vf-OS: virtual factory Operating System

WP1: Vision, Scenarios, and Requirements

D1.4a: Existing SOTA Analysis (M6) - Vs: 1.0.2

Deliverable Lead and Editor: Victor Fons, UPV

Contributing Partners: All

Date: 2017-04

Dissemination: Public

Status : EU Approved

Short Abstract

This report acts as a 'coversheet' for the D1.4 SOTA Analysis which is delivered as the vf-OS Wiki (including a vf-OS glossary) at http://www.vf-os.eu/wiki. This wiki is of type 'OTHER' but for administration purposes this brief overview document is provided. The purpose of this Wiki/Task is to allow the partners a comprehensive and up-to-date investigation of existing state-of-the-art architectures, technologies, and solutions supporting collaboration in manufacturing and logistics among the supply chain and thus the concepts of vf-OS.

Grant Agreement: 723710

Document Status

Deliverable Lead	Victor Fons, UPV
Internal Reviewer 1	Danny Pape, ASC
Internal Reviewer 2	Benjamin Menghini, APR
Internal Reviewer 3	Stuart Campbell, ICE
Type	Deliverable
Work Package	WP1: Vision, Scenarios, and Requirements
ID	D1.4a : Existing SOTA Analysis (M6)
Due Date	2017-03
Delivery Date	2017-04
Status	EU Approved

History

See Annex B.

Status

This deliverable is subject to final acceptance by the European Commission.

Further Information

www.vf-OS.eu

Disclaimer

Project Partners:

ICE Information Catalyst

UNIVERSITÉ LUMIÈRE LYON 2

IK4 IKERLAN Research Alliance

MONDRAGON ASSEMBLY

UNINOVA Instituto de Desenvolvimento de Novas Tecnologias

VIASOLIS

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

CONSULGAL

Caixa Mágica Software

KNOWLEDGE BIZ CONSULTING

almende ORGANIZING NETWORKS

APR Applications Plastiques du Rhône

ASCORA Consulting - Development - Research

TARDY

Executive Summary

This report acts as a ‘coversheet’ for the D1.4 SOTA Analysis which is delivered as the vf-OS Wiki (including a vf-OS glossary) at http://www.vf-os.eu/wiki. This wiki is of type ‘OTHER’ but for administration purposes this brief overview document is provided. The purpose of this Wiki/Task is to allow the partners a comprehensive and up-to-date investigation of existing state-of-the-art architectures, technologies, and solutions supporting collaboration in manufacturing and logistics among the supply chain and thus the concepts of vf-OS.

Table of Contents

0 Introduction....1
1 Vf-OS Wiki Overview....3
2 Wiki....5
3 Glossary....8
4 Summary of Solutions....10
5 Version Changes....11
5.1 M18 Changes....11
5.2 M30 Changes....11
6 Conclusions....11
Annex A: History....13
Annex B: References....14
Annex C: Wiki How To....15

0 Introduction

0.1. vf-OS Project Overview

vf-OS – virtual factory Open Operating System – is a project funded by the H2020 Framework Programme of the European Commission under Grant Agreement 723710 and conducted in the period October 2016 until August 2019. It engages 14 partners (Users, Technology Providers, Consultants and Research Institutes) from 7 countries with a total budget of circa 7.5M€. Further information can be found at www.vf-OS.eu.

0.2. Deliverable Purpose and Scope

The purpose of this document "D1.4a Existing SOTA Analysis (M6)" is to act as a 'coversheet' for the D1.4 SOTA Analysis (including the vf-OS Glossary) which is delivered as the vf-OS Wiki at http://www.vf-os.eu/wiki. This wiki is of type 'OTHER' but for administration purposes this brief overview document is provided.

Due to these reasons, this document/process does not follow the standard and comprehensive reporting layout but is a light pointer to the Wiki with some brief background narrative. It is NOT the intent to replicate the Wiki on paper.

The purpose of the Wiki and the overall task is to allow the partners, primarily the technology providers, the opportunity for a comprehensive and up-to-date investigation of existing state-of-the-art architectures, technologies, and solutions supporting collaboration in manufacturing and logistics among the supply chain and thus the concepts of vf-OS. This included the study of the current state of relevant technologies, including cloud computing, IOT, embedded systems, sensors and CPS components, data analytics, Big Data sources, Security, and mobile data sources. The investigation was in close cooperation with the end users and the solutions providers.

In addition, due to the relationship with ongoing FOF projects CREMA and C2NET which are nearing the end of their life time, it was decided to import all their content (for CREMA) and some content for C2NET, to both extend the timeline of their material and take advantage of it.

0.3. Target Audience

The document itself aims primarily at the EU for traceability of the outcomes of type OTHER. The Wiki is public and is relevant to all those working on the project and in the domain such as other RTD projects or organisations which would like to be updated on the vf-OS domains and concepts.

0.4. Deliverable Context

This document is overarching across most if not all vf-OS deliverables but especially influencing and receiving influence from those of WP2 (Architecture) and WP3-7 (RTD).

0.5. Document Structure

This deliverable is broken down into the following sections:

- **Section 1**: **Vf-OS Wiki Overview**: An introduction to the Wiki and this document
- **Section 2**: **Wiki**: Overview of Wiki Article Structure
- **Section 3: Glossary:** Overview of Glossary Structure
- **Section 4: Summary of Solutions:** Summary of technologies per component

- **Section 5 belowVersion Changes:** Placeholder for Wiki changes
- **Section 6: 5 below:** Conclusions to the document and next considerations

0.6. Document Status

This document is listed in the Description of Action as “public” since it represents non-confidential material which is of primary use for the project but may be of use to others as well.

0.7. Document Dependencies

This document is one of a series of three deliverables (D1.4abc) for delivery in the project at M6, M18 and M30. It is noted that the vast majority of the work, circa 80-90%, will have been undertaken in the initial 6 month period (for reporting in D1.4a). This is since the primary purpose of the task is to investigate to prepare for WP2 (Virtual Factory Operating System Architecture). The remain deliverables are to provide updates due to the evolution or augmentation of the technologies and subjects investigated and thus ensure the partners knowledge (and Wiki) are ‘current’

0.8. Glossary and Abbreviations

A definition of common terms related to vf-OS, as well as a list of abbreviations, is available in the supplementary and separate document “vf-OS Glossary and Abbreviations”.

Further information can be found at http://www.vf-OS.eu/glossary.

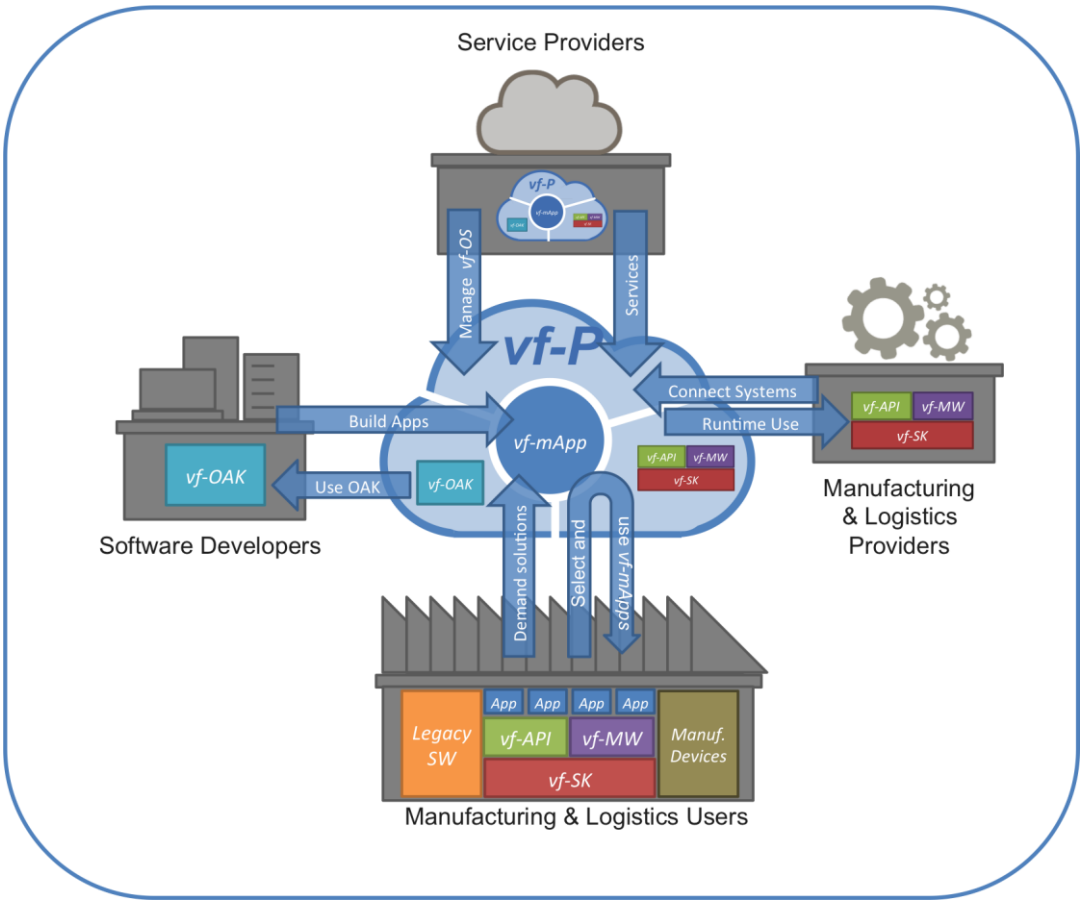
0.9. External Annexes and Supporting Documents

None

0.10. Reading Notes

None

1 Vf-OS Wiki Overview



The goal of vf-OS is to develop an Open Operating System for Virtual Factories composed of a kernel, application programming interface, and middleware specifically designed for the factory of the future. An Open Applications Development Kit (OAK) will be provided to software developers for deploying Manufacturing Smart Applications for industrial users, using the vf-OS Manufacturing Applications Store all operated through a Virtual Factory Platform.

The vf-OS Wiki describes existing solutions giving ground to vf-OS technology solutions. The Wiki identifies standards, commercial solutions and Open Source solutions that could have an impact on the vf-OS solutions. These solutions are grouped together around in various categorisations.

The vf-OS main solutions/components that set the boundaries covered by this Wiki are:

Term	Definition
vf-IO	IO Toolkit is a set of modules that virtualise factory's real assets and connect them to their virtualised images in the vf-OS. vf-IO implements Plug-and-Play mechanisms and device drivers for seamless/open access and smart virtualisation of the factory resources; it is composed by devices drivers, API Connectors, Security and Data Access. Thus, the vf-IO is composed of the modules that enable connectivity to assets like legacy ERPs, CRMs, CPSs, smart objects or wireless sensor networks.
vf-SK	Virtual Factory System Kernel is the core of the operating system, responsible for providing key system wide vf-OS resources and providing a set of specific

	services, which will be open and accessible; it is a specific set of libraries and infrastructure for vApps to be built upon and interact with each other; it is composed of a Framework, Generic Enablers and Manufacturing Enablers.
vf-MW	Virtual Factory Middleware is a set of modules for integrating data from arbitrary sources, ie vf-IO modules. The vf-MW provides a data infrastructure, data storage, data harmonisation, and data analytics features. The use of Cloud-based data storage avoids vendor lock-in issues and minimises the risk of system failures. Consideration will also be given if a particular Cloud-based data storage is not available anymore. Accessibility of data will be facilitated through connectors and wrappers.
vf-Store	Virtual Factory Manufacturing Application Store offers fundamental services of a modern eCommerce platform for consumers and developers. First, vf-Store enables software developers to offer assets (demanded or initiative), and second, users are able to search for, obtain, and rate existing vApps. Furthermore the vf-Store acts as a mediator between developers and users. Therefore, the vf-Store is the central point for developers to get in contact with users. In addition to view/set ratings, reviews, and technical information about the asset's behaviour, the vf-Store supports users to get in contact with developers to offer ideas for new assets.
vf-App	Virtual Factory Manufacturing Smart Application is a vf-OS application, which aim is to enable and optimise communication and collaboration among supply networks, but also within the organisation, of all manufacturing sectors in all the stages of manufacturing and logistic processes: demand forecast, planning, supply, manufacturing, distribution, storage, replacement, and recycling.
vf-OAK	Open Applications Development Kit is composed of a SDK to develop applications, a System Dashboard, the OAK Frontend Environment, the OAK Development Studio and a Developer Engagement Hub to engage developers. The SDK implements all the necessary APIs needed to develop vf-OS Assets. The OAK System Dashboard represents core software services for allowing system monitoring and configuration; the OAK Frontend Environment provides a framework that facilitates a general 'look, feel, and composition' to vf-OS Assets and will assist rapid development, by providing a compilation of UI elements including business logic to the OAK Studio; the vf-OS Development Studio is a desktop development environment that facilitates Technology Providers to compose their applications for running on vf-OS. Additionally, the Developer Engagement Hub is a collaboration platform for the vf-OS community to support each other.
vf-P	Virtual Factory Platform is a holistic service platform, which is the foundation for all services and end user applications that vf-OS will provide. vf-P encapsulates and acts as the interface between the components, connectors, OAK functions, marketplace, the service framework and the end user applications/developers. The vf-P is able to run locally and in Cloud environments.

The Wiki is composed of two main areas: The Wiki and the Glossary which are described in the following sections along with a summary of the technology findings. In the annex is a short how-to guide used by the project partners. The wiki is based on a popular open source tool called 'Mediawiki' and is hosted by UPV. There is a navigation tree at the right of the page and also search box at the top of the page that can be used to find and navigate to the defined knowledge areas.

vf-OS has also found mutual cooperation with the FOF projects CREMA and C2NET (due to an overlap of partners) where content has been embedded in the vf-OS wiki which both assists vf-OS (due to commonalities) and enables the SOTA work of these projects to 'live-on' after their termination in 2017-12. The "living-on" of the vf-OS wiki will also be determined at a later stage in the life of vf-OS.

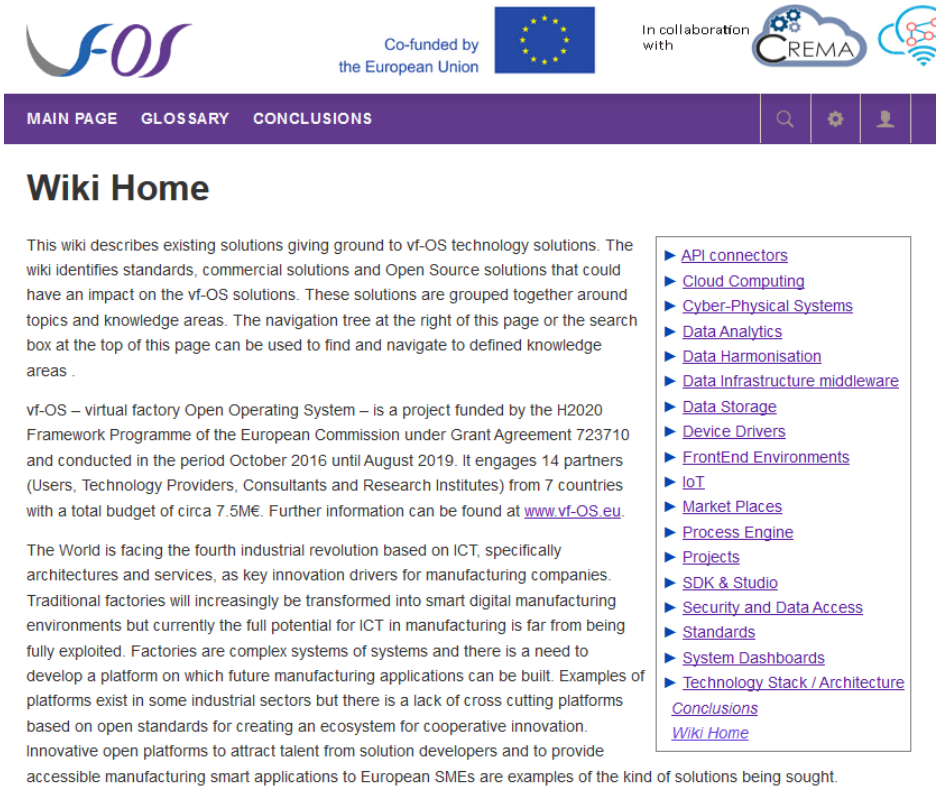
Cooperation with the Connected Factory CSA was also initiated and explored by vf-OS but due to project timings and scope it proved not to be possible. However, the channels are open and vf-OS is open to sharing/linking/merging content in either direction.

This document is one of a series of three deliverables (D1.4abc) for delivery in the project at M6, M18 and M30. However, it is noted that the vast majority of the Wiki work, circa 80-90%, will have been undertaken in the initial 6 month period (for reporting in D1.4a). Further major updates of the Wiki will be recorded in this living document in Section 5.

2 Wiki

The main part of the Wiki is entered via www.vf-os.eu/wiki and the topics mentioned. These topics can be drilled down further as illustrated in the next images.

Top level:



Drill down of the Data Analytics term:

- ▶ API connectors
- ▶ Cloud Computing
- ▶ Cyber-Physical Systems
- ▼ Data Analytics
  - *Big Data (Crema)*
  - *Data Analytics*
  - *Flink*
  - *Hadoop*
  - *ICE Data Gateway*
  - *Samza*
  - *Spark*
  - *Storm*
- ▶ Data Harmonisation
- ▶ Data Infrastructure middleware
- ▶ Data Storage

Selecting Flink as an example: It can be seen that the majority of entries are then composed of a summary of the 'technology', articles/references, related projects/initiatives to the technology and final the relevance to vf-OS.

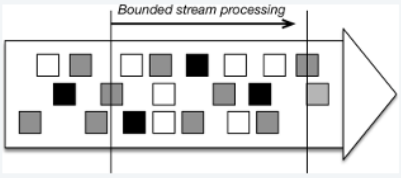
MAIN PAGE GLOSSARY CONCLUSIONS

Flink

Summary | Articles / References | Projects / Initiatives | vf-OS impact [edit]

Apache Flink is an Open Source big data stream processing framework that is designed so that it can also handle batch tasks. To do this, it treats batches to be a bounded data stream. This stream-first approach is referred to as being a Kappa Architecture, and is a simplification of the more commonly known Lambda Architecture. Flink's batch processing model operates as an extension of it's stream processing model. Because of this, Flink handles batches in a different manner to other processing systems, and instead treats batches as a bounded stream. Rather than defining a sliding window, as might be using in stream processing, a global window, which contains all data records to be processed, is defined.

Flink was initially developed as part of the Stratosphere project, before being donated to the Apache Software Foundation as an incubating project in April 2014. It was at this point that the name was changed to Flink, which was selected due to the fact that it reflected the nature of the stream processor: in German, "flink" means fast or agile. Flink completed the incubation very quickly, and became a top-level Apache project in December 2014.



[5]

Main Components

Flink systems are comprised of the following elements:

Articles: (for Flink)

Flink

Summary | Articles / References | Projects / Initiatives | vf-OS impact [edit]

[1] "Introduction to Apache Flink", E. Freidman & K. Tzoumas, O'Reilly, 9781491977163, 2016-09-01

[2] Stratosphere Last accessed Feb. 2017

[3] Kappa Architecture Last accessed Feb. 2017

[4] Hadoop, Storm, Samza, Spark, and Flink: Big Data Frameworks Compared Last accessed Feb. 2017

[5] Benchmarking Streaming Computation Engines at Yahoo! Last accessed Feb. 2017

Projects/Initiatives (for Flink):

Flink

Summary | Articles / References | Projects / Initiatives | vf-OS impact [edit]

- **Apache Flink** : [6]

Category: Data Analytics

vf-OS Impact: The impact tab has a standardised table identifying how the entry topic is relevant to vf-OS – for example, it impacts one or more specific vf-OS components.

Flink

Summary | Articles / References | Projects / Initiatives | vf-OS impact [edit]

Within vf-OS it is crucial to ensure that the Data Analytics components will be able to handle the array of tasks that the various components will perform, and that they are also able to handle the quantity of data that will be produced. As such, it is important to make sure that the data processing framework selected is robust enough to be able to handle a very high throughput of data, while also retaining a high level of fault tolerance.

Topic	How vf-OS can benefit from
vf-OS component this concept is relevant for	Data Analytics
Solutions and results that can be applied to vf-OS	Apache Flink
Main solutions to be considered	Apache Flink
IPR issues	Apache Flink (Apache License. Version 2.0)

3 Glossary

The Wiki also contains the project glossary. As each deliverable is approved by the consortium for submission to the EU the editor of that document is mandated to update the Glossary with new terms and potential changes to terms. The Glossary and these changes are audited by partner ALM. The Glossary can be accessed at www.vf-os.eu/Glossary.

The glossary is divided into four areas (Guidelines, Components, Supporting terms, Abbreviations) which are illustrated below.

Glossary intro:

Glossary

Guidelines | Components | Supporting terms | Abbreviations [edit]

The glossary and abbreviations section of the Wiki is maintained in a collaborative effort by all authors of deliverables in the vf-OS project. Especially each editor will need to go through the deliverable text, check consistent usage of terminology and abbreviations, extract missing terms for this wiki and update this wiki accordingly. During the internal deliverable review process for each deliverable, ALM will go through the Wiki terms and check them on internal consistency, clearness and prevent duplication and/or ambiguity. So, please update the wiki before providing the document for internal review.

The Glossary is split in sections:

1. Components - This contains top level definitions of components in vf-OS, sort of the internal nomenclature
2. Supporting terms - This contains our understanding of the industries standard terminology
3. Abbreviations - This contains a set of abbreviations used along vf-OS documents

Glossary: Components:

Guidelines | Components | Supporting terms | Abbreviations

Please keep this list in Alphabetic order!

vApp

Virtual Factory Manufacturing Smart Application is a vf-OS application, which aim is to enable and optimise communication and collaborat
the stages manufacturing and logistic processes: demand forecast, planning, supply, manufacturing, distribution, storage, replacement an

vf-IO

Virtual Factory Input/Output Interface is a set of modules that virtualise factory's real assets and connect them to their virtualised images i
and device drivers for seamless/open access and smart virtualisation of the factory resources; it is composed by devices drivers, API conr
composed of the modules that enable connectivity to assets like legacy ERPs or CRMs, CPS', smart objects or wireless sensor networks.

vf-MW

Virtual Factory Middleware is a set of modules for integrating data from arbitrary sources, ie vf-IO modules. The vf-MW provides a data in
analytics features. The use of Cloud-based data storage avoids vendor lock-in issues and minimises the risk of system failures. Considera

Glossary: Supporting Terms:

Guidelines | Components | Supporting terms | Abbreviations

Please keep this list in Alphabetic order!

Big Data

Handling of data at a large scale in either volume, variety, volatility or velocity of production. This includes all technology, knowledg
scale of data.

Data Harmonisation

The Data Harmonisation effort aims at providing a way to place data into a common semantic framework, allowing integration and
semantic consistency and standardisation. Goal is to provide a tool to make global decisions, based on global situation awarenes

Distributed Approach

Taking an approach in which the system is formed by independent building blocks, working together by sharing information and se
prevent singleton data storage. Design is mainly bottom-up, instead of top-down.

Future Internet/FI-WARE

A technology stack aimed at providing a common API to IoT devices. This includes the FI-Ware implementation that offers a REST

Glossary: Abbreviations:

Guidelines | Components | Supporting terms | Abbreviations

Please keep this list in Alphabetic order!

API

Application Programming Interface

B2B

Business-to-Business

B2C

Business-to-Consumer

B2MML

Business-to-Manufacturing Mark-up Language

BDA

Big Data Analytics

4 Summary of Solutions

Follows is an extract from the Wiki which points to a summary of solutions/technologies that can be applied in the different areas of vf-OS. The links point to the comprehensive descriptions of the individual vf-OS Wiki Entries as mentioned in the previous section.

vf-OS component	List of solutions

vf-IO	• **API**: Swagger, Aerospike API, Google Firebase Database API, CachetHQ API
	• **Drivers**:
	  • M2M:MQTT, Mosquitto, Contiki, OPC UA, NodeOPCUA
	  • PLC: CODESYS PLCHandler, IEC 61131-3 Virtual Machine, IEC 61131-3 Spec, Twincat ADS Protocol
	  • RFID: TagCentric, Rifidi, Shopfloor Data collection (SFDC)
	  • SCADA: OPC UA, Mango
	  • Sensors: Waspmote, Scilab
	  • Standards: ISO 16100, IEC 61131-3, ISO/IEC 17963
	  • Wireless Sensor Networks: GSM, Waspmote, Nsmam
	
	• **IoT**: IoT Discovery , IoT Broker , Identity Management – KeyRock

vf-SK	• **Process Engine**: Activiti, CBPM
	• **Security and Data Access**:
	  • Authentication, authorization and accounting: FreeRadius, [Apache Rampart
	  • Cryptography: OpenSSL, NSS, EJBCA
	  • Information Security: SELinux, ISO 27000, IEC 62443,
	  • Mobile applications security: Google manifest specifications, WHQL model for the protection of drivers, W3C recommendation for HTML sandboxing
	
	• **Events/Messaging**: smart-ESB model, Secure Event Management (SEM), Dynamic CEP (DyCEP), JADE
	• **FITMAN Specific enablers**: Dynamic CEP (DyCEP), Semantic Mediator (SedMed), Collaborative Asset Management (CAM), Management of Virtualised Assets (MoVa),Shopfloor Data collection (SFDC), Dynamic Visualization and Interaction (DyVisual)
	• **FIWARE enablers**: IoT Discovery , IoT Broker , Identity Management – KeyRock, Cloud Messaging - AEON

vf-MW	• **Data Storage**:
	  • Data Query Platforms: Phoenix, Trafodion
	  • Distributed File Systems: HDFS, OpenStack-Swift
	  • New SQL Databases: CockroachDB
	  • NoSQL Data Storage Infrastructure: Hypertable, Cassandra, MongDB, Neo4J
	  • Spatiotemporal databases: Warp10
	
	• **Data Analytics**:
	  • Batch Processing Frameworks: Hadoop

	• Hybrid Processing Frameworks> Flink • Other projects' solutions: COMBILASER Self-learning system • **Data Harmonisation**: ICE Data Gateway, Talend • **Data Infrastructure middleware**: IDAS Backend Device Management, Orion Context Broker, zeroMQ, DDS, Talend, Global Sensor Network, Akka, CREMA MPM
vf-App	• **Standards identified on pilots**: IndoorGML,

vf-OAK	• **SDK & Studio**: Java SDK (JDK), Eclipse Platform, Android SDK and Studio, IBM Bluemix Cloud
	• **Front-End Environments**:
	  • Behaviour templates: Abcore
	  • Corporate Design: Electron, Ionic 2
	  • Dashboards: CREMA Dashboard
	  • Styleguide: Material Design, Human Interface Guidelines

vf-P	• **Architectures**: AUTOSAR, aXBench, FITMAN Smart Factory Reference Architecture • **Cloud Computing**: OpenStack, and some of its projects NOVA, NEUTRON, SWIFT, CINDER, KEYSTONE, GLANCE, HORIZON, HEA • **Platforms**: Eve Agent Platform

5 Version Changes

This document (D1.4a) is one of a series of three deliverables (D1.4abc) for delivery in the project at M6, M18 and M30. As such, further iterations of this document will include major changes (eg additional subject areas or dropped technologies) in the following sub-sections as well as any changes suggested by reviews.

5.1 M18 Changes

[For version D1.4b]

5.2 M30 Changes

[For version D1.4c]

6 Conclusions

The vf-OS Wiki and its analysis of the existing solutions are the grounding to the technology solutions covered in vf-OS – virtual factory Operating System. The deliverable is a brief overview for administrative purposes only and the ‘real’ deliverable is the vf-OS wiki itself at www.vf-os.eu/wiki.

The Wiki covers a range of solutions, standards, commercial solutions, and open source solutions and which could have an impact on the vf-OS set of solutions. These set of solutions are grouped according to the topic and knowledge area they are part of. For each topic/knowledge area, a description of the topic is provided, a list of categorised solutions provided, articles, and references supporting the description are identified,

projects and initiatives that are a reference in the knowledge area are listed and finally a deeper analysis on how some of the solutions identified are of relevance to relevant components of the vf-OS project.

This document is one of a series of three deliverables (D1.4abc) for delivery in the project at M6, M18 and M30. It is noted that the vast majority of the Wiki work, circa 80-90%, will have been undertaken in the initial 6 month period (for reporting in D1.4a). This is since the primary purpose of the task is to investigate to prepare for WP2 (Virtual Factory Operating System Architecture). The remaining deliverables are to provide updates due to the evolution or augmentation of the technologies and subjects investigated and thus ensure the partners knowledge (and Wiki) are ‘current’. As such, further iterations of this document will include the major changes (eg additional subject areas or dropped technologies).

Finally, should be noted the mutual cooperation with the CREMA and C2NET where content has been embedded in the vf-OS wiki which both assists vf-OS (due to commonalities) and enables the SOTA work of these projects to ‘live-on’ after their termination in 2017-12. The “living-on” of the vf-OS wiki will also be determined at a later stage in the life of vf-OS.

Annex A: History

Document History	
Versions	NB: This concerns the document only and not the Wiki itself V0.9.x: • Draft versions V1.01: • Pre-Final Version (including review) V1.02: • Final Version (including review)
Contributions	UPV: • Victor Alonso Anaya Fons ICE: • Stuart Campbell

Annex B: References

None

Annex C: Wiki How To

See next pages.

vf-OS: virtual factory Operating System

Vf-OS: Wiki How To

1 Guidelines

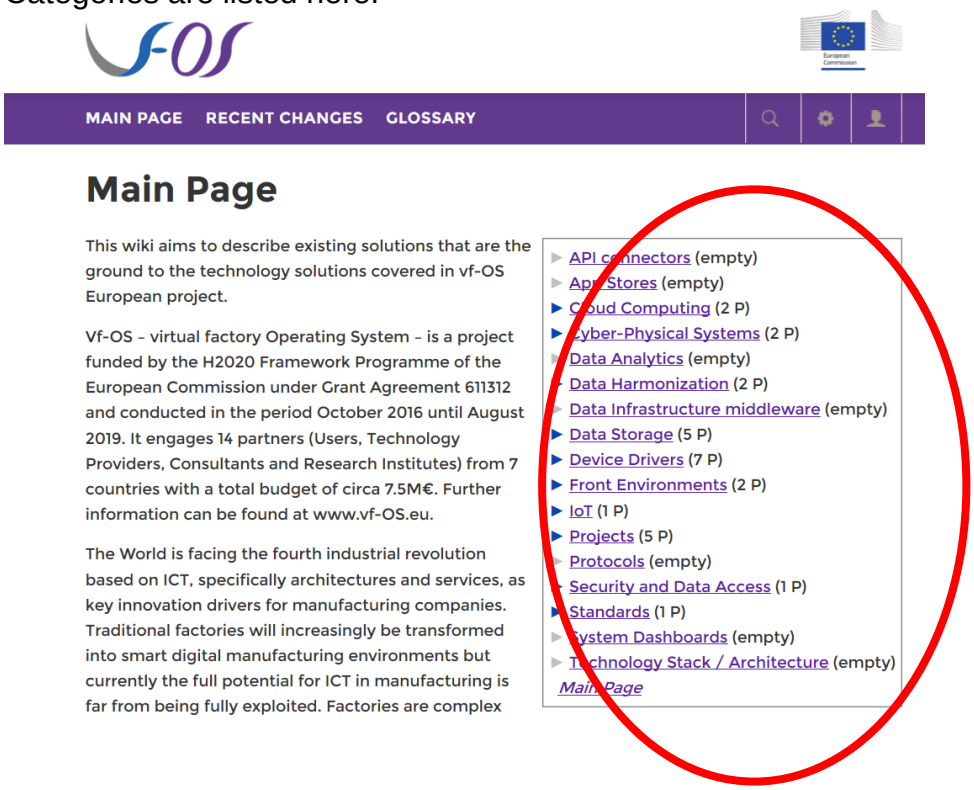
The Wiki is the main result of task 1.4 and it is a deliverable, which means that though it is a wiki, the content should have the same quality as a deliverable. The Wiki is regarding existing related solutions and giving ground to technologies that vf-OS build on top of them, or vf-OS integrates with them, or technologies in the same knowledge area as vf-OS.

As far as there are many technologies that could be related with vf-OS, solutions are grouped in topics (eg PLC, networked sensors, etc). Topics are associated to a page on the Wiki. A given topic (eg PLC) has a summary tab, a solutions tab, an articles/references tab, a projects/initiatives tab and a vf-OS tab – see below for further details.

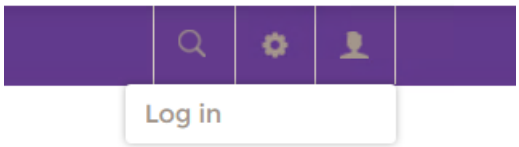
Topics are the main area where content is ground, however topics are also grouped in categories to make navigation of the Wiki easier. Categories are similar to folders in computers, and topics are in this case files. Categories only have a general description (half to one page), but solutions, articles, vf-OS impact, etc are created on pages, not in categories.

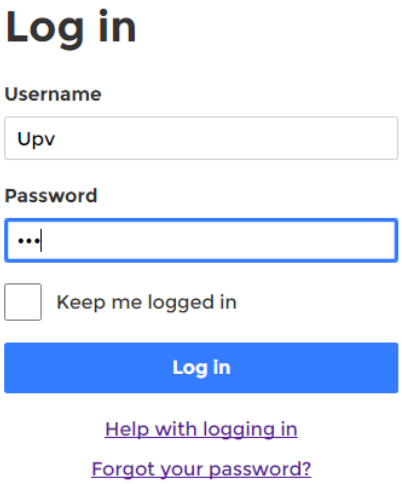
If there is a need for further categories or subcategories, the Wiki lead should be contacted (vicanfon@upvnet.upv.es).

Categories are listed here:

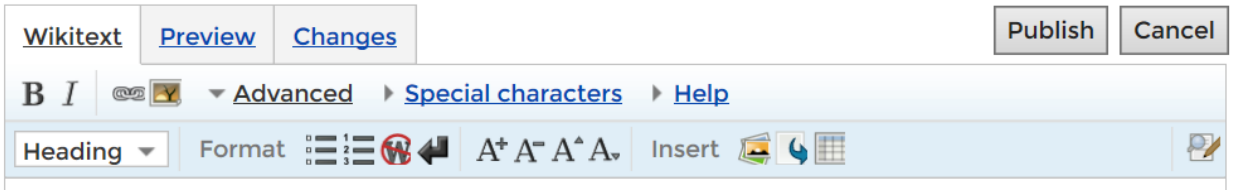


In order to create content an editor must log into the system.



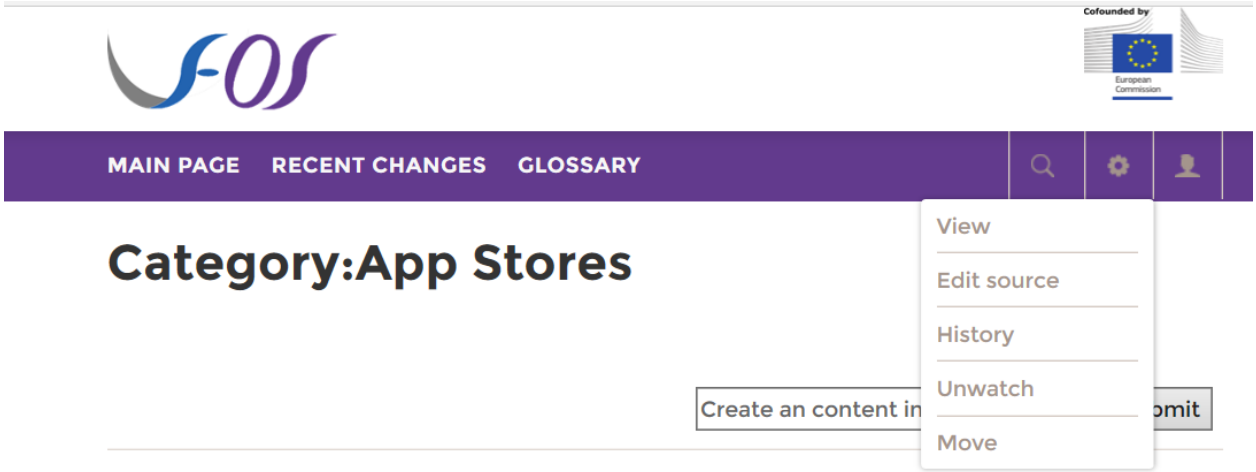


Writing on a wiki it is not so user friendly as writing on Word. vf-OS wiki make use of a specific syntax using specific symbols for including images, bolding words, etc. For this reason, a plugin has been added that makes access to the options easier.


Wikitext
Preview
Changes
Publish
Cancel
B I
Advanced
Special characters
Help
Heading
Format
Insert

2 Describe the Main Category

Navigate to the relevant category and if logged in click on the option edit source under the “cog’ icon.

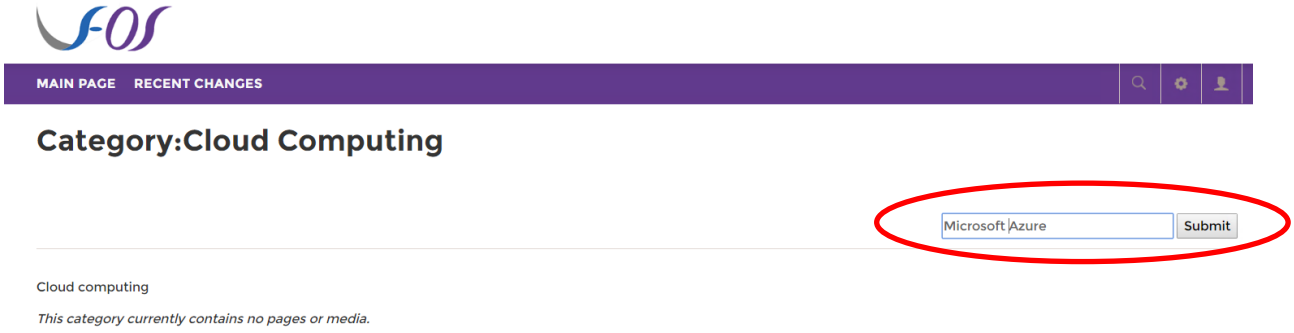


The content expected is the following:

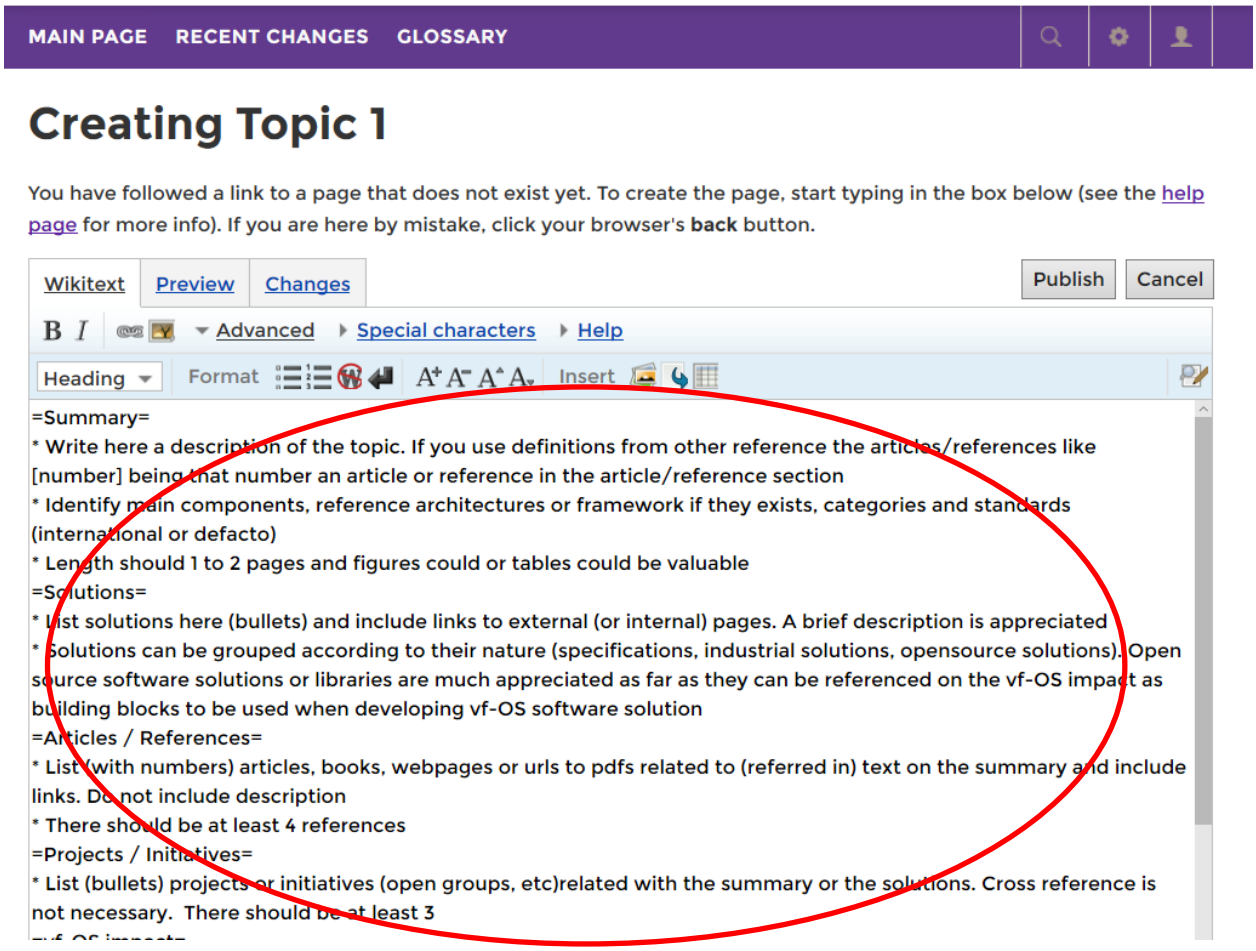
```
You are in a category. Categories group pages of specific areas of knowledge,
and a way to classify pages where the real extensive content is written.
Write here a description of this area of knowledge and main topics and maybe
their relationship. Then create using the textbox at your right pages for every
main topic of this category.
Pages will hold the real information with the description, solutions,
articles/references, projects and vfOS impact.
```

3 Create and describe topics under a category

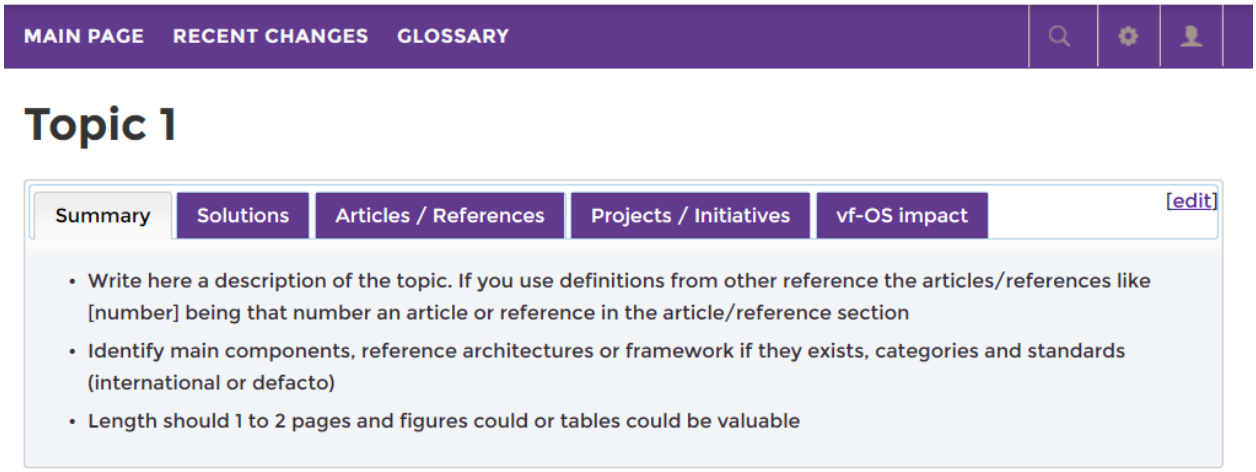
In a given category you can create a topic regarding that category. This is done writing the name of the topic on the box at the top right and clicking submit.



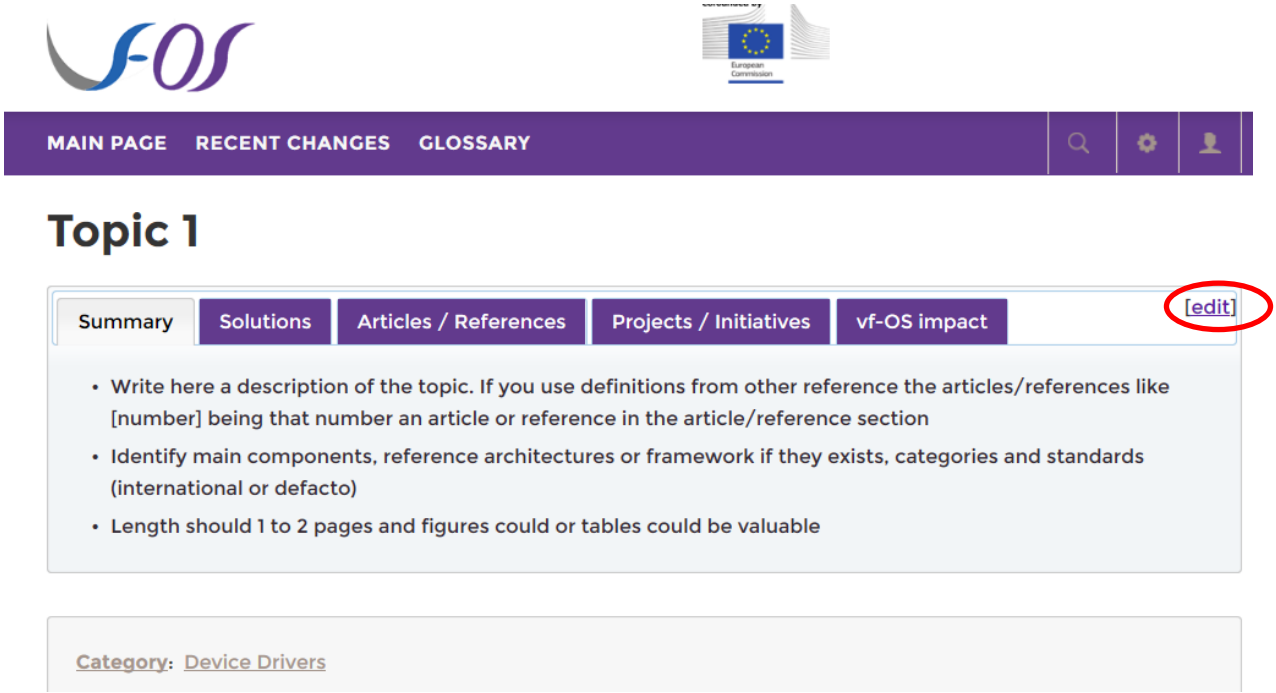
The system will create a page for the topic we want to write and will use a template. However the page that appears here uses mediawiki scary language. It is better if you click on this page publish to get a friendlier page.



After publishing the friendlier view of the template is displayed. From here navigate to each tab to be filled. Follows is an explanation of information to be gathered and extensions to be achieved.



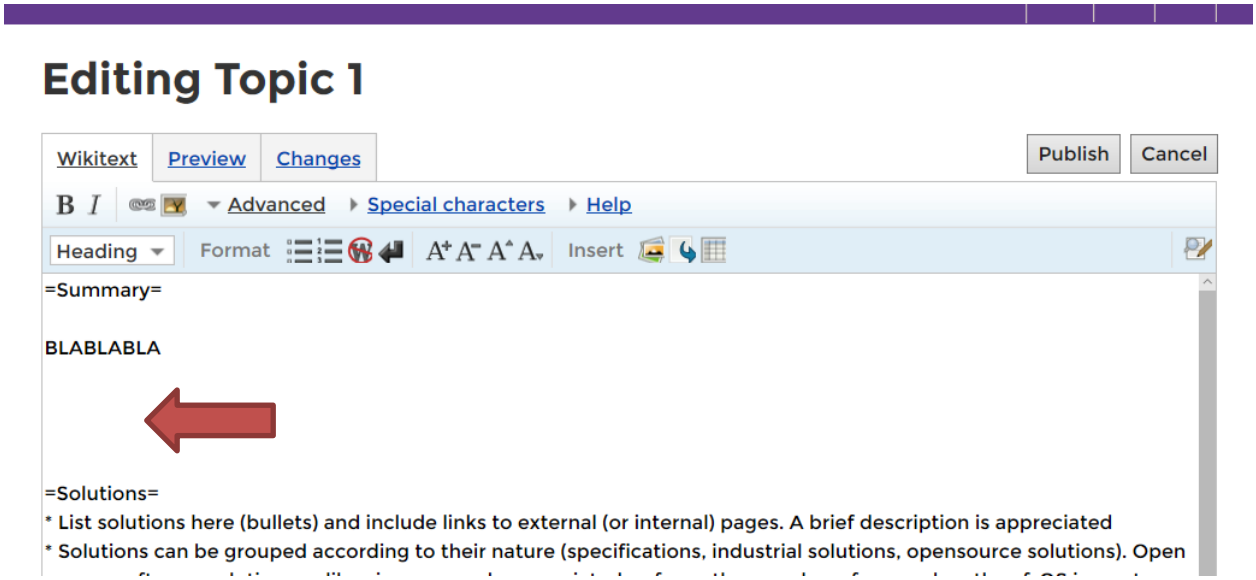
To write a given section click first on the tab (section) and then click the link button at the right. Fill only the content of this tab. Follow the guidelines written on the tab.



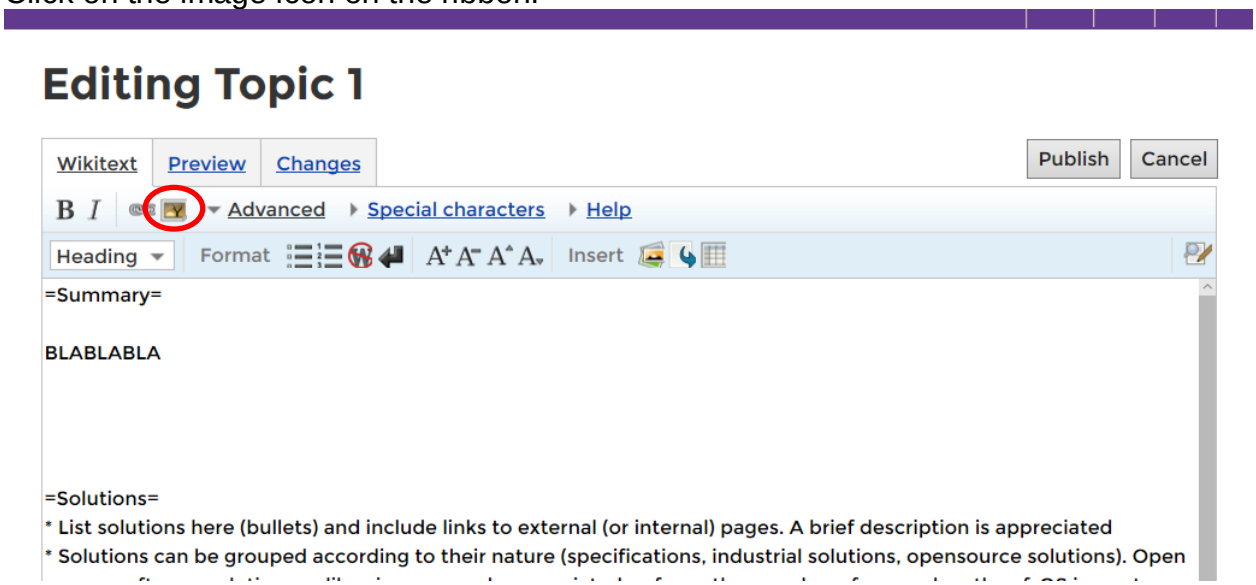
Examples to be considered when writing your own content are:
http://158.42.105.151/mediawiki/index.php/Machine_to_Machine_communication
and
http://158.42.105.151/mediawiki/index.php/PLC

4 Other editing issues

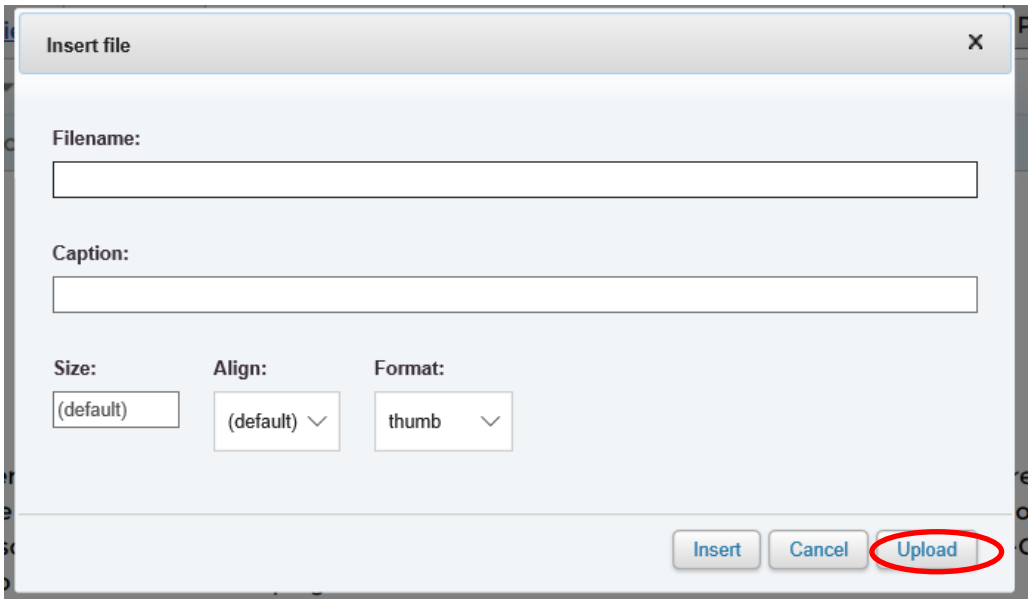
In order to add an image to a section (with architectures, frameworks, etc), first make sure that the image is in a reasonable size (400x400, or similar). If a bigger image is used and it is resized via the Wiki editor it will complain about an error so instead change the size using an image editor such as GIMP beforehand. With the image ready, navigate to the page and section, and while editing, position the cursor on the line where the image is to be added.



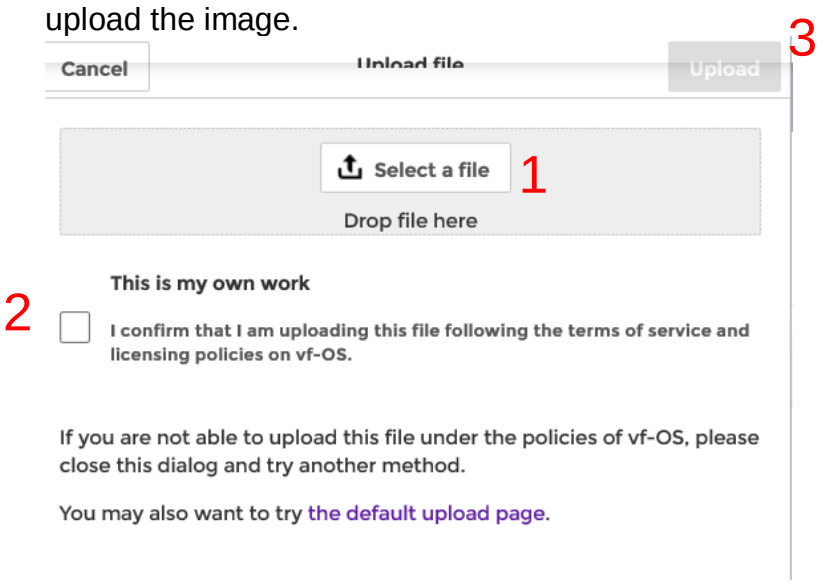
Click on the image icon on the ribbon.



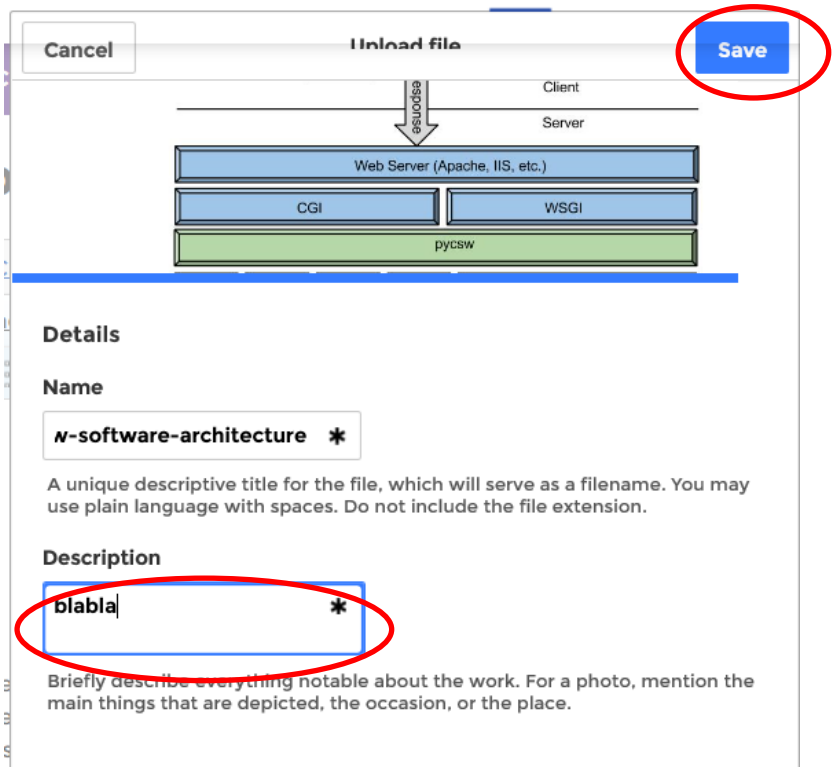
A popup appears, click on the upload button.



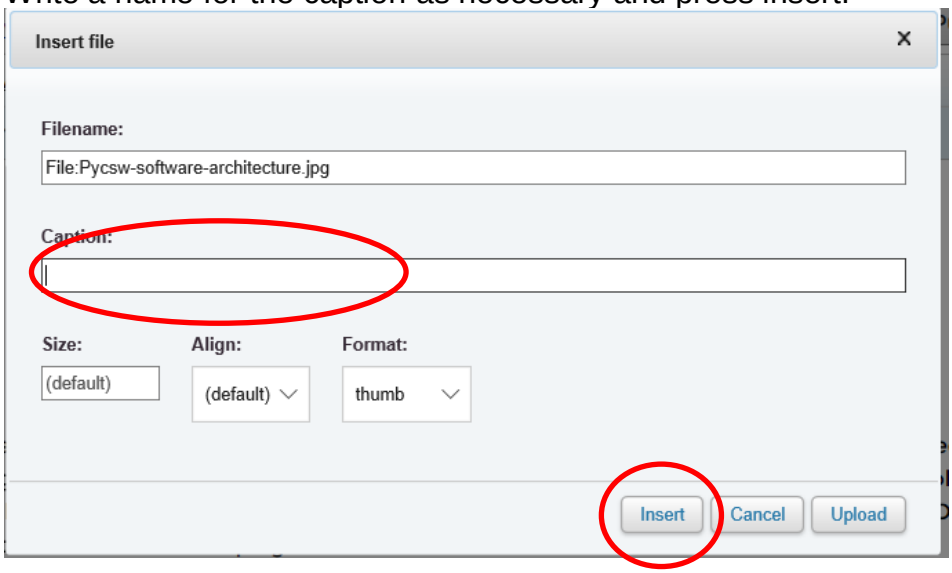
Browse the hard drive to the image to select the file. Check also the box below and then upload the image.



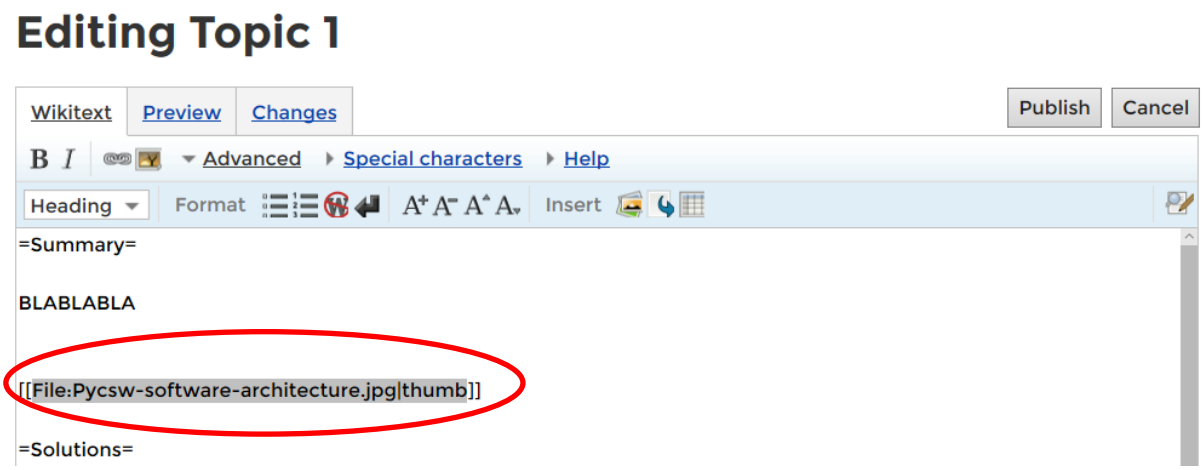
Write the description and save.



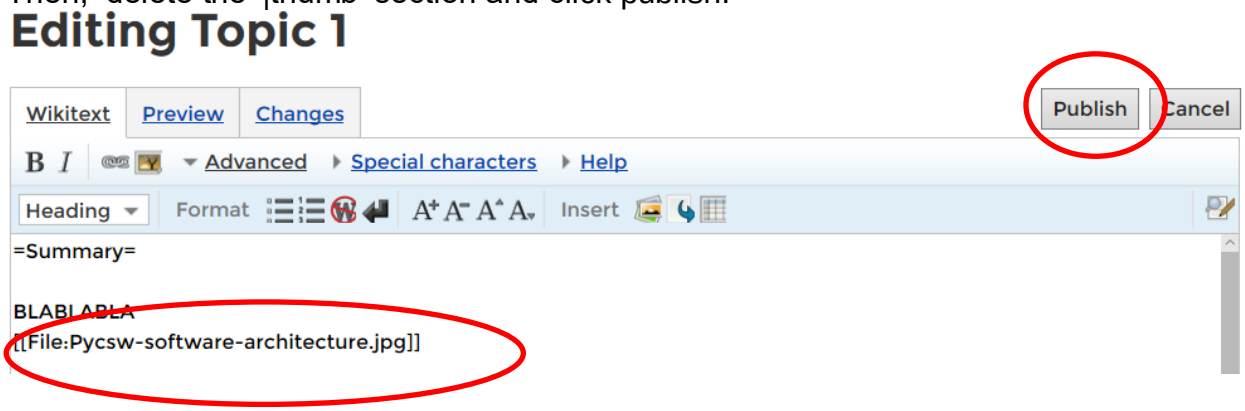
Write a name for the caption as necessary and press insert.



The mediawiki syntax is then inserted. Do not press publish yet.



Then, delete the '|thumb' section and click publish.



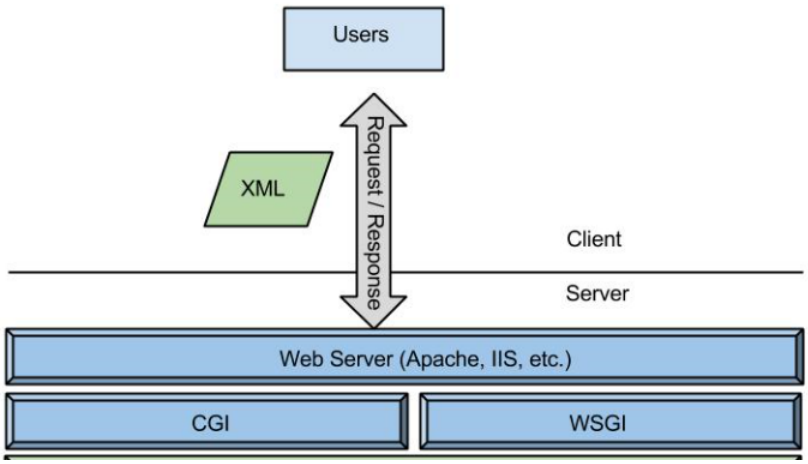
The image will be loaded:

MAIN PAGE RECENT CHANGES GLOSSARY

Topic 1

Summary | Solutions | Articles / References | Projects / Initiatives | vf-OS impact [edit]

BLABLABLA



www.vf-OS.eu