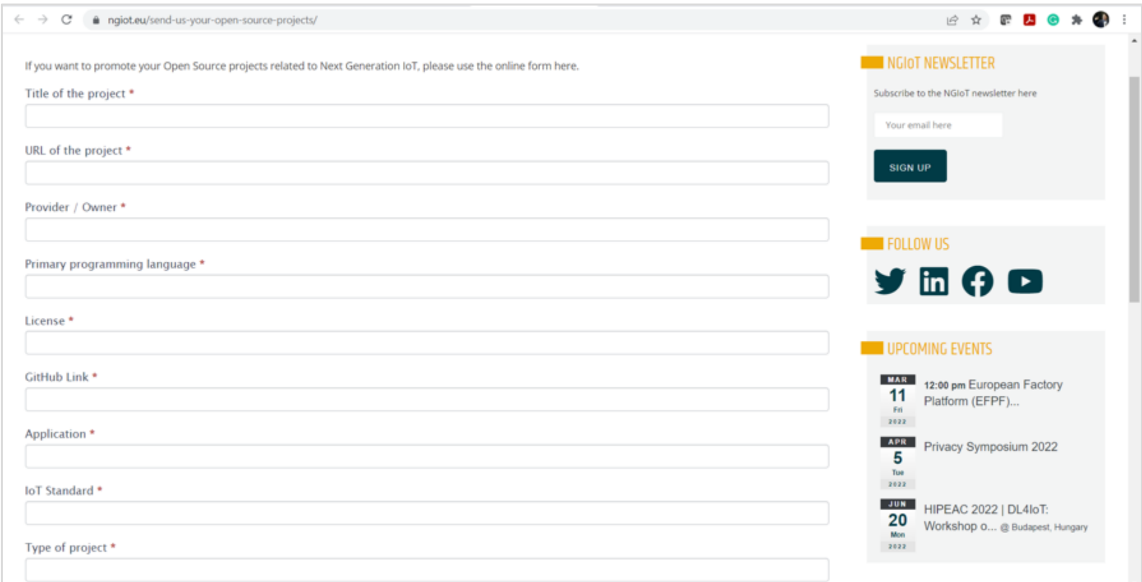


Figure 9: On-line Form for new OSS projects entries

The contents of the form are submitted to the EU-IoT team, which performs quality control and if appropriate inserts the new entry to the catalogue. Since December 2021, two entries have been included in the catalogue following respective submissions from founders of these projects.

5 CONCLUSIONS AND OUTLOOK

EU-IoT is committed to supporting the IoT community with access to information and resources about the development and use of open source software. In this direction, the project has performed an initial mapping and analysis of the IoT OSS ecosystem, in terms of projects, OSS related standardization, as well as OSS stakeholders. This analysis was outlined in earlier paragraphs of this document and has already resulted in:

- **The development of a searchable and extensible catalogue of OSS projects**. This catalogue is provided as a resource to the IoT community in order to ease the discovery of information about IoT OSS projects. It is also a channel for IoT project developers to promote and disseminate their OSS developments of the IoT community.
- **A whitepaper on the OSS Ecosystem**. The white paper provides an in-depth analysis of over 100 OSS projects, considering different aspects of the project such as their technology focus, the standards to which they are related, their programming language(s), as well as their scale and community characteristics. The whitepaper has been disseminated to the IoT community and already received considerable attention.

The development of the catalogue and of the whitepaper are the first, yet very important steps in the implementation of the EU-IoT strategy. These steps have led to added-value resources for the community. The project is committed to continue its efforts for supporting and promoting OSS IoT developments in the IoT ecosystem. In this direction, the following activities are planned:

- The extension of the catalogue with more projects, including the ones that will come from the community rather than from EU-IoT desk research.
- A more detailed mapping of open source projects to standards and standardization initiatives.
- A collection and analysis of the open source activities of H2020 IoT projects, notably activities linked to their developments and to their open source calls.
- An active dissemination and communication of the resources produced and their evolution in the coming months.

These activities will provide a basis for completing EU-IoT's OSS related contributions to the IoT community, which will be documented in the next and final version of this deliverable (i.e., D3.4 according to the project's DoA).

APPENDIX A: LIST OF PROJECTS ANALYSED

No.	Project	Short Description
1	ACME	An open source CSE Middleware for Education, intended to provide an easy to install, extensible, and easy to use and maintain CSE. This CSE implements a subset of the oneM2M standard specializations (see http://www.onem2m.org).
2	Advanced Message Queuing Protocol (AMQP)	AMQP is the Internet Protocol for Business Messaging. The Advanced Message Queuing Protocol (AMQP) is an open standard for passing business messages between applications or organizations. It connects systems, feeds business processes with the information they need and reliably transmits onward the instructions that achieve their goals.
3	Akraino	Akraino is a set of open infrastructures and application blueprints for the Edge, spanning a broad variety of use cases, including 5G, AI, Edge IaaS/PaaS, IoT, for both provider and enterprise edge domains.
4	AllJoyn	AllJoyn is an open source software framework that makes it easy for devices and apps to discover and communicate with each other. Developers can write applications for interoperability regardless of transport layer, manufacturer, and without the need for Internet access.
5	Apache Camel	Apache Camel is a powerful open source integration library based on Enterprise Integration Patterns. Rules for Camel's routing engine can be defined in either a Java based DSL or XML.
6	Apache Ignite	Apache Ignite is a distributed database for high-performance computing with in-memory speed.
7	Apache IoTDB	Apache IoTDB (Database for Internet of Things) is an IoT native database with high performance for data management and analysis, deployable on the edge and the cloud. Due to its light-weight architecture, high performance and rich feature set together with its deep integration with Apache Hadoop, Spark and Flink, Apache IoTDB can meet the requirements of massive data storage, high-speed data ingestion and complex data analysis in the IoT industrial fields.
8	Apache PLC4X	The universal protocol adapter for Industrial IoT. PLC4X is a set of libraries for communicating with industrial programmable logic controllers (PLCs) using a variety of protocols but with a shared API.
9	Apache Spark	Spark is a unified analytics engine for large-scale data processing. It provides high-level APIs in Scala, Java, Python, and R, and an optimized engine that supports general computation graphs for data analysis. It also supports a rich set of higher-level tools including Spark SQL for SQL and DataFrames, MLlib for machine learning, GraphX for graph processing, and Structured Streaming for stream processing.

No.	Project	Short Description
10	Arduino	Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.
11	Arduino Ethernet Shield	The Arduino Ethernet Shield 2 is open-source hardware that connects your Arduino to the internet in mere minutes. Just plug this module onto your Arduino Board, connect it to your network with an RJ45 cable (not included) and follow a few simple steps to start controlling your world through the internet. As always with Arduino, every element of the platform – hardware, software and documentation – is freely available and open-source. This means you can learn exactly how it's made and use its design as the starting point for your own circuits. Hundreds of thousands of Arduino Boards are already fueling people's creativity all over the world, everyday.
12	Arduino Yún	The Arduino Yún is a microcontroller board based on the ATmega32u4 and the Atheros AR9331. The Atheros processor supports a Linux distribution based on OpenWrt named Linino OS. The board has built-in Ethernet and WiFi support, a USB-A port, micro-SD card slot, 20 digital input/output pins (7 of them can be used as PWM outputs and 12 as analog inputs), a 16 MHz crystal oscillator, a micro USB connection, an ICSP header, and 3 reset buttons.
13	ARM mbed	Arm Mbed OS is a free, open-source embedded operating system designed specifically for the "things" in the Internet of Things. It includes all the features you need to develop a connected product based on an Arm Cortex-M microcontroller, including security, connectivity, an RTOS, and drivers for sensors and I/O devices.
14	Automotive Grade Linux	Automotive Grade Linux is a collaborative open source project that is bringing together automakers, suppliers and technology companies to accelerate the development and adoption of a fully open software stack for the connected car. With Linux at its core, AGL is developing an open platform from the ground up that can serve as the de facto industry standard to enable rapid development of new features and technologies.

No.	Project	Short Description
15	**BeagleBoard**	Built on the proven BeagleBoard.org® open source Linux approach, BeagleBone® AI fills the gap between small SBCs and more powerful industrial computers. Based on the Texas Instruments AM5729, developers have access to the powerful SoC with the ease of BeagleBone® Black header and mechanical compatibility. BeagleBone® AI makes it easy to explore how artificial intelligence (AI) can be used in everyday life via the TI C66x digital-signal-processor (DSP) cores and embedded-vision-engine (EVE) cores supported through an optimized TIDL machine learning OpenCL API with pre-installed tools.
16	**BipIO**	bip.io is a nodejs based web automation framework that runs 'bips'. A Bip is a web automation agent. A Bip can connect many different web services and perform useful work. It can act on your behalf or perform at your command. Bips can take actions in sequence or in parallel, and chain web services together as you like. They can be put to work via Web Hooks, Email or Trigger when something happens.
17	**Constrained Application Protocol (CoAP)**	The Constrained Application Protocol (CoAP) is a specialized web transfer protocol for use with constrained nodes and constrained networks in the Internet of Things. The protocol is designed for machine-to-machine (M2M) applications such as smart energy and building automation. The core of the protocol is specified in RFC 7252.
18	**Contiki-NG**	Contiki-NG is an operating system for resource-constrained devices in the Internet of Things. Contiki-NG contains an RFC-compliant, low-power IPv6 communication stack, enabling Internet connectivity. The system runs on a variety of platforms based on energy-efficient architectures such as the ARM Cortex-M3/M4 and the Texas Instruments MSP430. The code footprint is on the order of a 100 kB, and the memory usage can be configured to be as low as 10 kB.
19	**DeviceHive**	Open Source IoT Data Platform with the wide range of integration options. Connects any device via REST API, WebSockets or MQTT. The DeviceHive team supports libraries written in various programming languages, including Android and iOS libraries which make the platform device-agnostic. You can even connect such low-end wi-fi enabled devices as ESP8266. DeviceHive gives you the foundation to quickly build analytics the way you want it leveraging the world's best big data solutions: ElasticSearch, Apache Spark, Cassandra and Kafka for real-time and batch processing.
20	**DSA**	Distributed Services Architecture (DSA), is an open source IoT platform that facilitates device inter-communication, logic and applications at every layer of the Internet of Things infrastructure. The objective is to unify the disparate devices, services and applications into a structured and adaptable real-time data model.

No.	Project	Short Description
21	Dweet	Bug Labs offers a “dweet” messaging and alerts platform Dweet helps publish and subscribe to data. dweet.io doesn't require any setup or sign-up— just publish and go. It's machine-to-machine (M2M) for the Internet Of Things (IOT).
22	Eclipse 4DIAC	Eclipse 4diac™ in its current form has been started 2007 as open source project fostering the further development of IEC 61499 for its use in distributed Industrial Process Measurement and Control Systems (IPMCS) and further distribute research results from the original contributors. From the beginning it provided everything necessary to program and execute distributed IPMCS.
23	Eclipse Agail	The Eclipse Agail is a language-agnostic, modular software and hardware gateway framework for the Internet of Things with support for protocol interoperability, device and data management, IoT application execution, trusted data sharing and external Cloud communication.
24	Eclipse Californium	Californium (Cf) is an open source implementation of the Constrained Application Protocol (CoAP). It is written in Java and targets unconstrained environments such as back-end service infrastructures (e.g., proxies, resource directories, or cloud services) and less constrained environments such as embedded devices running Linux (e.g., smart home/factory controllers or cellular gateways). Californium (Cf) has been running code for the IETF standardization of CoAP and was recently reimplemented from scratch having all the experience.
25	Eclipse Concierge	Eclipse Concierge is a small-footprint implementation of the OSGi Core Specifications R5 standard optimized for mobile and embedded devices.
26	Eclipse Cyclone DDS	Eclipse Cyclone DDS is an implementation of the OMG Data Distribution Service (DDS) specification (see http://www.omg.org/spec/DDS/) and the related specifications for interoperability (see http://www.omg.org/spec/DDSI-RTPS/) With 'DATA' being 'the currency of the IOT', having a proper data-sharing technology will be a key-asset in any IOT-platform. The OMG DDS standard is recognized as a highly applicable standard for reliable and robust data-sharing in business- and mission-critical environments (see http://www.iiconsortium.org/IIRA.htm and http://www.iiconsortium.org/IICF.htm) and with that a great fit with the Eclipse-IOT ecosystem. Eclipse Cyclone DDS offers unique data-sharing capabilities compared to the already existing Eclipse solutions (i.e., for messaging).
27	Eclipse Duttile	Eclipse Ditto is a framework for providing the "Digital Twin" pattern for IoT applications in order to interact with IoT devices. That means that Ditto mirrors physical devices as digital representations in the cloud.

No.	Project	Short Description
28	Eclipse fog05	The End-to-End Compute, Storage and Networking Virtualisation solution. Eclipse fog05 is an open source project that aims at providing a decentralised infrastructure for provisioning and managing compute, storage, communication and I/O resources available anywhere across the network. Eclipse fog05 addresses highly heterogeneous systems even those with extremely resource-constrained nodes.
29	Eclipse hawkBit	Project hawkBit aims to create a domain independent back end solution for rolling out software updates to constrained edge devices as well as more powerful controllers and gateways connected to IP based networking infrastructure. Devices can be connected to the hawkBit server either directly through an optimized interface or indirectly through federated device management servers.
30	Eclipse Hono	Eclipse Hono™ provides remote service interfaces for connecting large numbers of IoT devices to a back end and interacting with them in a uniform way regardless of the device communication protocol.
31	Eclipse iceoryx	In domains like automotive, robotics or gaming, a huge amount of data must be transferred between different parts of the system. If these parts are actually different processes on a POSIX based operating system like Linux, this huge amount of data has to be transferred via an inter-process-communication (IPC) mechanism.
32	Eclipse Ignite|IoT	The Ignite|IoT methodology has two main perspectives: (i) IoT Strategy Execution: This perspective looks at IoT strategy from an enterprise perspective, including IoT strategy definition, IoT opportunity identification, IoT business case and IoT programme management; (ii) IoT Solution Delivery: This perspective looks at the individual IoT solution and the related project. Note that it defines the interfaces to the related asset and its organization, but usually excludes design and manufacturing of the asset itself.
33	Eclipse ioFog	Eclipse ioFog is a complete edge computing platform that provides all of the pieces needed to build and run applications at the edge at enterprise scale. The diversity and complexity of edge hardware is abstracted by ioFog Agent software. The management and orchestration of edge microservices is performed by ioFog Controller and its supporting set of components.
34	Eclipse Kapua	Modular IoT cloud platform to manage and integrate devices and their data. A solid integrated foundation of IoT services for any IoT application.
35	Eclipse Keti	Keti is a service that was designed to protect RESTfuls API using Attribute Based Access Control (ABAC). The solution itself is implemented as a cloud-native RESTful API that adheres to the guiding principles of the twelve factor app.

No.	Project	Short Description
36	**Eclipse Keyple**	Eclipse Keyple is an open source framework to facilitate the implementation of terminal processing operating smart card readers, and to defined advanced secure ticketing transaction.
37	**Eclipse Kuksa**	The open and secure Eclipse Kuksa project will contain a cloud platform that interconnects a wide range of vehicles to the cloud via in-car and internet connections. This platform will be supported by an integrated open source software development environment including technologies to cope especially with software challenges for vehicles designed in the IoT, Cloud, and digital era.
38	**Eclipse Kura**	Eclipse Kura™ is an extensible open source IoT Edge Framework based on Java/OSGi. Kura offers API access to the hardware interfaces of IoT Gateways (serial ports, GPS, watchdog, GPIOs, I2C, etc.). It features ready-to-use field protocols (including Modbus, OPC-UA, S7), an application container, and a web-based visual data flow programming to acquire data from the field, process it at the edge, and publish it to leading IoT Cloud Platforms through MQTT connectivity.
39	**Eclipse Leshan**	Eclipse Leshan is an OMA Lightweight M2M (LWM2M) implementation in Java. Leshan provides libraries which help people to develop their own Lightweight M2M server and client. The project also provides a client, a server and a bootstrap server demonstration as an example of the Leshan API and for testing purpose. Leshan relies on the Eclipse IoT Californium project for the CoAP and DTLS implementation.
40	**Eclipse Milo**	Milo is an open-source implementation of OPC UA. It includes a high-performance stack (channels, serialization, data structures, security) as well as client and server SDKs built on top of the stack.
41	**Eclipse Mita**	Eclipse Mita is a programming language for the embedded IoT. We combine a declarative setup of system resources (e.g. Bluetooth connectivity or a temperature sensor) with a modern imperative language. We introduce first-class primitives for sensor access and connectivity to other systems, which allows for quick exploration and integration into IoT ecologies. We support embedded algorithm through powerful primitives, e.g. lists, vectors and statistic functions. In order to prevent problems at compile-time Mita has a static strong type-system. Lastly, we offer direct interaction with sensor data and in-situ developer support by integrating with the embedded software development toolchains.
42	**Eclipse Mosquitto**	Eclipse Mosquitto provides a lightweight server implementation of the MQTT protocol that is suitable for all situations from full power machines to embedded and low power machines. Sensors and actuators, which are often the sources and destinations of MQTT messages, can be very small and lacking in power. This also applies to the embedded machines to which they are connected, which is where Mosquitto could be run.

No.	Project	Short Description
43	Eclipse MRAA	Conceived in 2014, the Eclipse MRAA project aims to deliver a high-level, easy-to-use set of APIs for I/O access on Linux* boards and systems, similar to Arduino* offerings for MCU boards. Initially, the project targeted support for platforms such as the Intel® Galileo, Intel® Edison and Raspberry Pi* boards, highly popular among IoT enthusiasts and makers. Over time, the project evolved substantially and won the appreciation of multiple IoT communities resulting in a number of new board and feature additions. In fact, the project's success prompted Arduino* to adopt the use of the library in the Arduino Create* cloud-based IDE when targeting x86 IoT platforms.
44	Eclipse NeoSCADA	Eclipse NeoSCADA (supervisory control and data acquisition) is a type of industrial control system (ICS). Simply speaking the primary target of Eclipse NeoSCADA is a way to connect different industrial devices to a common communication system and post-process as well as visualize the data to operating personnel.
45	Eclipse OM2M	The Eclipse OM2M project, initiated by LAAS-CNRS, is an open source implementation of oneM2M and SmartM2M standard. It provides a horizontal M2M service platform for developing services independently of the underlying network, with the aim to facilitate the deployment of vertical applications and heterogeneous devices.
46	Eclipse Paho	The Eclipse Paho project provides reliable open-source implementations of open and standard messaging protocols aimed at new, existing, and emerging applications for Machine-to-Machine (M2M) and Internet of Things (IoT).
47	Eclipse Paho Incubator	A permanent incubator for Paho. A permanent incubator is a project that is intended to perpetually remain in the incubation phase. Permanent incubators are an excellent place to innovate, test new ideas, grow functionality that may one day be moved into another project, and develop new committers. Permanent incubator projects never have releases; they cannot participate in the annual simultaneous release. Permanent incubators may have builds, and downloads. They conform to the standard incubation branding requirements and are subject to the IP due diligence rules outlined for incubating projects. Permanent incubators do not graduate.
48	Eclipse Ponte	Ponte is a multi-transport Internet of Things / Machine to Machine broker. As the current state it supports MQTT and REST APIs over HTTP and CoAP.
49	Eclipse Tahu	Eclipse Tahu is a set of references implementations for the Sparkplug specification. It addresses the existence of legacy SCADA/DCS/ICS protocols and infrastructures and provides a much-needed definition of how best to apply MQTT into these existing industrial operational environments.

No.	Project	Short Description
50	Eclipse Thingweb	The Eclipse Thingweb project currently hosts one sub-project and plans for a second one. Thingweb node-wot: node-wot is the official reference implementation of the W3C WoT Working Group and implements the so-called "Servient Architecture". node-wot provides a WoT Thing Description parser and serializer, several "Protocol Bindings" implementing the WoT Binding Templates, as well as a runtime system ("WoT Runtime") providing the WoT Scripting API for applications. It is based on Node.js and its fundamental module structure. node-wot also provides a browser bundle to visualize TDs and to enable the interaction with Things from the Web browser.
51	Eclipse tinydtls	tinydtls is a library for Datagram Transport Layer Security (DTLS) covering both the client and the server state machine. It is implemented in C and provides support for the mandatory cipher suites specified in CoAP.
52	Eclipse Unide	Eclipse Unide provides a lightweight Production Performance Management Protocol (PPMP) server-client implementations (using JSON, REST and other). The core of the protocol are two interface definitions: - PPMP Measurement Interface: This is a public REST API with the purpose of receiving time series and machine data. - PPMP Messages Interface: This is a public REST API with the purpose of receiving machine messages.
53	Eclipse UPM	The Eclipse UPM repository provides software drivers for a wide variety of commonly used sensors and actuators. These software drivers interact with the underlying hardware platform (or microcontroller), as well as with the attached sensors, through calls to Eclipse MRAA APIs. The Eclipse UPM project builds on the solutions of MRAA. While the MRAA project provides an abstraction layer for several IoT platforms, offering developer access to the physical pins and buses, UPM supplies developers with C/C++ sensor libraries with bindings to Java*, JavaScript* and Python*. UPM makes it easier to interface with the sensors bundled with the Intel® IoT Developer Kits and extends the MRAA library.

No.	Project	Short Description
54	**Eclipse VOLTTRON**	VOLTTRON™ is an open source platform for distributed sensing and control. The platform provides services for collecting and storing data from buildings and devices and provides an environment for developing applications which interact with that data.The Eclipse VOLTTRON software platform is Linux-based and capable of running on small, single-board computers as well as in the cloud. It deploys applications known as V-agents that gain access to energy data via automation systems, identifying and diagnosing problems such as erroneous temperature settings. The technology offers passive capabilities that report a problem, as well as active approaches that actually fix the problem. VOLTTRON can be configured to use any protocol needed to interface with devices or data sources.
55	**Eclipse Vorto**	Eclipse Vorto provides a language for describing models and interfaces for IoT Digital Twins. Digital twins are models of entities in the physical world such as a (multi) sensor device, smart power plant, and other entities that participate in IoT solutions. Modeling enables IoT solutions and IoT platforms to provision, use, and configure IoT devices and logical entities from multiple sources in a single solution. Using the vortolang and describing the entities's capabilities, IoT platforms and IoT solutions can leverage the semantics of these IoT entities.
56	**Eclipse Wakaama**	Wakaama is not a library but files to be built with an application. It is written in C and designed to be portable on POSIX compliant systems. Two compilation switches are used: LWM2M_CLIENT_MODE and LWM2M_SERVER_MODE. Defining LWM2M_CLIENT_MODE enables the LWM2M Client interfaces. Defining LWM2M_SERVER_MODE enables the LWM2M Server interfaces. Both can be defined at the same time. Wakaama is mono-threaded. An application can call a HandlePacket API to feed Wakaama with received LWM2M packets and a Step API to perform any pending LWM2M operations. Wakaama provides APIs for a server application to send commands to registered Clients. On client applications, Wakaama checks received commands for syntax and access rights and then dispatches them to the relevant objects.
57	**Eclipse Yasham**	Eclipse Yasham is an Open & Extensible Cloud-Native Smart Living Platform for Consumer IoT.
58	**Eclipse zenoh**	Eclipse zenoh provides is a stack designed to (1) minimize network overhead, (2) support extremely constrained devices, (3) supports devices with low duty-cycle by allowing the negotiation of data exchange modes and schedules, (4) provide a rich set of abstraction for distributing, querying and storing data along the entire system, and (5) provide extremely low latency and high throughput.

No.	Project	Short Description
59	**EdgeX Foundry**	EdgeX Foundry is a vendor-neutral open-source middleware platform hosted by the Linux Foundation, providing a common framework for industrial IoT edge computing. At the core there is a set of loosely coupled microservices organized in different layers.
60	**Ehcache**	Ehcache is an open source, standards-based cache that boosts performance, offloads your database, and simplifies scalability. It's the most widely-used Java-based cache because it's robust, proven, full-featured, and integrates with other popular libraries and frameworks. Ehcache scales from in-process caching, all the way to mixed in-process/out-of-process deployments with terabyte-sized caches.
61	**Exciting Printer**	An open source kit for exploring the possibilities of internet-of-things printing. In a nutshell, Printer is a software system that makes it easy for you to: build your own small internet-connected printers, produce customised content for them, and share that content with other people who also have small internet-connected printers.
62	**Extensible Messaging and Presence Protocol (XMPP)**	XMPP is the Extensible Messaging and Presence Protocol, a set of open technologies for instant messaging, presence, multi-party chat, voice and video calls, collaboration, lightweight middleware, content syndication, and generalized routing of XML data. XMPP was originally developed in the Jabber open-source community to provide an open, decentralized alternative to the closed instant messaging services at that time.
63	**Fledge**	Fledge developers build smarter, better, more cost effective industrial manufacturing solutions to accelerate Industrial 4.0 adoption. Fledge is an open source framework and community for the industrial edge focused on critical operations, predictive maintenance, situational awareness and safety. Fledge is architected to integrate Industrial Internet of Things (IIoT), sensors and modern machines with the cloud and existing "brown field" systems like historians, DCS (Distributed Control Systems), PLC (Program Logic Controllers) and SCADA (Supervisory Control and Data Acquisition). All sharing a common set of administration and application APIs.
64	**Flutter**	Flutter is a programmable processor core for electronics projects, designed for hobbysits, students, and engineers. Flutter features a fast ARM processor, powerful long-range wireless communication, built-in battery charging, and an onboard security chip, making Flutter an ideal choice for robotics, wireless sensor networks, consumer electronics, and educational platforms.
65	**Freeboard**	Bug Labs' Freeboard is an open-source drag-and-drop tool for designing IoT dashboards and visualizations.

No.	Project	Short Description
66	**HabitatMap**	HabitatMap is an environmental technology non-profit building open-source, free, and low-cost environmental monitoring and data visualization solutions. Our tools empower organizations and citizen scientists to measure pollution and advocate for equitable solutions to environmental health issues. We focus on low-income communities and communities of color living with disproportionate environmental burdens.
67	**Hazelcast**	Hazelcast is an open-source distributed in-memory data store and computation platform. It provides a wide variety of distributed data structures and concurrency primitives. Hazelcast IMDG: Open-source distributed In-memory object store supporting a wide variety of data structures such as Map, Set, List, MultiMap, RingBuffer, HyperLogLog. Cloud and Kubernetes friendly. Hazelcast Jet: Open-source distributed stream and batch processing with embedded in-memory storage and a variety of connectors such as Kafka, Amazon S3, Hadoop, JMS and JDBC.
68	**Herald**	Herald provides reliable Bluetooth communication and range finding across a wide range of mobile devices, wearables, and beacons allowing Contact Tracing and other applications to have regular and accurate information to make them highly effective.
69	**Home Assistant**	Open source home automation that puts local control and privacy first. Powered by a worldwide community of tinkerers and DIY enthusiasts. Perfect to run on a Raspberry Pi or a local server.
70	**InterUSS Platform Fund**	The InterUSS Project enables trusted, secure and scalable interoperability between UAS Service Suppliers (USSs) to further safe, equitable and efficient drone operations.
71	**IoTDM**	The Internet of Things Data Management (IoTDM) on OpenDaylight project is about developing a data-centric middleware that will act as a oneM2M compliant IoT Data Broker and enable authorized applications to retrieve IoT data uploaded by any device.
72	**IoTivity**	IoTivity is an open source software framework enabling seamless device-to-device connectivity to address the emerging needs of the Internet of Things. The IoTivity project delivers a reference implementation of the OCF standard specifications. The Open Connectivity Foundation (OCF) specifications are ISO/IEC recognized international standards.

No.	Project	Short Description
73	IoTSyS	IoTSyS is an integration middleware for the Internet of Things. It provides a communication stack for embedded devices based on IPv6, Web services and oBIX to provide interoperable interfaces for smart objects. Using 6LoWPAN for constrained wireless networks and the Constrained Application Protocol together with Efficient XML Interchange an efficient stack is provided allowing using interoperable Web technologies in the field of sensor and actuator networks and systems while remaining nearly as efficient regarding transmission message sizes as existing automation systems. IoTSyS was started within the frame of the FP7 IoT6 European research project and is hosted and further maintained by the Automation Systems Group at the Vienna University of Technology. It is also supported through the Internet Foundation Austria within the NetIdee open source grants and further developed within the project Secure and Semantic Web of Automation (SeWoA, FFG 840206).
74	IPPUSBXD	IPPUSBXD is a userland driver for IPP-over-USB class USB devices. It has been designed for Linux but uses a cross platform usb library allowing eventual porting to Windows and other non-POSIX platforms.
75	Kaa	The Kaa Cloud is an IoT platform-as-a-service (PaaS) accessible for all the registered users with KaaID. This is the Kaa Enterprise IoT Platform in the form of PaaS that you can start using in a matter of seconds! Designed for streamlined user experience, it gives you access to nearly all major Kaa capabilities, with no prior technical expertise required. That's why we believe that the Kaa Cloud is the easiest way to connect your devices to the cloud and enjoy the benefits of remote device management and analytics. It is perfectly suitable for enterprise-grade IoT applications as well as home-grown IoT projects and experiments.
76	KernelCI	kernelci.org is a community based, open source distributed test automation system focused on upstream Linux kernel development. Their goal is to unify all upstream Linux kernel testing efforts in order to provide a single place where to store, view, compare and track these results. It is their mission to detect, bisect, report and fix regressions on upstream Kernel trees before they even reach «mainline».
77	Kinoma	Kinoma, a division of Marvell Semiconductor, is a software engineering group providing an open-source, cross-platform ECMAScript stack aimed at developing software for Internet of Things products and other embedded devices. Additionally, Kinoma provides Kinoma Create, a hardware prototyping platform aimed at the independent maker community.
78	Linux Vendor Firmware Service	The Linux Vendor Firmware Service is a secure portal which allows hardware vendors to upload firmware updates. This site is used by all major Linux distributions to provide metadata for clients such as fwupdmgr and GNOME Software.

No.	Project	Short Description
79	**LinuxBoot**	LinuxBoot is a firmware for modern servers that replaces specific firmware functionality like the UEFI DXE phase with a Linux kernel and runtime.
80	**Macchina.io Edge Device SDK**	macchina.io Edge Device SDK provides a web-enabled, secure, modular and extensible C++ and JavaScript IoT application environment on top of Linux, consisting of middleware, protocols and rich APIs. macchina.io provides ready-to-use and industry proven software building blocks that enable applications to collect, process, filter, analyze and visualize sensor, machine and process data locally, directly where it originates, and where the action happens. macchina.io enables IoT edge and fog computing.
81	**Macchina.io Remote Manager**	macchina.io Remote Manager provides secure remote access to connected devices via HTTP or other TCP-based protocols and applications such as secure shell (SSH) or Virtual Network Computing (VNC). With macchina.io Remote Manager, any network-connected device running the Remote Manager Agent software (WebTunnelAgent, contained in this SDK) can be securely accessed remotely over the internet from browsers, mobile apps, desktop, server or cloud applications.
82	**Mainflux**	Mainflux is performant and secure open-source patent-free IoT platform with the complete full-scale capabilities for development of Internet of Things solutions, IoT applications and smart connected products. It accepts user and thing connections over various network protocols (i.e. HTTP, MQTT, WebSocket, CoAP), thus making a seamless bridge between them. It is used as the IoT middleware for building complex IoT solutions.
83	**Mainspring**	M2MLabs Mainspring is an open source application framework for building machine to machine (M2M) applications such as remote monitoring, fleet management or smart grid. t is written in Java and runs on a standard J2EE6 application server like glassfish 4.1 and JBoss 7 and higher. Data storage is provided by the extremely scaleable NoSQL database from the Apache Cassandra project.
84	**Mango OS**	Radix IoT's flagship platform, Mango OS, is a manufacturer-independent full-stack framework that allows clients – with one location to tens of thousands – to harness their data for better business outcomes. From data acquisition to remote facility monitoring and automation, Radix IoT users have access to an intuitive, out-of-the-box flexible platform to allow disjointed data to be turned into business solutions.

No.	Project	Short Description
85	**Node-RED**	Node-RED is a programming tool for wiring together hardware devices, APIs and online services in new and interesting ways. It provides a browser-based editor that makes it easy to wire together flows using the wide range of nodes in the palette that can be deployed to its runtime in a single-click.
86	**OASIS Message Queuing Telemetry Transport (MQTT)**	Providing a lightweight publish/subscribe reliable messaging transport protocol suitable for communication in M2M/IoT contexts where a small code footprint is required and/or network bandwidth is at a premium.
87	**OASIS SI**	Part of Open-source Architecture Semantic IoT Service-platform project, SI oneM2M Server is a server framework (IN-CSE) that supports interworking of devices and applications based on oneM2M, IoT international standard. Using SI oneM2M Server source code, you can build oneM2M-based device and application interworking server, and you can develop various oneM2M components such as AE, MN-CSE using oneM2M core source code.
89	**oneM2MTester**	oneM2MTester is a oneM2M Conformance testing project aiming to develop and distribute an open source oneM2M Conformance testing tool as well as associated user manuals. The oneM2MTester is initialized by Korea Electronics Technology Institute (KETI) in January 2016, and it is a global collaboration project with 11 members.
90	**Open Garden**	Open Garden is an Open Source alternative to commercial home automation to remotely control your indoor and outdoor plants. The platform consists of three different kits, each ready for a specific kind of growing plant scenario: indoor (houses and greenhouses), outdoor (gardens and fields) and hydroponics (plants in water installations).
91	**openHAB**	The open Home Automation Bus (openHAB) is an open source, technology agnostic home automation platform which runs as the center of your smart home. The open Home Automation Bus (openHAB) project aims at providing a universal integration platform for all things around home automation. It is a pure Java solution, fully based on OSGi.

No.	Project	Short Description
92	OpenIoT	The OpenIoT middleware infrastructure supports flexible configuration and deployment of algorithms for collection, and filtering information streams stemming from internet-connected objects, while at the same time generating and processing important business/applications events. OpenIoT is a joint effort of prominent open source contributors towards enabling a new range of open large scale intelligent IoT (Internet-of- things) applications according to a utility cloud computing delivery model. OpenIoT is perceived as a natural extension to cloud computing implementations, which will allow access to additional and increasingly important IoT based resources and capabilities. In particular, OpenIoT will research and provide the means for formulating and managing environments comprising IoT resources, which can deliver on-demand utility IoT services such as sensing as a service.
93	OpenMTC	OpenMTC is a reference implementation of the oneM2M standard, for conducting applied research and developing innovative M2M and IoT applications. Its horizontal service approach easily integrates devices from different Industrial IoT verticals, independent of the underlying hardware or network infrastructure. The OpenMTC SDK aims to provide developers with a convenient yet flexible tool to write oneM2M compliant applications. This includes network applications (NAs), gateway application (GAs), device applications (DAs), as well as interworking proxy entities (IPEs). The project is part of FIWARE.
94	OpenRemote	OpenRemote is a platform that simplifies connecting networked assets to mobile- and web applications. OpenRemote Protocol agents allow you to connect different data sources, live sensors, and even controls. Whether generic HTTP, MQTT, or very specific protocols like KNX, protocols shouldn't be in your way to create your business application. The OpenRemote rules engine allows for any automation, prediction or optimisation in your system. As it can be programmed with Groovy, Javascript, and a JSON or Flow rules object model, all your developers will quickly understand it. The Account management and Identity service gives you the means to decide who has access to which application or data source. Your applications are designed just once but simultaneously accessible on desktops, and mobile apps for iOS and Android. Designing in HTML5, adding our web components, and using our consoles, it's just an extension to your web strategy.

No.	Project	Short Description
95	OpenThread	OpenThread released by Google is an open-source implementation of Thread®. Google has released OpenThread to make the networking technology used in Google Nest products more broadly available to developers, in order to accelerate the development of products for the connected home and commercial buildings. With a narrow platform abstraction layer and a small memory footprint, OpenThread is highly portable. It supports both System-on-Chip (SoC) and Co-Processor designs.
96	OpenWSN	The goal of the OpenWSN project is to provide open-source implementations of a complete protocol stack based on Internet of Things standards, on a variety of software and hardware platforms. This implementation can then help academia and industry verify the applicability of these standards to the Internet of Things, for those networks to become truly ubiquitous.
97	OS-IoT	ATIS Open Source – Internet of Things (OS-IoT) is an open source software library that simplifies the development of IoT devices that connect to an open, interoperable ecosystem.OS-IoT has a unique focus in the oneM2M software ecosystem which is to support lightweight client applications that can be deployed on constrained devices
98	OSRF Gazebo	Gazebo accurately and efficiently simulates robots for your application, whether it's warehouse logistics, autonomous driving, or space exploration. You get a robust physics engine, high-quality graphics, and programmatic interfaces, including integration with (OSRF) ROS. And it's all open source.
99	OSRF ROS	ROS (Robot Operating System) is a software development kit that helps you build robot applications. From drivers to state-of-the-art algorithms, and with powerful developer tools, ROS has what you need for your next robotics project. And it's all open source.
100	Oxford Flood Network	Low-cost wireless sensors harness the power of the Internet of Things to give you updates about waterways, culverts, rivers, ditches and even groundwater. They're battery powered and connect wirelessly to a gateway which sends the data back to our system using the Internet. A web map visualises your waterways and their levels at https://map.flood.network
101	Particle	Particle is the edge-to-cloud IoT platform to connect, manage, and extract data from physical products.

No.	Project	Short Description
102	Physical Web/Eddystone	The Physical Web is an open approach to enable quick and seamless interactions with physical objects and locations. The Physical Web is an effort to extend the superpower of the web - the URL - to everyday physical objects. Our premise is that you should be able to walk up to any "smart" physical object (e.g. a vending machine, a poster, a toy, a bus stop, a rental car) and interact with it without first downloading an app. The user experience of smart objects should be much like links in a web browser, just tap and use. At its base, the Physical Web is a discovery service: a smart object broadcasts relevant URLs that any nearby device can receive. This simple capability can unlock exciting new ways to interact with the Web.
103	Picos	Picos is an actor-based programming system that supports people-centric, reactive programming on the Internet of Things.
104	PlatformIO	PlatformIO is a cross-platform, cross-architecture, multiple framework, professional tool for embedded systems engineers and for software developers who write applications for embedded products.
105	PrivateEyePi	Home Automation and Monitoring Projects for Raspberry Pi. This is a Raspberry Pi projects website aimed at the Raspberry Pi enthusiast wanting to build home security/automation systems and at the same time learn programming and electronics. All software and source code we provide is free of charge can be copied, shared and modified without restriction. There are no charges for the alarm system or any of the projects.
106	Px4	PX4 is an open source flight control software for drones and other unmanned vehicles. The project provides a flexible set of tools for drone developers to share technologies to create tailored solutions for drone applications. PX4 is hosted by Dronecode, a Linux Foundation non-profit.
107	RabbitMQ	RabbitMQ is the open source message broker. RabbitMQ is lightweight and easy to deploy on premises and in the cloud. It supports multiple messaging protocols. RabbitMQ can be deployed in distributed and federated configurations to meet high-scale, high-availability requirements. RabbitMQ runs on many operating systems and cloud environments, and provides a wide range of developer tools for most popular languages.

No.	Project	Short Description
108	Raspbian	Raspbian is a free operating system based on Debian optimized for the Raspberry Pi hardware. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi. The initial build of over 35,000 Raspbian packages, optimized for best performance on the Raspberry Pi, was completed in June of 2012. However, Raspbian is still under active development with an emphasis on improving the stability and performance of as many Debian packages as possible.
109	RIOT	RIOT is a free, open source operating system developed by a grassroots community gathering companies, academia, and hobbyists, distributed all around the world. RIOT supports most low-power IoT devices and microcontroller architectures (32-bit, 16-bit, 8-bit). RIOT aims to implement all relevant open standards supporting an Internet of Things that is connected, secure, durable & privacy-friendly.
110	seL4	seL4 is a high-assurance, high-performance operating system microkernel. It is unique because of its comprehensive formal verification, without compromising performance. It is meant to be used as a trustworthy foundation for building safety- and security-critical systems. It is available as open source on GitHub and supported by the seL4 Foundation.
111	SiteWhere	SiteWhere is an industrial strength, open source IoT Application Enablement Platform which facilitates the ingestion, storage, processing, and integration of IoT device data at massive scale. The platform leverages a microservices architecture which runs on top of cutting-edge technologies such as Kubernetes, Istio, and Kafka in order to scale efficiently to the loads expected in large IoT projects.
112	SPDX	SPDX is an open standard for communicating software bill of material information, including provenance, license, security, and other related information. SPDX reduces redundant work by providing common formats for organizations and communities to share important data, thereby streamlining and improving compliance, security, and dependability.
113	State of the Edge	State of the Edge is a vendor-neutral platform for open research on edge computing that is dedicated to accelerating innovation by crowdsourcing a shared vocabulary for edge. The project develops free, shareable research that is widely adopted and used to discuss compelling solutions offered by edge computing and the next generation Internet.
114	Tessel	Tessel 2 is a robust IoT and robotics development platform. Leverage all the libraries of Node.JS to create useful devices in minutes with Tessel.

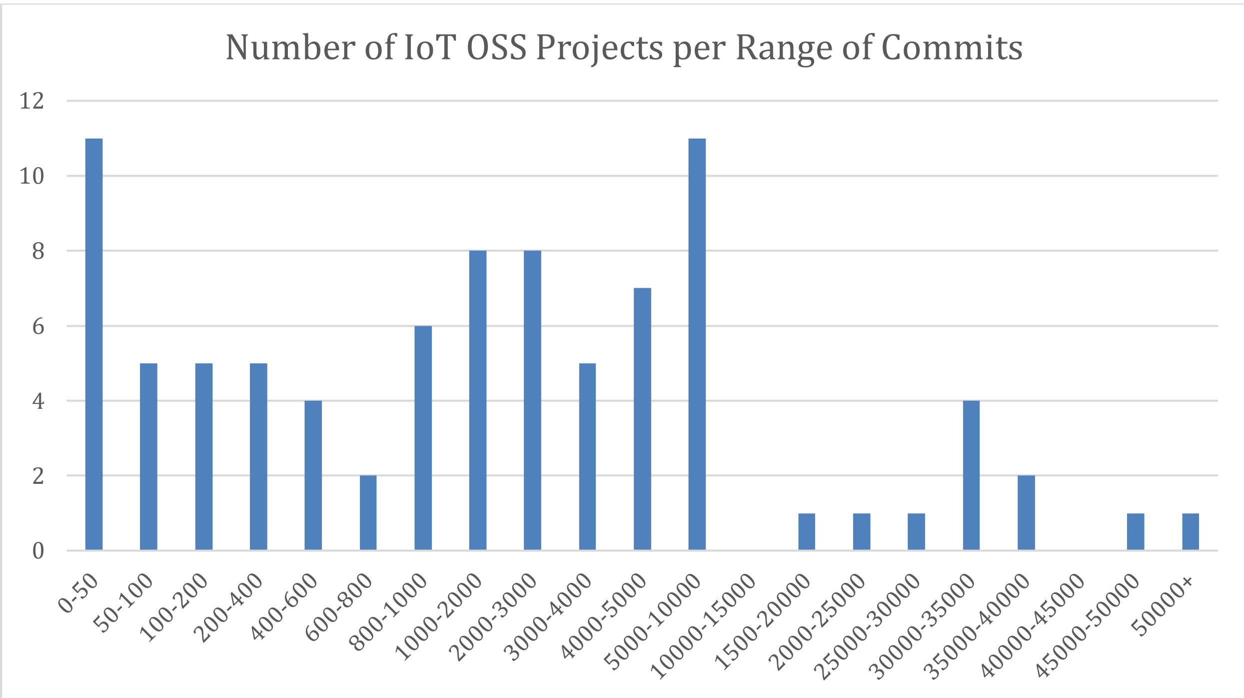
No.	Project	Short Description
115	The Thing System	The Thing System is a set of software components and network protocols that claims to support true automation rather than simple notifications. Its self-learning AI software can handle many collaborative M2M actions without requiring human intervention.Its steward software is written in node.js making it both portable and easily extensible. It can run on your laptop, or fit onto a small single board computer like the Raspberry Pi.
116	ThingBox	The ThingBox is a set of software ready to use, already installed and configured on an SDCard. for Raspberry Pi featuring Node-RED and The Internet of Things technologies, easy to use with a graphical interface. The ThingBox allows anyone to graphically create new unlimited applications interacting with connected objects from a simple web-browser.
117	Thinger.io	Ready to go scalable cloud infrastructure for connecting millions of devices. You can control them with easy to use admin console, or integrate them in your business logic with our REST API.
118	Thingful	Thingful is a search engine for the Internet of Things, enabling secure discoverability & interoperability between millions of public & private connected objects around the world. Our mission is to enable an interoperable Internet of Things, in which connected objects find and use each other's data with the active consent of their owners. Thingful enables IoT data owners to control how their data is used, and empowers them to make more valuable and effective decisions through secure cross-domain IoT search & access.
119	ThingsBoard	ThingsBoard is an open-source IoT platform for data collection, processing, visualization, and device management. It enables device connectivity via industry standard IoT protocols - MQTT, CoAP and HTTP and supports both cloud and on-premises deployments. ThingsBoard combines scalability, fault-tolerance and performance so you will never lose your data.
120	ThingSpeak	ThingSpeak™ is an IoT analytics platform service that allows you to aggregate, visualize and analyze live data streams in the cloud. ThingSpeak provides instant visualizations of data posted by your devices to ThingSpeak. With the ability to execute MATLAB® code in ThingSpeak you can perform online analysis and processing of the data as it comes in. ThingSpeak is often used for prototyping and proof of concept IoT systems that require analytics.
121	Tinq Project	Tinq is completely based on the Qeo publish/subcribe framework produced by Technicolor. Tinq is a software framework that allows devices to easily exchange data with other devices on the same network based on a publish-subscribe paradigm.

No.	Project	Short Description
122	Tizen	Tizen is an open and flexible operating system built from the ground up to address the needs of all stakeholders of the mobile and connected device ecosystem, including device manufacturers, mobile operators, application developers and independent software vendors (ISVs). Tizen is developed by a community of developers, under open source governance, and is open to all members who wish to participate.
123	Ubuntu	Ubuntu Core is Ubuntu for IoT and embedded environments, optimised for security and reliable updates. It's easy to deploy, tamper-resistant, and hardened against corruption. Its read-only root filesystem is built from the same packages used to build the wider set of Ubuntu distributions, and only differs in the way packages are delivered, and crucially, updated. This is all handled by snaps, a secure, confined, dependency-free, cross-platform Linux packaging system.
124	UDOO	UDOO is a family of Open Source Arduino-powered Mini PC compatible with Windows, Android and any Linux Distro. You can exploit both as embedded systems for DIY-electronics projects and as low power consumption, fanless computers for everyday use. The product line involves five single board computers: UDOO QUAD/DUAL (2013), UDOO NEO (2015), UDOO X86 (2016), UDOO BOLT (2019) and UDOO BOLT GEAR (2020).
125	Very Simple Control Protocol (VSCP)	VSCP (Very Simple Control Protocol) is a framework for IoT/m2m tasks. The framework defines methods to have a common device discovery, a common configuration, a common way to interface with remote devices and a common way to update firmware of devices built on different architectures. A server is available that runs on many platforms that have a webserver/websocket/rest/driver and tcp/ip interface with ssl security.
126	Waspmote	Waspmote is an Open Source sensor device to develop Internet of Things projects.
127	webinos	The webinos EU FP7 project defined and delivered an Open Source Platform and software components for the Future Internet in the form of web runtime extensions, to enable web applications and services to be used and shared consistently and securely over a broad spectrum of converged and connected devices, including mobile, PC, home media (TV) and in-car units.
128	WeIO	WeIO is an innovative open source hardware and software platform for rapid prototyping and creation of wirelessly connected interactive objects using only popular web languages such as HTML5 or Python. WeIO consists of the development board and WeIO IDE, locally served (directly from the board) web application that lets you program and control all inputs and outputs on the board from any browser.

No.	Project	Short Description
129	WIZnet	WIZnet (Wizard of Internet) is the IoT Device Platform company. It's unique technology – Hardwired TCP/IP provides better performance and stability than any other software Internet connectivity solutions. We can summarize the main features of Hardwired TCP/IP as Unattackable, High Performance and Easy to Use
130	Yaler	Yaler is a relay infrastructure for secure access to embedded systems. The Yaler relay infrastructure enables secure Web access to embedded systems behind a firewall, NAT or mobile network gateway. All you need on your device is a TCP socket. A simple HTTP handshake makes your Web service running on the device accessible from any Web browser, cURL or other HTTP client, allowing you to remotely monitor and control your device.
131	Yocto	The Yocto Project is an open source collaboration project that helps developers create custom Linux-based systems regardless of the hardware architecture. The project provides a flexible set of tools and a space where embedded developers worldwide can share technologies, software stacks, configurations, and best practices that can be used to create tailored Linux images for embedded and IOT devices, or anywhere a customized Linux OS is needed.
132	Zephyr	The Zephyr Project is a scalable real-time operating system (RTOS) supporting multiple hardware architectures, optimized for resource constrained devices, and built with security in mind. The Zephyr OS is based on a small-footprint kernel designed for use on resource-constrained systems: from simple embedded environmental sensors and LED wearables to sophisticated smart watches and IoT wireless gateways.
133	Zetta	Zetta is an open source platform built on Node.js for creating Internet of Things servers that run across geo-distributed computers and the cloud. Zetta combines REST APIs, WebSockets and reactive programming – perfect for assembling many devices into data-intensive, real-time applications.

APPENDIX B: SCREENSHOTS OF ANALYSIS SPREADSHEETS

License	GitHub	Comments	Days Since Last Push	Days Since Created	Github Stars	Commits	Watchers	Forks
BSD-3-Clause License	https://github.com/ankraft/ACME-oneM2M-CSE	Python 82.9% JavaScript 8.4% CSS 7.7% Other 1.0% using oneM2M standards	0	456	8	850	6	4
AMQP License Agreement	https://github.com/oasis-tcs/amqp-specs	For the OASIS AMQP Members: https://www.amqp.org/about/members	24	668	0	50	30	5
Creative Commons Attribution 4.0 International License. http://creativecommons.org/licenses/by/4.0/	https://github.com/alljoyn	Also: C Java Objective-C Python	1130	1153	27	4671	4	23
Apache-2.0 License	https://github.com/apache/camel/	https://projects.apache.org/project.html?camel	0	4370	3713	52401	275	4268
Apache-2.0 License	https://github.com/apache/ignite	https://projects.apache.org/project.html?ignite	0	2270	3823	27515	289	1682
Apache-2.0 License	https://github.com/apache/iotdb	https://projects.apache.org/project.html?incubator-iotdb	0	896	1288	4867	80	387
Apache-2.0 License	https://github.com/apache/plc4x	https://plc4x.apache.org/	0	1237	339	4117	53	152
Apache-2.0 License	https://github.com/apache/spark		0	2631	29613	30011	2109	23815



REFERENCES

[Biasin20] Biasin, E and Kamenjašević, E. 2020. Open Source Hardware and Healthcare Collaborative Platforms: Common Legal Challenges. Journal of Open Hardware, 4(1): 7, pp. 1–8. DOI: https://doi.org/10.5334/joh.31

[Dabbish12] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. "Social Coding in GitHub: Transparency and Collaboration in an Open Software Repository," Proc. CSCW '12, ACM, pp. 1277-1286, 2012.

[EC-OSS20] COMMUNICATION TO THE COMMISSION, OPEN SOURCE SOFTWARE STRATEGY 2020 – 2023, Think Open, C(2020) 7149 final, Brussels, 21.10.2020

[Eclipse18] Eclipse Foundation, "IoT Developer Survey 2018 Results" April 2018, Copyright (c) 2018, Eclipse Foundation, Inc. | Made available under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

[Eclipse19] Eclipse Foundation, "IoT Developer Survey 2019 Results", April 2019, COPYRIGHT (C) 2019, ECLIPSE FOUNDATION, INC. | MADE AVAILABLE UNDER THE ECLIPSE PUBLIC LICENSE 2.0 (EPL-2.0)

[Eclipse20] Eclipse Foundation, "IoT Commercial Adoption Survey 2019 Results", March 2020, COPYRIGHT (C) 2020, ECLIPSE FOUNDATION, INC.

[ECOSS20] European Commission, OPEN SOURCE SOFTWARE STRATEGY 2020 – 2023, C(2020) 7149 final, Brussels, 21.10.2020

[Eggers20] Günter Eggers, Bernd Fondermann, Berthold Maier, Klaus Ottradovetz, Julius Pfrommer, Ronny Reinhardt, Hannes Rollin, Arne Schmieg, Sebastian Steinbuß, Philipp Trinius, Andreas Weiss, Christian Weiss, Sabine Wilfling, "GAIA-X Technical Architecture", Federal Ministry for Economic Affairs and Energy (BMWi), June 2020.

[Joshi18] Rajive Joshi, Stephen Mellor, Paul Didier, "The Industrial Internet of Things Volume G5: Connectivity Framework", V1.01, Ferbuary 2018, IIC:PUB:G5:V1.01:PB:20180228, available at: https://www.iiconsortium.org/IICF.htm

[Nagel21] Lars Nagel (ed), "Design Principles for Data Spaces", Position Paper, International Data Spaces Association, April 2021, available at: https://design-principles-for-data-spaces.org/

[NGIOT-D1.1] Anna Brékine, Sébastien Ziegler, Francisco Molina Castro, Adriënne Heijnen, "D1.1. IoT Ecosystem Building Vision and Report", EU NGIOT project, Grant Agreement N°: 825082, March 2021.

[OpenLogic20] OpenLogic,"2020 State of Open Source Report Market Trends, Analysis, & Looking Ahead", OpenLogic by Perforce © Perforce Software, Inc. All trademarks and registered, available at: www.openlogic.com

[Qutqut18] Qutqut, Mahmoud & Al-Sakran, Aya & Almasalha, Fadi & Hassanein, Hossam. (2018). Comprehensive Survey of the IoT Open Source OSs. IET Wireless Sensor Systems. 8. 10.1049/iet-wss.2018.5033.

[Sheoran14] Jyoti Sheoran, Kelly Blincoe, Eirini Kalliamvakou, Daniela Damian, and Jordan Ell. 2014. Understanding "watchers" on GitHub. In Proceedings of the 11th Working Conference on Mining Software Repositories (MSR 2014). Association for Computing Machinery, New York, NY, USA, 336–339. DOI:https://doi.org/10.1145/2597073.2597114

[Soldatos20] John Soldatos, "A 360-Degree View of IoT Technologies", Artech House, ISBN: 9781630817527, December 2020.

[VisionMobile15] The essential guide to Open Source in IoT | © VisionMobile |CC BY-SA 4.0 | White Paper.