BEMO-COFRA

Brazil-Europe Monitoring and Control Framework

(Project No. 288133)

D2.5 Final Validation Report of the Platform

Published by the BEMO-COFRA Consortium

Dissemination Level: Public

Project co-funded by the European Commission within the 7th Framework Programme
and
Conselho Nacional de Desenvolvimento Científico e Tecnológico
Objective ICT-2011-EU-Brazil

Document version: 1.0 Submission date: 10 March 2014

Document control page

Document file: D2.5 The Final Validation Report of the Platform v0.6.docx
Document version: 1.0
Document owner: Trine F. Sørensen (IN-JET)

Work package: WP2 – Requirements Engineering and Validation
Task: T2.3 – Metrics Definition and Validation Plan
Deliverable type: R

Document status: ☒ approved by the document owner for internal review
☒ approved for submission to the EC

Document history:

Version	Author(s)	Date	Summary of Changes made
0.1	Trine F. Sørensen, Jesper Thestrup (IN-JET)	2014-01-13	Table of contents. Chapter 2.
0.2	Ferry Pramudianto (FIT) Hussein Khaleel (ISMB)	2014-02-21	Evaluation of the development and monitoring tools. Input to 4 and 4.1 4.3.2 WP4 test activities 5.2 WP4 Validation Results
0.3	Eduardo Souto (UFAM) Peeter Kool (CNET)	2014-02-24	Input regarding WP3 Input regarding WP6
0.4	Trine F. Sørensen (IN-JET)	2014-02-26	Ex. summary, introduction, conclusion, editing of content.
0.5	Markus Taumberger (VTT)	2014-03-5	Contribution from VTT
0.6	Jesper Thestrup (IN-JET)	2014-03-7	Version for internal review
1.0	Jesper Thestrup (IN-JET)	2014-03-11	Final version submitted to the European Commission

Internal review history:

Reviewed by	Date	Summary of comments
Juliana Nogueira Vilela	2014-03-10	Accepted with minor comments
Lucas Lira Gomes	2014-03-10	Accepted with minor comments

Index:

1. Executive summary

The main aim of the validation is to assure that the BEMO-COFRA services developed adhere to the necessary quality standards for professional services and meet the needs and requirements of the end users.

In general, the validation activities in BEMO-COFRA consist of three distinctly different elements: i) Verification, ii) User validation, and iii) Usability testing. Previously to the actual validation activities which are reported here, a validation plan was defined specifying how each component would be tested (e.g. the procedure) and what results were expected. This deliverable reports on the results from the testing of all the individual components that together make up the BEMO-COFRA platform.

A complete end-user validation of the entire platform will be carried out in connection with the demonstration of the final prototype in a real setting. This will take place during the project's final review meeting in March 2014. The demonstration will simulate a scenario at the welding station where the car body parts arrive on a skid to be welded.

The validation consisted of various detailed assessment tests of the requirements in order to test if they fulfilled the fit criteria, i.e. the expected and desired result.

The components that were validated were:

- Availability
- Reliability
- Robustness
- Self-organization
- Monitoring
- Control
- Interoperability
- Scalability
- User friendly reporting
- Network diagnosis
- Data fusion
- Event debugging tool
- Legacy System Proxies
- Event Manager
- Data monitoring in shop floor

The validation results demonstrate that all requirements and the components fulfil the fit criteria and have therefore been validated to a satisfactory level. Based on these results, the demonstration of the full BEMO-COFRA platform is also expected to fulfil its promise and meet the end-users' requirements.

Usability testing of the Model Driven Development Tool and the iPad Energy Monitoring Tool was also carried out with users. The tests consisted of both a hands-on tests and usability satisfaction questionnaires. These results were also satisfactory.

2. Introduction

Validation is a vital element in any development work; validation is basically a way to ensure that what is being developed not only works but works the way the users want it to work. Validation is a fundamental element in the evolutionary requirements engineering, specification and design methodology which the BEMO-COFRA project has adopted. Users have a central and crucial role in the definition of requirements, of quantitative and qualitative metrics, and the validation process. The main aim of the validation is to assure that the BEMO-COFRA services developed adhere to the necessary quality standards for professional services and meet the needs and requirements of the end users.

The development work in the project has had two iterative cycles. Lesson Learned have been collected twice and the requirements have been updated and new requirements added as a result. The results from each iterative cycle have thus directed the following development work.

A first prototype of the BEMO-COFRA platform was developed by the end of the project's first year. It demonstrated some of the main aspects and features of the BEMO-COFRA platform and allowed developers to test the integrated components. The results have fed into the final year development work where requirements and their related components have been further tested and validated in accordance with the predefined validation plan.

This deliverable presents the validation results from the testing activities of the different components that make up the complete platform. These activities have been carried out in the technical work packages in accordance with the validation framework and plan defined in *D2.3 Validation Framework* and *D2.3 Validation Framework ANNEX*.

An additional end-user validation of the entire platform will be carried out in connection with the demonstration of the final prototype at the project review in March 2014. The final demonstration will simulate a scenario at the welding station where the car body parts arrive on a skid to be welded. Due to the complexity and high cost of the robots used in car manufacturing industry it is only possible to perform one full demonstration of the BEMO-COFRA platform. The full demonstration will therefore take place at the project's final review meeting and, based on the demonstration, end-user validation will be carried out and documented in the Project Final Report.

2.1 Background

BEMO-COFRA platform provides a middleware able to expose smart objects, legacy devices and subsystems capabilities by means of web services. Syntactic and semantic interoperability among coexisting technologies in the overall monitoring and control framework is made available. WSAN devices, legacy subsystems and devices will thus be able to interact and cooperate, orchestrated by a manager in charge of enforcing a distributed logic with the overall monitoring and control network.

The BEMO-COFRA platform features a Service oriented Architecture (SoA) and a middleware able to expose smart objects, legacy devices and sub-systems' capabilities by means of web services thus supporting syntactic and semantic interoperability among different technologies coexisting in the overall monitoring and control framework. Wireless Sensor and Actuator Network (WSAN) devices, legacy sub-systems and devices will thus be able to interact and cooperate, orchestrated by a manager in charge of enforcing a distributed logic with the overall monitoring and control network.

The project adopted an evolutionary requirement engineering, specification and design methodology, which has had two iterative cycles. The results from each iterative cycle have fed into the ensuing development work. All requirements have followed a pre-defined workflow and the online tool, JIRA, has been used to manage requirements engineering, thus making it easy to trace each requirement's progress and status. The requirements workflow is a good tool to ensure the quality of requirements, as it separates the checking of the form and the contents of the requirements. Once a requirement is implemented, it needs to be validated by performing the appropriate testing activities. To this end, a validation framework and a plan for the specific testing and validation activities were defined (see *D2.3 Validation Framework* and *D2.3 Validation Framework ANNEX*) and each development work package has been responsible for carrying out the necessary and appropriate validation tasks.

2.2 Purpose, context and scope of this deliverable

This deliverable presents the results from the testing and validation activities carried out in the technical development work packages, e.g. WP4-WP6. It is important to note that although the architectural design phase is conducted in *WP3 Large scale distributed system architecture*, actual testing and validation activities related to the architectural components and their associated respective requirements have been carried out in WP4-WP6 as the components have been developed in these work packages. Similarly, as *WP7 Solution Integration and Deployment* is responsible for the final validation of the complete platform the results from WP7's validation activities will therefore be reported in the Final Project Report. As noted above, this is due to the fact that the BEMO-COFRA platform will only be tested in its entirety at the demonstration at the final review in March 2014.

This deliverable is concerned with the validation of the separate components that make up the BEMO-COFRA platform. The results are presented per work package thereby following the format introduced in *D2.3 Validation Framework ANNEX*.

The deliverable is structured as follows:

Chapter three gives brief overview of the adopted validation methods

Chapter four presents the components that have been tested. The requirements that make up each component are identified. The assessment procedure, the expected results and the validation results are described and where relevant. In chapter five, the results of usability testing are presented.

Finally, the conclusion in chapter six briefly sums up the results.

3. Validation Methods

The BEMO-COFRA validation framework outlines a structured approach to software testing and user validation, including the definition of quantitative and qualitative metrics, guidelines for usability testing and identification of success criteria for the field trials. The validation framework provides guidance for carrying out the validation activities and for making the decisions about redesign, error correction, start of implementation etc., on the basis of the validation results.

The user validation plan is inspired by the guidelines recommended by VNET5[1], including the three steps in the user validation process:

1. Planning user validation
2. Carrying out validation activities according to this plan
3. Making decisions founded on the validation results (e.g. redesign, correction of errors, implementation, and release).

The general validation approach for the project is described in *D2.3 Validation Framework*, whereas the specific methods for each validation activity were described in *D2.3 Validation Framework ANNEX*.

Overall, the validation activities in BEMO-COFRA consist of three distinctly different elements:

- Verification, i.e. a quality control (QC) process that is used to evaluate whether or not an artefact, product, service, or system complies with regulations, specifications, or conditions imposed at the start of a development phase
- User validation, i.e. a quality assurance (QA) process of providing a high degree of assurance that a product, service, or system accomplishes its intended requirements namely the expectations and requirements of its intended users
- Usability testing, i.e. field tests with users in order to assess the quality of use of the applications and user satisfaction.

The software verification (debugging and testing) is a quality control (QC) process that was used to evaluate whether or not a system component complied with regulations, specifications, or conditions imposed at the start of a development phase. It was always performed at the laboratory level by the technical partner(s) responsible for the component.

The user validation element was done at the laboratory level, with technical partners analysing each software module and verifying its consistency alone and inside the overall architecture. Then the assessment of performance measurements was carried out with both user partners and technical partners who have not contributed to the implementation, so that there was an evaluation of the (stable) components and prototypes from different points of view.

Finally, usability testing was assesses the quality of use in the field trials made with users. Usability testing was carried out in controlled conditions to guarantee that valid and interpretable results were obtained. While some usability testing has been carried out at present, the overall end-user validation will be carried out using a test bed where the platform will be deployed and tested. The user scenarios will provide a baseline and the actual end-users will be involved in the validation activities which will focus on both qualitative and quantitative metrics. As noted above, the test bed and the final demonstration and test of the BEMO-COFRA platform will be carried out in connection with the project review in March 2014. The results will be reported in the Project Final Report.

[1] www.vnet5.org

4. Validation Activities

In *D2.3 Validation Framework ANNEX*, the components that make up the BEMO-COFRA platform were identified and the tasks involved in validating these components were described. Each component has of a number of requirements that must be fulfilled and each requirement has a defined Fit Criteria. The Fit Criteria quantify how to measure that the requirement is met, i.e. how to test it. Based on Fit Criteria, detailed assessment procedures and their expected results were defined in *D2.3 Validation Framework ANNEX*.[2]

The following subchapters present the results from the validation activities that have been carried out in WP4-WP6.

4.1 WP3 Validation Activities and Results

The architectural design phase has been conducted in WP3. It involves the designs for architecture, their components and interfaces. The validation plan for the design phase includes the following tasks:

- Software design evaluation, which correspond to evaluate design of architectural components for correctness, consistency, completeness, and accuracy. The goal is assess the technical merits of the architectural components if they are in compliance to the requirements specification.
- Software design traceability analysis, which correspond to trace design documentation to requirements. This allows developers to analyse the correctness and consistency of the relationship between architectural components.

As noted in section 2.2., these tasks have been carried out in the development work packages (i.e. WP4, WP5, and WP6). The testing activities and their results are presented below.

[2] A description of all the requirements and their fit criteria can be found in the deliverables D2.2 Preliminary Requirements Report, WD2.2 Updated Requirements Report, and D2.4 Change Request and Re-engineering Report.

4.2 WP4 Validation Activities and Results

The activities in WP4 were divided in 3 major components, as previously presented in *D2.3 Validation Framework*. These components are further divided into features and requirements necessary for the developed system. The table below lists the different components and their related requirements, describes how they will be tested and what results are expected, and finally describes the actual validation results. It should be noted that the test activities were performed in an indoor environment at the laboratory where the various components have been developed; however, this does not affect the quality and generality of the tests.

Dependability

The following provides a brief description of the developed features related to the dependable network infrastructure and the tests performed for validation.

Component	Requirements	Assessment procedure	Expected Results	Results
Availability	BEMOCOFRA-6 BEMOCOFRA-7 BEMOCOFRA-8	*Multi-Radio Nodes* (MRNs) have a *network interface priority array* that determines which interface should be used for transmitting their data. If the interface with the higher priority is not available, the next interface in the priority array is automatically used and so on. In addition, MRNs are capable of sending data using all the available interfaces simultaneously. On the other hand, *Single-Radio Nodes* (SRNs) communicate using 6LoWPAN only. Disabling a certain radio interface of an MRN, or introducing interference to force an interface or channel switch will make that interface unavailable (temporarily in the second case); and when this happens, other available interfaces will be used to send the data, until the interface with the higher priority becomes available. Autonomous context-awareness mechanisms at the *WSAN Gateway* are	Maintaining network end-to-end connectivity for 99.9% of the time, i.e. less than 1min/day of downtime. Ensuring quick switching among multiple radios in less that 3s.	When the WSAN operated using three wireless interfaces in a collaborative manner, it was possible to achieve and maintain **end-to-end connectivity for100% of the time**, and so it was possible to keep the no network up all the time. When a certain radio interface is being switched it temporarily becomes unavailable, in the meanwhile, another radio interface carries-on with node communications until the switching interface becomes available.

Component	Requirements	Assessment procedure	Expected Results	Results
		implemented to perform the necessary functions, leveraging on the *multi-radio manager*. MRNs can send data using all the available interfaces if it receives a command by the user, via the administration tool.		
Reliability	BEMOCOFRA-14 BEMOCOFRA-28 BEMOCOFRA-30 BEMOCOFRA-64	Monitoring packets from MRNs and SRNs include sequence numbers that can be used to estimate packet loss rates, prevent duplicate information, and ensure the correct sequence of events. If the multi-radio manager detects high packet losses on the current interface, it sends an interface priority command to the network with the new interface priority array, letting the MRNs to use a different interface, and eventually reduce packet losses. Delays between the WSAN Gateway and all network nodes are calculated and monitored by the multi-radio manager; and if the latter detects that the delay of the current interface is higher than a certain threshold, it sends an interface priority command to the network with the new priority array, letting MRNs to use a different interface, and ensure a deterministic time window for communications. Estimation of the accuracy of retrieved information, retrieved from multiple	Successful arrival of 99.9% of the data. Data freshness and correctness. Adaptability of the WSAN to context information.	Using all three radio interfaces collaboratively, it was possible to keep **0% packet loss rate**. The multi-radio manager module was able to continuously monitor the packet loss rates and delays of the individual interfaces and manage the interface priorities for the WSAN. The result is that the monitoring data are received within the **correct sequence**, **no duplicates**, and within a **deterministic time window**.

Component	Requirements	Assessment procedure	Expected Results	Results
		sensor nodes monitoring the same event.		
Robustness	BEMOCOFRA-13	The WSAN implements *spectrum-sensing* (of IEEE 802.15.4 channels) and dynamic *channel allocation* capability. When interference is introduced to the current operating radio channel, using self-developed jamming devices based on IEEE802.15.4, the multi-radio manager is able to detect the presence of interference and reallocate the network to the new radio channel. In the meanwhile, if the interface whose channel is being switched is currently in use, another interface will be used to transmit data in MRNs.	Successful reallocation of the network to a new operating channel or to another radio interface, if interference is present.	The spectrum manager is able to successfully **detect interference** on IEEE 802.15.4 channels, and issue an interface switch command to reallocate the corresponding WSAN to the **best available channel**. The spectrum manager is able to successfully **detect overlapping channels** of clusters and reallocate one cluster to another **unused channel**.
Self-organization	BEMOCOFRA-29 BEMOCOFRA-30	WSAN nodes use a *network discovery* mechanism at start-up to search for and join the desired network. SRNs scan IEEE 802.15.4 channels searching for the network. MRNs use all the available interfaces simultaneously to find the network. If the MRN is able to join the network on any of its interfaces, it receives the operating information of the other interfaces, allowing them to immediately join their corresponding networks. The nodes also implement a *connection monitoring* mechanism, such that if a node doesn't receive any packets on a	Network discovery and network recovery without user intervention.	When powered-up, all WSAN nodes execute a network discovery mechanism. Nodes are able to successfully **search for and join the desired network**. SRNs search IEE 802.15.4 channels. MRNs search for the desired network using all available interfaces simultaneously, and thereby **minimizing the discovery time**. A connection monitoring mechanism is executed at each

Component	Requirements	Assessment procedure	Expected Results	Results
		certain interface it detects that it has lost connectivity with the network on that interface, and makes it unavailable for communications, until it joins the corresponding network.		node and was able to successfully **detect connection loss**, then starts the discovery process. MRNs exploit the availability of multiple interfaces to **minimize recovery time** of the disconnected interface.

Network management

The following provides a brief description of the developed features related to the network management and the corresponding tests performed for validation.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
Monitoring	BEMOCOFRA-27 BEMOCOFRA-52 BEMOCOFRA-53 BEMOCOFRA-54 BEMOCOFRA-55 BEMOCOFRA-56 BEMOCOFRA-57 BEMOCOFRA-58 BEMOCOFRA-59 BEMOCOFRA-60 BEMOCOFRA-65	A WSAN monitoring module has been developed that collects data from the network in two ways: i. Monitoring data sent by the nodes. ii. Node statistics in terms of packet losses and delays inferred by the monitoring module. Monitoring data are partially processed inside the nodes and managed in clusters to reduce monitoring overhead. In addition, a GUI has been developed and deployed to display the following: i. Real-time network topology. ii. Real-time spectrum occupancy. iii. Real-time event monitoring and node performance data.	Monitoring tool for WSAN status in terms of connected nodes list, lost nodes, packet losses, battery, interference status, uptime. Low monitoring overhead. Ability of real-time and long-term monitoring. Platform and application independent monitoring.	The monitoring module is able to successfully **store, organize, and display** data from the WSAN, in terms of: Interference status, event monitoring, list of node, types, addresses, roles, reporting times, status, packet losses, and delays. (Battery and life-time data are not considered anymore). Application and node status monitoring information are received and inferred from the same packet. This way, **traffic overhead is minimized for application and node performance information**.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
		iv. Long-term node performance data. A *WSAN Proxy* has been developed to expose the necessary event monitoring information to the LinkSmart enabled environment. The WSAN proxy has been designed to run as a standalone, loosely-coupled, and platform-independent application. In addition, the WSAN Proxy exposes node attributes through an interface that has been designed to allow flexible and easy modifications as necessary. All monitoring capabilities have been developed to be remotely accessible. The monitoring module described above has been deployed in the test-bed in to verify its ability to collect the information arriving from the network, organize the data provided by the individual sensor nodes, update the information on the GUI, and visualize it remotely.		However, it was not able to combine spectrum-sensing traffic with other applications. The monitoring module is able to display both **real-time and long-term information**. The monitoring module (the management tool in general) is **remotely-accessible and platform independent**.
Control	BEMOCOFRA-52 BEMOCOFRA-54 BEMOCOFRA-55 BEMOCOFRA-56 BEMOCOFRA-57 BEMOCOFRA-58 BEMOCOFRA-59 BEMOCOFRA-61	The developed WSAN proxy provides an interface that exposes WSAN-related resources to the user and allows controlling the operational parameters of individual nodes. The proxy forwards these commands to the WSAN nodes. A remotely-accessible control GUI has been developed for a user-friendly control actions. The control module provides the access to the following:	Access and control WSAN resources and correct functioning of the WSAN control module. Platform and application independent monitoring.	The WSAN control module is able to successfully **expose WSAN resources** and allow the user to **control WSAN operational parameters**. Commands were **correctly transported** to the target nodes, and were **correctly interpreted** by the WSAN nodes.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
	BEMOCOFRA-62	i. Start, stop, and change the interval of reports. ii. Perform actuation. iii. Change the radio interfaces priority. The WSAN Proxy and the control GUI were deployed in the test-bed to verify correct functioning in terms of message delivery to the destination node and the task execution.		The monitoring module (the management tool in general) is **remotely-accessible and platform independent**.
Interoperability	BEMOCOFRA-9	The WSAN Proxy has been designed to receive/publish events from/to other sub-systems in the station through the LinkSmart middleware. The WSNA Proxy was deployed in the test-bed to verify the reception of events from the PLC, and ensure that the corresponding actions are correctly executed. In addition, WSAN events were published and verified for correct reception by the other sub-systems.	Correct reaction and interpretation of a WSAN to the operational states of other system components. Efficient exposure of the industrial process monitoring information from the WSAN to other system components. Value form wireless sensor network can be saved in a PLC variable which is used by the robot to determine if the clamps on the item have closed.	The WSAN is able to successfully subscribe to PLC events. The WSAN is able to successfully publish events to the LinkSmart environment, both related to clamp state and skid-orientation monitoring.

Scalability

The following provides a brief description on how scalability has been achieved in the considered system, particularly, how dependability and network management features are addressed in large-scale WSAN, and what are the validation tests that were performed.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
Scalability	BEMOCOFRA-2 BEMOCOFRA-54	In the developed WSAN, scalability has been addressed by dividing the network into clusters, where each cluster is identified by a *cluster ID*, managed as	Maintain end-to-end connectivity of a large-scale WSAN.	The large-scale WSAN operates in clusters.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
		an individual sub-network, and handles its own traffic and operational parameters. Scalability has been considered in the two aforementioned system components: dependability and network management. Dependability features of the large-scale network have been designed to operate in a centralized way, while individual clusters are managed independently from one another, yet in a way that allows an efficient coexistence among the clusters. That is, node performance parameters, interface priorities, and spectrum management are managed separately in each cluster, however radio interfaces of neighbouring clusters cannot operate on the same radio channels. Network management capabilities have been designed to include both a node address ID and a cluster ID for both monitoring and control functions, in order to efficiently allow node identification in the large-scale network. A network with multiple clusters has been deployed to verify: first, the correct functioning of the dependability features, and second, the ability of the management tool to identify the individual nodes within the available clusters.		Dependability features were successfully handled by the multi-radio manager in a cooperative manner for the different clusters. **Different clusters are assigned different radio channels.** **Different clusters conditions are handled separately.** Network management was successfully extended to allow for the large-scale WSAN administration. A WSAN node is represented by a **combination of cluster ID and node ID.**

4.3 WP5 Validation Activities and Results

WP5 will develop the distributed control logic and enabling features, namely the monitoring and control tool for WSANs and the LinkSmart-enabled monitoring and control infrastructure. The distributed control logic that supports service orchestration will be defined together with mechanisms to map it on the actual hardware resources. The table below lists the different components and their related requirements, describes how they will be tested and what results are expected, and finally describes the actual validation results.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
User friendly reporting	BEMOCOFRA-4	Display the monitored data and generate different error conditions and check that the information is presented with different level of detail according the selected user type (worker, manager, etc.)	Monitored data of the shop floor is presented in the desired level of detail to the stakeholder. 99% of the users confirm that the presented information is understandable for them.	The Graphical User Interface **visualizes the monitoring information gathered from the BEMO-COFRA installation in a continuous-time 3D representation**. The main objects are the car body, welding robots, and the grippers. **Detailed information about each object** is available when clicking or touching them on the screen. Active objects are rendered in green colour whereas idle objects are rendered in grey and objects that are in erroneous state are rendered in red.
Data monitoring in shop floor	BEMOCOFRA-5 BEMOCOFRA-44 BEMOCOFRA-45 BEMOCOFRA-46	Reading the relevant parameters from the PLCs and where necessary from additionally added wireless sensor nodes.	Consumption of power and cooling water, electrode temperatures are measured, saved in a database, and used in combination with service history to predict maintenance intervals of the welding robots.	The Administration Tool Manager serves the web content (HTML5 and Javascript) representing the Administration Tool to the client browsers. The Administration Tool can **access all BEMO-COFRA resources** through the use of SOAP API, and optionally also receive **asynchronous events** over the WebSocket connection, if it becomes available in the BEMO-COFRA platform.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
Network diagnosis	BEMOCOFRA-22	Inspecting the network using the network management protocol while introducing errors to the system. Verify that problems can be detected. Network problems such as latency, delay, loss, duplication will be simulated through a traffic shaper (netem[3]). On the web service level Fiddler[4] will be used.	Measure and display the network performance to diagnose potential system problems.	The implemented system adopts the Simple Network Management Protocol (SNMP) in order to communicate network-related information. For this purpose, each **Sink MRN is equipped with an SNMP Agent extension** that implements the necessary translation of information into the SNMP format then, whenever required, data are exchanged with the SNMP Manager using standard Traps in the up-link and Requests in the down-link.

[3] http://www.linuxfoundation.org/collaborate/workgroups/networking/netem
[4] http://www.fiddler2.com/fiddler2/

4.4 WP6 Validation Activities and Results

WP6 has integrated the different components and solutions provided by WP3, WP4 and WP5 into a coherent Production and Monitoring platform based on Internet of Things technologies. This includes adapting the LinkSmart middleware and creating interfaces for legacy devices, as well as interfacing with wireless sensor networks to acquire sensor data from the manufacturing process. The aim was to establish a central integrated environment for monitoring and control of all production processes. The table below lists the different components and their related requirements, describes how they will be tested and what results are expected, and finally describes the actual validation results.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
Data fusion	BEMOCOFRA-37 BEMOCOFRA-33 BEMOCOFRA-35 BEMOCOFRA-32	An event generator will be used to generate energy consumption data The data fusion component compiles energy consumption reports at a configurable interval and sends them to one or more recipients. These reports should be configured by a normal user.	Periodical energy consumption reports of units, stations and production lines are created and sent. These reports should follow the report configured by the user.	Event Generator has been created for testing the data fusion component. The real-time energy consumption events arrived every second and aggregated into consumptions per production process.
Event debugging tool	BEMOCOFRA-66	Events are generated by a program and the debugging tool is set to intercept the events.	The tool shows stored events. User is able to filter for selecting events from the log using regular expression.	Events are stored and can be filtered using the tool.
Legacy System Proxies	BEMOCOFRA-39 BEMOCOFRA-15	An OPC server is set up and connected to PLC, a PLC proxy shows the actual variables values. The value of the PLC variables is changed from SIMATIC step 7. New values are entered from the proxy.	PLC Proxy fires event when OPC variable is changed from SIMATIC Step 7. Values entered from the proxy are written to the PLC.	Events are received when PLC variables change. The applications could change the variables value through the OPC Proxy
Data fusion	BEMOCOFRA-42	Historical data from COMAU will be used to simulate a production database. BEMO-COFRA can be set to transform data and export it to at least one ERP system.	Transformed data follows the defined format which is understood by the ERP system.	Historical data are transformed into comma separated values and could be imported into SAP Netweaver.

Component	Requirements	Assessment procedure	Expected Results	Validation Results
Event Manager	BEMOCOFRA-21 BEMOCOFRA-10	Notifications will be simulated and injected to the system according the workflow defined in the production line. I.e. The person responsible for a certain unit will get notified of any problems there.	System notifications must be sent to right person as defined by different end user roles.	The event manager is able to rout events into the corresponding user interface for the intended users.

5. Usability Evaluations

5.1 Model Driven Development Tool

The first prototype of the model driven development tool has been evaluated in a controlled experiment where the users were given an hour task to develop a prototype for monitoring a room temperature. The reminder of this section has been published in "Model Driven Development for Internet of Things Application Prototyping" (Pramudianto, Rusmita et al. 2013).

A software walkthrough was used as the evaluation method to measure the users' acceptance to the proposed architecture and the proposed MDD tool described in D6.3 Data Fusion Engine for production monitoring(Pramudianto 2013). The evaluation was done with 7 participants (six males and a female).

Furthermore, to investigate the user satisfactions to the notations used in the tool, the participants are required to perform the same task with the proposed tool and Eclipse's EMF tool. The order of the tool used by each participant was exchanged to minimize the learning effect. Then the users were asked to rate the overall experience working with the tool using DSL notations compared to the EMF using UML notations. After performing the tasks, the participants were told to fill in the IBM computer usability satisfaction questionnaires (Lewis 1995) that contains questions with 7-level Likert-scale options where "7" denotes "Strongly Disagree" down to "1" which means "Strongly Agree".

The questions are divided into four categories that include the overall experience with the framework (Overall), the functionalities of the tool (Tool Functions), the workflow to be done when working with the tool (Workflow), and the user interface of the tool (UI). The questions are again adopted from the (Lewis 1995).

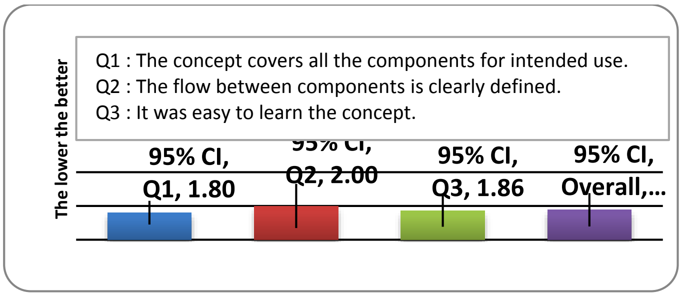


Figure 1: User satisfaction of the proposed architecture

As illustrated in Figure 1, the result of the questionnaire regarding the architecture shows that the users felt it was easy to understand (M=1.86, SD=0.64) and its functions were clearly defined (M=2, SD=0.82). Secondly, the proposed architecture helped the user developing the intended functional prototype (M=1.8, SD=0.4). Overall the users were satisfied to the architecture design (M=1.89, SD=0.67).

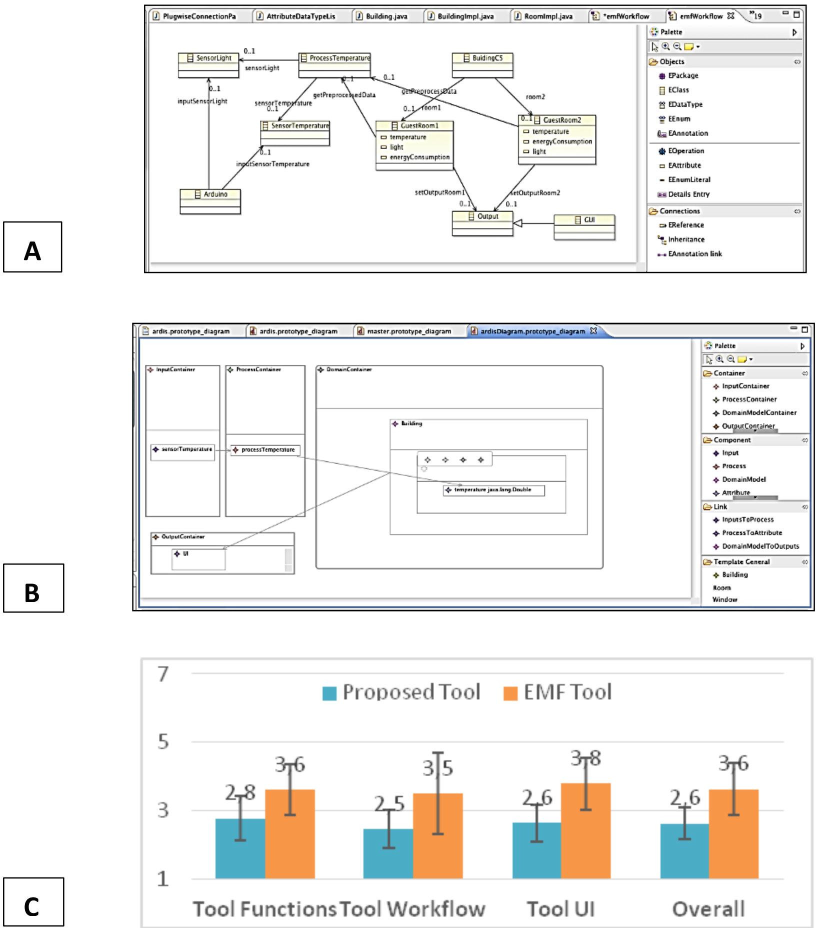


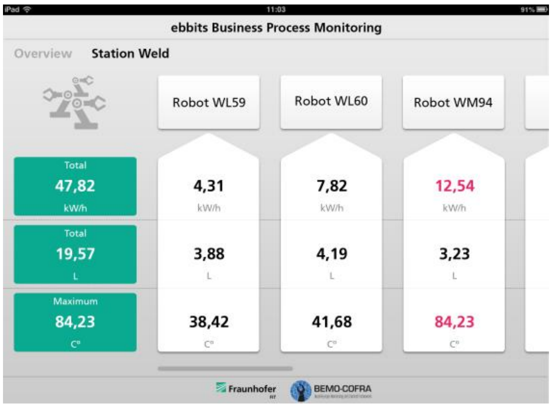
Figure 2: EMF with UML notation (A) vs. IoT Modelling Tool with DSL notation (B) and the evaluation results (C)

The comparison score between the proposed MDD tool and EMF Tool are presented in Figure 2. The proposed tool overall score better compare to modelling with UML diagram.

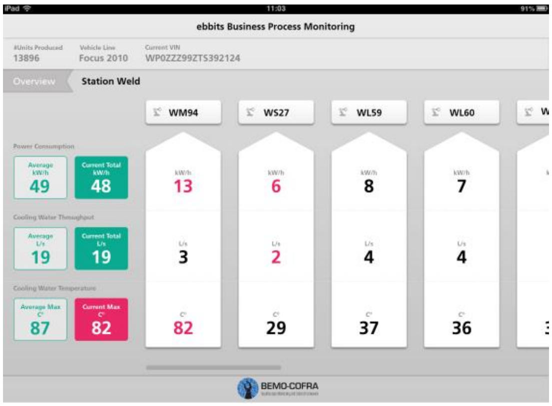
A paired sample T-Test analysis was performed to investigate if there is a statistical difference between user's satisfaction to the writer's work and EMF tool. The result of the questionnaire shows that regardless that the proposed tool scored a better means in all categories there was no significant differences of user satisfaction for the Functions, Workflow, and Overall [$T(7)=1.2$, $p>.5$), ($T(7)=-1.38$, $p>.5$), ($T(7)=-2.04$, $p>.5$) respectively]. Interestingly, the user opinions were significantly affected by the user interface of the tools ($T(7)=-2.66$, $p<.5$).

5.2 iPad Energy Monitoring Tool

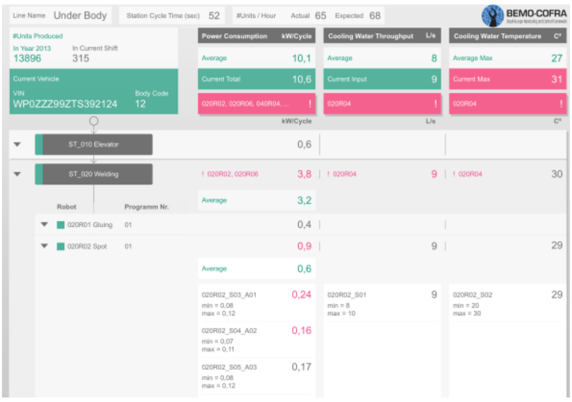
For evaluating the design and functionalities of the energy monitoring user interface we choose to perform a qualitative analysis instead of quantitative analysis since the user interface is still in a prototype stage. Qualitative analysis helps us to obtain the missing features required by the line managers to monitor the production cells on the shop floor. Based on the requirements collected on a workshop with COMAU and FIAT employees in Betim, Brazil and in Turin, Italy, several design prototypes were introduced and evaluated iteratively.



First prototype



Second prototype



Third prototype

Figure 3: Prototypes of the user interface for line monitoring

First workshop has been conducted with COMAU department in Italy. We introduced the prototypes and identified the missing features. During the workshop the users found that the first prototype was not clear since it does not show the hierarchy of the manufacturing lines which the users are used to. The second prototype was created and discussed with the same users. Although they found it better, they request an interface that could show them all important status on the first glimpse. The third prototype was created and also evaluated with COMAU in Betim. Their feedback are quite useful, several users were overwhelmed by the amount of information on the user interface.

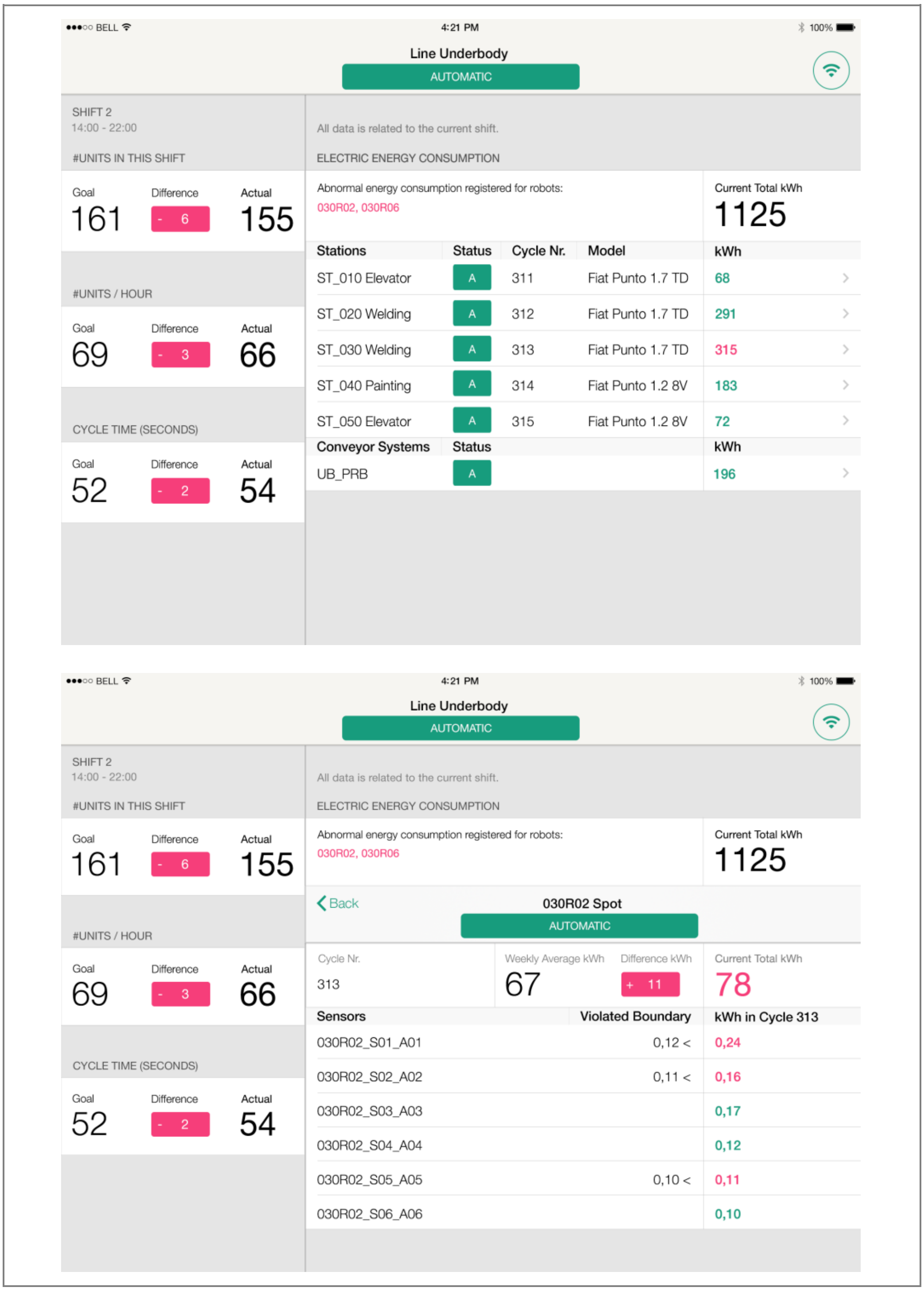


Figure 4: The final user interface design

The design of the monitoring tool was adjusted as depicted in the Figure 4 based on the feedback and finally they are approved by the users. We will evaluate this interface on the next Hannover Fair to obtain more general opinions from different type of users. In the future, we would also like to perform quantitative analysis by measuring time required by the workers to interact with the user interface.

6. Conclusion

The results from the validation testing demonstrate that the requirements and components that make up the BEMO-COFRA platform perform as requested. The usability testing allowed real users to test two of the tools available: Model Driven Development Tool and the iPad Energy Monitoring Tool. The results showed that users were satisfied with the design and usability of both tools.

The BEMO-COFRA platform will be subject to another complete test in March 2014 where it will demonstrate a simulation of the actions at the welding station in a car manufacturing plant. The results will be recorded in the Final Project Plan.

References

BEMO-COFRA, (2013)	The BEMO-COFRA Consortium, D2.3 Validation Framework
BEMO-COFRA, (2013)	The BEMO-COFRA Consortium, D2.3 Validation Framework ANNEX
BEMO-COFRA, (2013)	BEMOCOFRA Consortium, D6.3 Data Fusion Engine for production monitoring
Lewis, J. R. (1995).	IBM computer usability satisfaction questionnaires: psychometric evaluation and instructions for use, International Journal of Human - Computer Interaction 7(1): 57-78.
Pramudianto, F., et al. (2013)	Model Driven Development for Internet of Things Application Prototyping. The 25th International Conference on Software Engineering and Knowledge Engineering, Hyatt Harborside at Logan Int'l Airport, Boston, USA, Knowledge Systems Institute Graduate School.