vf-OS: virtual factory Operating System

WP8: vf Smart Application Piloting and Validation

D8.1c: Annex C: Pilot 1 Evaluation Report
Vs: 1.0.3

Deliverable Lead and Editor: Raul Poler, UPV

Contributing Partners: UPV, MASS, IKERLAN

Date: 2018-10

Dissemination: Public

Status: Consortium Approved

Short Abstract
This document provides general information about the vApps that are going to be developed in the vf-OS project related to Pilot 1 needs re Manufacturing and Logistics / Automation. This concerns describing the vApps and their functionalities, the interactions between manufacturing and logistic users and software developers and the story maps developed.

Grant Agreement:
723710

Document Status

Deliverable Lead	Raul Poler, UPV
Internal Reviewer 1	Carlos Agostinho, KBZ
Internal Reviewer 2	Eduardo Saiz, IKERLAN
Internal Reviewer 3	Óscar García, ICE
Type	Annex
Work Package	WP8: vf Smart Application Piloting and Validation
ID	D8.1c: Annex C: Pilot 1 Evaluation Report
Due Date	2018-09
Delivery Date	2018-10
Status	Consortium Approved

History

See Annex A.

Status

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

Further Information

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:

ICE Information Catalyst

UNIVERSITÉ LUMIÈRE LYON 2

IK4 IKERLAN Research Alliance

MONDRAGON ASSEMBLY

UNINOVA Instituto de Desenvolvimento de Novas Tecnologias

VIASOLIS

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

CONSULGAL

Caixa Mágica Software

KNOWLEDGE BIZ CONSULTING

almende ORGANIZING NETWORKS

APR Applications Plastiques du Rhône

ASCORA Consulting - Development - Research

TARDY

Executive Summary

Deliverable “D8.1c Annex Pilot 1 Evaluation Report” provides a detailed description of the evaluation activities within Pilot 1: Manufacturing and Logistics / Automation. The pilot vApps described in this document focus on providing support to maintenance and spare part stock management in automation production equipment.

The annex contains the reports corresponding to the three evaluation periods regarded in the DOA at M24, M30, and M36. Each evaluation report contains a description of the software used for the evaluation (Software Under Test), the evaluation questionnaires generated using the Goal-Questions-Metrics (GQM) methodology, and the results of the evaluation.

Table of Contents

1 Pilot 1 Detailed Calendar 1
2 M27 Evaluation Report 3
2.1 Software Under Test....3
2.2 GQM Questionnaire....4
2.3 GQM Results....4
3 M30 Evaluation 5
4 M36 Evaluation 6

1 Pilot 1 Detailed Calendar

This section presents the development methodology and detailed calendar for Pilot 1 vApps development and validation. A less detailed calendar is provided in D8.1c Validation Scenarios main document, where it is stated that:

- **Phase 1:** To be released at month M26, scopes vfFailureManager RR1 developed by UPV and validated at M27 by UNINOVA
- **Phase 2:** To be released at month M29 and validated at M30, scopes vfFailureManager R2, vfFailurePrevention R1 and vfProductionFollowUp RR1. vfFailurePrevention is developed by IKERLAN and validated by UPV. vfProductionFollowUp is developed by UPV and validated by CMS
- **Phase 3:** To be released at month M35 and validated at M36, scopes vfFailureManager R3, vfFailurePrevention R2, vfProductionFollowUp R2, vfStockPolicies R1 and vfMaintenanceCalendar R1. vfStockPolicies and vfMaintenanceCalendar are developed by CMS and validated by IKERLAN

Some of the key concepts to be considered are:

- vApps differ in number of iterations/versions since vApps development depend on the availability of vf-OS components necessary for that vApp. vApps with more iterations involve less development effort in some of their iterations, as previous vf-OS components are integrated in previous releases of the same vApp. Figure 1 shows that fact.
- The parallelism between vf-OS component developments (releases) and vApps developments (releases) forces vApps developers to freeze the baseline of vf-OS component software packages used for evaluation purposes at M25, M29, and M35. This means that the versions of vf-OS components used for integration purposes may not be the latest available versions.
- A complete evaluation of vApps is intended at the end of the project, where all the vf-OS components are in their last and complete development, and end users have fully deployed and run their business processes with the new software systems

The development of vApps is based on agile methodologies, where the global functionality of an app is split in smaller functional groups called user stories. Those user stories were provided in D8.1a, and a detailed specification of user stories per vApp was made for the Pilot 1 case in the Annex C of D8.1b. User stories are grouped in sprints, that are bi-weekly deadlines where several user stories are developed and technically tested. Work generated in a sprint will be acknowledged and showcased at WP8 bi-weekly meetings, where status and risks are communicated to MASS users. Development problems due to inconsistencies with vf-OS component functionalities will be posted at the same bi-weekly pace. Development of vApps will finish 1 month before the ending of the phase. Evaluation of the vApp will be performed in the last month along with light bug-fixing.

With all the previous considerations, the development calendar foreseen for Pilot 1 is shown in Figure 1.

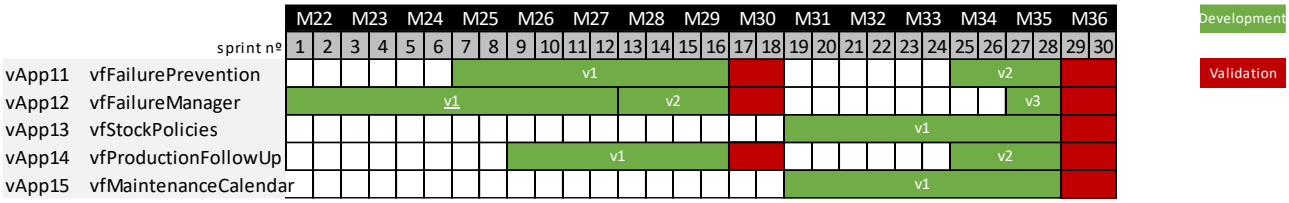

Figure 1: Pilot 1 development and validation calendar

2 M27 Evaluation Report

2.1 Software Under Test

As described in D8.1b, vApp12 - vfFailureManager purpose is to automatically detect failure states in Tabber Stringer (TS) machines and notify both the customer and supplier of a failure event. It also keeps historic information of past failures and the solutions that were applied. This information is used to support future maintenance operations.

vApp12 M24 is implemented in a photovoltaic panel manufacturing company where MASS has a new version of its TS installed. To be as independent as possible and not to interfere with the machine, an edge platform has been installed with an industrial PLC (IPC) and a NAS (Network-attached storage).

IPC is the only one that can read the variables of the TS, being able to read in real time with the EAP protocol (Extensible Authentication Protocol), and vf-OS platform will communicate with this device through protocols such as OPC-UA or ADS (Automation Device Specification).

MASS does not have access to its customer's line to perform M24 tests, so to test the functionalities of vf-OS, MASS has created an internal test environment in its facilities where, without communication with the machine PLC, the variables have been simulated.

On the otherhand, vf-OS developments have been deployed in the Docker installed on the NAS, this being also the scenario that is intended to be implemented in the real environment. Figure 2 depicts the real scenario and the test scenario for Pilot 1.

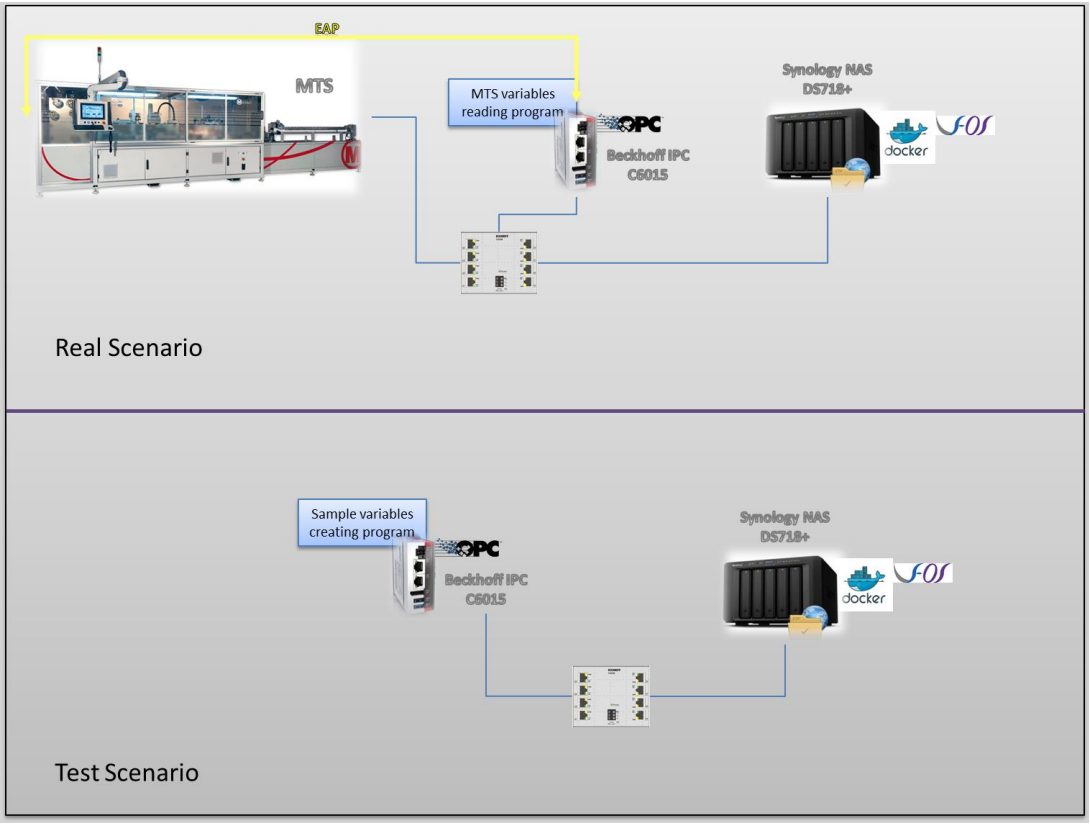

Figure 2: Pilot 1 M24 test scenario

The vApp12 M24 does not test the complete vApp functionality as stated in D8.1c, but a subset of already available components covering the backbone of the vApp. Figure 3 describes the different components regarded by the M26 evaluation.

Component	Functionality
Generator	The IOToolkit is used for generating the OPC UA driver instantiation necessary to communicate with the PLC.
Messaging	The alert recovered from the Tabber Stringer will be captured by the OPC UA driver and pushed to the Pub/Sub component to be shared with other vf-OS components.
TS OPC UA Driver	The driver is instantiated by the generator and is used to read and subscribe to changes to the PLC. The driver is configured and in charge of writing historic data for storing the historic alerts as required by MASS.
Security	The driver supports certificates and security policies required by the environment.

Figure 3: Pilot 1 test components

2.2 GQM Questionnaire

This section will be updated in M27.

2.3 GQM Results

This section will be updated in M27.

3 M30 Evaluation

Contents to filled in at M30.

4 M36 Evaluation

Contents to be filled in at M36.

Annex A: History

Document History	
Versions	V0.0.1-2: • ToC V0.0.3: • Section 1.1 filled with the vApp to be tested in M24 V0.1.0: • New section Pilot 1 detailed calendar introduced V0.1.1: • Changes to section 1 V0.1.1: • Calendar fixing V0.1.3: • First review and fixing V1.0.2: • Second revision V1.0.2: • Coordinator revision
Contributions	UPV: • Francisco Fraile • Víctor Anaya MASS: • Jean Philippe Aguerre • Jon Altube KBZ: • Carlos Agostinho (1st internal review) IKERLAN: • Eduardo Saiz (2nd internal review) ICE: • Carlos García (coordinator review)

Annex B: References

None

www.vf-OS.eu