**vf-OS: virtual factory Operating System**

**WP10: Impact**

**D10.4a: Workshops Reports (M26) - Vs: 1.0**

**Deliverable Lead and Editor:** João Sarraipa, UNINOVA

**Contributing Partners:** ICE, UPV, KBZ, IKERLAN

**Date:** 2018-11

**Dissemination:** Public

**Status:** Consortium Approved

**Short Abstract**

This deliverable report two major workshops conducted in the period till M26. Both were designed orientated towards main scientific and research outcomes supported with some technology available during this phase. The second reported workshop was also designed to explore possible synergies with similar projects and to analyse the results achieved so far. Different collaborations and initiatives resulted from such workshops.
**Document Status**

<table>
<thead>
<tr>
<th>Deliverable Lead</th>
<th>João Sarraipa, UNINOVA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internal Reviewer 1</td>
<td>Oscar Garcia, ICE</td>
</tr>
<tr>
<td>Type</td>
<td>Deliverable</td>
</tr>
<tr>
<td>Work Package</td>
<td>WP10: Impact</td>
</tr>
<tr>
<td>ID</td>
<td>D10.4a</td>
</tr>
<tr>
<td>Due Date</td>
<td>2018-11</td>
</tr>
<tr>
<td>Delivery Date</td>
<td>2019-03</td>
</tr>
<tr>
<td>Status</td>
<td>For EU Approval</td>
</tr>
</tbody>
</table>

**History**

See Annex A.

**Status**

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

**Further Information**

[www.vf-OS.eu](http://www.vf-OS.eu)

**Disclaimer**

The views represented in this document only reflect the views of the authors and not the views of the European Union. The European Union is not liable for any use that may be made of the information contained in this document.

Furthermore, the information is provided “as is” and no guarantee or warranty is given that the information is fit for any particular purpose. The user of the information uses it at its sole risk and liability.
Project Partners:
Executive Summary

The activities within the vf-OS project, as in other similar projects, benefit from the public awareness and debate with other specialists. That was a strong motivation behind the design of public workshops within the vf-OS project development with reports at middle and end of the project duration.

The present report, D10.4a – Workshop Reports, is the first of those two documents and reports the two workshops where vf-OS was presented and debated with specialists and other project leaders. From that sequence of events, the present report aims at documenting the materials presented, the dynamic of the workshops, the debate, and the outcomes. These activities come from the document of work; “Importantly, vf-OS commits to organise and lead two major workshops and will coordinate with other related projects to achieve this. One workshop in the middle of the project will identify current achievements and be orientated towards scientific/research/technology outcomes.” In this case two workshops where already organised with different audiences.

The hereby reported workshops were performed at the I-ESA 2018 conference and the other at the Intelligent Systems, IS 2018, conference.
Table of Contents

0 Introduction ............................................................................................................................ 1
1 Smart Services and Business Impact of Enterprise Interoperability @ I-ESA2018 ........ 4
  1.1 Workshop Rationale .......................................................................................................... 4
  1.2 Smart Services vf-OS ......................................................................................................... 4
  1.3 The six vf-OS aspects presented at the workshop ............................................................ 5
  1.4 Conclusions ...................................................................................................................... 5
2 Connected Smart Factories Workshop @ IEEE-IS2018 ................................................... 6
  2.1 Enabling Collaborations .................................................................................................... 6
  2.2 Smart Tools ..................................................................................................................... 7
  2.3 Big Data Management ..................................................................................................... 8
  2.4 Business Modelling ........................................................................................................ 9
  2.5 Discussion ...................................................................................................................... 9
Annex A: History ....................................................................................................................... 11
Annex B: References ................................................................................................................ 12
Annex C: I-ESA’18 Call for Workshops Proposals ................................................................. 13
Annex D: I-ESA’18 vf-OS Workshop Proposal ..................................................................... 16
Annex E: I-ESA’18 vf-OS Workshop Call for Papers ............................................................. 20
Annex F: I-ESA’18 Workshops Programme ......................................................................... 21
Annex G: I-ESA’18 vf-OS Workshop Papers ....................................................................... 27
Annex H: I-ESA’18 vf-OS Workshop Presentations ............................................................... 70
Annex I: I-ESA’18 vf-OS Workshop Pictures ....................................................................... 128
Annex J: IS’18 vf-OS Workshop Details ............................................................................... 132
Annex K: IS’18 vf-OS Connected Factories Workshop Notes .............................................. 133
Annex L: IS’18 vf-OS Workshops presentations .................................................................. 137
Annex M: IS’18 vf-OS Photos ............................................................................................... 333
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.

The World is facing the fourth industrial revolution based on ICT, specifically architectures and services, as key innovation drivers for manufacturing companies. Traditional factories will increasingly be transformed into smart digital manufacturing environments but currently, the full potential of ICT in manufacturing is far from being fully exploited. Factories are complex systems of systems and there is a need to develop a platform on which future manufacturing applications can be built. Examples of platforms exist in some industrial sectors but there is a lack of cross cutting platforms based on open standards for creating an ecosystem for cooperative innovation. Innovative open platforms to attract talent from solution developers and to provide accessible manufacturing smart applications to European SMEs are examples of the kind of solutions being sought.

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 Virtual Factory Platform is an economical multi-sided market platform with the aim of creating value by enabling interactions between four customer groups:

- **Software Developers (independent or within individual manufacturers)** which will build Manufacturing Apps either through innovation or from manufacturing user demand
- **Manufacturing and Logistic Users** which will explore the marketplace for already created solutions, ready to be run on the vf-OS
- **Manufacturing and Logistics Solutions Providers** which will provide ICT interfaces and manufacturing connections
- **Service Providers (vf-OS innovators and third parties)** will make available services (hosting, storage, connected cloud services, etc.) including those based on developed solutions
The Virtual Factory Platform will provide a range of services to the connected factory of the future to integrate better manufacturing and logistics processes. Manufacturing Applications Store will be open to software developers who, using the free Open Applications Development Kit provided, will be able to quickly develop and deploy smart applications to enable and optimise communication and collaboration among supply networks of all manufacturing sectors in all the manufacturing stages and logistic processes.

vf-OS aims to become the reference system software for managing factory related computer hardware and software resources and providing common services for factory computational programs. This operating system will be the component of the system software in a real factory system where all factory application programs will run.

0.2 Deliverable Purpose and Scope

The purpose of this document “D10.4a Workshop Reports” is to present the public discussions that were performed while developing the workshops. The scope of the document comprises presenting the key points of the presentations at the workshops and the activities developed.

0.3 Target Audience

The target audience of this document are the members of the project that can find the summary of what was presented at the vf-OS workshops. At the same time this document enables all other attendants of the workshop to have a document with the presentations and indications about the workshop’s execution. Finally, this document promotes awareness of the project by informing about both workshops organised in the scope of the vf-OS project.

0.4 Deliverable Context

The context of the present deliverable includes the reporting of both workshops:

- Workshop at I-ESA 2018 conference
- Workshop at Intelligent Systems 2018 conference

0.5 Document Structure

This deliverable is broken down into the following sections:

- **Section 1:** Smart Services and Business Impact of Enterprise Interoperability @ I-ESA2018: This section provides the summary of the workshop hosted at the I-ESA’2018 conference
- **Section 2:** Connected Smart Factories Workshop Report at IEEE-IS2018: This section provides the summary of the workshop hosted at the IEEE-IS2018 conference
- **Annexes:**
  - Annex A: History
  - Annex B: References
  - Annex C: I-ESA’18 Call for Workshops Proposals
• Annex D: I-ESA’18 vf-OS Workshop Proposal
• Annex E: I-ESA’18 vf-OS Workshop Call for Papers
• Annex F: I-ESA’18 Workshops Programme
• Annex G: I-ESA’18 vf-OS Workshop Papers
• Annex H: I-ESA’18 vf-OS Workshop Presentations
• Annex I: I-ESA’18 vf-OS Workshop Pictures
• Annex J: IS’18 vf-OS Workshop Details
• Annex K: IS’18 vf-OS Connected Factories Workshop Notes
• Annex L: IS’18 vf-OS Workshops Presentations
• Annex M: IS’18 vf-OS Photos

0.6 Document Status
This document is listed in the Description of Action as “public” since it has it reports public events and material presented to a wide audience.

0.7 Document Dependencies
This document has no dependency to other documents and will have implications in further versions.

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
The annexes do exist and are reported in the Table of Contents and in the Document structure 0.4.1.

0.10 Reading Notes
The present document reports the activity shaped in the DoA as “The vf-OS project will organise and participate in workshops at industrial conferences or during some of their internal meetings. vf-OS will both identify the most relevant conferences in the area (T10.1: Dissemination strategy and plan) and propose ‘minor’ workshops to be held during these events. Importantly, vf-OS commits to organise and lead two major workshops and will coordinate with other related projects to achieve this. The vf-OS project will organise and participate in workshops at industrial conferences or during some of their internal meetings. vf-OS will both identify the most relevant conferences in the area (T10.1: Dissemination strategy and plan) and propose ‘minor’ workshops to be held during these events. Importantly, vf-OS commits to organise and lead two major workshops and will coordinate with other related projects to achieve this.”

Uninova is responsible to organise such actions and all partners participate in the scientific workshops specially the RTD partners while the Industrial partners attend those workshops for the industrial community. Supporting documents and information can be found in Annexes at the end of this document.
1 Smart Services and Business Impact of Enterprise Interoperability @ I-ESA2018

The I-ESA Workshop was performed with the Scientific community attending the I-ESA conference 2018. The workshop was mainly based on the scientific papers (see them at Annex G:) and it was organised answering to the call for workshops presented at Annex C:. In the following the rationale and the main features related to the organised the workshop and finally some conclusions about it are presented.

1.1 Workshop Rationale

Today’s economic environment is being shaped by profound transformations as the world is facing the fourth industrial revolution based on ICT. This trend has impact in promoting specifically architectures and services, as key innovation drivers for manufacturing companies. The devices and processes in traditional factories will increasingly be transformed into smart digital manufacturing environments using new strategies and new processing flows using the flow of knowledge associated with the emerging trend of exponential technologies. Currently, however, there is a lack of capacity to fully implement the potential for ICT in systems and there is a need to develop a platform on which future manufacturing applications can be built.

European companies, especially the SMEs present needs around ICT solutions that are far from being satisfied at affordable costs and lack opportunities to deploy solutions on time to market. Some of the constraints SMEs are facing can be summarised in:

- The integration overheads for ICT that should be reduced
- The need to provide mobile devices that are intuitive, to provide traceability from cradle to grave through the supply chain and to facilitate monitoring, optimisation, and predictive maintenance to enhance availability and productivity.

1.2 Smart Services vf-OS

Manufacturing companies have, so far, been implementing point-solutions, each bringing a specific feature or fixing a specific identified need. Resulting from this approach is a highly heterogeneous manufacturing IT landscape. While these IT landscapes are already costly to administer, further addition of capabilities becomes even more costly because they must be fitted into the heterogeneous IT landscape already existing. There is a need for a homogeneous manufacturing IT landscape offering manufacturing IT solutions to support manufacturing companies, especially SMEs, in successfully responding to the challenge to be flexible and offer highly customised products, and without daunting implementation costs, thereby maintaining and boosting their competitiveness locally and in the increasingly globalised markets they participate in.

As an answer to these needs, an H2020 European project called “Virtual Factory Open Operating System” (vf-OS) funded under the topic FoF-11-2016 “Digital Automation” subtopic “Collaborative Manufacturing and Logistics”, started in October 2016 with a duration of three years. The aim of the Smart Services and Business Impact of Enterprise Interoperability workshop at the I-ESA conference is to demonstrate the research undertaken with the project during its first year. Therefore, vf-OS as organiser of a workshop related to this topic invited submission of related works, see the call for papers at Annex D:. 
1.3 The six vf-OS aspects presented at the workshop

The following listed aspects relates to the six presented papers, which texts and presentations can be seen at Annex G: and Annex H:

- “vf-OS Architecture” describes the vf-OS software architecture. Information and requirements of the real world are included in the structural planning of the complex software system.
- “Enablers Framework: Developing Applications using FIWARE” proposes the Enablers Framework, a component of the vf-OS project that acts as a bridge between applications and enablers, granting programmers an easy access to the services they need.
- “vf-OS IO Toolkit” describes the 10 Toolkit, which provides tools for OnPremise facilitation of the access to manufacturing devices and existing business software, while also facilitating the interaction with the rest of the vf-OS infrastructure.
- “Data Management Component for Virtual Factories Systems” envisages the vf-OS middleware platform and the Data Management Component for covering the issues related to data handling, pre-processing, extracting, and data flows management for Virtual Factories.
- “An Open Environment for Development of Manufacturing Applications on vf-OS” describes a novel environment that was envisioned for providing software developers with an advanced editor with syntax highlighting, providing a large set of artefacts and libraries including standard connectors and extension points, debuggers, and tools to optimise the performance of developed vApps.
- “A Novel Approach to Software Development in the Microservice Environment of vf-OS” describes the vf-OS approach consisting of Docker-based microservices, orchestrated through common REST services with special attention given to the challenge of running resources OnPremise versus purely InCloud.

1.4 Conclusions

The most important aspects were covered during the first year of the vf-OS project development. Those aspects were continuously and consistently being addressed by the technical developers at the vf-OS consortium. They present a variety of solutions for the identified problems presenting in the first section. The workshop served the aims to present those developments to the community, providing a space for debate for both: the project partners and the community, and capture comments and contributions from the workshop attendees. It is aimed to present further developments at the next workshop with next stage evolution of the work currently being performed at vf-OS Project. To find more about the atmosphere of I-ESA workshop a set of related pictures are available at Annex I:. Because of this workshop a cooperation with NIST of USA was established where the specific outcomes from it will be reported in WP10 as a dissemination activity.
2 Connected Smart Factories Workshop @ IEEE-IS2018

The Connected Smart Factories Workshop (26th – 27th September 2018) was organised alongside the IEEE-IS2018 conference in Madeira conducted and reported by Usman Wajid from the Information Catalyst and Marc Jentsch from the Fraunhofer FIT Institute. The workshop brought together industrial representatives from the digital manufacturing and IoT domain, along with the researchers from the areas of smart manufacturing, enterprise information systems, IoT, cloud computing, big data, standardisation, process engineering, business modelling and FIWARE. The aim of the workshop was to analyse and discuss the ongoing developments in the broad area of Connected Smart Factories.

The workshop was supported by 4 European funded projects (COMPOSITION\textsuperscript{1}, DIGICOR\textsuperscript{2}, NIMBLE\textsuperscript{3} and vf-OS\textsuperscript{4}) with a focus on different aspects of smart manufacturing, and with the common vision to contribute towards a Connected Smart Factory platform in Europe.

The workshop program was composed of 19 presentations organised into the 4 thematic sessions, which details about its challenges and features are presented in the next sections:

- Enabling Collaborations (Matchmaking, Automated Negotiations, Process Orchestration)
- Smart Tools (Digital Manufacturing, Smart Factory, IoT)
- Big Data Management (Interoperability, Analytics)
- Business Modelling (Ecosystem Strategies, Simulations)

The workshop was attended by 38 participants, who actively contributed towards the discussions during each of the presentation. The summary of the discussions is provided in the following sections. Further information about the workshop program and its details is reported in Annex J. Some notes reporting the main discussed topics are presented at Annex K.

2.1 Enabling Collaborations

Within this thematic area the presentations (Annex L:) focused on Automated Negotiations and Trust, Matchmaking, Process Orchestration and the role of Marketplaces.

\textbf{Automated Negotiations and Trust}: how do you set the trust values in new collaborations?
Understanding the links among business process and Multi-Sided platforms are critical to automate the negotiation phase. For new collaborations, the trust is generally based on the partner profile. More profile information helps to determine how trustworthy partner is. There can be several places from where the profile can be populated and also validated. Later, the trust values can be gathered from past interactions and transaction monitoring – more data generates better trust values.

\textbf{Matchmaking}: What are the expected benefits of automated match-making mechanisms?

\begin{footnote}
\textsuperscript{1} https://www.composition-project.eu
\textsuperscript{2} https://www.digicor-project.eu
\textsuperscript{3} https://www.nimble-project.org
\textsuperscript{4} http://www.vf-os.eu
\end{footnote}
Team formation and collaborations enable SMEs to jointly respond to the needs coming from OEMs. The benefits of automated matchmaking include: reduced time and cost to perform partner selection and related tasks, broaden access to supplier market, increased number of bids against open tenders and improved manufacturing capacity utilisation.

**Process Orchestration:** What key features of a process design and execution environment can effectively support agile collaborations in digital manufacturing platforms?

A shared (web-based) process environment can support effective collaborations by interlinking and executing distributed activities in a reliable and transparent way. The key attributes for such a process environment include: support for distributed and parallel process design, two-way communication between physical and virtual world to synchronise physical world activities with different activities/tasks in the process, real-time monitoring and alerting of process status, role-based access to processes, interoperability/standardisation of the data-exchange as well as security and privacy of information being processed through the process environment.

**Marketplace:** how to ensure integrity of data and (agent) behaviours in electronic marketplaces?

With the increasing number of electronic marketplaces, there is a need to develop trust models and governance mechanisms that ensure integrity of exchanged data as well as governance of the software agents that operate in the marketplace. For example, a rogue software agent can participate in multiple auctions e.g. to create artificial price hike or to damage the business transaction. The governance mechanisms can impose policies to control certain behaviours and the intelligent decision making of agents in the negotiation processes. This also includes the mechanisms for some kind of traceability on how decisions have been reached. Thus, the topic of governance is quite hot and open at the moment. While emphasising on the mechanisms for structured interactions and conflict resolution, the digital platform developers/owners also need to pay attention to the governance and trust-building mechanisms. Recent advancements in this area include the use of Blockchain for reputation models and to govern B2B interactions and transactions.

### 2.2 Smart Tools

Within this thematic area the presentations (Annex L:) focused on Smart Manufacturing and Control, Enterprise Integration, IoT Tools for Smart Factories and SDKs for Smart Factories.

**Smart Manufacturing:** what are the key focus areas for smart manufacturing tools?

There are extensive efforts on the data front, where new approaches are being devised to allow aggregated and meaningful data to be presented to the smart manufacturing tools and services. Standardisation is also a key area where there is increasing traction on some standards, such as OPC-UA for bringing shop-floor data to higher level services. However, the implementation of smart factory tools often overlooks the use of standards. This fact endangers the interoperability among platforms hampering the creation of a healthy ecosystem where large platforms offer services to other platforms.

**Enterprise Integration:** Why enterprise integration is relevant to the smart manufacturing domain?

The future of manufacturing is about Connected Smart Factories where intra-factory (value chain) and inter-factory (supply chain) need to be aligned to deliver products and services on time. When integrating data from the shop-floor into a central information management
system, BPMN based business process models can be used to connect shop-floor data with relevant systems in a smart way.

**SDK for Smart Factories**: How the SDK can be presented to the users?

The SDK can be offered to the users as a bundled set of libraries or as an integral part of a visual tool such as process environment (in the vf-OS project), where the process designer is used for visual composition of different services in an application.

**IoT Toolkit**: How to address the lack of test datasets required to develop and test IoT applications?

An example is the vf-OS IO Toolkit that facilitates driver development by using a drag and drop approach. However, the lack of relevant data for testing and training is a real problem faced by the developers. A way around this problem is to work with reference and/or generated datasets e.g. International Data Spaces ([www.internationaldataspaces.org](http://www.internationaldataspaces.org)) offers valuable resources here. The developed solutions can be tested through automated testing and benchmarking techniques.

**Planning Tools for Complex Supply Chains**: How a planning tool can help in distributed Supply Chains?

Big companies are transferring the coordination of tender processes to Risk Sharing Partners e.g., Airbus has selected a small number of RSPs from the Tier 1 suppliers list to manage the supply chain downstream. In such situations, the planning tools (e.g. tender facilitator) plays a crucial role to conduct the overall process, allocate partners to components and subcomponents as well as planning and scheduling of procurement processes.

### 2.3 Big Data Management

Within this thematic area the presentations (Annex L:) focused on the techniques for big data management in digital platforms.

**Data Management**: What problems are critical for the big data management platform?

The critical issues for managing big data include: establishing trust in the storage and processing aspects of data management, preserving data integrity, only sharing data with trusted entities, clarifying point of truth for distributed datasets, managing ownership of data throughout the value chain, and ensuring seamless access for application and services

**Data Harmonisation**: How can manufacturing companies, with limited technical knowledgebase, acquire relevant tools to manage their data?

The lack of relevant standards for harmonising different data formats is one of the issues that hinders the wider adoption of digital manufacturing platforms. This is also the issue for establishing interoperability between different platforms. UBL (Universal Business Language), DOM.ONTO or eCl@ss are interesting initiatives that provide ontologies to be applied in this field. In the case of manufacturing and IoT devices there is a lack of ontologies making it very difficult to provide data maps. One recommendation is to consult relevant experts. Another alternative is through crowdsourcing techniques, where more data-maps contribute towards richer reference models.

**Data Analytics**: What analytic techniques are needed or currently sought for?

Digital platforms handle and manage a lot of data, not much of it is used for market exploitation. Statistical models are quite underrated these days, although they are good
enough for most analytic scenarios. Artificial Intelligence based analytic techniques are getting a lot of attention these days. AI techniques kick-in or augment statistical analysis when there is uncertainty. At the end, complex analytics should be easier to maintain and user friendly.

2.4 Business Modelling

Within this thematic area the presentations (Annex L:) focused on Simulations for Business Modelling and the development of ecosystem around digital platforms.

**Business Model Simulations:** Can simulations be used to evaluate business models?

Simulation is an interesting approach to analyse the viability of new business models. The starting point for a new simulation will be determined by different KPIs and dynamic market conditions, which are carefully analysed before being fed into the simulation. Simulations can also be used to evaluate existing business models. The simulation of market factors alongside the performance KPIs of the platform allows us to be more realistic about what we could expect from current business models. From a platform owner perspective simulation also helps us to determine the minimum number of users we need to meet the operation cost of our platform. Moreover, when the simulation uses the market size (number of manufacturers and suppliers) of our use cases, we can see whether there could be revenue indeed or whether the operation cost has to be minimised according to the target market.

**Ecosystem:** How do you foster ecosystem growth?

Digital platforms are driven by the ecosystem of service providers and consumers. Building an ecosystem requires adequate emphasis on barriers and enablers. Barriers include awareness (of benefits, trust etc), slow rate of technology acceptance, interoperability and propensity to change. Enablers include better knowledge of the domain, access to interoperable tools and services, partnerships, and new business opportunities. Annex L: provides more detailed information about different presentations performed in the vf-OS workshop.

2.5 Discussion

The workshop concluded with a round table discussion focusing on the following topics:

**Intelligent Agents vs Intelligent Services:** All project discussed in the workshop had a common feature i.e. the use of intelligent agents to automate human tasks. In not so distant past, intelligent agents were being replaced by intelligent services. While theoretically they are different concepts, technologically they are quite similar except for their communication interfaces. In the past, it was thought that intelligent agents were already doomed because their communication interfaces were not that practical in contrast to intelligent services. However, now in all the projects discussed in the workshop, intelligent agents are present up front. The hypothesis is that intelligent agents, as a technology, have reached a sufficient maturity level to be seriously considered for robust, real-time, highly critical systems – such as the ones being developed in the ongoing European projects.

**Business and Governance Challenges:** Platform owners define the business rules based on the well-established business protocols from specific domains. These rules are often well known to the platform users or made visible at registration and transaction phases. However, governance rules and policies are often less visible. Governance is
about setting out rules of engagement and operations. DESCA (www.desca-2020.eu) is an example of governance rules used to manage the complex interdependency and collaboration needs in the distributed EU projects. In addition to collaboration, another key focus area for governance is data. For example, the data governance models of IDS provide the control mechanism towards data ownership and access.

**Impact of FoF-11-2016 projects on Connected Smart Factories:** A furniture manufacturing platform instance is starting to emerge out of the NIMBLE project. The vf-OS project is leading to a start-up focusing on the smart IoT applications, data harmonisation and analytics. The COMPOSITION project contributes with scientific impacts in the field of agent-based marketplaces. Furthermore, the pilot partner Boston Scientific is planning to roll out the tracking system that is developed within the COMPOSITION project. Finally, the DIGICOR platform is being rolled out to SME communities in the aerospace and automotive sectors by the two partner SME clusters. Annex M: illustrates this workshop atmosphere with some pictures.

*The authors would like to thank the following workshop chairs for their contributions: Violeta Damjanovic- Behrendt (SRFG), Wernher Behrendt (SRFG), Gash Bhullar (C2K), Eduardo Saiz (IKER), Carlos Agostinho (KBZ).*
## Annex A: History

### Document History

<table>
<thead>
<tr>
<th>Versions</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>V0.1:</td>
<td>First Structure of document produced</td>
</tr>
<tr>
<td>V0.2:</td>
<td>Input inserted related to annexes</td>
</tr>
<tr>
<td>V0.3:</td>
<td>Input inserted related to section 2</td>
</tr>
<tr>
<td>V0.4:</td>
<td>Input inserted related to sections 1 and others</td>
</tr>
</tbody>
</table>

### Contributions

<table>
<thead>
<tr>
<th>UPV:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Raul Poler – Document Structure and 1st Workshop contents</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>UNINOVA</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Joao Sarraipa – Main Editor</td>
<td></td>
</tr>
<tr>
<td>Fernando Ferreira – Document Structure; Sections 1</td>
<td></td>
</tr>
<tr>
<td>Elsa Jesus - Other sections</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ICE:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Usman Wajid – Workshop #2 related contents</td>
<td></td>
</tr>
<tr>
<td>Oscar Garcia – Internal review</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>IKERLAIN</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Eduardo Saiz – Input related to Workshop #2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>KBZ:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Carlos Agostinho – Workshop #2 notes</td>
<td></td>
</tr>
</tbody>
</table>
Annex B: References

There are no external references in this document.
Annex C: I-ESA’18 Call for Workshops Proposals

This annex presents the I-ESA’18 Conference Call for Papers, Call for Workshops Proposals and Call for Doctoral Symposium.
Call for Papers

Smart Services and Business Impact of Enterprise Interoperability

We invite papers for the 9th I-ESA conference on INTEROPERABILITY for ENTERPRISE SYSTEMS and APPLICATIONS. IESA attracts researchers, technologists and scientists from around the world to discuss latest developments in Enterprise Interoperability. Join us for technological breakthroughs, outstanding solutions and setting new standards in Interoperability. Be part of the world’s leading community of Interoperability and hand in your paper now.

www.i-esa.org

The programme includes research paper presentations, prominent international keynote speakers, a Doctoral Symposium as well as pre-conference workshops. Best Paper awards will be given. Other notable papers will be published in renowned journals while all accepted papers will be released in the conference proceedings.

Scope “Interoperability” has been defined by INTEROP-VLab as “The ability of an enterprise system or application to interact with others at a low cost in a flexible approach”. Being able to interoperate in collaborations is a step further for enterprises to react flexibly and efficient to market demands, building up consistent and fundamental internal structures and competences and being adaptive in a highly competitive market.

Topics

- Business Interoperability
- Future Internet and Digital Data Space
- Services for Enterprise Interoperability and Digital Twins
- Privacy, Security and blockchain in Enterprise Interoperability
- Interoperability in Industry 4.0
- Interoperability in Industrial Automation
- Platforms and infrastructures for Enterprise Interoperability
- Enterprise Modelling for Enterprise Interoperability
- Ontology and Semantics for Enterprise Interoperability
- Architectures and Frameworks for Interoperability
- Interoperability Application Scenarios
- Standards for Interoperability
- Education on Enterprise Interoperability

International Programme

Senior Committee

Chair
Prof. Klaus-Dieter Thoben
(Bremen Univer. BIBA)

Vice-Chair
Prof. Raul Poler
(Univer. Politecnica Valencia (ES))

Best Paper Award

Most Innovative Solution Award
Key Dates

**Conference** March 22nd to March 23rd, 2018

- Early bird discount* full paper submission due: September 15th, 2017
- Full paper submission deadline: September 30th, 2017
- Author notification due: November 15th, 2017
- Camera-ready copies due: December 15th, 2017

**Workshops** March 20th to March 21st, 2018

- Early-Bird discount* proposals submission due: October 15th, 2017
- Deadline for proposals: November 4th, 2017
- Acceptance notification due: November 16th, 2017
- Deadline for papers: January 31st, 2018
- Camera-ready copies due: April 30th, 2018

**Doctoral Symposium** March 19th, 2018

- Paper submission deadline: November 30th, 2017
- Author notification due: December 15th, 2017

Registration Fees

<table>
<thead>
<tr>
<th></th>
<th>Early Bird (from Jan 16th 2018)</th>
<th>Regular</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conference &amp; Workshop</td>
<td>530€</td>
<td>680€</td>
</tr>
<tr>
<td>Conference</td>
<td>450€</td>
<td>600€</td>
</tr>
<tr>
<td>Workshop</td>
<td>150€</td>
<td>300€</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Early Bird (from Jan 16th 2018)</th>
<th>Regular</th>
</tr>
</thead>
<tbody>
<tr>
<td>Conference &amp; Workshop</td>
<td>430€</td>
<td>530€</td>
</tr>
<tr>
<td>Conference</td>
<td>350€</td>
<td>450€</td>
</tr>
<tr>
<td>Workshop</td>
<td>50€</td>
<td>150€</td>
</tr>
</tbody>
</table>

* There will be an extra 15% discount of registration fees if a conference paper is submitted before September 15th 2017 and finally accepted for the conference or if a workshop proposal is submitted before October 15th 2017 and finally accepted for the workshops.

Venue

The conference will be held at the Fraunhofer Institute for Production Systems and Design Technology, which conducts applied research and development across the whole process spectrum of manufacturing industry. It is located in Germany's famous capital Berlin, well known for its memorials, rivers, lifestyle, and creative-scene, but even more so for the amount of entrepreneurship and progress that is taking place in the ever growing city of innovation. Being a place of both historical significance and modernity makes up the unique feeling Berlin has today.
Annex D: I-ESA’18 vf-OS Workshop Proposal

This annex presents the Workshop proposal prepared at vf-OS Project and sent to the I-ESA’18 organisers.

I-ESA 2018
CALL FOR WORKSHOP PROPOSALS

I-ESA 2018 invites proposals for workshops to be held in conjunction with the main conference. The workshops are focused on recent research and industry issues. The workshop topics are very close to those of the conference, but should be more oriented on project results. The particular goal is to facilitate an active exchange between the speakers and the audience in order to encourage further research and to publish the discussion outcomes in the proceedings. Workshops should also provide opportunities to promote the results of past or present International, European and national research projects in order to facilitate the exploitation of these results.

Suggested workshop topics (other relevant topic proposals would be welcomed):
- Funded collaborative projects (international, national or European),
- Application domains of Enterprise Interoperability research (Product Life Cycle Management and product-service integration, Industry 4.0, risk management, e-health, Smart Cities etc.),
- Research in Next Generation Enterprise Modelling: Semantic interoperability models for configurable manufacturing systems
- Design of Integrated Enterprise Models for industry use (case based team workshop)
- BigData interoperability and Cloud technologies
- Standardisation for sustainable interoperable systems, standards management, open standards

Important dates for the workshops
- Workshop proposals: November 4, 2017
- Acceptance notification due: November 16, 2017
- Workshop deadline for draft papers: December 15, 2017
- Workshop content publication: January 19, 2018
- Final draft papers due: March 6, 2018
- Workshops: March 20–21, 2018
- Publication of Proceedings: June 20, 2018

Please send your proposal by e-mail to the Workshop Chair ws-chair@i-esa.org
<table>
<thead>
<tr>
<th>1. Workshop title</th>
<th>Virtual Factory Operating System</th>
</tr>
</thead>
<tbody>
<tr>
<td>2. Information about the organiser(s)</td>
<td>Norma - Polver</td>
</tr>
<tr>
<td>Organisation: Universitat Politècnica de València</td>
<td>First name: Raul</td>
</tr>
<tr>
<td>Sector activity: Research &amp; Education</td>
<td>Country: Spain</td>
</tr>
<tr>
<td>Email: <a href="mailto:rpoler@cigp.upv.es">rpoler@cigp.upv.es</a></td>
<td>Phone number: +34629046988</td>
</tr>
<tr>
<td>Research interests and areas of expertise: Enterprise Modelling, Collaborative Networks, Supply Chain Management, Knowledge Management, Production Planning and Control, Decision Support Systems, Evolutionary Algorithms.</td>
<td>Name: Goncalves</td>
</tr>
<tr>
<td>First name: Ricardo</td>
<td>Organisation: UNINOVA</td>
</tr>
<tr>
<td>Sector activity: Research &amp; Education</td>
<td>Country: Portugal</td>
</tr>
<tr>
<td>Email: <a href="mailto:rg@uninova.pt">rg@uninova.pt</a></td>
<td>Phone number: +351212948300</td>
</tr>
<tr>
<td>Research interests and areas of expertise: Data Acquisition Systems, Architectures for Systems Integration, and Complex Systems Interoperability.</td>
<td>3. Objective of the workshop, goals, content and topics covered</td>
</tr>
<tr>
<td>Virtual Factory Open Operating System (vf-OS) is a H2020 project funded under the topic FoF-11-2016 “Digital Automation” subtopic “Collaborative Manufacturing and Logistics” started in October 2016, 3 year of duration. The objective of this workshop is to show the research performed in the project during its first year and to contrast with the research performed in other H2020 projects, academia and industry.</td>
<td></td>
</tr>
<tr>
<td>Topics covered:</td>
<td>- Cloud Manufacturing</td>
</tr>
<tr>
<td>- Enterprise Apps</td>
<td>- Cloud Computing</td>
</tr>
<tr>
<td>- Interoperability</td>
<td>- Data Analytics</td>
</tr>
<tr>
<td>- Operating System Virtualisation</td>
<td>- Multi-sided Platforms</td>
</tr>
<tr>
<td>- Virtual Factory Data and Connect</td>
<td>- Internet of Things</td>
</tr>
<tr>
<td>- Virtual Factory Middleware</td>
<td></td>
</tr>
</tbody>
</table>
### 4. Format (draft)
Number of speakers: 6  
Presentations duration: 30 minutes each  
Discussions and debates duration: 30 minutes  
Total duration of the workshop: 3.5 hours  
Half-day or full day?

### 5. Publicity
How will the WS be advertised?
At the vf-OS project Website, Facebook and LinkedIn sites as well as Through the Connected Factory CSA

### 6. Audience
How many participants are expected and what is their background?
30 participants  
Business processes modellers, software architects, software analysts, software developers

### 7. Workshop call for papers (tentative)
with the proposed members of the Program Committee. Workshop organisers have to set up a peer review of the proposed papers
Members of the Program Committee: Stuart Campbell, Raul Poler, Ricardo Gonçalves, Eduardo Sáiz, Nejib Maaila, Raquel Almeida

---

**Call for Workshop Papers**

**Workshop:**  
**Cloud Collaborative Manufacturing Networks**

**Workshop Chairs**
- Raul Poler, Ricardo Gonçalves

You are cordially invited to participate to the Workshop "Virtual Factory Operating System" in the International Conference on Interoperability for Enterprise Systems and Applications I-ESA 2018. Since several years, I-ESA is the major event regarding Enterprise Interoperability and the official conference of Virtual Laboratory for Enterprise Interoperability (INTEROP-VLab) with the sponsorship of the International Federation for Information Processing (IFIP). Accepted and presented papers will be published by ISTE Publications UK.

**Objectives**
Virtual Factory Open Operating System (vf-OS) is a H2020 project funded under the topic FoF-11-2016 “Digital Automation” subtopic “Collaborative Manufacturing and Logistics” started in October 2016, 3 year of duration. The objective of this workshop is to show the research performed in the project during its first year and to contrast with the research performed in other H2020 projects, academia and industry.

**Topics**

**Paper Submission**
Papers must be in English, describes original work and must follow the ISTE guidelines for authors. Paper length should be 6 pages for a full paper and 4 pages for a short paper. The selection of the paper will be performed based on the abstract and not on the full paper.
Authors should submit their contributions by email to the Workshop Chairs:
rpolar@cigip.upv.es, Stuart.Campbell@informationcatalyst.com, rg@uninova.pt

Key Dates

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>15th December 2017</td>
<td>First draft paper submission</td>
</tr>
<tr>
<td>19th January 2018</td>
<td>Notification of acceptance</td>
</tr>
<tr>
<td>6th March 2018</td>
<td>Final draft paper submission</td>
</tr>
<tr>
<td>20th, 21st March 2018</td>
<td>I-ESA 2018 Workshops</td>
</tr>
<tr>
<td>30th April 2018</td>
<td>Camera-ready</td>
</tr>
<tr>
<td>20th June 2018</td>
<td>Publication of Proceedings</td>
</tr>
</tbody>
</table>

Please consider that at least one author per paper must register to the Workshops by January 15th, 2018. If none of the authors is registered, the paper will not be included in the workshop programme and in the proceedings.

Further Information about this Workshop
Enquiries to: Raul Poler
{rpolar@cigip.upv.es}

Further Information about Conference
Website: www.i-esa.org

Please note that Workshop papers must be no longer than 6 pages.
Annex E: I-ESA’18 vf-OS Workshop Call for Papers

This annex shows the vf-OS Workshop Call for Papers in the frame of the I-ESA’18 Conference.

Workshop: Virtual Factory Operating System

Workshop Chairs
- Raul Poler, Ricardo Gonçalves

You are cordially invited to participate to the Workshop “Virtual Factory Operating System” in the International Conference on Interoperability for Enterprise Systems and Applications I-ESA 2018. Since several years, I-ESA is the major event regarding Enterprise Interoperability and the official conference of Virtual Laboratory for Enterprise Interoperability (INTEROP-VLab) with the sponsorship of the International Federation for Information Processing (IFIP).

Accepted and presented papers will be published by ISTE Publications UK.

Objectives

Virtual Factory Open Operating System (vf-OS) is a H2020 project funded under the topic FoF-11-2016 "Digital Automation" subtopic “Collaborative Manufacturing and Logistics” started in October 2016, 3 year of duration. The objective of this workshop is to show the research performed in the project during its first year and to contrast with the research performed in other H2020 projects, academia and industry.

Topics


Paper Submission

Papers must be in English, describes original work and must follow the ISTE guidelines for authors. Paper length should be 6 pages for a full paper and 4 pages for a short paper. The selection of the paper will be performed based on the abstract and not on the full paper.

Authors should submit their contributions by email to the Workshop Chairs: rpoler@cigip.upv.es, rg@uninova.pt

Key Dates

<table>
<thead>
<tr>
<th>Date</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>15th December 2017</td>
<td>First draft paper submission</td>
</tr>
<tr>
<td>19th January 2018</td>
<td>Notification of acceptance</td>
</tr>
<tr>
<td>6th March 2018</td>
<td>Final draft paper submission</td>
</tr>
<tr>
<td>20th, 21st March 2018</td>
<td>I-ESA 2018 Workshops</td>
</tr>
<tr>
<td>30th April 2018</td>
<td>Camera-ready</td>
</tr>
<tr>
<td>20th June 2018</td>
<td>Publication of Proceedings</td>
</tr>
</tbody>
</table>

Please, consider that at least one author per paper must register to the Workshops by January 15th, 2018. If none of the authors is registered, the paper will not be included in the workshop programme and in the proceedings.

Further Information about this Workshop
Enquiries to: Raul Poler (rpoler@cigip.upv.es)

Further Information about Conference
Website: www.i-esa.org
Annex F: I-ESA’18 Workshops Programme

This annex provides the I-ESA’18 Workshops Programme, including the vf-OS Workshop with code A3 and scheduled by Wednesday, March 21st 9:00 to 13:00.

I-ESA 2018 Workshop Programme

<table>
<thead>
<tr>
<th>Workshop</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>From Embedded Intelligence to Directed Manufacturing Decision Support</td>
</tr>
<tr>
<td>Organisers</td>
<td>Bob Young and Gash Bhullar</td>
</tr>
<tr>
<td>Papers and authors</td>
<td>Exploiting embedded intelligence in manufacturing decision support Paul Goodal, Heinz Lugo, Richard Sharpe, Kate Van Lopik, Saragini Pease, Andrew West and Bob Young</td>
</tr>
<tr>
<td></td>
<td>Test of Industrial IoT – Open the Blackbox Frank-Walter Jaekel and Jan Torka</td>
</tr>
<tr>
<td></td>
<td>Intelligent Decision Support Systems in Supply Chains: requirements identification Eduardo Saiz, Raul Poler and Beatriz Andres</td>
</tr>
<tr>
<td></td>
<td>A Total Solution Providers’ perspective on Embedded intelligence in Manufacturing Decision Support Systems Gash Bhullar</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Workshop</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>A2</td>
<td>Business Impact of applications of Enterprise Interoperability</td>
</tr>
<tr>
<td>Organiser</td>
<td>Frank-Walter Jaekel</td>
</tr>
<tr>
<td>Papers and authors</td>
<td>Enterprise Interoperability Management and Artifacts Frank-Walter Jaekel</td>
</tr>
<tr>
<td></td>
<td>Challenges for adaptable energy-efficient production processes Marc Allan Redecker, Alena V. Fedotova, Kay Burow, Quan Deng, Marco Franke, Zied Ghrairi and Klaus-Dieter Thoben</td>
</tr>
<tr>
<td></td>
<td>Interoperability Requirements for Adaptive Production System-of-Systems Georg Weichhart and Alexander Egyed</td>
</tr>
<tr>
<td></td>
<td>Platforms for the Industrial Internet of Things: Enhancing Business Models through Interoperability Soto Setzke, Nicolas Scheidl, Tobias Riasanow, Markus Böhm, and Helmut Krcmar</td>
</tr>
</tbody>
</table>
### Workshop A3: Virtual Factory Operating System (vf-OS)

**Organisers**
Raul Poler and Ricardo Gonçalves

**Papers and authors**
- Vf-OS Architecture
  - Danny Pope, Tobias Hinz, Oscar Garcia Perales, Francisco Fraile, José Luis Flores and Oscar J. Rubio
- Enablers Framework: an approach to develop applications using FIWARE
  - Pedro Corista, Joao Giao, Joao Sarraipa, Raquel Almeida, Oscar G. Perales and Nejib Maalla
- Vf-OS Toolkit
  - Victor Anaya, Nejib Maalla, Ludo Stellingwerff, José Luis Flores and Francisco Fraile
- Data management component for Virtual Factories Systems
  - Artem A. Nazarenko, Joao Giao, Joao Sarraipa, Oscar J. Saiz, Oscar G. Perales and Ricardo Jordim-Gonçalves
- An Open Environment for Development of Manufacturing Applications on vf-OS
  - Carlos Coutinho, Luis Lopes, Vitor Viana, Danny Pape, Gerrit Klasen, Bastianvon Halem, Ludo Stellingwerff and Andries Stam
- A novel approach to software development in the microservice environment of vf-OS
  - Luis Mantelgas da Cunha, Ludo Stellingwerff and Andries Stam Almende BV

### Workshop A4: Corporate Standardisation Management

**Organisers**
Kai Jakobs and Martin Zelm

**Papers and authors**
- Standardization: A multi priority approach
  - Eitan Naveh
- Why Interoperability R&D Needs to be Driven by Agile Integration Standards?
  - Nenad Ivezic
- Managing IT Standardization in Government Towards a Descriptive Reference
  - Dian Baita
- Review: What are Strategies for and Benefits of effective IT Standardization in Government?
  - Dian Baita
- Licensing terms for IoT standard-setting: Do we need “end-user” or “license for all” concepts?
  - Matt Heckman
<table>
<thead>
<tr>
<th>Workshop B1/B2</th>
<th>Industrial big data platforms enabling enterprise interoperability for smart services (Full day workshop)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Organisers</td>
<td>Marten van Sinderen and Sergio Gusmeroli</td>
</tr>
<tr>
<td>Papers and authors</td>
<td>Semantic interoperability for the Internet-of-Things: performance analysis of JSON for Linked Data</td>
</tr>
<tr>
<td></td>
<td>João Moreira, Marten van Sinderen, Luís Ferreira Pires and Patricia Dockhorn Costa</td>
</tr>
<tr>
<td></td>
<td>FIWARE for INDUSTRY: a data-driven Reference Architecture</td>
</tr>
<tr>
<td></td>
<td>Stefano de Panfilis, Ernőe Kovacs, Sergio Gusmeroli, Jorge Rodriguez and Jesus Benedictico</td>
</tr>
<tr>
<td></td>
<td>European Big Data Value Association Smart Manufacturing Industry position paper</td>
</tr>
<tr>
<td></td>
<td>Aníbal Reñones, Davide Dalle Carbonare and Sergio Gusmeroli</td>
</tr>
<tr>
<td></td>
<td>Big data platform architecture for synchronodal transport</td>
</tr>
<tr>
<td></td>
<td>Prince Singh and Marten van Sinderen</td>
</tr>
<tr>
<td></td>
<td>Contextual Data Analytics for Multivariate Anomaly Detection: an Industrial Use Case</td>
</tr>
<tr>
<td></td>
<td>Nenad Stojanovic, Pietrospi Petrali, Marko Dinic and Claudio Turrin</td>
</tr>
<tr>
<td></td>
<td>Fault Prediction in Aerospace Product Manufacturing: development of a model-based Big Data analytics service</td>
</tr>
<tr>
<td></td>
<td>Anna Maria Crespon, Marianna Lezzi, Carla Di Biccar and Mariangela Lazoi</td>
</tr>
<tr>
<td></td>
<td>A SAREF extension for semantic interoperability in the industry and Manufacturing domain</td>
</tr>
<tr>
<td></td>
<td>Laura M. Daniele Matthys Punter, Raúl García Castro, María Poveda and Alba Fernandez</td>
</tr>
<tr>
<td></td>
<td>Towards BIM-centred big data platform to support digital transformation of the AEC industry</td>
</tr>
<tr>
<td></td>
<td>Yvov Bosdriesz, Marten van Sinderen, Maria Iacob and Pieter Verkoost</td>
</tr>
<tr>
<td></td>
<td>Developing, Deploying and Experimenting with Industrial Big Data Platforms for Smart Furniture Product-Service Design</td>
</tr>
<tr>
<td></td>
<td>Ferrareti Lampathaki, Evmoria Bilir, Ariadni Michaelitsi-Psarrou, Angelos Arvanitakis, Javier Martin, Fernando Gigante, Vicente Sales and Maria Jose Nuñez</td>
</tr>
<tr>
<td></td>
<td>IoT Service-Based Machine: a data integration infrastructure for IoT applications</td>
</tr>
<tr>
<td></td>
<td>Helder Oliveira Gomes Filho, José Gonçalves Pereira Filho and João Luiz Rebelo Moreira</td>
</tr>
<tr>
<td></td>
<td>RS4IoT - A Recommender System for IoT</td>
</tr>
<tr>
<td></td>
<td>Caio Martins Barbosa, Roberta Lima Gomes, José Gonçalves P. Filho and João Luiz Rebelo Moreira</td>
</tr>
</tbody>
</table>
### Workshop B3

**Predictive Maintenance in Industry 4.0: Methodologies, tools and interoperable applications**

**Organisers**
Gregoris Mentzas, Karl Hribernik, Klaus-Dieter Thoben, Dimitris Kiritis and Ali Mousavi

**Papers and authors**
- Maintenance planning support tool based on condition monitoring with semantic modelling of systems and their healthy/deteriorated signature
  
  Alice Reina, Gökhan May, Eva Coscia, Sang-Je Cho, Jacopo Cassina and Dimitris Kiritis

- A Morphological and Process-based Model for Digital Assisted Maintenance
  
  Klaudia Kovacs, Fazel Ansari, Claudia Geisert, Eckart Ulßmann, Robert Glawar and Wilfried Sihn

- The ProaSense Platform for Predictive Maintenance in Automotive Lighting Equipment industry
  
  Alexandros Bousdeakis, Babis Magoutas, Dimitris Apostolou, Gregoris Mentzas and Primoz Puhar

- Standards as critical means of integration of advanced maintenance approaches to production systems
  
  Yves Keraron

- How to use sensor data for predictive maintenance of a complex transportation asset
  
  Bernd Bredehorst, Olaf Peters, Jeroen Versteeg, Markus Neuhaus and Carl Hans

- On-board Model-of-signals approach for condition monitoring in automatic machines
  
  Matteo Barbieri, Alessandro Bosco, Christian Conficani, Matteo Sartini, Roberto Diversi and Andrea Tili

- Predictive maintenance framework: implementation of local and cloud processing for multi-stage prediction of CNC machines health
  
  Panagiotis Aivaliotis, Konstantinos Georgoulas, Raffaele Ricatta and Michele Surico

- DRIFT Data-Driven Failure Mode, Effects, and Criticality Analysis (FMECA) Tool
  
  Davide Zanardi and Manuele Barbieri

- SERENA: Versatile plug-and-play platform enabling remote predictive maintenance
  
  Sotirias Makris, Nikolaos Nikolakis, Konstantinos Dimoulas, Apostolos Papavasileiou and Massimo Ippolito

- Approach for real-time predictive maintenance based on complex event processing
  
  Klaus-Dieter Thoben, Abderrahim Ait-Alla, Marco Franke, Karl-A Hribernik, Michael Lütjen and Michael Freitag

### Workshop B4

**Industry 4.0 Qualification - Higher Education for the Era of Industry 4.0**

**Organisers**
Moritz von Stietencron, Gabriele Montelisciani and Gualtiero Fantoni

**Papers and authors**
- Evaluation of Industry 4.0 Technology Applications
  
  Moritz von Stietencron, Bjørnar Henriksen, Carl Christian Røstad, Karl Hribernik, Klaus-Dieter Thoben

- A software application enhancing the workforce skills and competencies within Industry 4.0
  
  Enrico G. Caldaroni, Gianfranco E. Modoni

- Improving the efficiency of industrial processes with a plug and play IOT data acquisition platform
  
  Daniele Mazzei, Gabriele Montelisciani, Giacomo Baldi, Andrea Baù, Gualtiero Fantoni

- Knowledge transfer from students to companies: understanding industry 4.0 readiness levels
### Workshop C2
#### Modelling & Simulation to design Advanced Manufacturing System

**Organisers**
Guy Doumeingt, Carlos Agostinho, Gregory Zacharewicz, Yves Ducq, and Amir Pirayesh

**Papers and authors**
- Developing an Enterprise Modelling Ontology
  - David Chen
- Model Driven Requirements Elicitation for Manufacturing System Development
  - Amir Pirayesh, Guy Doumeingt, João Sousa, Carlos Agostinho, Cristiana Furtuzinhos
- A Comprehensive Architecture to Integrate Modelling and Simulation Solutions in Cyber Physical Production Systems
  - Carlos Agostinho, José Ferreira, Sudeep Ghimire, Grégory Zacharewicz, Guy Doumeingt
- Modelling and Simulation of Decision Systems
  - Raul Poler, Beatriz Andres, Guy Doumeingt, Amir Pirayesh

### Workshop C3
#### Method and Tools to support the development of Product Service System

**Organisers**
Guy Doumeingt, Sergio Gusmeroli, Amir Pirayesh and Giuditta Pezzotta

**Papers and authors**
- Identify new PSS Concepts: The Product Service Concept Tree tool
  - Giuditta Pezzotta, Fabiana Pinola, Roberto Sala, Antonio Marquart, Paulo Pina, Rui Neves Silva
- Product Service System (PSS) Innovation Process
  - Amir Pirayesh, Guy Doumeingt, Marco Seregni, Maria Jose Nainz Arillo, Carl Hans
- Technological and Organisational Pathways towards 2030 Product-Service Factories of the future
  - Chris Decubber, Sergio Gusmeroli, Domenico Rotondi, Luis Usatorre, Michele Sesana, Fenareti Lampathaki, Mike Freitag, Guy Doumeingt
  - Nerea Sopelano, Lara Gonzalez, Rikardo Mingocek, Andoni Laskurain, Oscar Lazaro

### Workshop C4
#### Interoperability for Crisis Management: Increasing Resilience of Smart Cities (ICRIM 2018)

**Organisers**
Antonio de Nicola and Frederick Benaben

**Papers and authors**
- Assessment of Climate Change-Related Risks and Vulnerabilities in Cities and Urban Environments
  - Jingquan Xie
- Semantic interoperability of Early Warning Systems: a systematic literature review
  - João Moreira, Luís Ferreira Pires and Marten van Sinderen
- Towards Semantic Generation of Geolocalized Models of Risk
  - Alex Coletti, Antonio de Nicola, Antonio di Pietro, Maurizio Pollino, Vittorio Rosato, Giordano Vicol, Maria Luisa Villani
- An Ontology-based Operational Emergency Response System for Interoperability in a Crisis Situation: POLARISC
  - Linda Elmhadhi, Mohamed-Hedi Karray and Bernard Archimède
- Analysing Interoperability in a Non-Functional Requirements Ecosystem to Support Crisis Management Response
  - Nicolas Daclin, Behrang Moradi, Vincent Chapurlat
### Tuesday, March 20th

<table>
<thead>
<tr>
<th>Time</th>
<th>Room</th>
<th>Track A Room 360</th>
<th>Track B Room 307</th>
<th>Track C Room S_105</th>
</tr>
</thead>
<tbody>
<tr>
<td>08:00 AM</td>
<td>Foyer</td>
<td>Reception</td>
<td></td>
<td></td>
</tr>
<tr>
<td>08:45 AM</td>
<td>001</td>
<td>Opening</td>
<td></td>
<td></td>
</tr>
<tr>
<td>09:00 AM</td>
<td>Foyer</td>
<td>A1: From Embedded Intelligence to Directed Manufacturing Decision Support R. Young, G. Bhullar</td>
<td>B1: Industrial big data platforms enabling enterprise interoperability for smart services, (Full day) M. vSinderen, S. Gusmeroli</td>
<td></td>
</tr>
<tr>
<td>10:30-11:00 AM</td>
<td>Break Foyer</td>
<td></td>
<td></td>
<td>C2: Modelling &amp; Simulation to design Advanced Manufacturing System G. Doumeingts et al</td>
</tr>
<tr>
<td>1:00 PM</td>
<td>Testing area</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:00 PM</td>
<td>Lunch</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:00-4:30 PM</td>
<td>Testing area</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>End: 6 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Wednesday, March 21st

<table>
<thead>
<tr>
<th>Time</th>
<th>Room</th>
<th>Track A Room 360</th>
<th>Track B Room 307</th>
<th>Track C Room S_105</th>
</tr>
</thead>
<tbody>
<tr>
<td>09:00 AM</td>
<td>Foyer</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:30-11:00 AM</td>
<td>Break Foyer</td>
<td></td>
<td></td>
<td>C3: Method and Tools to support the development of Product Service System G. Doumeingts et al</td>
</tr>
<tr>
<td>1:00 PM</td>
<td>Testing area</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2:00 PM</td>
<td>Lunch</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:00-4:30 PM</td>
<td>Testing area</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>End: 6 PM</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Annex G: I-ESA’18 vf-OS Workshop Papers

This annex provides the authors version of the 6 papers presented at I-ESA’18 vf-OS Workshop. All received papers were reviewed by two reviewers who provides feedback to the papers authors to improve it.

Virtual Factory Operating System

World is facing the fourth industrial revolution based on ICT, specifically architectures and services, as key innovation drivers for manufacturing companies. Traditional factories will increasingly be transformed into smart digital manufacturing environments. Currently, however, the full potential for ICT in manufacturing is far from being fully exploited. Factories are complex systems of systems and there is a need to develop a platform on which future manufacturing applications can be built.

European SMEs’ needs around ICT solutions are far off to be satisfied at affordable cost and time, some are: to reduce integration overheads for ICT, to provide mobile devices that are intuitive, to provide traceability from cradle to grave throughout the supply chain and to facilitate monitoring, optimisation and predictive maintenance to enhance availability and productivity.

Over the last decades, manufacturing companies have been implementing point-solutions, each bringing a specific feature or fixing a specific issue. Resulting from this approach is a highly heterogeneous manufacturing IT landscape. While these IT landscapes are already costly to administer, further addition of capabilities becomes even more costly because they have to be fitted into the heterogeneous IT landscape already existing. There is a need for homogeneous manufacturing IT landscape offering manufacturing IT solutions, to support manufacturing companies, especially SMEs, for successfully responding to the challenge to be flexible and offer highly customized products, and without daunting implementation costs, thereby maintaining and boosting their competitiveness locally and in the increasingly globalized markets they participate in.

Chapter written by Raúl POLER and Ricardo JARDIM-GONÇALVES
As an answer to these needs a H2020 European project called “Virtual Factory Open Operating System” (vf-OS) funded under the topic PoF-11-2016 “Digital Automation” milestone “Collaborative Manufacturing and Logistics” started in October 2016 with 3 year of duration. The aim of this workshop was to show the research performed in the project during its first year. Six papers have been presented at the vf-OS workshop:

“vf-OS Architecture”: This paper describes the vf-OS software architecture. Information and requirements of the real world are included in the structural planning of the complex software system.

“Enablers Framework: an approach to develop applications using FIWARE”: This paper proposes the Enablers Framework, a component of vf-OS project that acts as a bridge between applications and enablers, granting programmers an easy access to the services they need.

“vf-OS IO Toolkit”: This paper describes the IO Toolkit, which provides tools for on-premises facilitating the access to manufacturing devices and existing business software, while also facilitating the interaction with the rest of the vf-OS infrastructure.

“Data Management Component for Virtual Factories Systems”: This paper envisages the vf-OS middleware platform and the Data Management Component for covering the issues related to data handling, pre-processing, extracting, and data flows management for Virtual Factories.

“An Open Environment for Development of Manufacturing Applications on vf-OS”: This paper describes a novel environment that was envisioned for providing to software developers an advanced editor with syntax highlighting, providing a large set of artifacts and libraries including standard connectors and extension points, debuggers, and tools to optimise the performance of developed Apps.

“A novel approach to software development in the microservice environment of vf-OS”: This paper described the vf-OS approach consisting on “Docker”-based microservices, orchestrated through common REST services with special attention given to the challenge of running resources OnPremise versus purely InCloud.

*Virtual Factory Operating System* workshop chairs kindly acknowledge all the members of the program committee. In particular, they thank Stuart Campbell, Eduards Buia, Néjib Moulla and Raquel Almeida for their support in the scientific success of the event. The research leading to the results presented in this workshop papers has received funding from the European Union H2020 Program under grant agreement No. 723710 “Virtual Factory Open Operating System” (vf-OS)
As an answer to these needs a H2020 European project called "Virtual Factory Operating System" (vf-OS) funded under the topic FoF-11-2016 "Digital Automation" subtopic “Collaborative Manufacturing and Logistics" started in October 2016 with 3 years of duration. The aim of this workshop was to show the research performed in the project during its first year. Six papers have been presented at the vf-OS workshops.

"vf-OS Architecture": This paper describes the vf-OS software architecture. Information and requirements of the real world are included in the structural planning of the complex software system.

"Enablers Framework: an approach to develop applications using FIWARE": This paper proposes the Enablers Framework, a component of vf-OS project that acts as a bridge between applications and enablers, granting programmers an easy access to the services they need.

"vf-OS IO Toolkit": This paper describes the IO Toolkit, which provides tools for on-premises facilitating the access to manufacturing devices and existing business software, while also facilitating the interaction with the rest of the vf-OS infrastructure.

"Data Management Component for Virtual Factories Systems": This paper envisages the vf-OS middleware platform and the Data Management Component for covering the issues related to data handling, pre-processing, extracting, and data flows management for Virtual Factories.

"An Open Environment for Development of Manufacturing Applications on vf-OS": This paper describes a novel environment that was envisioned for providing to software developers an advanced editor with syntax highlighting, providing a large set of artifacts and libraries including standard connectors and extension points, debuggers, and tools to optimise the performance of developed Apps.

"A novel approach to software development in the microservice environment of vf-OS": This paper described the vf-OS approach consisting on "Docker"-based microservices, orchestrated through common REST services with special attention given to the challenge of running resources OnPremise versus purely InCloud.

"Virtual Factory Operating System" workshop chairs kindly acknowledge all the members of the program committee. In particular, they thank Stuart Campbell, Eduardo Sata, Nejib Mouilla and Raquel Almeida for their support in the scientific success of the event. The research leading to the results presented in this workshop papers has received funding from the European Union H2020 Program under grant agreement No. 723710 "Virtual Factory Open Operating System" (vf-OS).
vf-OS Architecture

This paper discusses the software planning of the Industry 4.0 project vf-OS. Information and requirements of the real world are included in the structural planning of the complex software system. The waterfall model is used, in which several phases of the planning build on each other. Starting with the system architecture, the individual components are defined in vf-OS and the individual connections to each other and to the outside world are specified. The application of various technical solutions is also dealt with in this phase. Based on this, the functional and technical specifications were drawn up, which describe the internal processes in more detail and also define the communication level. In today's industry 4.0 the emphasis is on security, since sensitive data in particular must be protected and the risk of abuse has increased as a result of the connection of industrial processes to the Internet. In order to minimize or avert these dangers, vf-OS has developed a concept for security and privacy that lists the main sources of danger and their solutions.

1.1. Global Architecture Definition

vf-OS is using a Service Oriented Architecture (SOA) approach in which the different vf-OS components implement individual functionalities and thus the composition of all these inter-related components form the complete vf-OS architecture to ground the vf-OS ecosystem. In order to apply this approach, all components implement and publish a REST interface allowing the exchange of data (primarily) with the messaging bus to be implemented within the project. vf-OS will support Event Driven SOA features so that the different components can decide their interaction pattern and react to internal and external events. Following this approach,

Chapter written by Danny PAPE, Tobias HINZ, Oscar GARCIA PERALES, Francisco FRAILE, José Luis FLORES, and Oscar J. RUBIO
the components of vf-OS can behave either as services or as event producers and consumers.

The diagram in Figure 1 provides an overview of the high-level architecture of vf-OS and provides a formal split of functional components. This diagram classifies the components based on their role to vf-OS (design, runtime, etc.) and its relationship with external resources. The “Kernel box”, situated at top-left of the diagram, indicates that all the components marked with the same color can be deployed within the vf-OS kernel when vf-OS is installed as an OnPremise solution.

Thus, the architectural building blocks are the following:

- Environment Block: The vf-OS Platform acts as an executing environment of vf-OS architecture. It acts as a container, which includes all other components.

- Application Development Block aka design time comprises the different vf-OS components that will be used for the development of vf-OS Manufacturing Assets/Apps.
1.2. Functional Specification

The functional specification describes how the vf-OS platform works from the perspective of the user. More specifically, for every component, the functional specifications contain a description of the functionality and behavior from the perspective of the user, the sequence of actions taking place during the execution of the component to satisfy its functionality and the mockups of the user interfaces that support the interactions with different vf-OS users.

The behavior and functionality of the components is described using story maps [PAT 1-4]. Story maps organize user stories for software development in a matrix, following the user workflow from left to right and the user priority from top to bottom. This visualization helps users and developers to build a common understanding on the behavior of the component and the expected functionality of the different software releases. User stories capture the functionality that needs to be developed with enough granularity to identify inconsistencies or overlapping of functionalities between components. The same methodology is applied to the pilot applications. This provides an effective feedback mechanism to from the pilots to the individual components.

The story maps are linked to UML sequence diagrams used to depict the interactions between components and to UI mockups that describe the interaction with users. This way, every activity in the story map sequence has a high level description of the interaction with external components and/or users.

Finally, the models used in the component and the inner interactions of the component are described using UML class diagrams and a detailed component
architecture diagram to enrich the high-level description provided in the global architecture definition.

1.3. Technical Specification

The technical specification is the primary document detailing the communication amongst vf-OS components. It defines and describes the API endpoints and data models to be transferred in vf-OS. Therefore, it takes into account the knowledge of the Functional Specification and builds the foundation for development efforts.

The target group of the technical specification are developers who need to build software that interacts with vf-OS components. To ease this approach the outsourcing interfaces of each component are accessible in a RESTful manner [RIC 08]. An online documentation tool has been used to do this where each vf-OS component describes its endpoints and data models. This approach keeps the documentation flexible and update-able for later use during the development or if there need to be changes in a later phase.

To provide all information needed by the developer each API endpoint is separated into three parts:

- **Description**: API endpoints are introduced with a short description to show the purpose of the related endpoint.
- **Request**: contains the endpoint URI and the necessary parameters (information and data) to be transmitted to the endpoint component for a successful response
- **Response**: describes both cases of responses (success and error). In case of success, it describes the expected data and the corresponding standardized HTTP status code (e.g. HTTP/1.1 200 OK); in case of an error a description is returned including a HTTP status code (e.g. HTTP/1.1 404 Not Found).

1.4. Holistic Security and Privacy Concept

The main aim of the vf-OS infrastructure is to provide a set of information services enabling industrial processes. From the security point of view, protecting industrial processes is very challenging, due to the high economic incentives for hackers to interfere with them (e.g. stop, corrupt, spy). This has resulted in a rising wave of cyber-attacks (e.g. denial of service, ransomware, espionage) targeting industrial services. Therefore, the vf-OS infrastructure must be adequately protected before it is deployed in production. This is not a trivial task, since vf-OS is composed of a variety of inter-related components and services that may expose large attack surfaces.
Hence, guaranteeing continuous and appropriate availability, integrity and confidentiality (AIC) levels along the whole vf-OS infrastructure calls for a solid, holistic security and privacy model. This model elicits know-how from reference software development life cycles, applicable regulations (General Data Protection Regulations), key industrial IT standards (mainly ISA-62443) and recommendations from top-class security institutions (e.g. NSA, NIST). The objectives pursued are to minimize attack surfaces, maintain security risks at acceptable levels and keep security threats at bay, while minimizing the negative impact of security measures on the functionality and performance of vf-OS services.

- Mainly inspired by SELinux, NSA’s Open Source Security Enhanced Linux, the vf-OS holistic security model integrates five simple, ruling principles:

  - The security component shall act as a security proxy, intercepting, authenticating and authorizing all actions on the platform in order to guarantee the protection of the vf-OS resources.

  - Following the example of Marketplaces such as Google Play or iOS Store, every external software development shall be provided as an Asset so that they can be examined and controlled in a uniform manner.

  - Most security processes (credentials exchange, tokenisation, cryptographic verifications) are automated and transparent, but there may be humans in the loop. Particularly, the process of vApp installations will typically involve granting vApp access permissions (to resources) by the administrator(s) of the vf-OS Platform instance. These access permission policies are managed by using a central console.

  - There is sandboxing by default. Any access (to a resource) that it is not explicitly permitted, because it has been authorised, is prohibited.

  - Authorisation policies follow a mixed system based on Role-Based Access Control (RBAC) and Attribute-Based Access Control (ABAC) called RABAC model. This will be addressed by means of XACML, and can be extended to support ABAC as well.

The holisitc security model envisioned above translates into a vf-OS security component, whose architecture is defined by the following components:

- A Primary Security Component: Comprised of a Security Command Center that concentrates the majority of security processes (authentication, authorization, and continuous security monitoring); an Installer Broker Service, to download and install vf-OS asset; an Identity Service to provide authentication, and an authorization Service to provide global security policies. This component is the main target for cyber-attacks and represents the largest attack surface.

- Marketplace Identity Manager, located in the vf-OS Marketplace, it is intended as the dependable source of applications, externals packages, components etc.
1.5. Conclusions

This paper describes the overall architecture and the specifications of vF-OS which are needed for the subsequent programming work. It ensures that the selected vF-OS technologies fit the requirements and that interfaces for all vF-OS software components are defined. Important security and privacy issues are promoted to their own focused task as they are cross-cutting concerns which largely need to be fulfilled by all vF-OS software components but of course they will be integrated with the overall specification stack. With the following completed core phases of software planning, the programming work of the individual components can begin.

- Architecture to define relations and data streams between vF-OS components and excluding APIs.
- Functional specification to define and prioritize functionalities depending on the requirements analysis
- Technical specification to define the access rules of each component.
- Holistic Privacy and Security Concept to prevent the system for misuse.

1.6. Acknowledgements

The research leading to these results has received funding from the European Union H2020 Program under grant agreement No. 725710

1.7. References


Enablers Framework: an approach to develop applications using FIWARE

Industry is moving towards smart systems and automation, which leads modern factories to demand a set of tools capable of sustaining their needs. Generic and specific sets of enablers, are emerging as core industry 4.0 components, as they provide the software services that factories of the future require. Although enablers can be at the FoF foundation, a system able to integrate them together in an interoperable way does not exist. To integrate the several enablers with their matching applications, it is necessary to create an adaptable system able to overcome this gap. This article proposes the Enablers Framework, a component of v-fOS project that acts as a bridge between applications and enablers, granting programmers an easy access to the services they need. This framework will thus open the possibility to several business opportunities to arise, since the integration of such versatile components is an alternative to the heterogeneous components used in the industrial panorama.

1.1. Introduction

The World is facing the fourth industrial revolution based on ICT, specifically architectures and services, as key innovation drivers for manufacturing companies. Traditional factories will increasingly be transformed into smart digital manufacturing environments but currently, the full potential of ICT in manufacturing is far from being fully exploited. Factories are complex systems of systems and there is a need to develop a platform on which future manufacturing applications can be built. Examples of platforms exist in some industrial sectors but

Chapter written by Pedro CORISTA, Joao Gaxo, Joao SARRAIPA, Oscar GARCIA PERALES and Ning MOALLA
there is a lack of cross cutting platforms based on open standards for creating an ecosystem for cooperative innovation. The approaches that have been used in industry rely on point to point solutions that compete, by providing new specific features or fixing specific issues. Using these solutions result in having a highly heterogeneous industrial IT panorama, where each factory may have similar solutions with different architectures and programming languages. Having such heterogeneous solutions create problems on adapting current solutions from an existing factory to another, as a specific solution is developed to a specific end. There is the need for a homogeneous solution that allows to support the needs from manufacturing companies and fill current approaches' gaps, vF-OS (virtual factory Operating System) aims to be that solution. The goal of the vF-OS project is to develop an Open Operating System for Virtual Factories, responsible for managing factory related computer hardware and software resources and providing common services for factory computational programs. This system will be the component in a real factory system where all factory application programs will run. To fulfill this objective, the system must have a component which follows kernel’s principles and logic. A kernel is a component that acts as a resource manager by taking care of the various programs that are running concurrently in the system. [MAU 2008]. As seen in [GAR 2010] one of Kernel’s functions is to interface all the applications, granting the bridge between them and the resources they need. Following this concept, vF-OS has a component called Enablers Framework (EF) that interfaces the applications (vApps) and the services they require. Such a component allows to integrate the necessary tools, without the user needing to experience the associated installation complexity. This paper proposes a definition for the EF. The first section highlights the EF’s purpose and its role as a wrapper, it also explains and details its main components. The second section is related to the business opportunities that EF provides, considering software developers, manufacturing users and ICT providers.

1.2. Enablers Framework

EF is the component that acts as a bridge between service providers and service consumers. There already exist approaches to work with FIWARE, that allow to integrate enablers with applications, as seen in [COL 2015]. EF aims to provide support for enabler integration and also installing and managing instances of an enabler. To understand the overall functionality of EF’s, it was necessary to define what are the service providers for the applications that consume them (vApps).

The service providers were chosen from the set of solutions adopted from the FIWARE project [FIW 2018 a] and are classified as generic enablers (GE) or specific enablers (SE). GE can be found at [FIW 2018 b] and provide functionalities that ease the connection to the Internet of Things (IoT) technologies and devices, process data and media in real-time at large scale, perform Big Data analytics or
incorporate advanced features for the interaction with users. SE can be found at [FIT 2018] and provide functionalities, which are more specific to the manufacturing domain such as manufacturing assets virtualization, collaborative manufacturing asset management, 3D scanning and virtualization, social data analysis, etc. Using the functionalities of the several GE and SE, a programmer can have a high range of tools that can act as core software components, while designing the vApps. How EF integrates the existing enablers is explained in the next sections.

1.2.1. EF as a bridge

The EF component, provides a solution for integration, exposition of and uniform access to functionalities of the different enablers in a single service-based component. The enablers, specifically GE and SE, can expose heterogeneous sets of service interfaces that could be used by vApp developers posing a need to understand and implement them. EF component acts as a wrapper engine for the different enablers and the vApps consume its services. Such an approach presents high interoperability advantages, as any enabler can be accessed through the platform, and integrated in any developed application. The developer doesn’t have to worry about the enablers installation process in the target machine, as the EF oversees its whole process. This way he can focus on the development process, saving time and resources. The first step to use an enabler is to register it through EF’s interface. From that moment on, the user can create an instance of the enabler and use it. The instantiation process allows users to choose whether the enabler instance is public or private.

The instance creation and installation are processed by creating a Docker container from an enabler image, as seen in [MER 2014]. To run an enabler on a personal computer or public server it is necessary to have access to its virtual instance. The Docker solution allows to instantiate enablers, following an approach similar to the one done in [STU 2015] for microservices, presenting an alternative to actual Virtual Machines (VM). Docker is a platform that lets the user run an application in an isolated environment meaning that it is possible to run multiple containers at the same time on the same host. Using such approach, it allows any enabler to be packaged into isolated containers with necessary code, runtime environment, system tools, system libraries and settings bundled all together. This strategy can lead towards the realization of efficient, lightweight, self-contained software packages and guarantees that software always runs the same way, regardless of where it is deployed. A Docker based approach secures the enablers instantiation, granting that all enablers can be installed, thus contributing to the integration process.
1.2.2. Component functionalities

After an instance is created, a user can access the enabler he needs through the EF. The access to each enabler is granted by following an approach analogous to proxy communication. Whenever an application needs a service it communicates with the framework, that contacts the corresponding enabler, performing the necessary request and delivering its result to the vApp. The communication between the applications and the framework can be made using Messaging and Pub/Sub components, all through the Process Execution. Between the applications and EF exists a Process Executor that orchestrates what the application will do. Communication between enablers and the framework is performed using IoT protocols such as REST and NGSI. To perform the enablers integration, EF uses two main modules: Request Handler (RH) and Enabler Registry Lookup Services (ERLS) (see Figure 1).

The RH abstracts and implements all necessary functionalities for translating requests to/from vApps and enablers that are served by the EF. The internal components of this module parse the requests and create a suitable communication channel for accessing the functionalities of the enablers and build necessary response messages, based on the result of the function invocation. This module uses the technical details of the enablers from the ERLS and relies on suitable proxies to invoke the method requested by vApp. ERLS abstracts all the data models corresponding to registrations details and provides functionalities for performing Create, Read, Upgrade and Delete (CRUD) operations on them. This module is also responsible for providing service details of the enabler to the RH.

Fig. 1. Enablers integration using Enablers Framework
The two presented modules interactions grant the enabler integration. After receiving a service request from an application, the RH contacts ERLS to verify if such enabler is registered in the framework. Upon verifying it, ERLS informs RH about the target enabler’s services, this information is then used to exchange information between the framework and the enabler. The framework informs then the vApp on the service’s response or delivers an error if the target enabler or service are unreachable.

1.3. Implementation Scenario

This section provides a scenario that explains how the vApps can consume the enabler’s services they need. The implementation scenario uses the EF’s components presented in section 2.2, along with the Process Enabler (PE) (figure 2). The PE is a tool used to orchestrate the different tasks that a vApp will carry out. When developing a vApp (Design Time), the Process Designer (the designer side of the Process Enabler) will be used to define the different sub-tasks (e.g. call an analytics algorithm) and/or sub-processes (e.g. persist data in the storage) that will be necessary for executing the vApp, and assign the necessary EF libraries to achieve integration. This process model, in the form of a BPMN, will be then incorporated to the vApp source code so vf-OS is able to execute the different process steps. Thus, the vApp Designer will contact the vf-OS Marketplace, vf-Store, to access the definition and the services of the different assets that the vApp developer wants to make use of. These assets could indeed be an enabler which offers certain services and then, this enabler will be accessed through the EF at runtime. At runtime, when the vApp is being executed, the vf-OS Platform will call the Process Execution (PEX) PE. This way, any vApp will be executed following the same pattern. Then, the PEX will open the BPMN definition of the vApp and orchestrate the calls to the different assets specified in this definition. When it is time to call the services offered by an enabler, regardless if its GE, SE or vf-OS Enabler, the PEX will call the EF that will be in charge of relaying the call to the enabler specified. Then, when the enabler replies, the EF will take its response, pack it and send it to the PEX so the response packet can continue its route in the BPMN flow of the vApp.
1.4. Business opportunities

vf-OS aims to become a business multi-side platform enabling value by creating interactions between external producers and consumers. This approach provides an open, participative infrastructure for these interactions and sets governance conditions for them. Its main goal is to make producers’ and consumers’ information available to each other, in a timely, accurate and transparent way, diminishing their information asymmetry, and making the digital exchange of products and services a better interaction experience for both. The EF and its underlying enablers are a significant part of the added value of vf-OS, which business value relates to: 1) End user enablers that could be directly utilised in the development of vApps (design time), or solely to facilitate an easy plug and play of other resources (CPS, APIs etc.); 2) Scalable Services through enablers that can provide services, which vApps or other components can consume; 3) The vf-OS Ecosystem where the EF as the bridge between service providers and consumers facilitates their integration with the vf-OS platform and its associated services.

Being developed under FIWARE guidelines Enablers must follow an open source approach, but this doesn’t exactly mean that the enabler services execution through the EF should be taken for free. On top of this free approach, enablers developers could sell support contracts on top of enablers deployment for continuous maintenance, specially customization. Thus, it is expected that all these transactions occur through vf-OS Store, being the platform the main sponsor of the marketing and selling activities of all its vApps and enablers specific services.
1.5. Conclusions and future work

Modern industry demands new tools capable of providing resources to its needs. Generic and specific enablers provide the services that factories of the future require. This article presented a framework, part of the vf-OS project, able to integrate enablers with the applications, acting as a bridge between them. By relying on a Docker-based approach the framework provides several instances of the desired enabler, without the user having to pass through its installation process. The enabler can then be accessed through the framework, that provides the means to access the enabler’s services. The orchestration of the services can be achieved through Process Enabler, that assigns the necessary tasks which the application will carry out. The PE will then generate a BPMN which will be integrated in the vApp to execute the defined tasks. The integration of the enablers implies new business opportunities as it opens new business opportunities for its customer segments directly. Additionally, enablers can be used directly in shop floors for easy establishment of IoT connections, or indirectly by enrichment of the vApps functionalities.

1.6. Acknowledgements

The research leading to these results has received funding from the European Union H2020 Program under grant agreement No. 776871 “Virtual Factory Open Operating System” (vf-OS).

1.7. References

vf-OS IO Toolkit

The growing variety of programming languages in software engineering feeds the permanent need of interoperability enablers between IT solutions. In agile and integrated software development, it is vital to ensure the communication between existing IT system and new ones. In addition, the connected smart factories paradigm extends the perimeter of enterprise information system to cover the direct integration of sensors and machines. vf-OS is an H2020 European project aimed at providing a development environment with an integrated, virtual view on manufacturing processes. Within vf-OS, the IO Toolkit provides tools for on-premises facilitating the access to manufacturing devices and existing business software, while also facilitating the interaction with the rest of the vf-OS infrastructure. As such it allows developers to build secure and controlled gateways into the factories, as part of the capability set for their applications.

1.1. Introduction

The World is facing the fourth industrial revolution based on ICT, specifically architectures and services, as key innovation drivers for manufacturing companies. Factories are complex systems of systems and there is a need to develop a platform on which future manufacturing applications can be built.

vf-OS, virtual factory Open Operating System, is an Open Operating System for Virtual Factories composed of a kernel, application programming interface, and middleware specifically designed for the factory of the future. The purpose is to attract talent from solution developers and to provide accessible manufacturing smart applications to European SMEs.

Chapter written by Victor ANAYA, Njib MOALLA, Ludo STELLINGWERFF, José Luis FLORES and Francisco FRAILE
Enterprise interoperability [JAK 12] remains the common research umbrella for software-oriented initiatives aiming to provide development frameworks and toolkits. To ensure engineering efficiency, structured interoperability methodologies are requested to improve the sustainability of software-oriented initiatives and solutions.

1.2. vf-OS IO Toolkit

The IO Toolkit is the development framework provided by vf-OS to Manufacturing and Service Providers wanting to generate new device drivers and API Connectors with the purpose to widen the range of shop floor devices and line of business systems that can be connected to applications developed for the vf-OS platform (called vApps).

Any new developed device driver or API connector (generically called IO Component) can be published to the Marketplace (see top-right part of Fig 1).

If a vApp uses a specific IO Component, then the Manifest file of the vApp describes this dependency, so that users can be informed before installing the vApp. Software Developers can thus develop functionalities that interact with any manufacturing asset that is connected to vf OS via an IO Component.
1.3. IO Components

An IO component is a generic term to refer to Device Drivers or API connectors. A Device Driver is a kind of IO component that interacts with physical industrial devices to integrate sensors’ data and send commands to devices. An example is a driver for consuming data from devices supporting the OPC UA protocol. On the other hand, API Connectors are IO components that interact with legacy software in factories (such as ERPs, databases or tailored ICT solutions).

The common structure followed by every IO component is shown in the next layered schema:

![IO Component Diagram]

**Fig 2. I/O components functionalities**

1.3.1. Top Interfaces

Top interfaces are the interfaces that any IO component must implement to be vf-OS compliant. Those interfaces, bootstrapped by the IO toolkit provide mechanism to invoke and consume data and functionality provided by drivers and APIs. The different interfaces are:

- REST API Server Composition: providing specific REST APIs that new drivers and API connectors must follow to be called by the vf-OS platform components.

- Messaging endpoints: providing drivers and connectors ways to interact with the vf-OS Messaging component when sending and receiving new messages.

- Pub/Sub endpoints: providing ways to interact with the vf-OS Pub/Sub component, providing triggers, events and subscription functionalities.
4  - Metadata template composition: to generate a metadata template file used by vApp developers.

- Registration endpoints: providing functionality to drivers and API connectors to be registered themselves in the Platform and the Marketplace.

- Logging endpoints: to register logs that will be consume by the System Dashboard and the Marketplace.

1.3.2. Logic

The IO Toolkit will provide a set of functionalities available when developing the internal logic of a driver or API connector. Every new driver and component will have available a set of libraries when developing its internal logic. Those libraries are functionally aggregated as:

- Internal Storage: to connect and query the internal databases used to store configuration and operational data, both relational and non-relational databases.

- Key-value store: to access key-value stores to manage associative key-value pairs.

- Edge computing: to process sensor data as close to the data source as possible, instead of processing raw sensor data in the cloud.

- API Access Control: for authentication and security, which can be used alone or in combination to issue credentials and control access.

- API Lifecycle management: The IO Toolkit must provide API lifecycle management tools (prototype, publish, deprecate, and block).

1.3.3. Bottom Interfaces

Bottom interfaces are the proprietary interfaces part of well-known communication standards used by devices and business applications in manufacturing companies.

The IO Toolkit will provide out of the box the bottom interfaces (Table 1).

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPC UA</td>
<td>OPC UA is a machine-to-machine communication protocol for industrial automation developed by the OPC Foundation.</td>
</tr>
<tr>
<td>MQTT</td>
<td>ISO standard (ISO/IEC PRF 20922) publish-subscribe-based &quot;lightweight&quot; messaging protocol for use on top of the TCP/IP protocol.</td>
</tr>
<tr>
<td>Modbus</td>
<td>Ethernet serial communication protocol extensively used in SCADA.</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------------------------------------------------</td>
</tr>
<tr>
<td>GPIO</td>
<td>Generic hardware interface integrated in single board computers (e.g. Raspberry Pi), widely used in fast prototyping.</td>
</tr>
<tr>
<td>MTConnect</td>
<td>Protocol to exchange data between industrial network components.</td>
</tr>
<tr>
<td>MS Excel</td>
<td>Spreadsheets are widely used in SMEs to build simple data analysis and reporting applications.</td>
</tr>
<tr>
<td>STEP</td>
<td>STEP a file format used in CAD to work with 3D models. The STEP file format is widely used in the design of mechanical parts and is supported by the most popular CAD Software tools.</td>
</tr>
<tr>
<td>ODBC</td>
<td>Standard API to access database management systems.</td>
</tr>
<tr>
<td>OData</td>
<td>REST protocol to build API Connectors with leading business software applications and systems (e.g. IBM WebSphere, Microsoft Dynamics CRM, Acumatica ERP, SAP ERP).</td>
</tr>
<tr>
<td>SAP ERP</td>
<td>OData-based API Connector.</td>
</tr>
<tr>
<td>NAV ERP</td>
<td>Microsoft Dynamics NAV is an ERP system developed by Microsoft.</td>
</tr>
</tbody>
</table>

### 1.3.4. IO Toolkit Security

The IO Toolkit Security incorporates a mapped set of security requirements and a documented unit testing plan in security for the development of new vF-OS IO Components. This set of security requirements are based on the recommendations of the most relevant and authoritative organizations in the field of security such as SysAdmin Audit, Networking and Security Institute (SANS), OpenWeb Application Security Project, or National Institute of Standards and Technology.

Based on these recommendations, the requirements are created and provided as a set of checklists. These will require concrete specifications of aspects such as the use of the corresponding libraries for implementing OpenID [OPE 17] for fulfilling the holistic authentication and authorization model of vF-OS (XACML [OAS 17]) or the specification of valid time periods of sessions, error management, or memory management policies.

Finally, the IO Toolkit Security will provide endpoints for the vF-OS Security Command Centre REST APIs (Identity Service, Policy Administration, and Continuous Security Monitoring).

### 1.4. Containerization

The vF-OS platform technical infrastructure [VFO 17] is based on microservices, and Docker [DOC 17] is the core technology used with that purpose.

Developed drivers and API connectors will be deployed by the vF-OS marketplace as Docker containers. The benefits that Docker provides the IO Toolkit
Execution Services are a standardised API for connecting various services, the possibility to port applications to different environments and adapt to several deployment patterns, and several security features such as signed containers, through the Docker Trust feature.

1.5. Conclusions

The vf-OS IO Toolkit is the core component in the vf-OS platform providing the functionality to consume and interact with physical device drivers and existing business software solutions. The IO Toolkit will provide a containerisation solution to run drivers and API connects as microservices in the context of the vf-OS technical architecture. The driver could be run in a specific webserver on a docker machine providing very flexible deployment alternatives. The IO toolkit will provide a catalog of driver and connector templates that will empower developers on the development of new Device drivers and API connectors on the vf-OS Platform. The IO Toolkit will provide a convenient development documentation, along with a standard versioning scheme for the Toolkit and the produced components. Finally, vf-OS requires new drivers and connectors to be properly validated. With that purpose, the IO Toolkit will provide a testing framework to check the compliance of the IO component development.

1.6. Acknowledgements

The research leading to these results has received funding from the European Union H2020 program under grant agreement No. 723770 "Virtual Factory Open Operating System" (vf-OS).

1.7. References


Data Management Component for Virtual Factories Systems

The vf-OS project aims to develop a middleware platform for the virtual factory of the future. The vf-OS is composed of several modules providing interoperability mechanisms for systems to exchange real-time data. Moreover, it enables the managing of data flows, transforming data, providing open API’s to ease integration process. The proposed Data Management Component (DMC) intends to cover the issues related to data handling, pre processing, extracting, and data flows management for Virtual Factories. It integrates four subcomponents: a data infrastructure middleware element that handles the data communications; a data storage mechanism able to work with high loads and triples; a semantic and mapping function to establish model information integration; and an analytical module composed by machine learning and prediction mechanisms to enable knowledge extraction from the vast amount of data generated by the associated virtual factory system.

1.1. Introduction

Modern Internet of Things (IoT) and Cyber-Physical Systems developed to cover needs of manufacturing domain are facing with difficulties related not only to amounts data generated which need to be properly stored, transmitted and delivered on demand, but also with challenges of structuring and analysing data in order to create additional value. The vf-OS project tightly related to concepts of IoT and CPS aiming at provision of an open platform for Virtual Factory (VF) and collaborative environment for different entities on different levels of manufacturing process. vf-OS ecosystem will also inherit the problem of handling enormous ascending and

Chapter written by: Artem A. NAZARENKO, Joao GIAO, Joao SARRAIPA, Oscar J. SAIZ, Oscar GARCIA PERALES and Ricardo JARDIM-GONÇALVES
descending data flows. Some of these challenges are covered by other VF-OS platform components, where one of them is responsible for data connection (I/O functionality) and data movement which comes from its corresponding middleware module. However, before data are absorbed or consumed by applications, they need to be mapped from input to output type – this will be facilitated via semantic harmonisation and transformation techniques. Finally, for data consumption analytics interfaces provided by the infrastructure must scale to extract relevant data subsets according to user’s expectations in the simplest possible way. To access the data, the proposed component also includes all the elements required for data stream management: reading, cleaning, storing, indexing, enrichment, search & retrieval, fusion, maintenance, and correspondence of open APIs. This data infrastructure will serve analytic and decision-making services to the Virtual Factory Open Operating System.

1.2. Virtual Factory Data and Connect

The focus of the proposed DMC [n.1] is on providing a set of semi-independent but related services taking the inputs of a variety data at large scale, with different characteristics such as transmission speed, etc., and providing set of nontrivial analytic operators. It is composed by four subcomponents: the Data Infrastructure Middleware, the Data Storage, the Data Harmonisation, and the Data Analytics. The Data Infrastructure Middleware provides the core access that utilise Data Storage functions for permanent and cross application access. Data Harmonisation provides transformation services, based on semantics, and is particularly connected to the first subcomponent for data connectivity. All these four will be supported by the Data Analytics.

![DMC Architecture](image)

*Figure N.1 – DMC architecture*
1.2.1. Data Infrastructure Middleware

This component specifies and implements a data bus that will support the other subcomponents and the overall vf-OS application for data storage, transformation, and analytic operations. The data infrastructure will contain adapters in order to aggregate data from various enterprise information sources including: machines, hardware sensors (which might include high-precision camera data, accelerometers, vibration, and temperature sensors), software sensors from Enterprise Resource Planner (ERP) systems and external business context data, etc. and take use of other vf-OS activities. Since sensorial data typically generates large amounts of micro measurements, the supporting data infrastructure needs to provide a high throughput technology pipeline for acquisition, pre-processing and aggregation of collected data.

At the early stage two main concepts for middleware were considered REST and message-oriented (AMQP). Talking about advantages and disadvantages, first to be mentioned is that REST has synchronous nature and AMQP broker implies rather asynchronous processing, for instance, message might be requested right after it has been published in the queue or later, as it is kept until being consumed. REST has an advantage that the web-kind behaviour can be transmitted to its resources. In some cases, when direct communication client-server is not possible, REST approach will be completely useless, while AMQP allows ignoring mutual awareness between exchange parties. Another clear advantage considering IoT domain is that AMQP supports both point-to-point as well as publish/subscribe patterns, while REST implies only point-to-point connection. Considering two ways of implementation with broker or broker-less, the first one was chosen due its characteristics extremely relevant for vf-OS: history logging, traceability, message queuing and reordering.

![Messaging model for vf-OS components using RabbitMQ broker](image)

Figure 1.2. – Messaging model for vf-OS components using RabbitMQ broker

To implement the abovementioned functionality message-oriented approach or middleware was selected: the AMQP protocol implemented by RabbitMQ message broker. According to the definition provided by designer of AMQP architecture [OHA 07] it is: “a binary wire protocol and well-defined set of behaviours for transmitting application messages between systems using a combination of store-
and-forward, publish-and-subscribe, and other techniques.” It is based on four basic
definitions: (i) Message – basic unit of transmitted data, the main part of which is
not interpreted by the server (only headers are processed); (ii) Exchange point –
delivery service for messages, all messages are sent to this point to be then distrib-
uted into different queues [n.2]. There are three main types of exchange points: Di-
rect Exchange – sending a message directly at a single queue, Topic Exchange –
producing the copies of the message and sending to all clients, Headers Exchange –
checking all headers on matching with query predicates from interested clients. (iii)
Queue – here the messages are being kept, until the client request. (iv) Bindings –
deliver necessary information about routing, in different types of exchange it is pro-
vided by different entities.

Main components represented on [n.2] are as follows message broker (Rab-
bitMQ) deployed on the server side implementing the AMQP protocol for message
exchange and on the client side both message producers as well as message consum-
ers. The client side is only responsible for creating or requiring messages and also
adding some headers with relevant information about type of exchange and target
node/nodes the rest is being performed by broker. A Publish-subscribe middleware
will help to implement Event Driven Architecture (EDA) and is fully compatible
with major industry standards (i.e. JBI, SCA, BPEL or WSDL).

1.2.2. Data Storage

This component implements the storage services for vf-OS assets. It supports
three major dimensions of “Big Data” when dealing with intensive streaming data,
namely: Volume (scale of data being processed), Velocity (data transmission speed
and optimized reaction time), and Variety (supporting heterogeneous types of data).
It implements a scalable data storage system, capable of handling real-time sensor
data and events, based on an underlying infrastructure that transparently absorbs
very large amounts of data, as well as other types of non-real-time heterogeneous
data. The vf-OS Platform has different needs of Data Storage, e.g., sensor data,
events, alarms, time series, maps, models, data files, log files and structured data.
Each of them has its own requirements in terms of velocity of storage and querying,
volume and updateability of the data, consistency, availability, partition tolerance,
and other dimensions. So, it is not possible to think of a solution based on a single
storage system. Here is where polyglot persistence comes into play.

In 2006, Neal Ford coined the term “polyglot programming”¹, to express the idea
that applications should be written in a mix of languages to take advantage of the
fact that different languages are suitable for tackling different problems. The same

¹http://nealford.com/messagora/2006/12/05/Polyglot_Programming.html
1.2.3. Data Harmonisation

Data Harmonisation aims at (i) extracting more information from incoming data and (ii) preparing the data in a form/schema suitable for other vf-OS components and vf-OS Assets. It will enable semantic enrichment with background knowledge and data mining on real-time streams, received through the Messaging component. The data enrichment objective is to generate from the observed data additional derived attributes/features, either using external background knowledge or internal relationships within data. This could include the use of external ontologies, statistical properties, (models) of data and/or temporal characteristics of data. The key aim is to encode functional transformations of data to help the data (pre-)processing as long as the vApp needs it, as an example, by carrying out analytic techniques of T5.4 to detect non-linear and other non-trivial patterns within the data (otherwise non-detectable by traditional analytic techniques).

Issues related to data extraction from heterogeneous sources with further transformation of them are solved by so-called Extract-Transform-Load (ETL) Systems. The main goal of ETL systems is to make data more useful for further processing and analytics. This might be achieved through linking data and creating triples or semantic triples, which can be described as a statement of the subject-predicate-object form. ETL workflow consists of three main phases, as it follows from the definition: extraction, transformation and load. According to [BAN 14], data extraction aims at getting data from the source, which can be presented in form of flat file or base source. Transformation is a process of “cleaning” data or bringing them to the form corresponding to the scheme, including some actions such as: normalization, filtering, etc. And the last stage is loading them to the data storage.

Since vf-OS is oriented towards the software developers wanting to develop vApps for the Manufacturing Industry, it is expected that the ETLs would be part of the execution of a vApp. However, vf-OS follows a different approach: instead of

---

https://martinfowler.com/bliki/PolyglotPersistence.html
programming the ETLs directly in the source code of the vApp, the ETLs will be programmed as independent services and, thus, deployed as standalone services that could be re-used by other vApps and launch several instances of the same service if deemed necessary. As such, the Data Harmonisation is divided in two components: one for design-time, the Data Mapping, and another for run-time, the Data Transformation. Thus, the ETLs will be “programmed” under the Data Mapping component and deployed as these standalone services and executed within the vApp as part of the Data Transformation component.

1.2.4. Data Analytics

This sub-component covers the creation of building blocks for off-line analytical processing of inputs coming from the Manufacturing environment. This will include machine learning algorithms supporting supervised and unsupervised scenarios. The core of the analytics algorithms will be based on the combination of the modern statistical-machine-learning linear-algebra based algorithms (e.g. SVM, CRF, LDA, Mixture-Models) and traditional data-mining algorithms (e.g. decision trees and rules, k-means, association rules). This will cover typical classification and segmentation scenarios for enriched representations coming from semantic based ontological descriptions to capture nonlinearities in data. The key research innovation will be provided by using “multi-level” analysis on the top of more traditional machine learning algorithms simultaneously observing the data on multiple aggregation levels [VFO 17].

In a factory there will be two different types of analytics tasks to be performed: those cases depending on historical data sources, e.g. readings of sensors for a given period of time in the past, and those ones that need to be calculated on real-time as they will be involved in triggering alarms conducting to time (or spend) critical actions. As such, two different modules will be integrated into the Data Analytics component focusing each of them in these two, although complementary, areas. Similarly, to the Data Harmonisation, the Data Analytics will be composed of analytics libraries so that the vf-OS user, i.e. the developer of vApps, would be able to re-use the existing analytics libraries that will be provided by vf-OS. This means that there will be a set of libraries each of which covering complementary functionality. There will be libraries for the various analytics approaches as for modelling and training like Random Forest predictors, which will be built on top of a SOTA analytics tool [H2O 18] that offers an API to access to such different algorithms.
1.3. Conclusions and future work

Current research work represents the DMC as the part of vf-OS project’s being performed within Horizon 2020 initiative. The proposed solution being developed to satisfy both basic as well as more complex needs of VF’s in managing data flows. This includes organisation of data exchange based on Message Oriented Middleware, extracting, transforming and representing data with Data Harmonisation and Data Analytics subcomponents and storage of heterogeneous data both in terms of content and format. Important requirements for VF’s such as vast amount of data being produced as well as consumed, scalability and adaptability of infrastructure, need to cope with different data in terms of formats, origin, heterogeneity and distributiveness of data sources are also considered. The main efforts are directed towards providing a scalable, open-source, modular platform for developing various industrial applications within a shared VF ecosystem. Resulting component, in fact an independent platform, will be able to solve relevant issues related to quick and agile deployment of smart applications through enabling of necessary tools and software modules around data flows handling. Further work will contain implementation of DMC within vf-OS for different manufacturing scenarios. However, DMC can be also used as an independent component being integrated with already existing solutions and thus offering the wide range of possible implementations as a part of other systems and solutions.

1.4. Acknowledgements

The research leading to these results has received funding from the European Union H2020 Program under grant agreement No. 723710 “Virtual Factory Open Operating System” (vf-OS).

1.5. References

An Open Environment for Development of Manufacturing Applications on vf-OS

Developing a project aimed at providing an Operating System to a manufacturing system requires having a suite of tools that can specify, develop, build and distribute the applications that will be at the heart of this manufacturing system. These applications shall be able to receive stimuli from and to provide actuation over the manufacturing assets, combining this virtualised manufacturing system with the power of the novel ICT resources, enablers and state of the art techniques for development and building of systems and applications that have been used in industry, which involve WYSIWYG editors with syntax highlighting, SDKs which provide a large set of artefacts and libraries including standard connectors and extension points, but also debuggers, tools to optimise the performance and make the best use of the latest technology. Additionally, innovative methodologies for development promote strong involvement of the development community and the developers’ engagement. This paper describes a novel environment that was envisioned for this purpose, involved in the scope of the H2020 European Project vf-OS.

1.1. Introduction

The upcoming of the new industrial revolution “Industrie 4.0” [IND 18] presents a new paradigm for the development of manufacturing services, tightly aligned with the most advanced ICT innovations. At the very heart of this development are new techniques that promote the collection of data from multiple sensors and sources of

Chapter written by Carlos COUTINHO, Luis LOPES, Vítor VIANA, Danny PAPE, Gerrit KLASSEN, Bastian von HALEM, Oscar GARCIA PERALES, Ludo STELLINGERFF and Andries SYAM
input information and the ability to infer knowledge from it (Internet of Things – IoT), and virtualisation of the manufacturing assets. The new smart digital manufacturing environments will allow the development of applications that are able to interact with the virtual factory assets, receiving inputs and actuating on the real factory systems. This will allow them to benefit from the advances in the methodologies and development of software that traditionally were not applicable to the manufacturing domain, with multi-purpose agile dashboards and frontends that deal with accessibility and that are more responsive and provide more valuable information. New methodologies which include agile techniques, the use of popular and proven development frameworks, the support of development communities that provide manpower for testing and for innovative ideas, suggestions and bug corrections. Additionally, the development of applications shall be more pleasant, with a set of tools like IDEs and editors with code syntax highlighting, debuggers, drag and drop features, integration with version control, issue trackers and other mechanisms that engage and motivate the developer into the creation of applications, a set of libraries accessible through the development SDK that will provide access to innovative features such as new enablers, dependencies analysis, dashboards and especially the connection to the internet and to a whole new level of ICT assets.

The developers will be able to produce “What You See Is What You Get” (WYSIWYG) applications that use frontend elements commonly used on web-based applications, based on the latest technologies and best-practices of the market.

The H2020 project vf-OSS [VFO 18] proposes to create an operating system targeted to the manufacturing business. This naturally involves the development of mechanisms that interact with the real factory assets, enablers for connecting to internet and web features, and an environment that fosters the development of applications for the manufacturing business. The vf-OS development framework is named virtual factory Open Application development Kit (OAK) [VFO 17]. It is composed by a set of libraries, assets and tools which are described next.

1.2. vf-OAK SDK

The vf-OAK Software Development Kit (SDK) is a centralised environment for the development of applications and, generically, for the centralised access of the vf-OS assets and functionalities. The SDK itself will not have a user interface per se, instead, it will be accessed as a set of APIs to access the main development resources. That way, the SDK will be able to provide to the vf-OAK Studio (see section 1.3) and to other Application/Development components the resources and services that they require.
The vf-OAK SDK comprises an extensive fully-documented API framework that provides to developers the means to easily generate applications (vApps) and services. It also includes the development of a service API which offers methods to access functionalities of all technical components in the vf-OS environment that the service developers need, such as the easy integration of data from the Cloud-based information infrastructure and the storage of service-specific data within the Cloud.

The basis for the development of the SDK is to provide a component or framework of elements that foster and improve the development of solutions and vApps for vf-OS. The idea is to be able to help a developer to create vApps with this framework such that the SDK will provide what needs to be included in the vApp, and then other OAK components will be responsible for defining behaviours, process flows and other application definitions, in a user-friendly interface, based on Frontend customised elements, that aims to promote and foster the development.

Particularly speaking, the SDK is divided into several parts considering the components that are to be connected. The SDK core is responsible for handling and exporting the communication layer between the OAK Studio or other components for actions e.g., publish/subscribe, messages, or security. The SDK for vApps has a process interpreter which can transform the results from the Process Designer tool (XML files) to JavaScript language. And the SDK for the I/O Toolkit accesses the real manufacturing devices, sensors and other mechanisms via the vf-IO interfaces.

1.3. vf-OAK Studio

The vf-OAK Studio provides an Integrated Development Environment (IDE) with a Graphical User Interface (GUI) for users that want to develop vApps for the vf-OS platform. It also provides all the necessary tools and means for developing a vApp, and supports the application deployment on end user devices, and its publication in multiple Marketplaces. Besides including tools for code development, with editors that include features like Business Process Management Notation (BPMN), syntax highlighting and drag-and-drop, debugging and analysis of the developed code, the vf-OAK Studio includes the interfaces for Process Designer, Frontend Composer, and provides access to the vf-Store marketplace.

The vApp creation starts by setting the necessary configuration which includes dependencies, marketplaces and permissions that are going to be used by the vApp. Then the vf-OAK Studio provides a working space composed by a Process Designer (see section 1.1) and a list of services and processes available in the vf-Store marketplace which are specified during the project configuration.
The implementation of vApps will be based on the composition of pre-built blocks, where the developer can select an available process/service from the marketplace, and drag-and-drop it as a block to the Process Designer. Each block can be configured by the user. If a process or service needed by the user is not available in the list, the user can create and develop a new customised block. The logic for the implementation of this custom block can be done using or composing other blocks or code. When finished, the custom block can be saved locally, so it can be used on the vApp, or deployed to one or more marketplaces, so that other developers can make use of it.

The vF-OAK Studio also includes a Frontend composer (see section 1.5) which manages the user interfaces used by the vApp. From this tool, the user can use the available templates, e.g., default login page, or create new ones. This User Interface (UI) composer also supports logic implementation used by the UI, e.g., filters the received data and shows the results or gets the inputs entered by the user.

1.4. Process Designer

vF-OS will use a Process Designer for complementing the development of the vApps. The Process Designer is responsible for allowing users to model multiple manufacturing workflows for orchestrating the various assets available within a collaborative framework. The tool will be an online workspace supporting a BPMN-like model and will be usable by vApps whenever process design and orchestration is appropriate.

The Process Designer will be a visual online reactive canvas, allowing the developer to pull in existing models from a library representing the virtualised manufacturing assets and all the elements which will be necessary for the correct development of a vApp. These elements can be any of the vF-OS Assets: Generic FIWARE Enablers (Application, Services, e.g., single-sign-on (SSO), but also Specification/Methodology), Manufacturing Enablers (Applications, Services and Specification/Methodology), vF-OS Services (e.g., Transformation), vF-OS Component (e.g., Process Engine [VFOS 18]), and of course, other vApps.

Each asset may support additional properties that can be defined. The defined workflow can include other sub-workflows and be saved and versioned within the Data Storage as BPMN-model definitions. This will allow a developer to design an ad hoc workflow that will model and shape the desired vApp. Examples of vApps could be the execution of a specific business process or the integration both inter-plant and cross-company data exchange processes. These services, which will be orchestrated at runtime, will be augmented with additional information needed to define the required processes.
The vf-OS Process Designer will be a web-based BPMN tool accessible through the vf-OS OAK. It will take advantage of cutting-edge web technologies. The foundations of the Process Designer will make use of an open source BPMN modeller framework such as BPMN-IO, which is a rendering toolkit and web modeller for BPMN. It allows easy creation of BPMN2.0 diagrams using a web based modelling library which can be extended to add the functionality for vf-OS.

The toolbox inside the Process Designer will offer vf-OS manufacturing assets (Services) that are available from the vf-Store. From this repository, a process designer can then select and drag from the Toolbox onto the design canvas to begin modelling a process. It will contain all the standard BPMN modelling elements, such as Parallel or Exclusive Gateways, Processes, and many others needed to create a BPMN diagram.

1.5. Frontend

The Frontend Environment is part of vf-OAK and builds a library of graphical elements that are optionally linked to simple functionalities. These elements are cross-platform compatible and they serve as a help for developers to quickly and safely create consistent looking vApps. The libraries from the Frontend Environment are integrated into the vf-OAK Studio, and can be used directly, by using the elements in the shape of classes. By using the Frontend Environment, the developers can choose between the following repositories:

- UI Element Repository contains single UI elements following the style guide of vf-OS that can be used during vApp development. Particular attention was paid to the corporate holistic layout of vf-OS for branding purposes;

- UI Template Repository provides compositions of UI elements from the UI Element Repository. Each template appears as a single UI element but is composed of several UI predefined elements. A lightweight example of a UI template is a login form, which is composed of a grid, textbox, password-textbox, and action buttons;

- Behaviour Repository contains several default behaviours for activities, which are related to UI interactions. Those behaviours are easy to integrate and applicable for e.g., forms, downloads, registration, and notifications;

- Behaviour Template Repository provides a composition of several behaviour templates. A single behaviour template consists of lined-up behaviours that are processed synchronously. Lined-up behaviours ease the use of handling events and support developers with additional default events;
Holistic Template Repository provides holistic templates that unite UI templates and behaviour templates. These templates follow the style guide and include logic to handle user interactions automatically.

Furthermore, in vf-OS, the Frontend Environment is used both in design and at runtime. In design time the developer can choose elements from the library and can integrate them in their vApps with help of the vf-Studio. But in runtime, demanded user interactions can call simple functionalities from public vf-OS interfaces, such as the analytics component, or the vf-SDK.

1.6. System Dashboard

The vf-OS System Dashboard is a runtime, central dashboard and task manager, aimed at monitoring, warning, configuring, and adapting system resources. As such, it provides information on the runtime behaviour of the vf-Apps, useful for both the end user and the developers of the apps.

The primary component of the dashboard is a web-based user interface, in which the status of all user accessible vf-OS components and deployed assets can be monitored and controlled.

Behind this UI, a controller service is provided which aims to collect the required information for the dashboard and which contacts the execution services for the starting and stopping of vf-Apps and their dependencies.

A notification service is also part of the dashboard, allowing the user (and/or developer) to be informed of off-nominal issues or other triggers. The Notification Engine stores and manages notifications, and applies several external messaging mechanisms (e.g., email, messaging, notification icon) to warn users of such issues.

1.7. Developers Engagement Hub

To complement the actions of development, creation, debugging and integration, the modern methodologies for supporting the development advocate that as good or better than having a great development team is to have an active and engaged community of users, partners, customers and developers that contribute to the success of the projects.

Hence, the team proposed the inclusion of a centralised hub which has the purpose of gathering and promoting the contact, interaction, collaboration and discussion of interested stakeholders. These communities often help with their requirements, their suggestions and ideas, but also greatly with their own
development or testing of the developed vApps. This is the largely successful philosophy around Open Source, but to be able to do this in a correct way, these communities need to be correctly biased and engaged to the developing project. The vf-OAK suite therefore includes a platform that has as sole purpose the capitalisation of the developing project, providing mechanisms that promote a clear and straightforward description of what is being developed, and its surrounding concepts, methodologies, requirements, and as much more information as possible.

The vf-OS Developers Engagement Hub will promote the dissemination of the developed code and its evolution using source code control (e.g., Git repositories), and tools which will mostly deal with dissemination and communication (collaboration), which may include wikis, templates, tutorials, multimedia, forums, mechanisms for capturing and describing new features and bugs (issue trackers), management tools, and many others which may support the development and make it simpler like business and continuous integration, connections to larger ERP and CRM tools, statistics and other tools which may both facilitate the development, its documentation and dissemination.

1.8.2 Conclusion

The purpose of the H2020 vf-OS European Project is to provide manufacturing businesses with an open Operating System that will allow them to – similarly to what happens in a regular computer – create, develop, build and load applications that span and cover the whole manufacturing operations and processes, enabling the capture of the business knowledge and promoting its analysis and potential to achieve greater value.

1.9. Acknowledgements

The research leading to these results has received funding from the European Union H2020 Program under grant agreement No. 723776 "Virtual Factory Operating System" (vf-OS).

1.10. References


A novel approach to software development in the microservice environment of vf-OS

Within the vf-OS project, a common approach is taken for the deployment of computing resources and data models. As such vf-OS represents a modern approach to the software middleware layer, specifically aimed at the manufacturing domain. The approach consists of *Docker*-based microservices, orchestrated through common REST services. Because of the domain, special attention is given to the challenge of running resources OnPremise versus purely InCloud, and the challenges of letting such heterogeneous environments work together. As this environment poses challenges for the development process of applications, a novel approach to the development of such applications is needed. In this paper, this approach is described. During the vf-OS project, experience will be obtained, validating this approach.

1.1. Introduction

The World is facing the fourth industrial revolution based on ICT, specifically architectures and services, as key innovation drivers for manufacturing companies. Factories are complex systems of systems and there is a need to develop a platform on which future manufacturing applications can be built.

vf-OS, virtual factory Open Operating System, is an Open Operating System for Virtual Factories composed of a kernel, application programming interface, and

Chapter written by Luís MANTEGAS DA CUNHA, Luís STELLINGWERFF and Andries STAM
middleware specifically designed for the factory of the future. The purpose is to attract talent from solution developers and to provide accessible manufacturing smart applications to European SMEs.

vf-OS offers a manufacturing oriented cloud platform, supporting a multi-sided market ecosystem that provides a range of services for the connected factory of the future, allowing manufacturing companies to develop and integrate better manufacturing and logistics processes. [VFO 18]

Due to the highly distributed microservices architecture used in vf-OS, such an environment poses some challenges for the software development process which will produce the assets. This paper describes the approach taken to handle these challenges and describes how this can be evaluated over the lifetime of the project.

1.2. vf-OS platform

The design of the vf-OS platform environment is formed based on the requirements set out in the public requirements document D1.5. [VFR 18] It provides the environment in which vf-OS assets are installed, run and accessed. These assets encompass services, tools and applications. In the case of end-user visible applications, these are called vApps. Such assets interact with each other through web-technologies, such as REST-services, web-based GUIs and modern message busses. See Figure 1.1.

![vf-OS asset interaction](image)

The platform allows the assets to run both InCloud or OnPremise. This heterogeneous model is the result of the specific requirements for the manufacturing domain. In most use-cases there is data produced within the factories and through the legacy systems of the end-users. This data can be used within cloud applications.
This introduces quite stringent security challenges, to allow the users to control their data dissemination. Similarly it requires security measures to prevent undesired access to machinery and other local resources. The manufacturing domain poses several unique privacy and control issues, especially due to the inter-corporate interactions. The relationship between a factory, its subcontractors and its customers is complex, with many contracts on liability, guarantees, services and time constraints.

Within vf-OS this is solved by introducing a model where there are multiple vf-OS platform instances, with a strict inter-platform communication model. One of the implementation options for this model is shown in Figure 1.2. It shows a factory, which will run a platform instance locally, which will communicate with a cloud platform instance, through a controlled proxy asset. The customer application will then run in the cloud platform and only get very controlled access to the data from the factory’s premises. The SCC component in the platform is the Security Contact Center, part of the security model of vf-OS.

![Figure 1.2. vf-OS inter platform communication](image)

The vf-OS assets will have a standardized structure and packaging format. The basic executable entity will be based on Docker images. This Docker image will be enveloped in a wrapper structure, containing metadata, like access rules, dependency information and security signatures. The assets will be storable in the vf-OS Marketplace and can be bought and deployed into the Execution environment, provided by the Platform.

This distributed environment makes it harder to create consistent, coherent applications, especially with regard to debugging, versioning and other software lifecycle aspects. An important part of the goal of the vf-OS project is aimed at providing an answer to these challenges.
1.3. Development approach

The vApp development is the most visible asset development process, as facilitating this process is the stated goal of the project. But the development of the other assets within the vf-OS environment is very similar. It is therefore a good approach to use a single development model for both application development and for the development of other assets. This means that different types of developers will interact with the development model, requiring some flexibility in supporting tools and building blocks.

To tackle this challenge of building consistent, coherent applications within the highly distributed environment of vf-OS, a integrated development environment has been chosen. This environment is called the vf-OS Studio. As all parts of the vf-OS project, it will be available as a set of assets, forming an IDE-like application. An overview of the building blocks of the studio is given in figure 1.3.

![vf-OS Studio components](image)

As can be seen, the main assets are Javascript-based services. Each service will run in its own docker container, providing some generic APIs for system interacting. For example, each asset will provide data for the vf-OS Dashboard. Similarly each asset exposes an API for interacting with the Execution environment. Assets may choose to also carry a web-based GUI, in the form of containing a web server. (e.g. an NGINX host (NGI 18)). Several SDKs and JavaScript libraries are provided by the project for interacting with other assets. Examples of such libraries are the publish/subscribe mechanism, the messaging framework and access to storage facilities.

To support the distribution and marketing of the developed vApps, the project offers a MarketPlace. This MarketPlace has several tasks in the software
development process: a) a marketing and sales channel; b) a services registry; and c) hosting of the assets for deployment. Because of the single development model, incorporating this marketplace, these tasks all contribute to the ease of developing vApps.

External services can be represented in the single development model as well, by encapsulating them into vF-OS Assets. For this purpose the project provides a support module, called the External Services Provisioning framework.

1.4. Conclusions

We have explained a solution to a particularly interesting problem, arising from the manufacturing domain and the micro-services architecture which vF-OS employs to handle the domain. The described approach of standardized packaging, marketing, deployment and interaction between micro-services, is quite novel. Future steps are evaluating this approach, using the projects intermediate releases, use-cases and experimentation. The results of this evaluation will be reported in similar publications.

1.5. Acknowledgment

This paper presents work developed in the scope of the project vF-OS. This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 723310. The content of this paper does not reflect the official opinions of the European Union. Responsibility for the information and views expressed in this paper lies entirely with the authors.

1.6. References


Annex H: I-ESA’18 vf-OS Workshop Presentations

This annex provides the slides used by the speakers at the I-ESA’18 vf-OS Workshop, starting with an introduction to the vf-OS Project and following with the 6 papers presentations.

The title of the Workshop was “Smart Services and Business Impact of Enterprise Interoperability”
**Aspiration**

<table>
<thead>
<tr>
<th>Software Operating System Environment</th>
<th>vf-OS Equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Kernel</td>
<td>Virtual Factory System Kernel</td>
</tr>
<tr>
<td>Processor, Memory, Internal Bus</td>
<td>Framework, Enablers</td>
</tr>
<tr>
<td>I/O</td>
<td>Virtual Factory I/O</td>
</tr>
<tr>
<td>Interfaces, Device Drivers, Peripherals</td>
<td>Device Drivers, APIs, Security</td>
</tr>
<tr>
<td>File and Data Handling Interfaces</td>
<td>Virtual Factory Middleware</td>
</tr>
<tr>
<td></td>
<td>Data Infrastructure, Storage, Harmonisation, Analytics</td>
</tr>
<tr>
<td>SDK</td>
<td>Open Applications Development Kit</td>
</tr>
<tr>
<td>Application Development, System Monitor</td>
<td>System Dashboard, Frontend Environment, Studio, Developer Engagement Hub</td>
</tr>
<tr>
<td>Applications</td>
<td>vApps</td>
</tr>
<tr>
<td>ERPs, CRM, MESes, WMs</td>
<td>Collaboration in real-time, monitoring</td>
</tr>
</tbody>
</table>

**Some initial concepts**

<table>
<thead>
<tr>
<th>Term</th>
<th>vf-OS Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>vApp</td>
<td>Manufacturing Smart Applications (vApps) developed in the three pilots act as the experimental base of the vf-OS Platform</td>
</tr>
<tr>
<td>Enabler</td>
<td>An Enabler is a third-party application that encompasses certain stand-alone functionality that has been already developed, ie within FIWARE, or developed by vf-OS</td>
</tr>
<tr>
<td>vf-OS Asset</td>
<td>A vf-OS Asset is compound of Enablers, Drivers, APIs, External Services and vApps</td>
</tr>
</tbody>
</table>
virtual factory Operating System

Application

Use cases

Pilot A – Manufacturing & Logistics/Automation:
- **Goal:** To build a set of Smart Applications (vApps), integrated into vf-OS, for the **advanced management of spare-parts** in the automation production equipment sector.

Pilot B – Manufacturing Assembly/Collaboration:
- **Goal:** To **accelerate and maintain collaboration** channels between a network of collaborative SMEs in two complementary business domains: plastic and metal.

Pilot C – Construction/Industrialization:
- **Goal:** To facilitate the **documentation management** of a construction site by assimilating to an industrial factory where workers manufacture, assemble goods, operate machines, and carry out tasks processing products.
Exploit and Sustain

Sustainable business model based on the use of the \textit{vf-P}, \textit{vf-Store}, and other \textit{vf-OS} components

- **Open Access**: \textit{vf-OS} and all its components will be by default open source
- **Sustainable Marketplace and Store**: \textit{vf-OS} will take advantage of the great success of the mobile apps business model
- **Start-up Company**: Provide a sustainability model including the establishment of a start-up company to exploit data and data services

Multi-Sided Platform
Overview

Service Providers

Software Developers

Manufacturing & Logistics Users

Manufacturing & Logistics Solution Providers

Partnership

<table>
<thead>
<tr>
<th>ID</th>
<th>Participant organisation name</th>
<th>Country</th>
<th>Nature</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Information Catalyst for Enterprise Ltd</td>
<td>UK</td>
<td>SME</td>
</tr>
<tr>
<td>2</td>
<td>Ituran SCL</td>
<td>Spain</td>
<td>Technological Center</td>
</tr>
<tr>
<td>3</td>
<td>UNINNOVA - Instituto de Desenvolvimento de Novas Tecnologias</td>
<td>Portugal</td>
<td>Research</td>
</tr>
<tr>
<td>4</td>
<td>Universitat Politècnica de València</td>
<td>Spain</td>
<td>University</td>
</tr>
<tr>
<td>5</td>
<td>Caixa Mágica Software, S.A.</td>
<td>Portugal</td>
<td>SME</td>
</tr>
<tr>
<td>6</td>
<td>Université Lumière Lyon 2</td>
<td>France</td>
<td>University</td>
</tr>
<tr>
<td>7</td>
<td>ASCORA GmbH</td>
<td>Germany</td>
<td>SME</td>
</tr>
<tr>
<td>8</td>
<td>Almende B.V.</td>
<td>The Netherlands</td>
<td>SME</td>
</tr>
<tr>
<td>9</td>
<td>Mondragón Assembly</td>
<td>Spain</td>
<td>SME</td>
</tr>
<tr>
<td>10</td>
<td>Via Solis UAB</td>
<td>Lithuania</td>
<td>SME</td>
</tr>
<tr>
<td>11</td>
<td>Consulgal - Engenharia e Gestao S.A.</td>
<td>Portugal</td>
<td>Large enterprise</td>
</tr>
<tr>
<td>12</td>
<td>Knowledgebizz Lda</td>
<td>Portugal</td>
<td>SME</td>
</tr>
<tr>
<td>13</td>
<td>Applications Matiques du Maine SAS</td>
<td>France</td>
<td>SME</td>
</tr>
<tr>
<td>14</td>
<td>Tardy SAS</td>
<td>France</td>
<td>SME</td>
</tr>
</tbody>
</table>
Contact us

http://vf-os.eu

https://www.youtube.com/channel/UCN-5AXq1aXjXItq8JuoW1w

https://www.linkedin.com/in/vf-os-project/

https://www.facebook.com/vfoseuropeanproject/

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 739789.
virtual factory Operating System

Architecture

Authors: Danny Pape, Tobias Hinz, Oscar García Perales, Francisco Fraile, José Luis Flores and Oscar J. Rubio

Agenda

- Global Architecture Definition
- Functional Specification
- Technical Specification
- Holistic Security and Privacy Concept
Global Architecture Definition

- Service Oriented Architecture
- REST Interfaces
- Event Driven SOA features
- TOGAF oriented

High Level Architecture
Architecture Building Blocks

- Environment Block
  - vf-OS Platform as a Container
- Application Development Block
  - vf-OS design time components
- Application Services & Middleware Block
  - vf-OS runtime components
- Application Deployment Block
  - vf-OS Microservices (Docker)
Solution Building Block

- **vf-OS Assets**
  - Generic Enablers (FIWARE)
  - Specific Enablers (FITMAN)
  - vApps
  - External Services
  - I/O Connectors
    - Software
    - Hardware

Functional Specification

- User requirement evaluation
- Convert requirements to functionalities
- Prioritize functionalities
- Analyse functionality behaviour
Functional Specification Roadmap

- **Story maps**
  - User workflow (left to right)
  - Priority (top to bottom)

- **UML Sequence Diagrams**
  - Depict interactions between components
  - Developing workflows in UI mock-ups

- **UML Class Diagrams**
  - Focus on component interactions

Technical Specification

- **API endpoint definitions**
  - RESTful interfaces

- **Data model definitions**
  - That are relevant for communication between components

- **Living online documentation (Slate)**
  - Inclusive Code Repository and Continuous Integration
API Endpoint composition

- Endpoint Description
- Request Information
  - how to use this endpoint
  - URL: POST /v1/paymentnotifications (example)
  - Table of required and non-required parameters
- Response Information
  - Return value
  - http status code

Holistic Security and Privacy Concept

- Cyber Attack Protection
- vf-OS considers
  - IT Standards (mainly ISA-62443)
  - General Data Protection Regulations (GDPR)
  - Recommendations of top-class security institutions (e.g. NIST, NSA)
5 Simple Rules

- Security component acts as a proxy
- Each new asset will be examined
- Access Permission Policies (User Management)
- Sandboxing by default
- Authorisation Policies
  - RABAC

Security Architecture

- Primary Security Component
  - Security Command Centre
  - Installer Broker Service
  - Identity Service
  - Monitoring

- External Security Subcomponents
  - Marketplace Identity Manager and PKI
  - OAK Studio/Toolkit
  - Engagement Hub
Questions?
Enablers Framework: an approach to develop applications using FIWARE

Joao Sarraipa
UNINOVA

Current Industrial Panorama

- Too much end to end solutions
- No global standards
- It is costly to adapt existing technologies
- Too much technologies
vf-OS project

Objectives:

Provide a standard solution for the Industrial Panorama

Manage factory related computer hardware and software resources

Provide and integrate common services for factory computational programs

Enablers Framework

Main Role

Act as a bridge between service providers and service consumers.
Enablers as Service Providers

Why to use them?
Enablers offer reusable and shared modules, which include protocols and interfaces for operation and communication, able to cover several usage areas among different sectors.

Coverage Areas

<table>
<thead>
<tr>
<th>Cloud Hosting</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data context/management</td>
<td>Interface with Networks and Devices (I2ND) Architecture</td>
</tr>
<tr>
<td>Internet of things</td>
<td>Advanced Web-based User Interface</td>
</tr>
<tr>
<td>Applications, Services and Data Delivery</td>
<td></td>
</tr>
</tbody>
</table>

Factory Service Providers

Enablers

- Specific Enablers
- Generic Enablers

- HITMAN
- FI-WARE
- vf-OS
Enablers Framework as a Bridge

Components Functionalities
Implementation Scenario

Enablers Framework Business Opportunities

- Facilitate enablers creation to be utilised in the development of vApps or to facilitate plug and play of other resources.
- Interface the access to Enablers services to be consumed by vApps and other components.
- Contribute to vf-OS Ecosystem and act as the bridge between service providers and consumers.

E.g. Robot Producer

To teach karate

Define Control robot movements

enabler

To be used in academic teaching
Conclusions

FIWARE

- FIWARE is an open sustainable ecosystem, built around public, royalty-free and implementation-driven software standards. We make and share open source technology for smart solutions.

- The proposed Enablers Framework, intends to integrate enablers with the applications, acting as a bridge between them.
  - Acting as an approach to develop applications using FIWARE
- It relies on a Docker-based approach to provide (if needed) several instances of the enablers, without the user having to pass through its installation process.
- The enabler can then be accessed through the EF API
- The orchestration of the services can be achieved through the Process Enabler, which generates a BPMN to be integrated in the vApp to execute the defined tasks.

The End

Questions?

Joao Sarraipa
jfss@uninova.pt
virtual factory Operating System

vf-OS IO Toolkit

2018-03-21

Victor Anaya, Nejib Moalla, Ludo Stellingwerff, José Luis Flores and Francisco Fraile

Index

- Motivation
- Drivers and API connectors in vf-OS – the IO component
  - A perspective from app developers
  - A perspective from manufacturing and logistic users
  - A perspective from drivers and API connector developers
- Security concerns
- Conclusions
Motivation

- vf-OS is an operating system that will permit applications to use existing manufacturing resources (devices and business applications)

Context of manufacturing apps(devices)

- Wide range of existing devices in manufacturing
  - Temperature
  - Humidity
  - Speed
  - Pressure
  - Flow Rate
  - Speed
  - etc

- Many protocols
**Context of manufacturing apps(legacy)**

- Siloed legacy system.
  - SAP
  - SharePoint
  - Microsoft Dynamics NAV
  - ORACLE
  - AUTOCAD

- Any with specific languages, APIs and environments to develop
  - COM objects
  - DLL
  - JAR / RMI
  - REST
  - C/C++
  - XML

**vf-OS input and output**

- Provide tools to **app developers** for:
  - Developing apps abstracting from complexity of different protocols
  - Developing apps that coexist and profit from existing business applications

[Diagram showing vf-OS platform with vApp 1, vApp 2, and vApp 3, with IO component connecting to devices in manufacturing and legacy applications.]
**Solution - vf-OS IO component**

- Black-box abstraction

![Diagram showing IO component and components](image)

- Devices
  - OPC UA
  - MQTT
  - MODBUS
  - GPIO
  - MTOCONNECT

- Systems
  - MS EXCEL
  - STEP
  - ODBC
  - ODATA
  - API
  - NAV EII

**Using drivers and API connectors**

![Diagram showing manufacturing and logistic companies](image)
Developing new IO components

- Driver and API connector developers

Abstraction Layer

Devices
- OPC UA
- MQTT
- MODBUS
- GPIO
- MTCONNECT

Systems
- MS EXCEL
- STEP
- ODBC
- CDATA
- SAP
- NAV ERP

Solution - vf-OS IO component

- Generate a unified abstraction

Top interfaces
- REST
- Messaging Pub/Sub
- Registration
- Logging

logic
- Computing
- Access Control
- Internal storage
- Lifecycle
IO Toolkit

- Command-line prompt tool to generate skeleton of IO components.

IO toolkit generator
IO components on the marketplace

Deployment example
Security Concerns

- **External services**
  - REST alignment with OWASP

- **Internal services**
  - ISA 99/IEC 62443 secure network architecture, service encapsulation

- **Asset development**
  - ISA 99 / IEC 62443 component development, digital signatures

- **Privacy and Personal information**
  - Security Control Centre, cryptographic recommendations

Conclusions

- **IO component** is the component in charge of wrapping and integrating devices and legacy applications in vf-OS apps.

- **IO components** are self-hosted, secured, server-based, docker components.

- Out of the box vf-OS plans to provide up to 20 drivers / API connectors.

- IO toolkit is the easy-to-use tool to extend vf-OS with new IO capabilities.
virtual factory Operating System

Thanks you

vanaya@cigip.upv.es

Victor Anaya, Nejib Moalla, Ludo Stellingwerff, José Luis Flores and Francisco Fraile
virtual factory Operating System

Data Management Component for Virtual Factories Systems

Joao Giao
UNINOVA

Data Management Component Objectives

The vf-OS applications will consume and produce tremendous amounts of data.

Objectives:

- Handle all vf-OS communication;
- Data storage capable of storing all vf-OS data;
- Harmonise vf-OS data;
- Analytic methods for enrichment, analysis and data interpretation.
Data Infrastructure Middleware

**Data Infrastructure Middleware** relies on Message Oriented approach based on RabbitMQ using the AMQP protocol. Develop the message and publish/subscribe components.

General representation of Data Infrastructure Middleware

**Producer-Consumer Model for vf-OS**

Communication steps:
1) Producer sends one message to one exchange and to one or more queues (with routingKeys);
2) Broker redirects the message to designed exchange;
3) The queues that are linked to the routingKeys will receive the message;
4) Consumers that are listening to specific queue get the message.
Data Storage

Data Storage has to provide storing services for heterogeneous data from heterogeneous sources.

- Relational data
- Time Series data
- Document oriented data
- RDF data

vf-OS usage:
- Relational data - relational data from vApps as well as other relational information of the vf-OS Platform itself;
- Time Series data - to allow storage and querying time series data. E.g: Sensor data;
- Document-Oriented data - to store, retrieve and manage document-oriented information, also known as semi-structured data;
- RDF data - to store and query subject-predicate-object triples to be used in the conceptualization taxonomy (Data Harmonization module).

Data Harmonisation

Data Harmonisation has to provide taxonomy connections for vf-OS data, through the detection of non-linear and non-trivial patterns within the data.

Design time:
1. Receive vf-OS data
2. Data arrangement
3. Link the data concepts
4. Store data concept connection for future use

Run:
1. Use pre-stored data relations to link vApps’ data
Data Analytics

Data Analytics has to provide analytical processing for sensor data.

Types of Data to be considered:
- Real-time data (alarms, critical actions);
- Historical Data (stored data for a certain period).

Types of algorithms to be used:
- Machine learning algorithms;
- Traditional data-mining algorithms (e.g. decision trees and rules, k-means, association rules).

In development

Conclusions

- The Data Management Component, to be developed during the vf-OS’ WP5, intends to manage data flows for vf-OS platforms.

Message Oriented Middleware
Data communication management

Data Harmonisation and Data Analytics
Transforming and representing data

Data Storage
Store vf-OS data

Main Goals
Scalability
Adaptability
The End

Questions?

Joao Giao
jgs@uninova.pt
virtual factory Operating System

An Open Environment for Development of Manufacturing Applications

2018-03-21 – Berlin, Germany

Carlos Coutinho
CEO

Evolution of ICT & SW

INTERNET OF THINGS
Cloud Computing

D10.4a: Workshops Reports - Vs: 1.0 - Public 105 / 334
Industrie 4.0

Industrie 4.0 – Sensing Industry
Virtual Factory

Traditional Operating Systems

User
Application
Operating System
Hardware
Open Apps Dev Kit (OAK)

OAK Software Development Kit
OAK SDK

- Concentrator for vf-OS resources
- Aggregator of APIs
- Tools & Procedures
- vApp Build and Deploy Libraries
- EFFRA coordination of WG-05 for SDKs

OAK SDK

- Abstraction of the Factory
- Abstraction of the Business
- Distributed Building
- Container isolation
- Interoperability between different systems
OAK Studio

- Web-based GUI of the SDK
- Container-Targeted
- Full-featured IDE
  - Syntax Highlighting
  - Drag & Drop
  - Debug
  - ...
- Code Editor & Build Interface
**OAK Process Designer**

- Model Process Workflows
- Orchestrate pre-built assets
- BPMN style definitions
- Reuse of vApps
OAK Front End

Front End Editor

- Allows simplified input of abstraction layers
- How-tos and auto-complete guides
- Based on Angular and Bootstrap
- Consumable and integratable
- MicroService of the vf-OAK studio
### Front End Editor

**Input:**
- Configuration (languages, etc.)
- XML / HTML-alike styling for the structure

**Output:**
- Clickable prototype
- Generates code for further usage

---

**Example:** `<app></app>` =

```
username
password
```

Not registered? Create an account.
System Dashboards

- Command & Control environment
- Central Dashboard
- Task Manager
- Provides information about runtime vApps

OAK Dashboards
Development Process

Developers Engagement Hub
Developers Engagement Hub

- Share Code & Development
- Disseminate the vf-Apps
- Promote Reuse
- Promote Development
- Engage Community

Developers Engagement Hub

- Community Building tools
- Open Source Git Repositories
- Multiple projects and Sub-projects
- Container Technology
- Continuous Integration
- Project Management
Developers Engagement Hub

- Dissemination:
  - Wikis
  - Blogs
  - Forums, etc.
  - Multimedia

- Collaboration:
  - Issue-trackers per Topic
  - Chats
  - Social Networks
  - Gamification

OAK Development Timeline

- OAK SDK
- OAK Studio
- Process Designer
- Front End
- System Dashboards
- Developers Engagement Hub
Questions?

Carlos Coutinho
CEO / R&D Manager
Carlos.coutinho@caixamagica.pt
www.caixamagica.pt
vf-OS Workshops Berlin, Germany - 21th March 2018
A novel approach to software development in the microservice environment of vf-OS

Luis Manteigas da Cunha, Ludo Stellingwerff and Andries Stam

Presentation

- Introduction
- vf-OS platform
- Development approach
- Conclusions
- Questions
Introduction

vf-OS

- It is an Open Operating System for Virtual Factories
- Offers a manufacturing oriented cloud platform
- Supports a multi-sided market ecosystem that provides services for the connected factory of the future
- Allows manufacturing companies to develop and integrate better manufacturing and logistics processes

vf-OS is composed of

- Kernel
- Application programming interface
- Middleware specifically designed for the factory of the future

Introduction

Challenge

Due to the highly distributed microservices architecture used in vf-OS, such an environment poses some challenges for the software development process which will produce the assets
vf-OS platform

Assets encompass
- Services
- Tools
- Applications

Assets interact with each other through web-technologies
- REST-services
- Web-based GUIs
- Modern message busses

vf-OS platform

The platform allows the assets to run both InCloud or OnPremise

Data might be produced within the factories and it can be used within cloud applications

Security challenges
- To allow the users to control their data dissemination
- Requires security measures to prevent undesired access to machinery and other local resources
vf-OS platform

An implementation option:

Locally platforms can communicate with a cloud platform through a controlled proxy asset.

The customer application will then run in the cloud platform and only get very controlled access to the data from the factory’s premises.

vf-OS platform

vf-OS assets

- Standardized structure and packaging format
- The executable entity will be based on Docker images
- Images will be enveloped in a wrapper structure, containing metadata, like access rules, dependency information and security signatures.
- The assets will be storable in the vf-OS Marketplace and can be bought and deployed into the Execution environment, provided by the Platform.
vf-OS platform

Problem
This distributed environment makes it harder to create consistent, coherent applications, especially with regard to debugging, versioning and other software life-cycle aspects.

Goal
An important part of the goals of the vf-OS project is aimed at providing an answer to these challenges.

Development approach

vApp is the end-user application as an asset.

The vApp development is the most visible asset development process.

But the development of the other assets within the vf-OS environment is very similar.

It is therefore a good approach to use a single development model for both application development and for the development of other assets.
Development approach

**vf-OS Studio**

- IDE-like application
- The main assets are Javascript-based services
- Several SDKs and Javascript libraries are provided for interacting with other assets
  - Pub/Sub mechanisms, messaging framework and access to storage facilities
- Each service will run in its own docker container

**Development approach**

**MarketPlace**

The MarketPlace has several tasks in the software development process:

- A marketing and sales channel
- A services registry
- Hosting of the assets for deployment
Development approach

External services

External services can be represented in the single development model as well, by encapsulating them into vf-OS Assets.

For this purpose the project provides a support module, called the External Services Provisioning framework.

Conclusions

- The core challenge is software development in such a microservices environment, with the added challenge of security in such a distributed environment.
- The solution is to have such an integrated studio, including external services and a proxy model for security.
- The novelty lies in the web-based tooling (e.g.: Eclipse CHE) combined with modern containerization (Docker), Microservices and an OS-like middleware for messaging.

Future steps are evaluating this approach, using the projects intermediate releases, use-cases and experimentation. The results of this evaluation will be reported in similar publications.
Questions?

contacts:
luis@almende.org, ludo@almende.org, andries@almende.org

Thank you!

contacts:
luis@almende.org, ludo@almende.org, andries@almende.org
Annex I: I-ESA’18 vf-OS Workshop Pictures

This annex illustrates the I-ESA’18 vf-OS Workshop with some pictures: the workshop opening, the workshop attendees and the papers presentation starting.
Annex J: IS’18 vf-OS Workshop Details

Workshop details

Wednesday, 26 September 2018

ENABLING COLLABORATION
1. Automated Negotiation and Trust in the NIMBLE Platform (Violeta, 3RFG)
2. Tender Decomposition and matchmaking in the Digicor platform (Grigory)
3. Process Orchestration in vf-OS (Oscar Garcia)
4. Marketplace in the COMPOSITION platform (Guiseppe) 12:30 - 12:45

SMART TOOLS
5. Smart Monitoring and Control in the DIGICOR (Gash Bhullar)
6. Enterprise Integration in the COMPOSITION platform (Marc)
7. Smart Factory SDK of vf-OS (Carlos)
8. IoT Toolkit of vf-OS (Victor)
9. Planning in a Distributed Supply Chain (Airbus)

FIWARE SOLUTIONS AND DEMOS
10. vf-OS Enablers Framework (UNINOVA)
11. FIWARE for industry and OPC-UA architecture (Claudio Polasciano)
12. The use of FIWARE enablers in SME cluster portal (Gash)

Thursday 27 September 2018

13. Business Models for Multi-Sided Platforms (Cesar)
14. Business Model Simulations for a Smart Factory Platform (Raúl)
15. Case Study from L4MS an Open Platform for innovation in Logistic-OPIL (Angelo)

BIG DATA MANAGEMENT
17. The Use of Deep Learning techniques in the COMPOSITION platform (Paolo Vergori, Instituto Superiore Mario Boella, Italy)
18. Product and Service Catalogue Management in the NIMBLE platform (Wernher Behrendt, Salzburg Research, Austria)
19. IoT Data Harmonization in vf-OS (Oscar Garcia, Information Catalyst, UK)

-------- ROUND TABLE DISCUSSION ---------------------------
Annex K: IS’18 vf-OS Connected Factories Workshop

Notes

Agenda 1st DAY
11:00 – Introduction to Connected Smart Factories Workshop – Usman Wajid (Information Catalyst, UK)
11:15 – Enabling Collaborations (Chair: Violeta Damjanovic-Behrendt)
  • Automated Negotiations and Trust in the NIMBLE platform (Violeta Damjanovic Behrendt, Salzburg Research, Austria)
    – Links among Federated business process and Multi-Sided platforms. Collaborations to be explored in order to match challenges
    – Multi-sided platform approach in NIMBLE (vf-OS WP9)
    – Automated negotiation -> Which are the main added values comparing the current business process solutions provided by commercial solutions (e.g. ERPs)?
    – Registration in the platform to collaborate
  • Tender Decomposition & Matchmaking in the DIGICOR platform (Grigory Pishchulov, University of Manchester, UK)
    – Team formation and collaboration mainly among SMEs to respond needs coming from big sectors like the Aerospace
    – Call for tenders -> How to manage common interests of several partners in the same tender parts or components that include these parts? -> What are the rules for doing it in these cases?
    – Call for Tenders -> Who is responsible for managing the platform (the final user (Airbus), tier 1, a third party, …)
  • Process Orchestration in the vf-OS platform (Oscar Garcia, Information Catalyst, UK)
    – Why do you use of code generation instead of BPMN engines?
    – How is the connection of tasks with the organisation structure (deparments, staff, )?
  • Marketplaces in the COMPOSITION platform (Giuseppe Pacelli, Instituto Superiore Mario Boella, Italy)
    – Agent-Based marketplace approach of COMPOSITION (useful for vf-OS WP7 -> vf-Store?)
    – Agent Management System with requester and supplier agents (something useful for ASC?)
    – Security infrastructure (to discuss among ASC, IKERLAN and …)
    – Blockchain -> something to consider in the vf-OS component vf-Store?
    – Clear possibilities of collaboration -> Approaches for ZDMP?
    – What kind of intelligence do agents have in the negotiation? They are some policies associated to the agents
    – Use of reputation model that could be used for DIGICOR (possibility)
    – Are there some kind of traceability of the decision making process? Now there is a log of the process
12:45 – Lunch Break
14:00 – Smart Tools (Chair: Gash Bhullar)
  • Smart Monitoring and Control in the DIGICOR platform (Gash Bhullar, SMECluster, UK)
    – Aerospace and Automotive sectors
    – Core platform to offer services to other platforms (e.g SME cluster platform)
- Third party applications connectivity (link to vf-OS External services connection component vf-OS WP5?)
- OPC-UA compliant for Smart monitoring
- Production monitoring, IoT Server (COMAU), and Kanban tool
- Open source implementation of OPC-UA
- Interoperability among platforms

- Enterprise Integration in the COMPOSITION platform (Marc Jentsch, Fraunhofer FIT, Germany)
  - Intra Factory & Inter Factory
  - Use Cases: Predictive maintenance in a polishing machine (performance improvement), Fill level notification in scrap container (logistics improvement), Equipment monitoring and Line Visualisation (monitoring the full shopfloor processes) & Component tracking (avoid component lost)
  - BPMN used in the third use case
  - RFID Tags for components location?

- SmartFactory SDK of vf-OS (Carlos Coutinho, Caixa Mágica, Portugal)
  - Differences among SDK and IDE -> Studio as the front end of the SDK
  - Reference implementation of the SDK with the idea of being used in other projects
  - Are DIGICOR, NIMBLE & COMPOSITION interesting in SDK?

- IO Toolitof vfOS (Víctor Anaya, Universitat Politècnica de València, Spain)
  - Drivers development using vf-OS approach IO Toolkit (interesting for DIGICOR?)
  - Drag and drop approach
  - Possibility to collaborate in this field with DIGICOR?

- Planning in a distributed supply Chain (Xabier Rakotomamonjy, AIRBUS, France)
  - Coordination transferred from Airbus to Risk Sharing Partners (1st Tier supplier list restricted to a small number of RSP).
  - Offer preparation and management (connected to the Tender presentation of DIGICOR)
  - What is the role of the Tender facilitator? Conducting the process, allocation of partners to components and subcomponents, planning & scheduling, … What is the level of automation?

General discussion
- Problems to test IO Toolkit in real manufacturing scenarios
- How to create a SDK to be used in other projects?
- What is the role of the Tender facilitator?
- How to manage by contract the use of data belonging to different companies? Use of security approaches (command centre in vf-OS)
- Data sharing and privacy as the main barrier for collaboration

15:45 – Coffee Break
16:15 – FIWARE Solutions and Demos (Chair: Carlos Agostinho)
- vf-OS Enablers Framework (João Sarraipa, UNINOVA, Portugal)
  - Enablers including User Interface or not? Current vf-OS position is that an enabler should not include an user interface (it will provide by other vf-OS components. Gash raises a different opinion.
- FIWARE for Industry and OPC UA architecture: an industrial use case (Claudio Palasciano, POLIMI, Italy)
  - CPS-ized production system
- The use of FIWARE enablers in SMECluster portal (Gash Bhullar, SMECluster, UK)
  - Some kind of collaboration among DIGICOR (SMECluster) and vf-OS?
18:00 – End of First Day

Agenda 2nd DAY

11:00 – Business Modelling (Chair: Eduardo Saiz)

- Business Models for Multi-Sided Platforms (Cesar Marin, Information Catalyst, UK)
  - Morphological box for Business models definition (Check it WP9 for potential application)
  - Tender costs improvement as the main revenue generated
  - Currently, only a fix scenario is evaluated -> Focus on DIGICOR profits

- Business Model Simulations for a Smart Factory Platform (Raúl Poler, Universitat Politècnica de València, Spain)
  - Starting with the vf-OS Stakeholders -> Marketplace -> Different approach than in DIGICOR
  - Considering different scenarios
  - Business Model -> How to start with the calculations? Surveys with stakeholders …
  - Not platform functionalities improvements are modeled in the future
  - Simulation tool as a Stakeholders engagement tool
  - Investors interest (tool to convince)
  - Validation the model with some real cases
  - To discuss the use of the simulation tool by DAEDALUS/NIMBLE, …

- Case Study from L4MS an Open Platform for innovation in Logistic-OPIL (Angel Margulo, Engineering, Italy)
  - Productivity but also flexibility
  - Stakeholders: Mid-Caps, Competences centres, Business developers, Technologies suppliers
  - Application experiments
  - Open Platform and a Marketplace -> What kind of business models are defined for going to the market?
  - Business developers for helping to open new business
  - Go the market not only to the companies that demand this kind of services

- Technology platforms and ecosystems, who drives the success? The case of the DAEDALUS platform (Andrea Barni, SUPSI-ISTePS, Switzerland)
  - IEC61499 Standard
  - vf-OS Technology suppliers -> Marketplace. Could be shared with vf-OS marketplace (vf-Store): control software, APIs, drivers, …?
  - IBM App Store -> Big coincidence with vf-Apps approach
  - DEADALUS Competence Centre
  - Individual Business Models and Business Plans. Are there some Joint Business Models?

General questions

- Multi-side platform – Roles of different stakeholders and benefits for them
- How complicated is to apply this tool to other projects? Customisation, experts software developers, licensing
- What about the figures? Different scenarios implies different outcomes (prediction)
- How to control the evolution of real figures when the product is introduced in the market
- What is the business model to exploit the platform (commercialisation, product development, support, …? Only individual-joint exploitation, Key project partner, Start-up, …
- How about data security and privacy? Some especial contract, technologies involved, …
12:45 – Lunch Break
14:00 – Big Data Management (Chair: Marc Jentsch)
- The Use of Deep Learning techniques in the COMPOSITION platform (Paolo Vergori, Instituto Superiore Mario Boella, Italy)
  - Artificial Neural Networks in Industry 4.0 -> Deep Learning
  - BSL Use Case: Predictive maintenance (2 ovens) -> dataset with 625,000, 350,000 samples, 60 sensors, ..
  - Interesting for prediction vApp? (MASS) (WP8) -> To sent presentation to Asier and Juan
  - Successful tests in Laboratory -> Ready to go with real validation
  - ELDIA: Agent-based marketplace estimation (4 raw materials markets)
- Product and Service Catalogue Management in the NIMBLE platform (Wernher Behrendt, Salzburg Research, Germany)
  - Enable the federation of B2B Platforms
  - Product Catalogue: UBL/DOM.ONTO/eCl@ss Ontologies
- IoT Data Harmonisation in vf-OS (Oscar Garcia, Information Catalyst, UK)
  - Data Manufacturing Maps, Data formats
  - Who can provide the semantic maps? Recommendation for something from the companies with the required knowledge
  - How about crowdsourcing? As much maps contributions the reference model will be richer
  - Have you done some statistics?
  - How often have to do these maps?
15:45 – Coffee Break
16:15 – Round Table Discussion (Chair: Usman Wajid)
- Challenges for Connected Smart Factories
  - Start ups: Governance procedures -> High percentage of communality
  - Availability of guidelines to support its definition
- Impact of FIWARE on Connected Smart Factories
  - vf-OS Enablers Framework -> To be used in DIGICOR
- Analysis of automation and digitalisation in the manufacturing value chain
- Common goals and collaboration opportunities

18:00 – End of Second Day
Annex L: IS’18 vf-OS Workshops presentations

This section presents the different presentations performed in the vf-OS workshop organised during the Intelligent Systems 2018 conference at Madeira, Portugal.

AUTOMATED NEGOTIATION AND TRUST IN THE NIMBLE PLATFORM

DR. VIOLETA DUMJANOViC-BEHRENDT
SALZBURG RESEARCH, IOT GROUP, AUSTRIA

CONNECTED SMART FACTORIES WORKSHOP

ABOUT THE PROJECT

- H2020 FoF NIMBLE (Collaborative Network for Industry, Manufacturing, Business and Logistics in Europe) project
- NIMBLE: B2B platform for IoT-based companies to register & publish manufacturing services, search for suitable partners, negotiate contracts & supply logistics
- 4 use cases: furniture, wooden houses, white goods, and textiles
- Platform based on permissive open source license (Apache)
- Ambitious adoption plan
- 17 partners – clustered around the 4 use cases
- 8M € budget
## Concept of Work

- Multi-Sided Platform (MSP)
- Federated cloud-based businesses
- Reduced search costs and shared transaction costs among participants
- Collaborative business processes
- Cybersecurity risks and possible harm to MSP participants

### NIMBLE platform

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Core functionality (registration of users and comp.; catalogue pub.;)</td>
<td>All GUIs functioning</td>
</tr>
<tr>
<td>Tools</td>
<td>Support for different platform instances (federation services)</td>
</tr>
<tr>
<td>Search functionality</td>
<td>Semantic search</td>
</tr>
<tr>
<td>Simple negotiation</td>
<td>Advanced negotiation</td>
</tr>
<tr>
<td>Security</td>
<td>Agent-based negotiation</td>
</tr>
<tr>
<td>User authentication, Role-based AC</td>
<td>Customizable business workflows</td>
</tr>
<tr>
<td>Security</td>
<td>Context-based AC; Sec. dashboard</td>
</tr>
<tr>
<td>Industry connectivity</td>
<td>Trust and reputation</td>
</tr>
<tr>
<td>Data channels for remote monitoring</td>
<td>Support for additional standards, e.g. oneM2M</td>
</tr>
<tr>
<td>Tool store</td>
<td>Testing environment</td>
</tr>
<tr>
<td>Third party APIs</td>
<td>Advanced infrastructure diagnostics</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Centralized logging and monitor</td>
</tr>
<tr>
<td>Use cases</td>
<td>Integration of extern. adopters</td>
</tr>
<tr>
<td>Integration</td>
<td>Access from the 3rd party apps</td>
</tr>
<tr>
<td>Developers support</td>
<td>More detailed documentation</td>
</tr>
<tr>
<td>Basic documentation</td>
<td>Extensive documentation</td>
</tr>
<tr>
<td>Community support</td>
<td>Access to testing version</td>
</tr>
<tr>
<td>Early adopters programme for companies and platform providers</td>
<td>Subscription to federated platforms</td>
</tr>
</tbody>
</table>
FUNCTIONALITIES TO BE AUTOMATED

- Release 3 (July 2018) functionality: https://www.youtube.com/watch?v=T-eVQDIhjM
- reviewing negotiation processes for inefficiencies & reinventing business models, strategies and platform technological capabilities

NIMBLE functionalities to be automated
1. Negotiation of order placement and execution
2. Negotiation of supply chain and logistics processes
3. Negotiation and collaboration through data channels
4. Trust negotiation

AUTOMATED NEGOTIATION OF ORDER PLACEMENT AND EXECUTION
- UBL order process
- Current functionalities of the platform:
  1. inventory management
  2. order placement
  3. order routing
AUTOMATED NEGOTIATION OF ORDER PLACEMENT AND EXECUTION

- **Challenge 1**: the ability of the platform to track and verify products at any stage - in design, development, production, in shipment and possibly - at customer sites

- Visualization on the location, status and environment of assets using the IoT + Computer Vision + data analytics applications
- **Sensors & smart cameras** embedded in production machines, logistics facilities, vehicles, raw materials and products themselves to enable tracking and monitoring conditions in real time, as well as the speed and timeliness of shipment. They contribute to the product quality and (early) defect elimination
- **Data analytics** contribute to inventory replenishment and the requirements of customer’s demand for specific products

-> NIMBLE data channels !!!

AUTOMATED NEGOTIATION OF ORDER PLACEMENT AND EXECUTION

- NIMBLE data channels
- see Release 3, from 24:50: https://www.youtube.com/watch?v=T-eVQDihijM

- Apache Kafka (open source messaging infrastructure & streaming platform)
- Zookeeper (service for maintaining configuration information, naming, providing distributed synchronization and group services)
- Schema Registry (metadata service registry)
AUTOMATED NEGOTIATION OF ORDER PLACEMENT AND EXECUTION

- Open data channel (supplier side)
Automated Negotiation of Supply Chain and Logistics Processes

- **Challenge 2**: the ability of the platform to recommend other supply chain and logistics elements before placing the final decision on what to order, from whom, and how to organize transportation (shipment tracking)
  
- Search & recommender system based on ranking algorithms that include context, behavioural similarity features, (semantically meaningful) locations and time dependent features, popularity, etc.
- Ranking that incorporate metadata such as categories of businesses (e.g. wood manufacturing close to the furniture producer)
- Bayesian Personalized Ranking algorithm

\[
p(A|B) = \frac{p(B|A)p(A)}{p(B)}
\]
AUTOMATED NEGOTIATION OF SUPPLY CHAIN AND LOGISTICS PROCESSES

Some examples of data to be used for NIMBLE ranking algorithms:

Distance between the buyer and the seller

Properties of the seller
• How popular/highly rated is the company?

Properties of the buyer
• How many times has the buyer checked in here? When? Behavior (liked/disliked the seller or any seller’s product in the past)

Time of the day
• Is the company open right now? How popular is the company?

Properties of the company’s social network

AUTOMATED NEGOTIATION OF SUPPLY CHAIN AND LOGISTICS PROCESSES

• Challenge 3: the ability of the platform to trace supply chain and logistics using blockchain

• The digital ownership certificate - the blockchain stores an ownership relation by storing physical properties of the asset, e.g. a serial number, the wallet of the owner …

• Tracking down assets and assembly in the supply chain – for each transaction, the blockchain can provide information on the involved parties, the timestamps, notifications about the execution of various steps …

• Trusted devices - have a cryptographic identity and should be resistant to misuse. They can be used to automatically update the tracking information in case of a goods delivery.
AUTOMATED TRUST NEGOTIATION

- **Challenge 4**: the ability of the platform to create and use trust and reputation features of business entities for search and recommendation

- Collecting ranking data (explicit ranking (could be influenced by social trends), implicit ranking (what events occurred between the users?)), user behaviour/taste …

- Model training phase – with various levels of “personalization”, from fully personalized (based on personal taste/needs) to the most popular content (non-personalized content)

- Filtering based on similarity measurements (between items, between users…)

- Model evaluation phase – e.g. implementing the Mean Average Precision in the trust system
CONCLUSION

- **Challenge 1**: NIMBLE Data Channels for tracking and verifying products at any stage, at manufacturer side - in design, development, production
- **Challenge 2**: NIMBLE supply chain and logistics recommender service - what to order, from whom, and how to organize transportation
- **Challenge 3**: NIMBLE blockchain-based tracing and tracking service
- **Challenge 4**: NIMBLE automated trust-based search and negotiation

THANK YOU FOR YOUR ATTENTION

**Violeta Damjanovic-Behrendt**
Senior Researcher
Salzburg Research, Austria
E-Mail: violeta.damjanovic@salzburgresearch.at

**Salzburg Research Forschungsgesellschaft m.b.H.**
Internet of Things / Industrial Internet
Jakob-Haringer Strasse 5/II
5020 Salzburg
Austria
NIMBLE

JOIN THE PLATFORM:

HTTPS://NIMBLE-
PLATFORM.SALZBURGRESEARCH.AT/NIMBLE/FRONTE
Tender decomposition and matchmaking in the DIGICOR platform

Grigory Pishchulov, Sonia Cisneros-Cabrera, Pedro Sampaio, Zixu Liu, Sophia Kununka, Nikolay Mehandjiev

University of Manchester, UK
St. Petersburg State University, Russia

Connected Smart Factories Workshop
9th international Conference on Intelligent Systems, 25-27 September 2018, Madeira

Agenda

1. Introduction
2. Call for tenders
3. Tender decomposition
4. Matchmaking
5. Conclusion
Introduction

Matchmaking

<table>
<thead>
<tr>
<th>Company</th>
<th>Product</th>
<th>Capabilities</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company 1</td>
<td>Lavatories</td>
<td>Make</td>
</tr>
<tr>
<td>Company 2</td>
<td>Lavatories</td>
<td>Design, Manage</td>
</tr>
<tr>
<td>Company 3</td>
<td>Door Modules</td>
<td>Make, Deliver</td>
</tr>
<tr>
<td>Company 4</td>
<td>Housing</td>
<td>Design, Make</td>
</tr>
<tr>
<td>Company 5</td>
<td>Outer Walls</td>
<td>Make, Deliver</td>
</tr>
<tr>
<td>Company 6</td>
<td>Outer Walls</td>
<td>Design, Manage</td>
</tr>
<tr>
<td>Company 7</td>
<td>Outer Walls</td>
<td>Deliver</td>
</tr>
<tr>
<td>Company 8</td>
<td>Ceiling Panels</td>
<td>Design, Deliver</td>
</tr>
<tr>
<td>Company 9</td>
<td>Ceiling Panels</td>
<td>Design, Make</td>
</tr>
<tr>
<td>Company 10</td>
<td>Ceiling Panels</td>
<td>Make, Manage</td>
</tr>
</tbody>
</table>

Introduction

Artwork: Majumder and Srinivasan (2008)
Call for tenders (CfT)

1. Target item
   » ontology

2. Associated goals
   » design
   » make
   » deliver

3. Capabilities required
   » certification
   » min. turnover
   » min. number of employees

4. Capabilities desired
   » industry branch, technology, materials, target regions

Agenda

1. Introduction

2. Call for tenders

3. Tender decomposition

4. Matchmaking

5. Conclusion
Tender decomposition: items

Tender decomposition: goals

Plan & manage

Design

Source

Make

Deliver

Plan & manage

Design

Design

Make

Assemble

Make

Deliver

Plan & manage
Tender decomposition

1. Item decomposition
   » stepwise BOM explosion → generate all rooted subtrees
   » each time, decompose a subset of items into sub-items
   » add sub-items to the list of items → a new decomposition
   » put decomposition in the queue

2. Goal decomposition
   » each item may have several associated goals
   » some goals may be decomposed, and some remain as is
   » multiple goal decompositions per item decomposition

Agenda

1. Introduction
2. Call for tenders
3. Tender decomposition
4. Matchmaking
5. Conclusion
Matchmaking

Criteria

1. Inclusion criteria
   » apply at the company level
   » exclude individual companies from search

2. Grouping criteria
   » apply at the team level
   » check eligibility of composed teams

3. Evaluation criteria
   » apply to individual companies within a team
   » apply to the team as a whole

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Inclusion</th>
<th>Grouping</th>
<th>Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Item–goal capabilities</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Annual turnover</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>No. of employees</td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Certification</td>
<td>X</td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Industry branch classification</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Materials</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Technology</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Preferred contract types</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Target regions</td>
<td></td>
<td></td>
<td>X</td>
</tr>
<tr>
<td>Locations</td>
<td></td>
<td></td>
<td>XXX</td>
</tr>
</tbody>
</table>
Matchmaking

Evaluation criteria → team risk

- **Company-level**
  - *item–goal capability*: exact instance, product class, other instance?
  - *certificates*: how well does the company cover CfT requirements?
  - *low coverage* → *high risk*
  - *similarly*: industry branch classifications, technology, materials, preferred contract types, target regions

- **Team-level**
  - *number of companies*: increases team risk
  - *manufacturing locations*: proximity of partners to each other reduces team risk

**Example**
Conclusion

- **Outcomes**
  » tender decomposition and matchmaking algorithm
  » decomposition of items and goals, any product complexity
  » team composition and evaluation subject to multiple criteria
  » same algorithm for replacement of team members, CfT announce

- **To do**
  » user evaluations
    • functionality and UI
    • matchmaking criteria

Thank you for your attention!

Your opinion matters to us
APPENDIX

SCOR model:

Plan

Source  Make  Deliver

Return

Enable (manage)

Source: APICS (2017)

Product delivery strategies:

Design  Fabrication & procurement  Final assembly  Shipment

Deliver-to-order  OPP

Assemble-to-order  OPP

Make-to-order  OPP

Engineer-to-order  OPP

Tender decomposition

Item decomposition: list of lists

<table>
<thead>
<tr>
<th>ID1</th>
<th>ID2</th>
<th>ID3</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>item1</td>
<td>item1</td>
<td>item1</td>
<td>...</td>
</tr>
<tr>
<td>item11</td>
<td>item11</td>
<td>item11</td>
<td>...</td>
</tr>
<tr>
<td>item12</td>
<td>item12</td>
<td>item12</td>
<td>...</td>
</tr>
<tr>
<td>item13</td>
<td>item13</td>
<td>item13</td>
<td>...</td>
</tr>
<tr>
<td>item111</td>
<td>item112</td>
<td>item113</td>
<td>...</td>
</tr>
</tbody>
</table>

Tender decomposition

Goal decomposition: list of lists of lists of lists

<table>
<thead>
<tr>
<th>GDlist1</th>
<th>GDlist2</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GD1</th>
<th>GD2</th>
<th>GD3</th>
<th>...</th>
</tr>
</thead>
<tbody>
<tr>
<td>item1</td>
<td>item11</td>
<td>item12</td>
<td>item13</td>
</tr>
<tr>
<td>goal1</td>
<td>goal2</td>
<td>goal3</td>
<td>...</td>
</tr>
</tbody>
</table>
3. **Preventing combinatorial explosion**
   - one-step look ahead
   - do not decompose an item if:
     - none of platform members can fulfil a goal for a sub-item,
     - even after having the sub-item decomposed
   - "can fulfil an item–goal" = can fulfil the goal for:
     - exactly this item, or its class, or any item from that class

4. **Speeding up**
   - whitelisting item–goals that can be fulfilled
   - blacklisting item–goals that can’t
   - blacklisting item–goals due to blacklisted children
Process Orchestration in the vf-OS Platform
2018-09-26 – Funchal, Portugal
Oscar Garcia Perales
Operations Director, Information Catalyst (ICE)

But before getting to the meat...
The Ages of the OS: iOS

...and now: vf-OS

virtual factory Operating System
If you want to create an Android application you require an SDK with java programming and an Android Marketplace to exploit it.

If you want to create an iOS application you require an iOS SDK with the swift language and the Apple App Store to exploit it.

If you want to create a vf-OS application you require the vf-OAK and utilise the vf-OS Platform to exploit it.

---

**Software Operating System Environment** | **vf-OS Equivalent**
--- | ---
Kernel, Processor, Memory, Internal Bus | Virtual Factory System Kernel Framework, Enablers
I/O Interfaces, Device Drivers, Peripherals | Virtual Factory I/O Device Drivers, APIs, Security
File and Data Handling Interfaces | Virtual Factory Middleware Data Infrastructure, Storage, Harmonisation, Analytics
SDK Application Development, System Monitor | Open Applications Development Kit System Dashboard, Frontend Environment, Studio, Developer Engagement Hub
Applications ERPs, CRM, MES, WMS | **vApps** Collaboration in real-time, monitoring
Objective within vf-OS:
- Provide an environment for developers to create vApps

What is needed for building a vApp?
- **Runtime Environment:**
  Place to execute a vApp and access components and services
- **OAK Development:**
  Tools to design and develop vApps
- **Data Management:**
  Components that deal with data, eg store, analyse, ETL, etc
- **I/O Toolkit:**
  Components to access machines & software, and security
- **Process Designer and Enablers:**
  Components to design internal workflows of a vApp and provide additional functionality from FI-WARE
vApps and how they are developed

What is a vApp?
- Applications specially developed for solving particular industrial needs.

How are they developed?
- Re-use existing libraries
- Developing new functionality
- Assembly of these functionalities
- Executing the vApp
- Routing messages
- Monitoring performance
- Storing intermediary and final results
Process Orchestration in the vf-OS Platform
2018-09-26 – Funchal, Portugal
Oscar Garcia Perales
Operations Director, Information Catalyst (ICE)

vf-OS Process Enabler

- Process Designer
  - A component to design Process Models

- Process Execution
  - A component to design Process Models
**vf-OS Process Enabler**

- **Process Designer**
  - A component to design Process Models, but...
  - What if a vApp can be seen as a process model, then...
  - Component to help designing/developing a vApp

- **Process Execution**
  - A component to run Process Models, but...
  - What if source code can be linked to the Model
  - What if source code can be generated based on a BPMN process model, then...
  - The Execution helps the developer in constructing source code

**vf-OS Process Designer**
vf-OS Process Designer

- Model Process Workflows
- Orchestrate pre-built assets
- Based on BPMN style definitions
- Reuse of vf-OS Assets

Interpreting BPMN when developing vApps

- In BPMN world, interaction with external services is described visually by Service Tasks
- Each Service Task is a rounded box in a diagram
- Inspection of service task definition to generate the corresponding call
- Parameters for the calls are accessed through:
  - Set those parameters in the message start
  - Manually typing the parameter values when the Service Task is dropped
  - Supply them through an incoming script task outputs
  - Let them flow from an incoming service task outputs
Let’s build a vApp (as far as possible!)

Justification

Business need:
- Create vApp capable of triggering alarms when a Tabber Stringer machine sensor hits a given value
Start LIVE DEMO

Well… a bit fake, but do come to our booth in the Registration room
### Process Designer

- Visual aspect
- Configuring a vApp

### Process Execution

- Source code generator
- Code to be used at the core of the vApp
Process Execution

- Linking custom functions to process steps

Questions?
Ecosystem for Collaborative Manufacturing Processes – Intra- and Interfactory Integration and Automation

Marketplaces in the COMPOSITION platform

Giuseppe Pacelli
pacelli@ismb.it
Istituto Superiore Mario Boella

Summary

- The COMPOSITION project
  - Goals
- COMPOSITION Marketplace
  - Concept Overview
  - Enabling technologies
  - Innovations
  - Impact and Re-usability
- Pilots
  - Scrap Metal Collection Scenario
  - Service Matching Scenario
- Demo
The COMPOSITION Project

- H2020 - FOF-11
  - Grant Agreement No 723145
- 12 partners
- 6 EU Countries
- 3 years
  - 09/2016 – 09/2019

Goals

- Create a (semi-)automatic ecosystem, incorporating and inter-linking both the Supply and the Value Chains
**Challenges**

**Value Chain**
- Unveil and make usable knowledge “embedded” in the factory shop-floor
- Exploit data, at all levels
  - Shop floor
  - SCADA/MES
  - ERP
- Enhance response latency to changing market requirements

**Supply Chain**
- Define an ecosystem of connected and interoperable factories
- Allow for quick and seamless supply chain formation / reconfiguration
- Open new business opportunities

---

**COMPOSITION Marketplace**

---
Marketplace enabling technology

- **Multi Agent System**
  - Agent Management System, White Pages Service
  - Requester and Supplier Agents
  - Composition eXchange Language

- **Security infrastructure**
  - Access Control
  - Channel encryption (TLS)
  - RBAC for agents

- **Blockchain**
  - Non-repudiation
  - Audit trail
  - Message Provenance & Trust
  - Material traceability
Innovations

- Extension of the “Agent-container” definition
  - “An agent container is a set of intelligent agents interacting through the same, shared broker (can be a cluster) and referring to shared platform services such as the Directory Facilitator and the Agent Management System.” In COMPOSITION a more advanced version of such an agent, namely the MatchMaker, operating based on ontology models is adopted.

- Open and closed marketplaces
  - “A COMPOSITION Open Marketplace is a COMPOSITION Marketplace open to any stakeholder having valid COMPOSITION credentials”.
  - “A COMPOSITION Closed Marketplace is a COMPOSITION Marketplace owned by one stakeholder and typically offered to a trusted subset of other COMPOSITION stakeholders.”

Impact and Reusability

- Improved responsiveness to dynamic market demands
- Improved collaborative extra logistical chains

- Composition eXchange Language easily extendable to cover different scenarios
- Several negotiation protocol could be easily supported and added through simple software modules
Pilots

Supply chain formation / adaptation

Manager at KLEEMANN receives notification about current fill level and the initiation of the feeding process in the market place. Manager can control the processes or abort the action. Interested companies place their offers and are pre-selected from the COMPOSITION System. The manager (KLEEMANN) is then asked to select the favorable candidates. Notifications are sent and a pick-up arrangement is done automatically.
Pilots – ATLANTIS

• Service / need matching
• Pre-negotiation

26/09/2018  Marketplaces in the COMPOSITION platform

Demo

Bidding Process Management

Requester KLE

Supplier ELDIA

Supplier Competitor

26/09/2018  Marketplaces in the COMPOSITION platform
Thanks for your attention

Contacts

Pacelli Giuseppe,
Istituto Superiore Mario Boella,
Pervasive Technologies Research Area,
pacelli@ismb.it
www.ismb.it

All rights reserved. All material in this presentation is owned in full by the COMPOSITION Project. Permission is granted to print material published in this presentation for personal use only. Use for any other purpose, and in particular for commercial use or distribution, is strictly forbidden in the absence of prior written approval.

COMPOSITION has received funding from the European Union’s Horizon 2020 Framework Programme for Research and Innovation under Grant agreement No 723545.

Possible inaccuracies of information are under the responsibility of the project. This presentation reflects solely the views of its authors. The European Commission is not liable for any use that may be made of the information contained therein.

Please see us here: www.composition-project.eu

26/09/2018

Marketplaces in the COMPOSITION platform
Decentralized Agile Coordination Across Supply Networks

Smart Monitoring and Control in the Digicor Platform
26th – 27th September 2018, Madeira
Gash Bhullar, Control 2K Limited

Introduction

Featuring Mid-term Review Slides from Digicor Technical Roadmap
Presented on 19th April 2018 in Prague by Filip Jiru, Certicon

- Platform Overview
- Goals and Challenges
- Road map overview
- Factory Connectivity and Smart Monitoring
- Concluding remarks
Goals and Challenges

The DIGICOR goal:
To provide extensible secure collaborative platform including communication with factories

Challenges:
- Third party tools vs. security
- Innovative Industry 4.0 approaches vs. readiness of companies
- Research project aiming to production
- Several distinct use cases
- Distributed development team
- Unknown target deployment

Good design decisions are critical

Design and architectonic decisions

Microservice architecture
Each service is developed by a separate team that holds responsibility

Sticking to technology standards
Promoting standards such as REST, OPC-UA, ISA-95, MPEG-M

Leveraging event driven architecture
To enable loose coupling, implementing event sourcing

Using production ready technologies and practices
Kubernetes, AWS, Continuous Delivery
Architecture overview

Technical road map overview

<table>
<thead>
<tr>
<th>Component</th>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digicor Portal</td>
<td>Partial implementation</td>
<td>All GUIs functioning</td>
<td>Configurable layout</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Personalization</td>
</tr>
<tr>
<td>Tools</td>
<td>Partial functionality</td>
<td>All reference tools working</td>
<td>Advanced algorithms</td>
</tr>
<tr>
<td>Security</td>
<td>User authentication</td>
<td>Complete access control</td>
<td>Full data security</td>
</tr>
<tr>
<td>Factory connectivity</td>
<td>One way (Factory-&gt; Digicor)</td>
<td>Two way communication</td>
<td>Pub-Sub OPC UA</td>
</tr>
<tr>
<td></td>
<td>Limited data model</td>
<td>ISA-95 implementation</td>
<td>Controlling factories from Digicor</td>
</tr>
<tr>
<td>Tool store</td>
<td>CD implementation</td>
<td>Testing environment</td>
<td>Tool approval process</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>Infrastructure services in place and running</td>
<td>Centralized logging and monitoring</td>
<td>Payment gateway</td>
</tr>
<tr>
<td>Use cases Integration</td>
<td>None</td>
<td>Single sign on</td>
<td>Advanced Infrastructure diagnostic</td>
</tr>
<tr>
<td>Developers Support</td>
<td>Service template</td>
<td>Accessing Digicor services from outside Digicor platform</td>
<td>Full integration of all three use cases</td>
</tr>
<tr>
<td>Community support</td>
<td>Guided presentations</td>
<td>Access to testing version</td>
<td>Extensive documentation available online</td>
</tr>
</tbody>
</table>

Deployed in Kubernetes in AWS
Digicor Portal road map

- Partial implementation using mockups
- All GUIs are functioning
- Configurable layout
- Personalization

Frequent feedback from customers is required

Factory connectivity road map

- One way (Factory to Digicor) communication
- Limited data model
- Bi-directional communication
- Full ISA-95 implementation
- Mapping factory sources
- Pub-Sub OPC UA
- Controlling factories from Digicor

Suitable use cases should be consulted with innovative companies
**Tool store road map**

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Continuous Delivery implementation</td>
<td>Separate parallel testing environment</td>
<td>Tool approval process</td>
</tr>
<tr>
<td>Deploy to testing</td>
<td>Deploy to testing</td>
<td>Deploy to testing</td>
</tr>
<tr>
<td>Deploy to production</td>
<td>Deploy to production</td>
<td>Deploy to production</td>
</tr>
<tr>
<td>Tool subscriptions</td>
<td>Payment gateway</td>
<td></td>
</tr>
</tbody>
</table>

Essential for microservice architecture & production environment

**Use cases integration road map**

<table>
<thead>
<tr>
<th>Phase 1</th>
<th>Phase 2</th>
<th>Phase 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>Single sign on</td>
<td>Full integration of all three use cases</td>
</tr>
<tr>
<td>Digicor portal</td>
<td>Accessing Digicor services from outside Digicor platform</td>
<td></td>
</tr>
<tr>
<td>SME portal</td>
<td>Digicor portal</td>
<td></td>
</tr>
<tr>
<td>SME cluster</td>
<td>Digicor portal</td>
<td></td>
</tr>
<tr>
<td>CTU</td>
<td>Digicor portal</td>
<td></td>
</tr>
<tr>
<td>Flexxweb Comau</td>
<td>Digicor portal</td>
<td></td>
</tr>
<tr>
<td>Digicor portal</td>
<td>Digicor portal</td>
<td></td>
</tr>
</tbody>
</table>

Development of tools serving needs of all use cases should be promoted
Specific Usecases in DIGICOR

DIGICOR Factory Connectivity: Use Cases

Production monitoring
Two-way communication between FC & FG
Provide connectivity between factory data sources and DIGICOR tools

IoT Server
The DIGICOR production data model can reference the robot condition data provided by the COMAU IoT server to better monitor the status of the CN -Equipment class-

Send PTP motion program to robot
Stream joint angles to 3D view of robot

Kanban tool
OPC UA Pub-Sub can be used to exchange data between IndustrieWeb and the Kanban tool
The DIGICOR production data model can be utilized for data exchange -inventory buffer class-

Factory connectivity: Smart Monitoring

DIGICOR platform

The technical components

DIGICOR platform UI
Production monitoring
Factory Gateway

Factory Connector

Event Store

Factory premises

OPC UA Pub-Sub
OPC UA servers/clients

AWS

Eventuate

Factory premise

OPC UA Pub-Sub
Over AMQP (ActiveMQ) broker

Factory Gateway

Eventuate
Conclusion

- Continuous need to have seamless connectivity to Enterprise Data
- DigiCor like many other platform still needs a Digital Backbone to allow aggregated data to be presented to applications and services
- Infrastructure needs to be secure for company data but open for security enabled services

Thank you for your attention!
COMPOSITION - Ecosystem for Collaborative Manufacturing Processes – Intra- and Interfactory Integration and Automation

IoT Applied to Factories of the Future
Marc Jentsch
Fraunhofer FIT

Value Chain

Supply Chain

26.09.2018  Intelligent Systems 2018 - Connected Smart Factories
## Introduction of Pilot Partners

- **Boston Scientific**
  - Medical Device Manufacturer
  - Optimization of Internal Processes

- **Kleemann**
  - Lift Manufacturer
  - Optimization of Waste Processes

- **EANIA**
  - Recycling Company
  - Customer Relationship with Kleemann

## Use Cases

<table>
<thead>
<tr>
<th>Tier</th>
<th>Use Case</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tier 1</td>
<td><strong>UC-BSL-2 Predictive maintenance</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-KLE-1 Maintenance decision support</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-KLE-4 Scrap metal collection and bidding process</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-ELDIA-1 Fill-level notification – Contractual wood and recyclable materials management</strong></td>
</tr>
<tr>
<td>Tier 2</td>
<td><strong>UC-BSL-5 Equipment monitoring and line visualization</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-KLE-2 Delayed process step</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-BSL-3 Component tracking</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-ATL-3 Searching for recommended solutions</strong></td>
</tr>
<tr>
<td>Tier 3</td>
<td><strong>UC-KLE-3 Scrap metal and recyclable waste transportation</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-BSL-7 Automatic long term tracking of high value materials for physical security</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-BSL-4 Automatic solder paste touch up</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-KLE-7 Ordering raw materials</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-ATL-1 Selling software/consultancy</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-ATL-2 Searching for solutions</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-ATL/NXW-1 Integrate external product into own solution</strong></td>
</tr>
<tr>
<td></td>
<td><strong>UC-NXW-1 Decision support over marketplace</strong></td>
</tr>
</tbody>
</table>
UC-KLE-1 Maintenance Decision Support

- Forecast Motor Breakdowns of Polishing Machine

- Combine Historical Breakdown Data with Vibration Sensor Data
- Vibration 0.038 m/s² data at 1.344 kHz (3 axes)

Figure 21: Bossi Motor (internal)

Figure 22: Bossi Motors (external)
UC-KLE-1 Maintenance Decision Support

Bossl Machine Probabilities
- Normal Operation: Current Probability 36.32%
- Mechanical Failure: Current Probability 50.47%
- Electrical Failure: Current Probability 13.42%
- Hydraulic Failure: Current Probability 9.41%

Vibrometer Graphs
- Bossl - Acceleration
- Bossl - Acceleration (X)

26.09.2018
Inelligent Systems 2018 - Connected Smart Factories

UC-ELDIA-1: Fill-level Notification

Fill-level

Send Trucks
**UC-ELDIA-1: Fill-level Notification**

Sx1272mbas Lora communication module attached on the mcu PCB, Antenna HRLV-MaxSonar EZ (MB1043) Ultrasonic Sensor and a 4 AA Battery Pack.

LoRa gateway access to Ethernet and power within 100m radius from the bins.

---

**Material Management Containers**

<table>
<thead>
<tr>
<th>Container</th>
<th>Content</th>
<th>Full in</th>
<th>Current fill level</th>
<th>Last emptied</th>
<th>Price if sold today</th>
<th>Filling prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Metal bars</td>
<td>6 days</td>
<td>50%</td>
<td>12 days ago</td>
<td>120 € / ton</td>
<td>6 days / 5 days</td>
</tr>
</tbody>
</table>

---

**Container B**

<table>
<thead>
<tr>
<th>Container</th>
<th>Content</th>
<th>Full in</th>
<th>Current fill level</th>
<th>Last emptied</th>
<th>Price if sold today</th>
<th>Filling prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scrap metal</td>
<td>1 day</td>
<td>36%</td>
<td>2 days ago</td>
<td>140 € / ton</td>
<td>1 day / 14 days</td>
</tr>
</tbody>
</table>

---

**Container C**

<table>
<thead>
<tr>
<th>Container</th>
<th>Content</th>
<th>Full in</th>
<th>Current fill level</th>
<th>Last emptied</th>
<th>Price if sold today</th>
<th>Filling prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Scrap metal</td>
<td>5 days</td>
<td>20%</td>
<td>8 days ago</td>
<td>100 € / ton</td>
<td>5 days / 10 days</td>
</tr>
</tbody>
</table>
UC-BSL-5 Equipment Monitoring and Line Visualisation

- Real time visualisation of production line’s efficiency
  - Representation of Involved Equipment
  - Light Tower Equipment Status
  - Actual Production
  - Target Production

20/20  12/20  10/20  3/20  2/20  1/20  0/20
Laser  DEK    SPI    Mount  2D    Reflow  3D

BPMN: Business Process Model and Notation
- Graphical Notation for Specifying Business Processes
- Available Software Tools to Execute Logic behind a BPMN Model
UC-BSL-5 Equipment Monitoring

Virtual factory Operating System – www.vf-OS.eu

UC-BSL-3 Component Tracking

- Pieces of Equipment get lost

Challenge | Technology | Application
----------|------------|------------
26.09.2018 | Intelligent Systems 2018 - Connected Smart Factories | 15
UC-BSL-3 Component Tracking

- BLE Tracking System
- Proximity-based: RSSI (Received Signal Strength Indication)

Test A
Tracking a single tag
Thanks for your attention

All rights reserved.

All copyright for this presentation are owned in full by the COMPOSITION Project. Permission is granted to print material published in this presentation for personal use only. Its use for any other purpose, and in particular its commercial use or distribution, is strictly forbidden in the absence of prior written approval.

COMPOSITION has received funding from the European Union’s Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No 723345.

Possible inaccuracies of information are under the responsibility of the project. This presentation reflects solely the views of its authors. The European Commission is not liable for any use that may be made of the information contained therein.

Please see us here: www.composition-project.eu
 SDK on vf-OS

Timeline: M13 – M36

- T6.1: SDK: CMS
  - D48/D6.1 (SW, PU)
  - D49/D6.2 (SW, PU)
  - D50/D6.3 (SW, PU)
  - D51/D6.4 (Doc, PU)
  - D52/D6.5 (Doc, PU)
  - D53/D6.6 (Doc, PU)

Within Task T6.1, the vf-OS SDK (D6.1.1) supports service developers with the functionalities needed to define, design, develop, and orchestrate manufacturing-related services. The vf-OS Applications/Services SDK, templates and manifest will be developed. Based on the knowledge and technology from previous projects such as Simplicity (in the mobility sector) and SDO4All (generic services), the project participants will develop an extensive fully documented API framework that provides developers with the means to easily generate applications and services. The API contains the means to combine services in order to provide value-added service compositions. An important part of OAK will be the Application's Manifest, which will characterise the applications developed in the vf-OS Marketplace of WP7. The Service API offers methods to access functionalities of all technical components that service developers need, such as the easy integration of data from the Cloud-based Information infrastructure, and the storage of service-specific data within the Cloud. After the applications/services have been developed, they can be registered in the WP7 marketplace. Within vf-OS, applications will be developed in WP5. The deliverable D6.1.2 documents the OAK SDK, enabling developers to efficiently develop applications.

vf-OAK Environment

- Software Development Kit (SDK)
  - Environment supporting the development, building and integration of vApps
  - Centralised access to the vf-OS assets
  - APIs and Documentation for interoperability

![Diagram of SDK on vf-OS](image-url)
Progress so far...

- **Software Development Kit (SDK)**
  - Defined basic infrastructure for the SDK
  - Documentation platform and SDK Skeleton
  - Started the Integration with Messaging, Pub/Sub and Data Storage

- **First Deliverable (M24) completed:**
  - First draft of integrated APIs
  - Integration and documentation of Messaging, Pub/Sub, Data Handling and Transformation, Drivers and Enablers
  - Evolution depending on the asset development

**WG DA-05 Status**

- The WG-05 already performed one teleconf meeting
  - Participants
    - Carlos Coutinho, Vítor Viana
    - Usman Wajid
    - Chris Decubber
    - Carlos Agostinho, Ricardo Gonçalves
    - Raquel Melo
  - Contacts have already been going (May/June) to have a follow-up meeting
    - Chris and perhaps Sergio Gusmeroli will provide additional partners working on SDKs, even if not part of EFFRA
    - Gash Bhullar showed interest in participating
    - It was decided to postpone until after the summer holidays
    - The meeting will take place in the following weeks
WG DA-05 Status

- Topics
  - Scope and known projects that could contribute
    - vf-OS
    - DigiCor
    - EFFRA: Connected Factories
    - FIWARE – SDK
    - Fitman (FIWARE)
  - State of the art on SDKs
    - Generic enabler to access all functionality of a context
    - How to make different APIs available?
    - Flutter, Ionic and similar multi-platform development SDKs
    - Create a cross-platform environment
    - Interoperability and open-access documentation and sources
  - Possibility for developing a reference implementation
  - Possibility to publish research paper

WG Scope

- Objective:
  - To propose and define an environment Software Development Kit (SDK) for developing applications to the manufacturing area
- To define the SDK standards:
  - Current vision of the SDKs
  - Future SDK vision
  - Projects involved
  - New vision
    - Multi-platform
    - Raise the SDK level to Model?
  - Associated tools
    - YEOMAN: Templates of SDK, version and dependencies
    - SWAGGER: Documentation
    - NPM: Version Control system and manage dependencies
**Objectives:**
- Create new standards
- Create a reference implementation
- Define a common approach on all FoF projects
  - A problem is the time frame for the vf-OS project
  - Will it be possible to develop anything at all?
  - Clarify with Chris the target of the EFFRA WGs
  - Is it just to define a roadmap for the future?
  - Or a common approach for future projects?
  - Definition of frameworks that gather projects
  - Define a Meta-SDK that would have access to other SDKs resources

**Actions:**
- Propose and Define SDK
  - This needs to be manageable and credible considering the project timeframes
- Data Model
  - Define generic use-cases and requirements (interoperability?)
- API
- Exchange Formats
- Define a standard set of tools for creating new SDKs
- Define a common/reference platform for SDKs
- Is this applicable only to manufacturing?
  - How can we include other parties outside EFFRA?
  - Added-value goes a lot beyond manufacturing projects
vf-OAK Software Development Kit (SDK)

Carlos Coutinho
CEO

www.vf-os.eu
Objective:
- To be a manufacturing operating system where applications can be developed easily by generic developers while reducing the time-cycle proving an ecosystem through a store.
How to develop vApps benefiting from existing manufacturing devices and software applications with easiness
Using drivers and API connectors

Manufacturing and logistic companies

vf-OS IO component for driver developers

- Shared abstraction layers configured using the toolkit

IO Components
- Security
- Top Interfaces
- Logic
- Bottom Interfaces

vf-OS API
Agnostic logic

Top interfaces
- REST
- Messaging Pub/Sub
- Registration
- Logging

Logic
- Computing
- Access Control
- Internal storage
- Lifecycle
vf-OS IO component for driver developers

Other vf-OS Components
- IO Components
- Security
- Top Interfaces
- Logic
- Bottom Interfaces

readSensorData: async function,
subscribe: async function,
sendCommand: async function,
releaseConnection: async function,
getDriverMetadata: function

A case study: Failure Manager

Business need:
- Create vApp capable of triggering alarms when a Tabber Stringer gets an error propagating it to two different partners to improve the maintenance service
Conclusions

- vf-IO component is a framework to make easy to code or bring existing drivers to vf-OS
- Provides a code generator for vf-OS drivers and API connectors developers
- Allows the pre-configuration of drivers before deployment
- Provides a Driver and API deployment and configuration tools (web)
Thanks you
vanaya@cigip.upv.es

Victor Anaya, Nejib Moalla, Ludo Stellingwerff, José Luis Flores and Francisco Fraile
Planning in a distributed supply chain

2018-09-26 IEEE-IS

Xavier Rakotomamonjy
AIRBUS

Challenges & needs

Supply Chain Criticality

- Production ramp-up at Airbus need huge capacity increase at supplier side too
- Coordination is transferred from Airbus to Risk Sharing Partners (RSP)
- 1st Tier suppliers list is restricted to a small number of large RSP capable to fulfil the requirements
- Suppliers in aerospace sector have to fulfil complex requirements along the project life cycle
Offer preparation & collaboration phase

Step 1: OEM set a new call to product with the associated dimension (production volume, quality, time constraints...)

Step 2: Several SME responds to the call by creating a join offer

Step 3: OEM make a trade-off between different offers by assessing several properties of the tender (consistency, availability, history...)

Step 4: Start collaboration

Planning functions

WHAT
Collaborate with multi-partners
Handle project performance
Manage task complexity

HOW
Forecast project activities and deliverables

WHEN

PROJECT PLANNING
CPM
PERT

PRODUCTION SCHEDULING
Planning Front End

1. Decision for consortium
2. Create a general offer
3. Finalize consortium offer
4. Sign purchase contract
5. Setup Collaboration
6. Monitor ASR

Decentralized Agile Coordination Across Supply Networks
vf-OS Enablers Framework

Demo

2018-09-26 – Madeira, Portugal

Joao Sarraipa and Joao Giao
UNINOVA

Enabler Framework Objectives

- Provides functionalities to access the services exposed by enablers
- Code and policies to facilitate the creation of enablers
- Access, monitoring and lookup services
- Interface to Enablers consumption
Enablers Framework Architecture

- **Request Handler**
  - Service Proxy
  - Access Services for Enablers and vApps
  - Performance Measurement

- **Enabler Registry**
  - CRUD functionality
  - Monitoring & Lookup Services

- **User Interface:**
  - Enablers Framework Configuration;
  - Enabler Registry

Enablers Framework Added Value

- **Software reusability**
  - Generic and Manufacturing Enablers (FIWARE and FITMAN Projects)
  - vF-OS Specific Enablers

- **Uniformization**
  - Docker based approach to install and use instances of Enablers

- **Integration**
  - accessing enablers services through a common API independently its protocol
  - NGSI1/2; REST

- **Accountability**
  - Technical performance and log registration features
Enablers Framework - Demo steps

1. Show the registration of an enabler using REST protocol (Notification Enabler)
2. Registration of an enabler using NGSI protocol (ORION)
3. Registration of an enabler that is running in a different server (FITMAN – 3DSCAN)

FIWARE Enablers:
1. Orion Enabler Services @ Enablers Framework
2. Orion Test of a Service through Enablers Framework
Enablers Framework Metrics

<table>
<thead>
<tr>
<th>SN</th>
<th>vApp</th>
<th>Framework ExecTime</th>
<th>Enabler ExecTime</th>
<th>Status</th>
<th>Error Message</th>
<th>Request Size</th>
<th>Response Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>378905/F10547 36782</td>
<td>7737</td>
<td></td>
<td>Success</td>
<td></td>
<td>514</td>
<td>272</td>
</tr>
<tr>
<td>2</td>
<td>378905/F10547 38257</td>
<td>8177</td>
<td></td>
<td>Success</td>
<td></td>
<td>516</td>
<td>274</td>
</tr>
<tr>
<td>3</td>
<td>378905/F10547 30948</td>
<td>0</td>
<td>Error</td>
<td></td>
<td>JSON data does not correspond to Schema</td>
<td>516</td>
<td>504</td>
</tr>
<tr>
<td>4</td>
<td>378905/F10547 31154</td>
<td>0</td>
<td>Error</td>
<td></td>
<td>JSON data does not correspond to Schema</td>
<td>516</td>
<td>504</td>
</tr>
<tr>
<td>5</td>
<td>378905/F10547 56676</td>
<td>0</td>
<td>Error</td>
<td></td>
<td>JSON data does not correspond to Schema</td>
<td>516</td>
<td>504</td>
</tr>
<tr>
<td>6</td>
<td>378905/F10547 19033</td>
<td>5838</td>
<td></td>
<td>Success</td>
<td></td>
<td>516</td>
<td>274</td>
</tr>
<tr>
<td>7</td>
<td>378905/F10547 30504</td>
<td>7963</td>
<td></td>
<td>Success</td>
<td></td>
<td>514</td>
<td>272</td>
</tr>
<tr>
<td>8</td>
<td>378905/F10547 98687</td>
<td>73137</td>
<td></td>
<td>Success</td>
<td></td>
<td>512</td>
<td>268</td>
</tr>
</tbody>
</table>

Enablers Framework Integration

Integration with the Process Designer:

- Generate NodeJS code to develop a vApp using enablers services schema provided by Enablers Framework
Enablers Dockerisation and Versioning

- Added Value
  - To facilitate control in configuration and registration of versions
  - To facilitate deployment for private use and its management
FIWARE for Industry and OPC UA: an industrial use case

Claudio Palasciano
Politecnico di Milano

Madeira, 26 September 2018

Who I am

BE in CPPS project: Business Experiments in Cyber Physical Production Systems
PLANTCockpit project: Production Logistics and Sustainability Cockpit
EMC2-Factory project: The Eco-Factory: cleaner and resource-efficient production in manufacturing
Sust.MFG/C: Smart Fabbrica Intelligente: Smart monitoring and planning, virtual production systems
Intellimech / Kilometro Rosso project: eco-factory per l'Industria del futuro, 2014
IECON 2013 / APMIS 2014 conferences special sessions: Emerging solutions for eco-factories
Industrial Technology 2014 Conference on nanotechnologies, biotechnology, advanced materials and new production technologies

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 856803.
CPS-ized machine line

Machine i

Information flow

Actuator  Sensors  Processing unit  Memory

CPS-ized line control

State of the Art

- **Trend**: reduce the idle time of machines in favor of stand-by states
- **Main focus**: single machine
  - Statistical models (inter-arrival times, processing time)
  - Local information
  - Production line quite unexplored

CPS-ized production line

- **Focus**: exploit potential of ICT enablers (information processing and M2M communication)
- **How**:
  - Real time information (digital twin)
  - Shared information among machines
  - Each machine decides in real-time their specific energy behavior based on the digital twins of all machines.
Interoperable OPC UA standard

OPC UA (*): standard architecture at application layer for connection of an automation cell with the factory level, integrating machinery and workstations.

OPC UA CHARACTERISTICS
• Flexible client-server approach
• Interoperable communication at industrial automation level
• Object-oriented ‘address space’ modeling (data model + methods)
• Semantically enhanced variables: struct type may include identifier, name, description

(*) https://opcfoundation.org/about/opc-technologies/opc-ua/}

Energy saving serial line control

Machine i-2    Machine i-1    Machine i

Actuator    Sensors    Processing unit    Memory
Energy breakdown

Threshold Time TT

Available Time

Heuristic Energy Control Policy

Time available \( t^* \geq TT \) (threshold time)
Limit of proposed policy

Machine digital twin

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description and unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i = 1..N$</td>
<td>machine identifier</td>
</tr>
<tr>
<td>bufferObject$[i]$</td>
<td>part/job identifier</td>
</tr>
<tr>
<td>PT$[i]$</td>
<td>processing time of machine $i$ (exogenous)</td>
</tr>
<tr>
<td>TT$[i]$</td>
<td>time threshold for activating stand-by in $i$ (exogenous)</td>
</tr>
<tr>
<td>cjObject$[i]$</td>
<td>identifier of current job of machine $i$</td>
</tr>
<tr>
<td>cjExpectedExitTime$[i]$</td>
<td>expected exit time from $i$ of job cjObject$[i]$</td>
</tr>
</tbody>
</table>
Industry 4.0 Lab in POLIMI

7 machine modules and 1 robot (8 stations)

Product assembly mock-up

The main task is to produce simplified smartphone or remote control

Housing → Plating → Fuses → Cover
Scenario: energy aware control

Autonomous energy aware control

- Line stations share digital twins
- Drill station automatically defines its status vs.:
  - Its Buffer status
  - Magazine station State + Time Left for Operation of previous station

Architecture
OPC UA Agent

IoT Manufacturing data
OPC UA SERVER PUBLISHER

OPC UA Agent
CONTEXT PRODUCER
OPC UA CLIENT SUBSCRIBER

ORION Context Broker

Energy management application

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No 68003.
Thank you very much

OPC UA Agent

- The OPC UA Agent enables capturing data from OPC UA devices on the shopfloor and provides them to the upper levels of a FIWARE-based system. Therefore, the main focus of this component is on the communication from field devices implementing an OPC UA server to FIWARE, allowing the communication to the FIWARE Orion Context Broker.
- The agent for the southbound side registers itself to monitor the OPC UA variables (as OPC UA client), whereas for the northbound side it is a context producer in order to send context data to Orion Context Broker.
ORION Context Broker

- Orion Context Broker allows to manage context information: persists last observation about context data, handles subscriptions in order to notify context consumers about changes.

- OPC UA Agent represents the Context Provider

- Cygnus represents the Context Consumer
Cygnus is a connector, based on Apache Flume, in charge of persisting certain sources of data in certain configured third-party storages, creating a historical view of such data.

Cygnus gets notifications from OCB (previously subscribed) and with a custom sink writes context data on Postgres db using custom tables (SensorsLog and OperationLog)
### Persistence repository (TBD)

**Machine def**

- **Resource**
  - ResoResourceID: integer
  - ResoResourceName: string
  - ResoResourceOpt: string
  - IP: string
  - WebPage: string

- **PowerLog**
  - timeStamp: integer
  - ResourceID: integer
  - ActivePower: float
  - Flow: float
  - Pressure: float

**Machine state log**

- **SensorsLog**
  - ResourceID: integer
  - sensor: integer
  - StationEntry: boolean
  - ReadyAtStationB: boolean
  - DoneWorking at: boolean
  - ExitStationB: boolean

**ResourceOperation**

- **OperationNo**: integer
  - WorkingTime: integer
  - TargetPower: float
  - TargetFlow: float
  - TargetPressure: float
  - Operation: integer

**WorkPlanDef**

- **WorkPlanNo**: integer
  - WIPlanDescription: ...

### OPC UA variable mapping

<table>
<thead>
<tr>
<th>Source</th>
<th>Transform</th>
<th>OPC UA identifier</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power</td>
<td>Power</td>
<td>Node[0].Power.v</td>
<td>2022-01-01 00:00:00</td>
</tr>
<tr>
<td>Status</td>
<td>Status</td>
<td>Node[0].Status.v</td>
<td>2022-01-01 00:00:00</td>
</tr>
</tbody>
</table>

**N.B.:** columns Namespace and OPC UA identifier are shown as NodeID in UAExpert
Current data collection in Access DB

Monitoring minimum frequency = 1 sec

- **FestoDB** contains MES data tables on production orders and machines
- **FestoEnergyDB** contains data tables on energy consumption (tblPower#) and compressed air consumption (tblSensors#) where # (1..86400) is the update frequency in seconds

Possible functionalities

Real-time and off-line stats visualization

- Resource monitoring about processing state of the line, energy compressed air consumption (actual/target), energy consumption per order/cycle/product
- Statistics about orders and machines (lead time, throughput, machine utilization, bottlenecks...)
- Operational conditions and warnings (machine error states energy consumption peaks, substained actual/target delta, etc) for maintenance support
Decentralized Agile Coordination Across Supply Networks

• The use of FIWARE Enablers in SMECluster Portal
  • 26th – 27th September 2018, Madeira
    • Gash Bhullar, Control 2K Limited

Introduction

Overview of Topics

- SMECluster Platform History and Overview
- SMECluster - Community Platform with Tool Delivery and Orchestration framework
- FIWARE Connectivity options
- Concluding remarks
Project Contributing to the development of www.smecluster.com

SYNERGY – Supporting Highly Adaptive Network
Enterprise Collaboration Through semantically enabled knowledge services
https://cordis.europa.eu/project/rcn/85320_en.html

CADIC – Cross-Organisational Assessment and Development of Intellectual Capital. Focus on providing an infrastructure open and extensible innovation for B2B environment, where Intellectual Capital (IC) can flow to the mutual benefit of all concerned
http://www.cadic-europe.org

CREMA – Provides an Everything as a Service model
http://www.crema-project.eu/

DIGICOR – The DIGICOR project develops novel collaboration concepts and implements an integrated platform that significantly reduces the burden to map up production networks and collaboration between SMEs
http://www.digicor-project.eu/

HEADING TOWARDS THE KNOWLEDGE ECONOMY - ROADMAP

DIGICOR - Decentralised Agile Coordination Across Supply Chains

2018_Sep ©Sematronix Limited Joint FoF and FiWARE Workshop at IS2018 in Madeira 25th Sept 2018 www.smecluster.com
BUSINESS ACTIVITIES

SMECluster

PRODUCTS AND SERVICES

LOCAL / NATIONAL OPERATIONS

EUROPEAN OPERATIONS

EU PROJECTS

Industreweb provides the "Glue" for Industry 4.0
http://www.industreweb.com

Provides IT Services

Control 2K LIMITED
Technology and Services Providers

Member

EFFRA
The European Factories of the Future Research Association (EFFRA) is a non-for-profit, industry-driven association
https://www.effra.eu

Technology Expertise
http://www.control2k.co.uk

SMECluster
Business Platform
http://www.smecluster.com

SEMATIONIX LIMITED
Cluster / Strategy Managers

Business Strategy
- To convert knowledge obtained from EU Projects into business products and services that can be offered commercially to create a sustainable future for smaller and medium sized companies
- To provide collaboration and business opportunities for members of the Cluster

Welsh Cluster

Erlent is the innovation Hub for Wales for Industry 4.0 Development
www.dmce.co.uk

Access control to SMECluster via member websites

WELSH AUTOMOTIVE FORUM
the voice of the Automotive Industry in Wales

B2B
One of the biggest challenges faced by buyer companies and organisations is the difficulties that exist today to match the capability of the members organisations to the tender and business opportunities available Locally, Nationally and Globally.

In Wales for example, sites such as Site2wire, Contracts4Wales and TEO provide these services. These sites are local to Wales but with the modern knowledge economy emerging, further opportunities exist on a global scale. These sites only list the opportunities. These need to be located and matched to the skill sets of the cluster.

New opportunities for collaborative working will be available soon, including Tender Opportunities so please email us to find out more: deborah.stewart@welshautomotiveforum.co.uk

Business Opportunities
Login

Email:
wm.williams@welshautomotiveforum.co.uk

Password:
******

Login
Community Benefits / Pitfalls of Clustering

Benefits
- Access to more opportunities
- More flexibility to win business
- More effective use of resources
- Knowledge sharing
- Build up business to strengthen stand alone
- Easier to find / develop supply-chain ‘hot spot’
- Greater networking / prospects pipelines

Pitfalls
- Commitment / Time
- Fear of structure – Understanding / lack of explanation
- Fear of subcontracting – Contractual Interaction
- Losing status / pecking order
- Ownership and liability not understood
- Lack of preparation to work collaboratively

Master Vendor with Subcontractors
Most common true definition of collaborative tendering
Used by the Public Sector
- Easiest structure for engagement for SME’s to be successful
- One contractual engagement with buying organisation
- Fear of ‘subby bashing’ (Sub-contractor)
- Contractual structure is poorly understood / carried out

Survey for SME in the DIGICOR-Project

What are the main activities to acquire new customers?

- Miscellaneous: 0%
- Shopping portals, electronic market places: 20%
- Internet research: 50%
- Mailing: 20%
- Personal discourse of potential customers at business meetings: 40%
- Network and cluster activities: 50%
- Presentations: 40%
- Exhibitions: 40%

Hanse-Aerospace Wirtschaftsdienst GmbH
E-Mail: hanseavd@hanse-aerospace.net
Internet: www.hanse-aerospace.net
General Concept of Tendering from SYNERGY project – the beginnings of apps as services

SMECluster - Community Platform with Tool Delivery and Orchestration framework
FITMAN
– THE FIWARE PROJECT FOR MANUFACTURING

Trial Focus

Virtual Factories

8

Business Opportunity & SME Network Matching
FITMAN Platform in TANET Trial

• Call for Tender Opportunities (BS.1)

Represent assets of a company to support facilitators in creating clusters to complete tenders.

GE / SE’s Used in Smart and Virtual

FIWARE for Industry Open Source Components tested and Used in ALL TANet Trials:

- Data Collection – SDC
- IoT Protocols – ZPA
- IoT Gateway Handling – CEP
- Zigbee Data – ZPA
- ... Many others Tested

Specific Enablers:
- Coll. Assets Management – CAM
- Supply Chain and Bus. Ecosys. Apps – SCApp
- Data Interop. Platform Services - DIPS
- Management of Virtual Assets – MoVa
- Generation & Trans. of Virt. Assets – GeToVA
New SE’s used

• AK 3 (MoVA) - Using as a complement/possible replacement for CAM

• AK 24 (GeToVA) - Use to automate the import of suppliers into SMECluster
Next time the Member logs on they are no longer prompted for their LinkedIn URL as the data is now stored against their profile.

Suppliers can be searched using the box provided.

Supplier details and Domain Keywords imported from MoVA.
Opportunities are retrieved from MoVA. GeToVo can import these Opportunities from Sell2Wales and other sources.

The Cluster Search in MoVA retrieves all suppliers that match the keywords on the Opportunity Requirements.

Section 1: This shows all the suppliers that can fulfill the requirements solely.
MoVA – RESTful-Interfaces

Data Maintenance

MoVA

Data Retrieval

Data Compilation

Tender Platform

Query: Find Cluster

New opportunity

New Requirements

SMECluster Portal

Manage Suppliers

Manage Domain Keywords

SMECluster featured Dashboard implementation
for EU FLEXINET Project

FoodMart Sales Statistical Report
SMECluster Additional Tool Delivery and Orchestration framework for other providers – MindSphere

- With MindConnect MindSphere provides multiple, varied and easy-to-implement connectivity solutions to onboard a wide range of assets (Siemens and 3rd-party) in both brown- and greenfield:
- Focusing on brownfield installations, dedicated IoT Connectors like MindConnect Nano and MindConnect IoT2040, which provide plug and play connection of Siemens and 3rd party products thanks to communication via open standards like OPC UA
- MindConnect embedded agents in Siemens industrial products like SIMATIC and SINUMERIK focusing on greenfield projects
- For every further product, including 3rd-party, MindConnect LIB, which enables self-programming of connectivity modules by interacting with the MindSphere southbound APIs

Example APIs @ MindSphere

- **Timeseries API**
  - Use Timeseries APIs to read, write and edit Timeseries values from storage
  - Aggregated Timeseries (M4)
  - Min/Max Values
  - Modify Timeseries

- **Trend Prediction API**
  - Use Trend Prediction APIs to create predictions based on one aspect

- **Anomaly Detection API**
  - Use Anomaly API to detect correlation analysis of two aspects

- **Outlier Detection API**
  - Use Outlier Detection API to detect failures through e.g. Gradient Check

Enable not only data visualization but also data analytics: Implement application for data visualization and analyze data for dependencies, like energy consumption and production information
FML Technology Enabler plan

In terms of the Virtual trial, the previous tables give an idea of how the SMECluster Platform will be expanded:

- Making platform and service maintenance faster, and more cost effective, and more reliable, since less custom code is used;
- Making the service offering more flexible and adaptable to technology developments, since evolving technology embedded in GE and SE instances will provide access seamlessly.
- Continue adding new features to support development of “ Middleware” for manufacturing data collection services.
- Provide a reselling service for the FITMAN / FIWARE Enablers
DIGICOR Use case interconnectivity road map

Phase 1
- None

Phase 2
- Single sign on
- Accessing Digicor services from outside Digicor platform

Phase 3
- Full integration of all three use cases

Development of Tools serving needs of all use cases should be promoted

Same model of Value Network as reseller of services
Presentation of the FIWARE-IDS Enabler at IS2018
Data-Driven Digital Manufacturing Workshop

Ulrich Ahle
CEO FIWARE Foundation
Member of the IDS Board

Connecting different platforms which could include the DIGICOR Tool store or SMECluster Platform

Zero defect manufacturing use case: Scope

The Zero Defect Manufacturing use case demonstrates how factories can benefit from IDS reference architecture and FIWARE open-source technology by obtaining enhanced functionalities for monitoring context data exported from the factory, enabling smarter decision-making.
Zero defect manufacturing use case: Smart decision workflow powered by FIWARE-IDS

Predictive Maintenance System Milling Machine

IDS CONNECTOR

Data App

Context Broker

Big Data Analysis

IDS CONNECTOR

Context Broker

+GF+

System Adapters

FACTORY

Milling Machine Behaviour
Normal (100%)

IDS APP STORE

Data Apps

System Adapters

Zero defect manufacturing use case: Smart decision workflow powered by FIWARE-IDS

Predictive Maintenance System Milling Machine

Mean Time To Failure: 639 days

Quality Control Cloud

Out of tolerance for last 10+ parts

IDS CONNECTOR

Context Broker

Big Data Analysis

+GF+ System Adapters

FACTORY

2018_Sep ©Sematronix Limited Joint FoF and FIWARE Workshop at IS2018 in Madeira 29th Sept 2018 www.smecluster.com

Factory Node Data Model – Can be adapted for FIWARE Enablers

- Mapping of Data Tree in IW to OPC UA Data Model

DIGICOR Platform could incorporate FIWARE enablers
“Thank You”

Gurbaksh Bhullar
www.smecluster.com

Sematronix Limited
Control 2K Limited
Technology Application Network Limited
Waterton Technology Centre, Bridgend. South Wales. CF31 3WT. UK.
(Simulation of) Business models for a multi-sided platform

Dr. Cesar Marin, Senior Computer Scientist
Information Catalyst (UK)

Contents

• Introduction

• Business models

• Simulation

• Concluding remarks
Introduction

• What is DIGICOR about?
Technology platform, tools & services that allow the creation and operation of collaborative networks across the value chain

Strategic Collaborations
DIGICOR collaboration platform provides vital communication and decision support services to companies, particularly SMEs, allowing them to plan and control collaborative production and supply networks.

Marketplace and Toolstore
DIGICOR marketplace provides matchmaking functionalities, allowing SMEs to find and interact with suitable partners. The primary function of the Marketplace is to facilitate the contracting of business with potential suppliers.

Governance Framework
DIGICOR governance framework and procedures sets out the terms of collaboration and rules of doing business on the DIGICOR platform. The framework allows SMEs to collaborate and exchange information in a trustworthy environment.

• Some achievements include

[DIGICOR screenshots showing user interfaces and applications]
Introduction

- How do we exploit these results?
  - DIGICOR as a whole?
  - Each component/tool exploited individually?

  - Business models were created with *some* insight information
  - ... and then what?

- Simulation of DIGICOR in a dynamic market environment

Business models

- DIGICOR business models (so far)

<table>
<thead>
<tr>
<th>Platform Type</th>
<th>Costs for Access</th>
<th>Use of Tools</th>
<th>Duration</th>
<th>Connections</th>
<th>Participation</th>
<th>Access to Standardized Documents</th>
<th>Operator Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Closed (only with registration)</td>
<td>Fully paid</td>
<td>All tools are free</td>
<td>Unlimited</td>
<td>Unlimited</td>
<td>Unlimited</td>
<td>Unlimited</td>
<td>Infrastructure as a Service (IaaS)</td>
</tr>
<tr>
<td>Partly open (only some parts with registration)</td>
<td>Party paid</td>
<td>Only a couple of tools are free</td>
<td>Limited to 1 year</td>
<td>Contingent with fixed number</td>
<td>Contingent with fixed number</td>
<td>Permanent access for some documents</td>
<td>Platform as a Service (PaaS)</td>
</tr>
<tr>
<td>Open (without registration)</td>
<td>Fully free</td>
<td>Pay per use</td>
<td>Limited to 1 month</td>
<td>10</td>
<td>10</td>
<td>Pay per use</td>
<td>Software as a Service (SaaS)</td>
</tr>
</tbody>
</table>

Variants:
- Operation Concept 1: "Free limited access"
- Operation Concept 2: "Partly free access"
- Operation Concept 3: "Fully paid access"
- Operation Concept 4: "Pay-per-use access"
Business models

• DIGICOR business models (so far) + costing

Morphological Box for Operation Models of the DIGICOR Platform

<table>
<thead>
<tr>
<th>Platform type</th>
<th>Costs for access</th>
<th>Use of tools</th>
<th>Duration of access</th>
<th>Number of connections</th>
<th>Number of participation</th>
<th>Access to standardized</th>
<th>Operator model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Concept 1: Free limited access</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
<td>-25,000.00 €</td>
</tr>
<tr>
<td>Service Concept 2: Partly free access</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
<td>-6,000.00 €</td>
</tr>
<tr>
<td>Service Concept 3: Fully paid access</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
<td>-12,000.00 €</td>
</tr>
<tr>
<td>Service Concept 4: Pay-per-use access</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
<td>-48,000.00 €</td>
</tr>
</tbody>
</table>

Cost Calculation for Operation of the DIGICOR Platform (1st DRAFT)

- Personal Costs (HAW)
  - project leader: 25% of full time -25,000.00 €
  - students: -6,000.00 €
- Technical Development (CER, SGL & IT)
  - hosting DIGICOR platform: -12,000.00 €
  - development of functions and interfaces: -48,000.00 €
  - maintenance / updates / support: 5% of costs for techn. dev. -3,000.00 €
  - other technical costs: -1,200.00 €
  - software systems for apps and functions: -1,200.00 €
- Marketing and Travel
  - trade fairs: -6,000.00 €
  - travel costs: -6,000.00 €
  - marketing material: -2,400.00 €
  - Costs total: -107,800.00 €
- Operation Concept 1: "Free limited access"
  - Operation Concept 2: "Partly free access"
  - Operation Concept 3: "Fully paid access"
  - Operation Concept 4: "Pay-per-use access"
  - Profits / losses (€)

Simulation

• “All models are wrong…”
• “…but some are useful” – (George Box, 1978)
  • We are looking for a business model simulation that provides insight into the potential revenue of DIGICOR outcomes

Other works in the area:
• (Ahrweiler, 2017) – Agent-based simulation for science, technology, and innovation policy
• (Evans et al., 2017) – Business model innovation for sustainability: towards a unified perspective for creation of sustainable business models
• (Lieder et al., 2017) – Towards circular economy implementation: an agent-based simulation approach for business model changes
Simulation

• For our simulations we consider the following:
  • Factories need supplies
  • SMEs look for factories interested in their products

  • Factories decide to try DIGICOR platform with a given probability
  • SMEs enter DIGICOR with a given probability IF the factory they are interested in is ALREADY using DIGICOR

  • Once inside DIGICOR they all behave as part of a natural ecosystem

Simulation

• For our simulations we consider the following:
  • Inside DIGICOR, factories open calls for tender
  • SMEs bid to participate in tenders

  • Factories have an expected quality of their suppliers
  • SMEs supply products with a quality
Simulation

For our simulations we consider the following:

- It can take days to complete a tender and days to find one
- When a tender is completed, a contract is formed and manufacturing starts

- Contracts increase happiness, no contracts decrease happiness
- Low happiness make a factory or SME leave DIGICOR which reduces their probability of re-entry

Simulation

For our simulations we consider the following:

- DIGICOR platform has an operation cost
- Membership type has a fee
Concluding remarks

What is left to be done
- Current simulation considers calls for tenders but other tools are also being developed in DIGICOR
- Tools to include in the next version: risk analysis, planning tool, manufacturing monitoring, …
- How will these tool impact the collaboration process?

Concluding remarks

What will we do with the simulation?
- Produce and analyse simulations of business models for each use case
- Produce and analyse simulation of DIGICOR as a whole and for individual tools
- Final output for exploitation and as scientific publication
Decentralized Agile Coordination Across Supply Networks

(Simulation of) Business models for a multi-sided platform

Thank you! Any questions?

Dr. Cesar Marin, Senior Computer Scientist
cesar.marin@informationcatalyst.com
Information Catalyst (UK)
Business Model Simulation for a Smart Factory Platform

Joint FoF and FIWARE Workshop
9th International Conference on Intelligent Systems 2018

Raul Polo
Research Centre on Production Management and Engineering
Universitat Politècnica de València

virtual factory Operating System – www.vf-OS.eu
The vf-OS Marketplace

Service Providers

Manufacturing & Logistics Providers

Manufacturing & Logistics Users

Software Developers

vf-F

vf-mApp

vf-OAK

vf-OM

vf-API

vf-MW

vf-SK

vf-SW

Connect Systems

Run-time Use

Build Apps
The Business Model

Transactional Graphs

vApps

Drivers

Services (Training)

Services (Support)

Services (Hosting & Computation)
Business Model Simulation
Business Model Simulation

Pricing model

<table>
<thead>
<tr>
<th>appComplexity</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>appComplexityF</td>
<td>0.17</td>
<td>0.64</td>
<td>1.41</td>
<td>2.47</td>
<td>3.82</td>
<td>5.46</td>
<td>7.35</td>
<td>9.53</td>
<td>11.99</td>
<td>14.71</td>
</tr>
<tr>
<td>appUniquePrice (min)</td>
<td>120.00</td>
<td>462.34</td>
<td>1,017.69</td>
<td>1,781.30</td>
<td>2,749.88</td>
<td>3,920.97</td>
<td>5,292.56</td>
<td>6,862.99</td>
<td>8,630.81</td>
<td>10,594.77</td>
</tr>
<tr>
<td>appInitialPrice (min)</td>
<td>2.00</td>
<td>7.71</td>
<td>16.96</td>
<td>29.69</td>
<td>45.83</td>
<td>65.35</td>
<td>88.21</td>
<td>114.38</td>
<td>143.85</td>
<td>176.58</td>
</tr>
<tr>
<td>appDailyPrice (min)</td>
<td>0.20</td>
<td>0.77</td>
<td>1.70</td>
<td>2.97</td>
<td>4.58</td>
<td>6.53</td>
<td>8.62</td>
<td>11.44</td>
<td>14.38</td>
<td>17.66</td>
</tr>
<tr>
<td>appUniquePrice (max)</td>
<td>25.00</td>
<td>963.20</td>
<td>2,120.19</td>
<td>3,711.03</td>
<td>5,726.92</td>
<td>8,164.69</td>
<td>11,026.17</td>
<td>14,297.89</td>
<td>17,980.85</td>
<td>22,072.43</td>
</tr>
<tr>
<td>appInitialPrice (max)</td>
<td>4.17</td>
<td>16.05</td>
<td>35.34</td>
<td>61.85</td>
<td>95.48</td>
<td>136.14</td>
<td>183.77</td>
<td>238.30</td>
<td>299.68</td>
<td>367.87</td>
</tr>
<tr>
<td>appDailyPrice (max)</td>
<td>0.42</td>
<td>1.61</td>
<td>3.53</td>
<td>6.19</td>
<td>9.55</td>
<td>13.61</td>
<td>18.38</td>
<td>23.83</td>
<td>29.97</td>
<td>36.79</td>
</tr>
</tbody>
</table>

Parameters

| priceComplexity (min) | 15 |
| priceComplexity (max) | 25 |
| priceLevel (min) | 0.8 |
| priceLevel (max) | 1.2 |
| multiUniquePrice | 60 |
| divDailyPrice | 10 |

Formulas

- appUniquePrice = appComplexityF * priceComplexity * priceLevel * multiUniquePrice
- appInitialPrice = appComplexityF * priceComplexity * priceLevel
- appDailyPrice = appComplexityF * priceComplexity * priceLevel / divDailyPrice

M&L Users modelling

- New M&L Users interested in buying vApps, Drivers and Services: $M_k$: daily investment in marketing

\[ P(U) = \frac{\ln(M_k)}{\alpha} \times \beta \]

- Selection of vApps among the offered in the marketplace:

\[ \text{Min}\{wUp} * U_{ip} + wlp * I_{ip} + wDp * D_{ip} + wCl * (3 - Cl_j)\}

- Similar procedure for Drivers and Services selection, with:

- $P(nD)$: probability vApp needs Driver
- $P(nS)$: probability vApp needs Service

www.vf-os.eu

Software Developers modelling

- New Software Developers interested in developing vApps:
  \[ P(\text{SD}) = \begin{cases} 
  \frac{P_0(\text{SD})}{n\text{Users}} & \frac{n\text{Users}}{n\text{SofDev}} \geq n\text{UpSD} \\
  0, & \frac{n\text{Users}}{n\text{SofDev}} < n\text{UpSD} 
  \end{cases} \]
  
- Increase of Certification Level: \( P(iCL) = \text{triangular}(0, 0.05, 0.1) \)
- Probability Unique Price: \( P(uPr) = \text{triangular}(0, 0.1, 0.2) \)
- Probability Initial Price: \( P(iPr) = \text{triangular}(0, 0.5, 1) \)
- Price Complexity: \( p\text{Comp} = \text{triangular}(p\text{Cmin}, p\text{Cmed}, p\text{CMax}) \)
- Price Level: \( p\text{Lev} = \text{triangular}(0.8, 1, 1.2) \)

Scenario value (\( \varepsilon \)) impact

- Probability of new M&L Users: \( P(U) = \frac{\ln\left(\frac{M_k}{\alpha}\right)}{\beta} \ast \varepsilon \)
- Probability M&L Users buying vApps: \( P_0(pA) = P_0(pA) \ast \varepsilon \)
- Probability M&L Users buying Services: \( P(nS) = P_0(nS) \ast \varepsilon \)
- Probability Soft. Dev. buying Services: \( P(bS) = P_0(bS) \ast \varepsilon \)
- Probability M&L Prov. buying Services: \( P(bS) = P_0(bS) \ast \varepsilon \)

\( \varepsilon = \begin{cases} 
  0.5 & \text{if pessimistic} \\
  1.0 & \text{if expected} \\
  2.0 & \text{if optimistic} 
  \end{cases} \)

Default values, used for the presented analysis, can be changed in the simulation form.

Similar modelling for M&L Providers and Service Providers (with some differences)
Business Model Simulation

Total incomes (5 years):
- Apps: 108 ME
- Drivers: 47 ME
- Services: 22 ME

vf-OS daily profit: 8,031 €

Key indicators for vf-OS Limited

Total incomes for the parties

Break even point day: 580
Total investment: 227,488,360
ROI: 8.4%
Conclusions

Simulation conclusions (figures after 5 years):

- **vf-OS is affordable for Buyers:**
  - M&L Users pay 124 € per day (average) per 11 vApps, 4.8 vDrivers and 1.7 vServices (average)

- **vf-OS Marketplace is attractive for Sellers:**
  - vf-OS Limited profit is around 8,500 € per day
  - Software Developers revenues are around 930 € per day (average)
  - M&L Providers revenues are around 800 € per day (average)
  - Service Providers revenues are around 380 € per day (average)
    - Note that Software Developers, M&L Providers and Service Providers could not be focused in vf-OS

- **vf-OS Limited is viable:**
  - Initial investment is around 230 K€
  - Break-even-point is around 17 months
Business Model Simulation for a Smart Factory Platform

Joint FoF and FIWARE Workshop
9th international Conference on Intelligent Systems 2018

Raul Poler
Research Centre on Production Management and Engineering
Universitat Politècnica de València
L4MS Open Call: Opportunities for co-creation with OPIL

Angelo Marguglio (ENGINEERING)
Head of Smart Industry and Agrifood Unit

Did you know that only...

2%

... of European SME’s use advanced manufacturing technologies?
Outdated intra-factory logistics are troubling European SMEs

- Innovations and new technological applications require
  - Strategic thinking
  - Simulation & planning
  - Re-skilling
  - Access to finance
- Resources that most SMEs don’t have

Logistics account for up to 50% of total manufacturing costs

Logistics account almost half of total manufacturing cost of an item*

Shorter production cycles, individualization & customization require *flexibility*

What is L4MS?  
... and how to get involved?
L4MS is a "one stop shop" for manufacturing industry with access to advanced digital solutions, funding opportunities, training and state-of-the-art piloting environments to help SME's and Mid-Caps develop new, smarter intra-factory logistics solutions.

Our goal is to reduce the installation cost and time of mobile robots by a factor of 10.

L4MS Marketplace: One-stop-shop for logistics

- Competence centers
- Business developers
- Technology suppliers

- TECHNOLOGY
- FINANCE
- BUSINESS MENTORING
- TRAINING
**Application Experiments**

- Foster new innovative service and business models
- Low cost customized and standardized AGV systems
- Advanced technologies without need for in-house expertise
- Real evidence and data for the investors

**Open Platform for Innovations in Logistics (OPIL)**

**REAL FACTORY**

- Data from IoT nodes

**OpIL**

- Optimizing
- Planning and Testing
- Monitoring and Control

**DIGITAL FACTORY**
L4MS Network

- Industry association
- Competence center
- Infrastructure
- Satellite node
- Expertise

What L4MS offers

- Matchmaking with system integrators, mobile robots manufactures & SME's
- Funding up to €250,000
- State-of-the-art test environment within your region
- Business developers for innovative business and service models
- Technology expert for adopting latest logistics automation solutions
- Training to re-skill workers at your door step
- Finance for scaling up the new business and service models
Apply now!

Submit applications at l4ms.eu

Sep - 30th Nov 2018

L4MS network

H2020 Innovation Action - This project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement No: 767042
Thank you!

@L4MS_Eu

L4MS

http://l4ms.eu
Technology platforms and ecosystems, who drives the success? The case of the DAEDALUS platform

Andrea Barni – SUPSI
andrea.barni@supsi.ch

26 Sept. 2018, Madeira

The components of the DAEDALUS platform

• Platform products/services rich, developed on the top of IEC 61499
• Runtime + SW development supporting elements
• Reference Hardware & Software implementations
Marketplace: the access to platform value

DAEDALUS fosters the creation of an ecosystem that relies on the IEC 61499 where the stakeholders of the automation domain are able to exchange new forms of value.

Products: applications and mechatronic systems
Marketplace: the access to platform value

**Products:** applications and mechatronic systems

**Value:** tangible (€) and intangible (***)

**Services:** application validation, etc.
Is it enough to foster ecosystem growth?

From stakeholders to complementors

“A complementor is a company/organisation which directly sell a product or service (or services) that complements the product or service of another company/organisation, by adding value to mutual customers.”

Brandenburger and Nalebuff
Is it enough to foster ecosystem growth?

The evolution of the automation value network towards DAEDALUS is strictly dependant by two factors:

1. **technology acceptance**: ability to make evidence of the improved efficiency and effectiveness of the work processes by means of IEC61499 adoption;

2. **propension to change**: ability to gradually guide companies in analysing and identifying the specific benefits for their business.

---

Is it enough to foster ecosystem growth?

The evolution of the automation value network towards DAEDALUS is strictly dependant by two factors:

1. **technology acceptance**: ability to make evidence of the improved efficiency and effectiveness of the work processes by means of IEC61499 adoption;

2. **propension to change**: ability to gradually guide companies in analysing and identifying the specific benefits for their business.

3. **Partnership**: ability to make the ecosystem living by incorporating business partners able to support ecosystem growth.

*From App Store history*

*When IBM and SAP got involved, they began building apps specifically geared towards enterprise customers. Customers could access these apps from a vetted App Store, which also was appealing to IT. The Cisco deal gave IT faster on-boarding of Apple devices on networks running Cisco equipment (which most enterprises use).*
1- Fostering technology acceptance

A common effort, led by the DAEDALUS Competence Centre, passing by activities aiming at:

- gain visibility at European level;
- provide standard-based reliability;
- involve interested target groups;
- demonstrate technology effectiveness through reference use cases.

2- Encouraging propensity to change

1. Development of a structured process guiding companies in understanding the impact of shifting towards a platform based technology;

2. Development of the instruments required to support the analysis.
Development of **reference business models** contemplating the impact of integrating DAEDALUS technologies in complementors’ one, and **validation by consortium partners**.

Development of **instruments** supporting the **preparation of business plans** contemplating the adoption of DAEDALUS technologies. **Qualitative** (supported interview) + **quantitative approach** (spreadsheet).
Encouraging propensity to change: approach

Development of a set of reference business plan implementations (at least one per complementor) to support companies external to the consortium in understanding benefits & changes in their business.

Encouraging propensity to change: approach

Use of the developed instruments to analyse the specific impact of Daedalus technologies’ on external organisations and pave the way for the transition towards platform integration.
Thank you for the attention

Andrea Barni
University of Applied Science of Southern Switzerland (SUPSI)
andrea.barni@supsi.ch
Ecosystem for Collaborative Manufacturing Processes – Intra- and Interfactory Integration and Automation

Big Data Management
The use of Deep learning techniques in the COMPOSITION platform
Paolo Vergori
ISMB

Agenda

• Why deep learning for factory of the future?
• The COMPOSITION’s challenge
  • UC-BSL-2 Predictive maintenance
  • UC-ELDIA-1 – Contractual goods and recyclable materials management
• Deep Learning approach in Composition
  • intra-factory scenario
  • inter-factory scenario
• Future work

27/09/2018
COMPOSITION @IS2018
Why deep learning for factory of the future?

- Prolific environment
  - large amount of potential data available
  - tangible results achievable in finite time span
  - potential market exploitation
- Overcoming current state of the art
  - outperforming statistical models
  - continuous learning for adapting to dynamic production environment
  - transferrable results in terms of research (not in terms of models or dataset)
- Strong correlation to upcoming existing technologies
  - improving of matchmaking results, supporting decision systems

Deep Learning approach in Composition

- Deep learning toolkit
  - twofold application as both intra and inter-factory intelligent tool
  - intra-factory data analysis for predictive maintenance
  - inter-factory agent-based marketplace market estimation
- Continuous learning
  - triggered by inter-factory agents and learning framework
- Fits in a comprehensive ecosystem
The COMPOSITION’s challenge - UC-BSL-2 Predictive maintenance

BSL: predictive maintenance on production equipment
- 2 analyzed reflow ovens
- very complex dataset with ~625,000, ~350,000 samples
  - 60 sensors readings in 20 zones
  - 5 newly sensors linearly independent
- few reported failures (~<20) over recorded timeframe
- very demanding in terms of accuracy
  - tentative for ~2.5h of timeframe
  - more realistically ~5 minutes of timeframe

The COMPOSITION’s challenge - UC-ELDIA-1 – Contractual goods and recyclable materials management

ELDIA: agent-based marketplace estimation
- 4 analyzed raw materials markets
- very small dataset with few samples
  - recorded started in 2016 only
- less than 50 reported values
- very demanding in terms of prediction
- steady prices recorded before
- very little contextual data to the analyzed market
Deep Learning approach in Composition – intra-factory scenario

- Intra-factory data analysis for **predictive maintenance** from heterogeneous sources
  - data analysis of existing machinery, forming the **historical dataset**
  - newly deployed sensors for acquiring more information
- High risk challenge:
  - work with an **incomplete dataset** with no historical information on new measured parameters
  - have **unbalanced classes** in a nearly real-time environment
- Dichotomy of solution:
  - deploy **untrained ANNs** and rely on theoretical convergence forgetting the historical data
  - **balance the classes** and propagate new parameters backwards

27/09/2018

**DLT – intrafactory – historical dataset**

![Graphs showing data analysis results](graph.png)
DLT – intrafactory – outliers and cleaned data

27/09/2018

Composition @ IS2018

DLT – intrafactory – dataset creation

27/09/2018

Composition @ IS2018
DLT – intrafactory – accuracy

27/09/2018
COMPOSITION @IS2018

DLT – intrafactory – ROC function

27/09/2018
COMPOSITION @IS2018
DLT – intrafactory – zoomed training dataset

27/09/2018

COMPOSITION @IS2018

DLT – intrafactory – test dataset – trial 2

27/09/2018

COMPOSITION @IS2018
DLT – intrafactory – test dataset – trial 3

27/09/2018

COMPOSITION @IS2018

19

DLT – intrafactory – test dataset – trial 4

27/09/2018

COMPOSITION @IS2018

20
DLT – intrafactory – newly deployed acoustic sensors

27/09/2018

COMPOSITION @IS2018

DLT – intrafactory – zoomed training dataset with audio [2013-2017]

27/09/2018

COMPOSITION @IS2018
DLT – intrafactory – predictions on test dataset with audio, look ahead 1 sample – trial 2

27/09/2018

COMPOSITION @IS2018

DLT – intrafactory – predictions on test dataset with audio, look ahead 1 sample – trial 3

27/09/2018

COMPOSITION @IS2018
DLT – intrafactory – predictions on test dataset with audio, look ahead 1 sample – trial 3 zoomed

27/09/2018

COMPOSITION @IS2018

DLT – intrafactory – predictions on test dataset with audio, look ahead 1 sample – trial 4

27/09/2018

COMPOSITION @IS2018
DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples - trial 2

27/09/2018
COMPOSITION @IS2018

DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples – trial 3

27/09/2018
COMPOSITION @IS2018
DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples – trial 3 zoomed

27/09/2018
COMPOSITION @IS2018

DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples – trial 4

27/09/2018
COMPOSITION @IS2018
DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples weighted – trial 2

27/09/2018

COMPOSITION @IS2018

31

DLT – intrafactory – predictions on test dataset with audio, look ahead 16 samples weighted – trial 3

27/09/2018

COMPOSITION @IS2018

32
Deep Learning approach in Composition – inter-factory scenario (I)

- Provide intelligence to the agents that form the agent-based marketplace
  - marketplace parameters estimation
    - historical data
    - matchmaking policies
    - learned transaction
- Continuous learning
  - triggered by inter-factory agents and learning framework
Deep Learning approach in Composition – inter-factory scenario (II)

Deep Learning approach in Composition – intra-factory scenario (III)

27/09/2018
COMPOSITION @IS2018

35

36
Future work (I)

- Experiment novel ANNs topologies
- More datasets to be evaluated alongside contextual markets information
- Enhance connection with matchmaking for extending the available information to be evaluated, consolidating the accuracy and improving reliability
- Extend current ontologies for supporting a broader range of possibly market correlated information, coming from heterogeneous sources (e.g. factors that influence stock market)

Future work (II)

Other challenges might be faced in different use cases by the DLT, based on different requirements:

UC-KLE-1: polishing machine predictive maintenance
  - existing historical dataset with few recorded parameters
  - no reported failures

UC-KLE-4: agent-based marketplace intelligence
  - round trip time calculation and estimation by statistical models in simulation and forecasting tools
  - bin fill level estimation with new deployed sensors
Thanks for your attention

All rights reserved.
All copyright for this presentation are owned in full by the COMPOSITION Project.
Permission is granted to print material published in this presentation for personal use only. Its use for any other purpose, and in particular its commercial use or distribution, is strictly forbidden in the absence of prior written approval.
COMPOSITION has received funding from the European Union’s Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No 723145.
Possible inaccuracies of information are under the responsibility of the project. This presentation reflects solely the views of its authors. The European Commission is not liable for any use that may be made of the information contained therein.

Please see us here: www.composition-project.eu
CONTENT

Brief Overview – NIMBLE Platform & NIMBLE Core Services

(on which big data is possible, but not yet realised)

NIMBLE Approach:
- Minimalistic common data model / “products first, semantics in small doses”
- UBL-based Catalogue Registry
- eCl@ss annotated Product Catalogue
- Semantic Search based on an RDF representation of catalogue data

Technical (and use) challenges:
- Use of domain-specific Ontologies in conjunction with eCl@ss and UBL
- Ingesting Local catalogues and product configurators
- Microservice Architecture and distributed database design
Core Business Services

- Registration of users/companies (+ role management)
- Uploading and management of product catalogues (+ semantic annotations)
- Semantic Search based on UBL, eCI@ss, and domain ontologies
- Finding B2B partners and negotiating contracts that establish supply chains
- Execute business processes compliant with UBL (or design specific ones)
- Share production data via a platform-enabled data channel management system
- Use an open API to connect 3rd party services for use by participants
PRODUCT CATALOGUE & ONTOLOGY

Basic elements of the product catalogue domain:

- **business entity** – a legal party or a person in a supply chain
- **resource entity** – a product or a service that is held by the business entity, e.g. an item (continuant resource entity), or logistics service (occurrence resource entity)
- **dependent entity** – details used to describe either business entity or resource entity, e.g. business role

---

PRODUCT CATALOGUE SCHEMA IN NIMBLE
UBL-BASED CATALOGUE REGISTRY

- UBL business process model drives information representation in NIMBLE

- a royalty-free library of standard electronic XML business documents used in supply chains, e.g. for ordering, delivery, invoice

- Note: Semantics by (XML) document exchange (for better or worse ...)

UBL-BASED CATALOGUE REGISTRY

- UBL document schema for catalogue/product management, related to supply chain operations

- Note: has provisions for relating products across different identity management systems

- Also note semantic open-ended-ness (e.g. Keywords [0 .. N])
PRODUCT CATALOGUES WITH ECL@SS

CONNECTING UBL / DOM.ONTO / ECLASS
INTEGRATION OF ECLASS INTO UBL

Note that without eCl@ss, UBL would be semantically empty for catalogues

A GENERAL MODELLING ISSUE: SEPARATION OF CONCERNS

UBL describes many Business Processes

UBL says little about Products

Eclass gives us a vocabulary for distinguishing products and (logistics) services

Eclass says little about business processes

Semantic Models (aka „domain ontologies“) of specific fields try to do „a bit of both“

This leads to “semantic entropy” (i.e. ambiguities of all sorts) when you merge ontologies
CHALLENGE – PRODUCT CATALOGUES & CONFIGURATORS

Challenges – Product catalogues come in different forms:

a) as static web pages

b) as bespoke database entities

c) as product information system entities (e.g. PIMCore)

d) as semantically annotated entities (in every research project)

e) as ad-hoc / out-of-the-box / ontology aware configurators

WHAT CAN A B2B PLATFORM LIKE NIMBLE DO TO HELP SMEs WITH THIS?

CHALLENGE – PRODUCT CATALOGUES & CONFIGURATORS

Approach taken in NIMBLE:

a) as static web pages → sorry, do it again via single or bulk product data entry

b) as bespoke database entities → Catalog-2-DB API

c) as product information system entities (e.g. PIMCore) → very tricky dependencies

d) as semantically annotated entities (in every research project) → similar as c)

 e) as ad-hoc / out-of-the-box / ontology aware configurators → working on an API for configurators
CHALLENGE – DOMAIN-SPECIFIC ONTOLOGIES

Challenges:

1st Generation Web: from STEP to XML (2000)

2nd Generation Web: Semantic Web (Web Ontology Language - OWL) (2005)

3rd Generation Web: use standard taxonomies, make them “Linked Data” (2010)

Issue: domain experts have detailed understanding of processes in their field. None of the above technologies/standards are able to capture this knowledge.

Most current technologies provide vocabularies and leave the meaning to the behaviour of the application.

The semantics is what the app does (or the user with the app).

CHALLENGE – MICROSERVICES & DISTRIBUTED DATABASES

Microservice 12-Factors approach
1. Codebase - One codebase tracked in revision control, many deploys
2. Dependencies - Explicitly declare and isolate dependencies
3. Config - Store config in the environment
4. Backing services - Treat backing services as attached resources (Database implications!)
5. Build, release, run - Strictly separate build and run stages
6. Processes - Execute the app as one or more stateless processes (Transaction implications!)
7. Port binding - Export services via port binding
8. Concurrency - Scale out via the process model
9. Disposability - Maximize robustness with fast startup and graceful shutdown
10. Dev/prod parity - Keep development, staging, and production as similar as possible
11. Logs - Treat logs as event streams (Technologist's view vs legal framework, e.g. GDPR)
12. Admin processes - Run admin/management tasks as one-off processes

REF.: https://12factor.net/
SUMMARY

- Matchmaking = database querying
- Matchmaking = finding vector similarities
- Matchmaking = satisfying logical constraints

- Product catalogues are (poor) knowledge representations of real things with real semantics
- Configurators are (better) knowledge representations of real things with real semantics
- Business relationships / supply chains are expressed in business PROCESSES
- UBL/Camunda is NIMBLE’s approach to modelling/executing business processes
- eCl@ss is our approach to semantically annotating product catalogues
- Domain ontologies need to be “aligned” w.r.t. upper ontology, processes, static assets

THANK YOU FOR YOUR ATTENTION

Wernher Behrendt
Senior Researcher
Salzburg Research, Austria
E-Mail: wernher.behrendt@salzburgresearch.at

Salzburg Research Forschungsgesellschaft m.b.H.
Internet of Things / Industrial Internet
Jakob-Haringer Strasse 5/II
5020 Salzburg
Austria
JOIN THE PLATFORM:

HTTPS://WWW.NIMBLE-PROJECT.ORG/PLATFORM
IoT Data Harmonisation in vf-OS

2018-09-27 – Funchal, Portugal

Oscar Garcia Perales
Operations Director, Information Catalyst (ICE)

Data Mapping
- A component to develop maps for ETL
- Semantic mapping between formats/items and meanings/concepts
- Will help developing standalone Transformation routines
- These routines will be published to the vf-Store and re-used within Process Designer

Transformation
- A component to run Transformations routines
- Provide transformed data to demanding component
vf-OS Data Harmonisation

- vf-OS Data Model
- Syntactic and Semantic integration
- Crowdsourcing for better mapping suggestions
- Self-executable libraries
**vf-OS Data Model**

- **Main idea:**
  - To provide clues to the user establish the mappings
  - Focused in connecting concepts/terms without any taxonomic structure
  - Conceptual structure with relations that relates concepts independently the why or the properties associated to such connections

- **Future Actions:**
  - Work in relate concepts potentially be mapped
  - Will require (re)adjusts as far as we use it to confirm it works perfectly to our purpose

---

**IoT devices Data Harmonisation**

- In IoT world, sensors and devices are continuously spitting data each of them in their own format
- Each sensor will require an specific Transformation routine
- Similar sensors may share similar data formats
- vf-OS Manufacturing Maps can be:
  - Published in the vf-Store, so they can be sold (shared) with other representatives of the community
  - Derived from the knowledge of the crowd
  - Deployed as self-executable libraries for re-use at vApp development phase
  - Parametrized by providing any Pub/Sub topic, Messaging channels or persistent storages
Let’s build a part of a vApp

Business need:
- Create vApp capable of triggering alarms when a Tabber Stringer machine sensor hits a given value
Data Mapping

- Process to generate a map: input, map, output

- Suggested links and suggested concepts
- Missing links are also shown
Annex M: IS’18 vf-OS Photos