Z - B R E 4 K

**Grant agreement nº: 768869**
**Call identifier: H2020-FOF-2017**

**Strategies and Predictive Maintenance models wrapped around physical systems for Zero-unexpected-Breakdowns and increased operating life of Factories**

**Z-BRE4K**

# Deliverable D2.1

Incorporation of AUTOWARE and certain extensions in Z-BRE4K

## Work Package 2

Operating System and Networked Machine Simulators

| | | |
|---|---|---|
| **Document type** | : | Report |
| **Version** | : | V2.1 |
| **Date of issue** | : | 30/06/2018 |
| **Dissemination level** | : | PUBLIC |
| **Lead beneficiary** | : | INNOVALIA |

***This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement nº 768869.***

# EXECUTIVE SUMMARY

| | |
|---|---|
| Abstract | ▪ This deliverable is the work obtained during the effort done in task T2.1 Operating System for Cognitive Manufacturing.<br>▪ The first goal is to recollect the architecture requirements and information achieved in WP1, having into account the Reference Architecture's description and the different components that will add up to the predictive maintenance that Z-BRE4K will lead to.<br>▪ The next step is to analyse AUTOWARE's RA and the different Open Source enablers and components developed in different international projects, enlisting the different ways they can be exploited and used for smart predictive maintenance.<br>▪ Solutions to extend AUTOWARE's interoperability to fulfil the heterogenic capabilities required through the combination of the Digital Shopfloor Alliance Reference Architecture and the IDS Communication Architecture with FIWARE Generic Enablers as core components.<br>▪ The final Z-BRE4K General Architecture can be adopted to the different use cases, where a series of steps to be able to apply it must be taken to reach different particular architectures fulfilling the predictive maintenance requirements.<br>▪ The final Z-BRE4K generic and particular information workflow have been defined. |
| Keywords | Architecture, Components, FIWARE, AUTOWARE, Industrial Data Spaces, Digital Shopfloor Alliance, Components, Information flow, Middleware, Cloud, Fog, Strategies, Predictive Maintenance |

# REVISION HISTORY

| Version | Author(s) | Changes | Date |
|---|---|---|---|
| 0.1 ToC | Aitor González (INNOVALIA) | ToC | 29/05/2018 |
| 0.2 | Aitor González (INNOVALIA) | Z-BRE4K RA V0.1 | 5/06/2018 |
| 0.3 | Oscar Lázaro (INNOVALIA), Paloma Taboada (TRIMEK) | Comments on Z-BRE4K RA V0.1 | 8/06/2018 |
| 0.4 | Aitor González (INNO) | Z-BRE4K RA V0.2 | 11/06/2018 |
| 0.5 | HOLONIX, ATLANTIS, EPFL | Comments on Z-BRE4K RA V0.2 | 12/06/2018 |
| 0.6 | Aitor González (INNOVALIA) | Z-BRE4K RA V0.3 | 15/06/2018 |
| 0.7 | HOLONIX, ATLANTIS, EPFL | Comments on Z-BRE4K RA V0.3 | 19/06/2018 |
| 0.8 | Aitor González (INNOVALIA) | Final Version of Z-BRE4K RA | 22/06/2018 |
| 1.0 | Aitor González (INNOVALIA) | V1 of Deliverable 2.1 | 28/06/2018 |
| 2.0 | Daniel Gesto (AIMEN) | V2 of Deliverable 2.1. Some corrections about format. | 29/06/2018 |
| 2.1 | Aitor González (INNOVALIA), ATLANTIS | V2.1 of Deliverable 2.1 Comments Incorporation | 29/06/2018 |

# TABLE OF CONTENTS

# ABBREVIATIONS

| Abbreviation | Name |
|---|---|
| **API** | Application Programming Interface |
| **BPMN** | Business Process Model and Notation |
| **CAD** | Computer-Aided Design |
| **CAM** | Computer-Aided Manufacturing |
| **CECM** | Cognitive Embedded Condition Monitoring |
| **CPPS** | Cyber-Physical Production System |
| **CPS** | Cyber-Physical Systems |
| **DB** | Data Base |
| **DSA** | Digital Shopfloor Alliance |
| **DSS** | Decision Support System |
| **ERP** | Enterprise Resource Planning |
| **ET** | Engineering Technology |
| **FM** | Failure Mode |
| **FMEA** | Failure Modes and Effect Analysis |
| **FMECA** | Failure Modes, Effects and Criticality Analysis |
| **FPGA** | Field-Programmable Gate Array |
| **GPU** | Graphics Processing Unit |
| **H/S** | Hardware and Software |
| **HPC** | High-Performance Computing |
| **IDS** | Industrial Data Space |
| **IDSA** | International Data Space Association |
| **IoT** | Internet of Things |
| **IS** | Information Systems |
| **IT** | Information Technology |
| **KPI** | Key Performance Indicator |
| **KRI** | Key Risk Indicator |
| **MES** | Manufacturing Execution System |
| **OEM** | Original Equipment Manufacturer |
| **OPC UA** | OPC Unified Architecture |
| **OS** | Operative System |
| **OT** | Operational Technology |
| **PLC** | Programmable Logic Controller |
| **RA** | Reference Architecture |
| **RAS** | Reliability, Availability and Serviceability |
| **REST** | Representational State Transfer |
| **RPN** | Risk Priority Number |
| **ROI** | Return of Investment |
| **SCADA** | Supervisory Control And Data Acquisition |
| **SME** | Small and Medium Enterprise |
| **SPARQL** | Simple Protocol and RDF Query Language |
| **TSN** | Time Sensitive Network |
| **XML** | Extensible Markup Language |

# 1 INTRODUCTION

## 1.1 Purpose and Scope of this Deliverable

This document defines and describes Z-BRE4K's architecture that all partners will implement and apply based on the defined strategies, use cases, and technological objectives. It includes aspects related to the identification of the major system components, how they interact with each other and how will the final Z-BRE4K structure will be. AUTOWARE RA will be exploited and extended through different components and enablers in order to adapt a cognitive manufacturing framework to predictive maintenance. The architecture is the most crucial part of the project since it provides a standard that fulfils all functional and non-functional requirements. The scope of this Deliverable is the understanding of a Z-BRE4K General Architecture which can be applied to the different use cases that take place during the project and future use cases from any industrial environment.

## 1.2 Content and Structure of this Deliverable

The deliverable is organized as follows:

- Section 2 describes the maintenance requirements the general architecture must fulfil.
- Section 3 describes AUTOWARE's Reference Architecture in which the Z-BRE4K General Architecture will be based, explaining the different enablers and APIs it will need.
- Section 4 describes the methodology taken to adopt AUTOWARE's RA to Z-BRE4K's predictive maintenance requirements and the extensions that will be implemented to accomplish it.
- Section 5 describes the components distribution and information flow between each other to be able to perform basic predictive maintenance, including the role of each of them in such performance.
- Section 6 will feed from the previous sections in order to end up with the Z-BRE4K General Architecture by exploiting and extending the AUTOWARE RA, demonstrating a General and a Use-Case Particular information workflows.

# 2 PREDICTIVE MAINTENANCE STRATEGIES REQUIREMENTS

Z-BRE4K projects looks to implement predictive maintenance strategies to avoid unexpected breakdowns thus increasing the uptime and overall efficiency of manufacturing scenarios. To this extent, several hardware and software solutions, will be implemented in three industrial demonstrators, adapting to the particular needs of each one.

In particular, Z-BRE4K will deliver a solution composed of eight scalable strategies at component, machine and system level targeting:

1) the prediction occurrence of failure (Z-PREDICT),
2) the early detection of current or emerging failure (Z-DIAGNOSE),
3) the prevention of failure occurrence, building up, or even propagation in the production system (Z-PREVENT),
4) the estimation of the remaining useful life of assets (Z-ESTIMATE),
5) the management of the strategies through event modelling, KPI monitoring and real-time decision support (Z-MANAGE),
6) the replacement, reconfiguration, re-use, retirement, and recycling of components/assets (Z-REMEDIATE),
7) synchronizing remedy actions, production planning and logistics (Z-SYNCHRONISE),
8) preserving the safety, health, and comfort of the workers (Z-SAFETY).

The Z-BRE4K solution implementation is expected to have a significant impact, namely:

- increase of the in-service efficiency by 24%,
- reduced accidents,
- increased Verification according to objectives,
- 400 new jobs created and over €42M ROI for the consortium.

In order to implement these strategies and reach these impact results, data coming from machine components, industrial lines and shop floors will be fed in the Z-BRE4K platform, which is featured by a communication middleware Operative System, a semantic framework module, a dedicated condition monitoring module, a cognitive embedded module, a machine simulator to develop digital twins, an advanced Decision Support System (DSS), FMECA module and predictive maintenance module, together with a cutting-edge vision H/S solution for manufacturing applications associated to advanced HMI.

The General Architecture must be able to support all the components developed under the Z-BRE4K project which lead to fulfilling the predictive maintenance strategies, being able to keep the information flow constant and well distributed between all the components. At the same time, it must permit an easy implementation in each Use Case Scenario (Gestamp, Philips and SACMI), leading the way towards each particular architecture for each use case and, in the future, different scenarios from other industrial systems. This means that the General Architecture must be highly flexible and easily adapted to new Use Cases, promoting the predictive maintenance towards its integration in SMEs.

Due to the high flexibility the architecture requires, the main communication middleware Operative System must be able to support a high number of different types of data coming from different types of sensors and control software.

At the same time, due to the high number of different components it must also support the need of a continuous communication between all of them, the interoperability must reach top-notch levels.

The solution has been an Open Source based software approach with a high interoperability and flexibility. The use of AUTOWARE's Reference Architecture and expanding it to fulfil the predictive maintenance requirements has been the main goal of the task T2.1.

# 3 AUTOWARE REFERENCE ARCHITECTURE

The Z-BRE4K architecture has been designed and developed **on the foundations of the AUTOWARE reference architecture and building blocks** enabling the convergence of Information Technology (IT), operational Technology (OT), Engineering Technology (ET) and the leveraging of interoperability of Industrial Data Spaces (IDS), for the support of a factory ecosystem. The objective is to develop a highly adaptive real-time Machine (network of components) Simulation platform that wraps around the physical equipment for the purpose of predicting uptimes and breakdowns – thus creating intuitive maintenance control and management systems.

Z-BRE4K aims at creating predictive maintenance strategies to avoid unexpected breakdowns, which implies an update of the reference architecture chosen towards such objectives. The main basis of the Z-BRE4K General Architecture will be that from AUTOWARE, but with some extensions to fulfil the predictive maintenance aimed by the project. The necessity to perfectly understand the AUTOWARE Reference Architecture and Blocks capabilities is a fundamental first step to understand the Z-BRE4K General Architecture.

The AUTOWARE Open OS was selected as Z-BRE4K's framework for cognitive CPPS service development and strategy implementation since the OS components have been piloted quite successfully in the past[1] and more importantly, AUTOWARE has been designed specifically with SMEs in mind. These two elements will allow Z-BRE4K strategies to be easily integrated over legacy machines and IT systems with minimum interference while SMEs will be able to easily integrate advanced predictive maintenance strategies in the very same IT framework used to deal with production optimization or zero defect manufacturing processes.

AUTOWARE Open Digital Automation OS has support for digital automation and full industry 4.0 support from shop-floor to the cloud. Its reference architecture rooted on solid foundations and intensive large scale piloting of technologies for development of cognitive digital manufacturing in autonomous and collaborative robotics and for modular manufacturing solutions based on RAMI 4.0 industry 4.0 architecture. AUTOWARE provides the necessary components and interfaces for:

- Extension of automation and control equipment for appized smart open control hardware (open trusted platforms) operation.
- Support to resilient and reliable time sensitive industrial (wireless) control communication and data distribution.
- Leveraged by OPC and rt-TSN edge computing and control framework.
- Access to cloud-based engineering and simulation capabilities.
- Smart service development for added value cognitive industrial services.

[1] more than 250 advanced manufacturing experiments (www.i4ms.eu)

## 3.1 AUTOWARE building blocks

### 3.1.1 Overview

The main components of the AUTOWARE framework which have been considered for the Z-BRE4K architecture are:

- AUTOWARE Reference Architecture.
- AUTOWARE Enablers (from FIWARE).

#### 3.1.1.1 Reference Architecture

The interesting aspects from the AUTOWARE Reference Architecture (RA) and main reasons it has been chosen for the Z-BRE4K since it aligns the cognitive manufacturing technical enablers, i.e. robotic systems, smart machines, cloudified control, secure cloud-based planning systems and application platform to provide cognitive automation systems as solutions while exploiting cloud technologies and smart machines as a common system. Since the goal of the AUTOWARE RA is to have a broad industrial applicability, map applicable technologies to different areas, and to guide technology and standard development, the Z-BRE4K Architecture will have the same capabilities.

The AUTOWARE RA has four layers/levels (see *Figure **1***), which target all relevant layers for the modelling of autonomous CPPS in the view of AUTOWARE:

- **Enterprise:** The enterprise layer is the top layer of the AUTOWARE reference architecture and encompasses all enterprise's factories.
- **Factory:** At the factory layer, a single factory is depicted. This includes all the various workcells or production lines available for the complete production.
- **Workcell/Production Line:** The workcell layer represents the individual production line of cell within a company. Nowadays, a factory typically contains multiple production lines (or production cells), where individual machines, robots, etc. are located in.
- **Field Devices:** The field devices layer is the lowest level of the reference architecture, where the actual machines, robots, conveyer belt, etc., but also controllers, sensors and actuators are positioned.

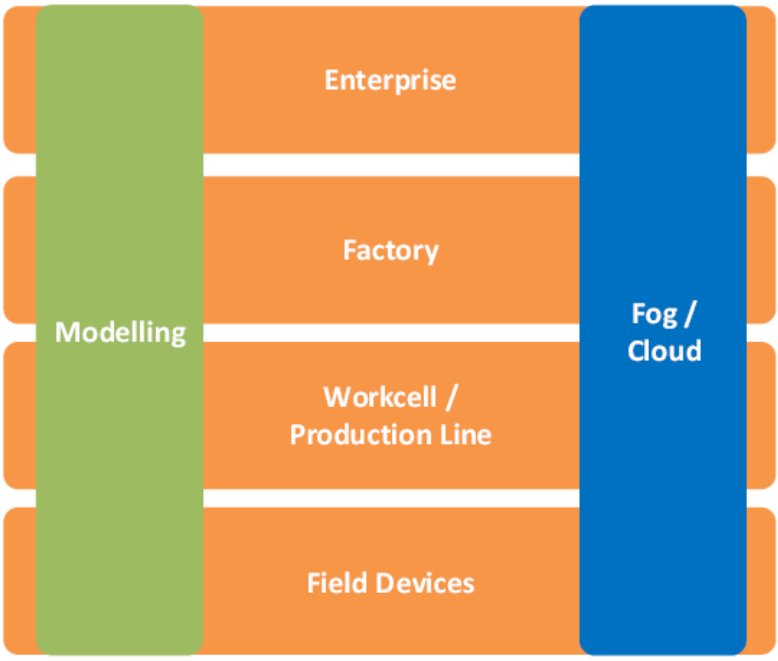


*Figure 1. AUTOWARE Reference Architecture*

As it was described in D1.3, the four layers are connected by two main pillars.

The first of them is the communication pillar named **Fog/Cloud** and uses wired (e.g. IEEE 802.1 TSN) and wireless communication to create direct interaction between the different layers by using Fog/Cloud concepts (blue column in *Figure **1***).

The second pillar, named **Modelling,** focuses on the modelling of the different technical components inside the different layers (green column in *Figure **1***). On each layer, different tools or technologies are applied and for all of them, different modelling approaches are available.

#### 3.1.1.2 Enablers

AUTOWARE provides a collection of enablers that facilitates the different users of the AUTOWARE Framework to interact with the system on different levels. A part from the enablers developed in the AUTOWARE project, there have been several international projects to promote the creation of new open source enablers for such an architecture. The most interesting ones have come from FIWARE and FITMAN and have been integrated into the AUTOWARE framework. Within AUTOWARE, there are three different enablers: technology, usability and Verification and Validation (V&V).

*Technology Enablers*

Within the AUTOWARE Framework, there is a collection of Technology Enablers, which can be identified as the technical tools, methods and components developed or provided within the AUTOWARE framework. Examples of technology enablers within the AUTOWARE project are e.g. robotic systems, smart machines, cloudified control systems, fog nodes, and secure cloud- and

fog-based planning systems as solutions to exploit cloud and fog technologies and smart machines as a common system.

In addition, to support connectivity and data management in CPPS, novel deterministic communication technologies and dynamic reconfiguration have been also considered as technology enablers within AUTOWARE. It also defines the integration and orchestration of various computing resources and CPPS trusted open platform at the scale of edge, cloud and High-Performance Computing (HPC). As a final technology enabler, AUTOWARE framework provides a reference implementation for a software defined autonomous services platform that can be applied to various industrial applications. This platform is based on fog node and cloud technology to provide the cognitive capabilities to the identified production systems.

*Usability Enablers*

Usability enablers are components that guarantee the usability of the AUTOWARE platform, technical enablers and tools by the manufacturing SMEs, resulting in an easy access and operation of the tools.

*V&V Enablers*

Although CPPS are defined to work correctly under several environmental conditions, in practice it is enough if it works properly under specific conditions. In this context, certification processes help to guarantee correct operation under certain conditions making the engineering process easier, cheaper and shorter for SMEs that want to include CPPS in their businesses.

In addition, certification can increase the credibility and visibility of CPPS as it guarantees its correct operation under specific standards. If a CPPS is certified to follow some international or European standards or regulation, it is not necessary to be certified in each country, so the integration complexity, cost and duration are highly reduced.

### 3.1.2 Main blocks

Once the complete AUTOWARE framework overview has been presented, the focus is on the most useful technological blocks for Z-BRE4K.

#### 3.1.2.1 Software Defined Autonomous Service Platform

Due to the recent development of numerous technical enablers (e.g. IoT, cloud, edge, HPC etc.), it is possible to take a service-based approach for many components of production information systems (IS). When using a service-based approach, instead of developing, deploying and running our own implementations for all production IS tasks, an external service provider can be considered and the end-user can rent access to the offered services, reducing the cost and knowledge needed.

AUTOWARE focuses on a service-based approach denoted as software defined autonomous service platform (in the following, also abbreviated as “service platform”) based on open protocols and implementing all the functionalities (physical, control, supervision, MES, ERP) as

services. As a result, the components can be reused, the solution can be reconfigured and the technological advanced can be easily followed.

The Z-BRE4K General Architecture will have a service-based approach so the Figure 2 will be of extreme importance when defining such architecture.

Figure 2 includes the reference architecture of the AUTOWARE service platform showing also how all the functionalities are positioned in the overall scheme of production IS. There are different functionalities (and therefore, services) on the different layers depending on the scope, but all of them are interconnected.

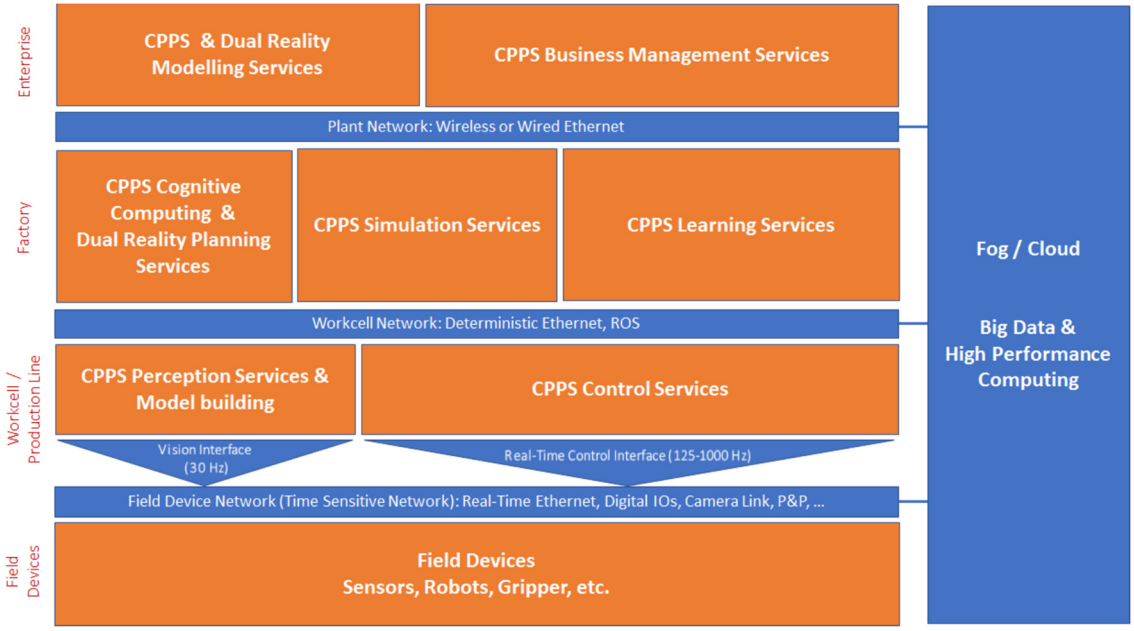


*Figure 2. AUTOWARE Software Defined Autonomous Service Platform*

### 3.1.2.2 Cloud Services Enablers

AUTOWARE considers several cloud services enablers for an easier implementation of the different services or functionalities. The use of these open source enablers permits the easier exchange of information and interoperability between different components and services, something really useful for future use cases. All of them have been developed inside FIWARE and FITMAN projects. The most useful enablers are:

- **FIWARE Orion Context Broker:** It produces, gathers, publishes and consumes context information. This will be the main communication system throughout all the Z-BRE4K architecture. It facilitates the exchange of context information between Context Information Producers and Consumers through a Publish/Subscribe methodology (see Figure 3*Figure 1*). This permits a high decentralized and large-scale context information management and high interoperability between the different components due to the use of a common NGSI protocol. The IDS architecture and connectors permit the use of such a powerful communication tool, making the use of IDS an extension of the AUTOWARE RA.

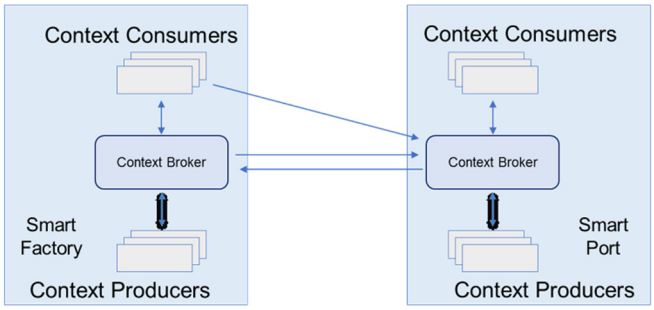


*Figure 3. Context Broker basic workflow*

Since it's a General Architecture what this Deliverable is aiming at, it won't be the only possible communication system thus it would miss its "generalness".

- **Backend Device Management – IDAS**: For the translation from IoT-specific protocols into the NGSI context information protocol considered by FIWARE enablers.
- **Cosmos:** For a more easy Big Data analysis over context integrated with most popular Big Data platforms and cloud storage.
- **WireCloud:** For the development of web components. It brings a powerful web mashup platform making it easier to develop operational dashboards which are highly customizable by end users.
- **DyCEP:** For Real-Time processing.
- **DyVisual:** For dynamic visualization and interaction.

As mentioned, all of this enablers were developed under FIWARE (Figure 4) and FITMAN projects.

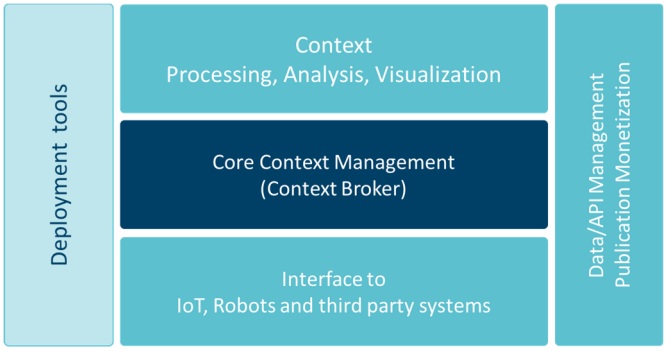


*Figure 4. FIWARE Architecture*

### 3.1.2.3 Fog Computing

AUTOWARE extended a cloud based architecture to a more flexible and efficient one based on fog computing, which is defined by the **OpenFog Consortium** as follows: "A horizontal, system-level architecture that distributes computing, storage, control and networking functions closer to the users along a cloud-to-thing continuum". This is of great relevance for Z-BRE4K due to the high number of cloud/fog computing components it will have.

The addition of an intermediate layer for Edge Computing and fog nodes were of great help to avoid bottlenecks and problems in more complex industrial scenarios, which could be future Z-BRE4K use cases. The following disadvantages have been eliminated thanks to such layer:

- Data bottlenecks that occur on the interface between IT and cloud infrastructure.
- Disability to guarantee pre-defined latencies in the communication.
- Sensor data are sent unfiltered to the cloud.
- Limited intelligence on the machine level.

Once the fog nodes have been placed, strict requirements on timing or even real-time constrains can only be achieved by avoiding long transmission of the data. Thus, the fog computing approach is inherently avoiding the latencies. This TSN is crucial in the lower layers where control interfaces requirements are around 125Hz – 1000Hz.

Figure 5 shows the embedding of the fog node into the AUTOWARE framework. Since Z-BRE4K will adopt the fog/cloud computing pillar (the blue column in *Figure 1*), it will have the following capabilities:

- Machine Control capabilities.
- Device Management capabilities.
- Data Gateway.
- Visualization capabilities.
- Application Hosting functionality.

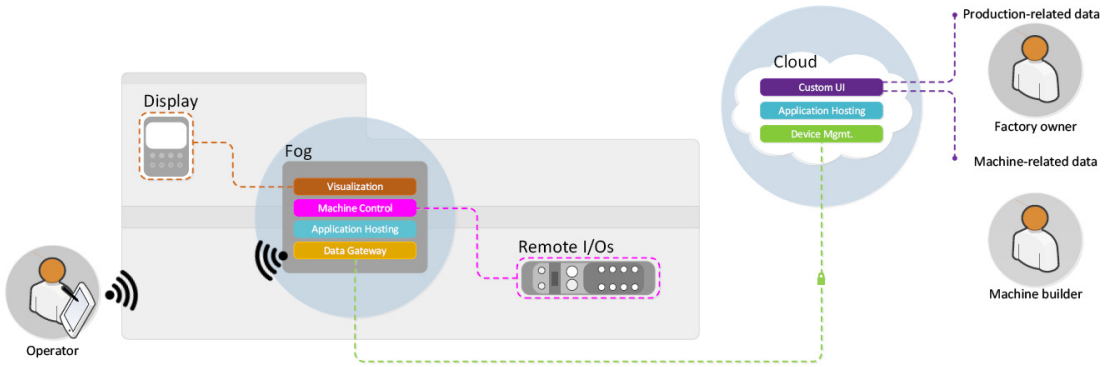


*Figure 5. Embedding of the fog node into the defined architectural framework*

The pillars of this architecture, which are common themes of the OpenFog reference architecture, include security, scalability, openness, autonomy, RAS (reliability, availability and serviceability), agility, hierarchy and programmability.

# 4 Z-BRE4K EXTENSIONS FOR AUTOWARE

AUTOWARE RA will have several extensions to be able to adopt its architecture to the predictive maintenance requirements. The Digital Shopfloor Alliance is an extension by itself of AUTOWARE which adds some crucial capabilities which Z-BRE4K will take advantage of.

The main extension will be a high trusted interoperability capability enabler that will permit a trusted information exchange through FIWARE's Orion Context Broker. Z-BRE4K architecture will then be the sum the Digital Shopfloor Alliance with the International Data Space Association (IDSA) which is the IDS architecture.


*Figure 6. Z-BRE4K Architecture Approach*

## 4.1 Digital Shopfloor Alliance

SMEs demand that digital automation can be provided at a cost, within short development cycles and be easily integrated "on-demand" for mission focused purposes. To respond to this demand, a significant effort needs to be provided in terms of providing

- The right framework for RoI analysis and technology selection.
- A reference framework for (cloud-edge automation) solution implementation.
- Open hardware platforms for dynamic automation app deployment.
- A marketplace for automation systems and solutions aggregation.
- A certification framework fast system integration and validated solution deployment.

The main purpose of the Digital Shopfloor Alliance is the creation of a full digital automation market offer available within SMEs constraints. No single project can build such vision in isolation, and only such synergies can create a positive market context; i.e. automation component providers and app developers covering the costs of certification on the basis of access to larger market shares with increased integration processes that can be now realized by system integrators at more competitive (lower) costs thanks to a more "structured" system property and behavior characterization of individual components, systems and SoS solutions.

With this vision in mind, none of the FoF-11 projects could build such attractive multi-sided market offer in isolation, yet all the components required are available and can be accommodated under the AUTOWARE digital automation framework, which will be transformed into a common Digital Shopfloor Framework to be further elaborated jointly.

The main idea has been to combine three international projects and architectures to fulfil such an ambitious objective:

- **DAEDALUS *(Distributed control and simulation platform to support an ecosystem of Digital automation developers):*** it enables the full exploitation of the CPPS concept, through the adoption of a completely distributed automation platform based on IEC-61499 standard in order to propose a functional model for CPS that blends coherently real-time coordination of its automation tasks with the "anytime" provision of services to other elements of the automation pyramid.

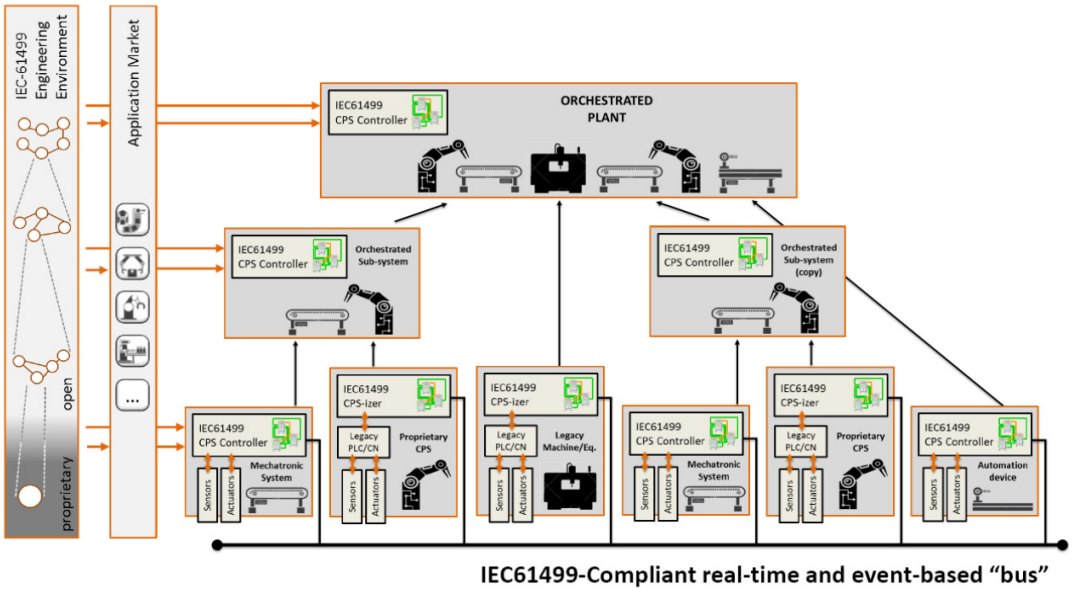


*Figure 7. Daedalus Architecture*

- **FAR-EDGE *(Factory automation edge computing operating system reference implementation)***: it suggests to move computing power where is actually needed, defining a decentralized automation architecture for the factory, resulting in a four-leveled architecture (field, edge, ledger and cloud) to create scalable and advanced manufacturing systems which implement automation techniques that deploy and reconfigure automation systems and production resources at the lowest possible cost. The Platform decentralizes monitoring, control and analysis of production processes, moving some key services down to the Edge level and thus helping the virtualization of the conventional automation pyramid.

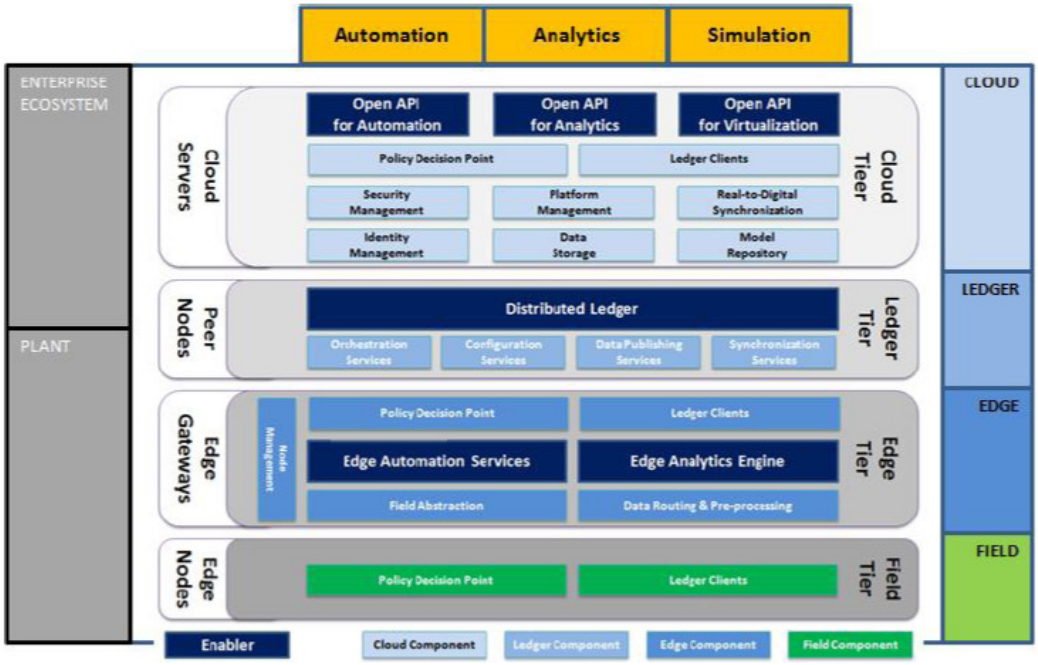


*Figure 8. Far-Edge Architecture*

- **AUTOWARE *(Wireless autonomous, reliable and resilient production operation architecture for cognitive manufacturing)***: as mentioned before in this Deliverable, it defines a complete open framework including a novel modular, scalable and responsive architecture for the factory automation, defining methods and models for the synchronization of the digital and real world based on standards and certified components. It also aligns several cognitive manufacturing technical enablers (mobile communications, data distribution, cognitive vision, flexible robotics, augmented virtuality, open fog, cybersecurity), which are complemented by usability enablers (CPPS engineering tools, real-time TSN network configuration, manual robotic tasks programming) of the platform by the manufacturing SMEs, making it easy to access and operate. In addition, to support connectivity & data management in CPPS, novel deterministic Ethernet technologies and dynamic reconfiguration have been developed.

The three projects (see Figure 9) provide a complete CPPS solution allowing SMEs to access all the different components in order to develop digital automation cognitive solutions for their manufacturing processes. AUTOWARE provides a complete CPPS ecosystem, including a reference architecture that perfectly fits with Far-Edge architecture based on splitting the computing in the field (considering the decentralized automated shopfloor defined inside Daedalus), the edge and the cloud. Daedalus also defines an intermediate layer (Ledger) to synchronize and orchestrate the local processes. Finally, AUTOWARE also enriches the different technical enablers to make easier the adoption of CPPS by SMEs as well as reliable communications and data distribution processes.

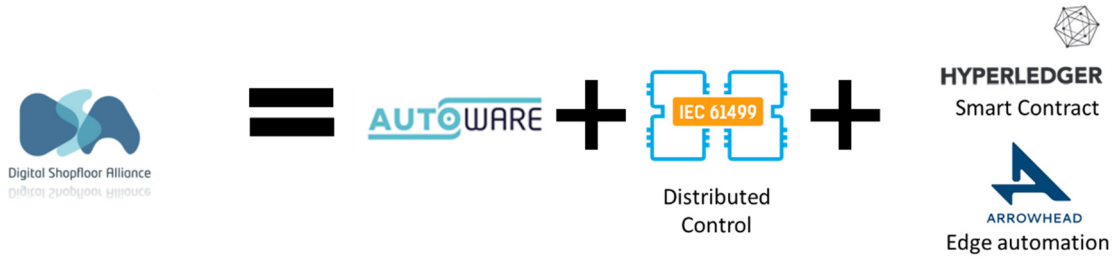


*Figure 9. Digital ShopFloor Alliance*

### 4.1.1 Digital Shopfloor value proposition

This combined solution reduces the complexity of the access to the different isolated tools significantly and speed up the process by which multi-sided partners can meet and work together. Moreover, the creation of added value products and services by device producers, machine builders, systems integrators and application developers will go beyond the current limits of manufacturing control systems allowing the development of innovative solutions for the design, engineering, production and maintenance of plants' automation.

The Digital Shopfloor Alliance for digital automation of SMEs delivers the following assets:

- A digital automation RoI analysis framework (AUTOWARE)
- A digital marketplace to aggregate IEC-61499 solutions & applications (DAEDALUS).
- Digital automation certification framework (AUTOWARE).
- Open fog & HPC infrastructure (AUTOWARE).
- Digital automation services & development community (FAR-EDGE).
- Digital automation ledger (FAR-EDGE).
- Automation system components, robotics, control, vision (DAEDALUS+AUTOWARE).

In fact, it responds to several stakeholders' needs (multi-sided ecosystem):

- Manufacturing SMEs can adopt digital cognitive automation systems within budget thanks to access to digital shopfloor marketplace. Furthermore, a fast development, experimentation and technical feasibility assessment of CPPS applications is ensured increasing their market visibility.
- Automation and machine tool/robot providers which can incorporate open CPS trusted platforms as part of an increased number of next generation smart production line components and solutions after component and system certification.
- Developers of cognitive (learning, analysis, knowledge management capability services) and automation apps for autonomous service support can benefit from certified deployment and well characterized performance in open fog platforms.
- Providers of cloud and HPC simulation and computation services can host the operation of advanced cognitive services to complement edge operations.
- Integrators and solution providers can build production line solutions for SMEs and OEMs faster and with more reliable components as this alliance provides a suitable

testing environment that ensures large scale validation of new technologies and services, including certified solutions for data storage as well as for communication.

The Table 1 below summarizes the identified socio-economic, as well as technological assets that could be contributed and aligned across initiatives to provide a compelling and appealing value proposition to the SME European market (initially) for an increased and faster digitization of the shopfloor.

*Table 1. Socio-economic and technology enablers of Digital Shopfloor Components*

| Digital Shopfloor Components | FAR-EDGE | DAEDALUS | AUTOWARE |
|---|---|---|---|
| Socio-Economic Enablers | | | |
| Multi-sided Framework | | | •• |
| Digital Automation App store | | •• | • |
| Reference Architecture | •• | | • |
| Certification Framework | | • | •• |
| RoI Business Case Framework | | | •• |
| Technology Enablers | | | |
| Distributed Control | | •• | • |
| OpenFog Node | | | •• |
| Edge-Analytics | •• | | |
| Ledger | •• | | |
| Arrowhead Edge | •• | | |
| Data distribution | | | •• |
| Mobile Communications | | | •• |
| Reconfigurable Cells/Cobots (ROS/DDS) | | • | •• |
| Modular Manufacturing (OPC-UA) | • | • | • |
| Augmented Virtuality & Visualisation | | | •• |
| HPC Engineering & Simulation | | | •• |

### 4.1.2 Digital Shopfloor Alliance Reference Architecture

The Digital Shopfloor Alliance is the combination of three projects with different architectures each. The pyramid-based architecture from Daedalus, the Automation/Analysis/Simulation vertical divisions from Far-Edge and the four-layer hierarchical division with fog/cloud interconnection from AUTOWARE is being a hard task. A first Reference Architecture can be seen in *Figure **10***.

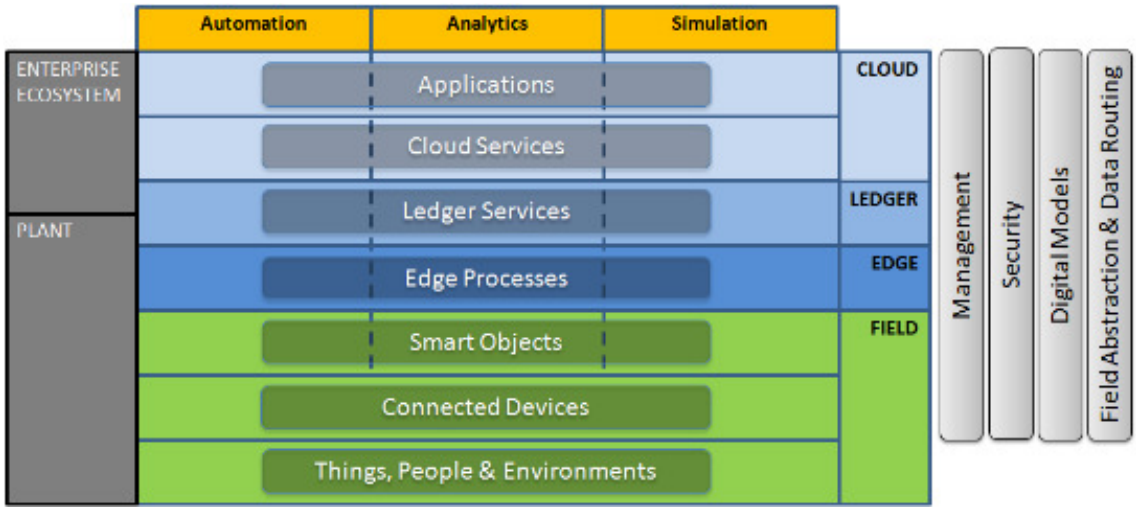


*Figure 10. Digital Shopfloor Alliance Reference Architecture*

## 4.2 IDS – Industrial Data Space

The Industrial Data Space foster secure data exchange among its participants, while at the same time ensuring data sovereignty for the participating data owners. The Industrial Data Space Association defines the framework and governance principles for the Reference Architecture Model, as well as interfaces aiming at establishing an international standard which considers the following user requirements:

- Data sovereignty.
- Data usage control.
- Decentralized approach.
- Multiple implementations.
- Standardized interfaces.
- Certification.
- Data economy.
- Secure data supply chains.

In compliance with common system architecture models and standards (such as ISO 42010, 4+1 view model, etc.), the Reference Architecture Model uses a five-layer structure expressing stakeholder concerns and viewpoints at different levels of granularity (see Figure 11).

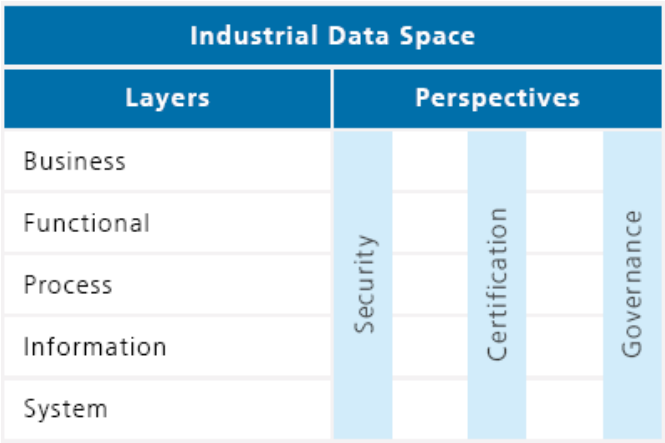


*Figure 11. General structure of Reference Architecture Model*

The IDS reference architecture consists of the following layers:

- The **Business Layer** specifies and categorizes the different stakeholders (namely the roles) of the Industrial Data Space, including their activities and the interactions among them.
- The **Functional Layer** comprises the functional requirements of the Industrial Data Space and the concrete features derived from them (in terms of abstract, technology-agnostic functionalities of logical software components).
- The **Process Layer** provides a dynamic view of the architecture; using the BPMN notation, it describes the interactions among the different components of the Industrial Data Space.
- The **Information Layer** defines a conceptual model which makes use of "linked data" principles for describing both the static and the dynamic aspects of the Industrial Data Space's constituents (e.g., participants active, Data Endpoints deployed, Data Apps advertised, or datasets exchanged).
- The **System Layer** is concerned with the decomposition of the logical software components, considering aspects such as integration, configuration, deployment, and extensibility of these components.

In addition, the Reference Architecture Model contains three cross-sectional perspectives:

- **Security:** It provides means to identify participants, protect data communication, and control the usage of data.
- **Certification:** It defines the processes, roles, objects, and criteria involved in the certification of hardware and software artifacts as well as organizations in IDS.
- **Governance:** It defines the roles, functions, and processes from a governance and compliance point of view, defining the requirements to be met by an innovative data ecosystem to achieve corporate interoperability.

### 4.2.1 System layer: technical components

The most interesting layer for the Z-BRE4K Architecture is the System Layer, where the roles defined in other layers (Business and Functional Layers) are now mapped onto a concrete data and service architecture in order to meet the requirements, resulting in what is the technical core of the IDS. From the requirements identified, three major technical components can be derived:

- Connector.
- Broker.
- App Store.

These are supported by four additional components, which are not specific to the IDS:

- Identity Provider.
- Vocabulary Hub.
- Update Repository (source for updates of deployed Connectors).
- Trust Repository (source for trustworthy software stacks and fingerprints as well as remote attestation checks).

### 4.2.2 IDS and FIWARE

The most interesting aspect about the IDS architecture is the opportunity to combine it with open source enabler. It is a common goal that a valid open source implementation of the IDS Architecture can be based on FIWARE, compatible with FIWARE Architecture principles.

The FIWARE Foundation is working towards making sure that core FIWARE Generic Enablers can be integrated together to build a valid open source implementation of the IDS Architecture

Both organizations are collaborating on the development of domain data models and communicating about the development of their respective specifications and architectures to keep them compatible.

The way FIWARE is being implemented within the IDS architecture can be seen in the following Figure 12.

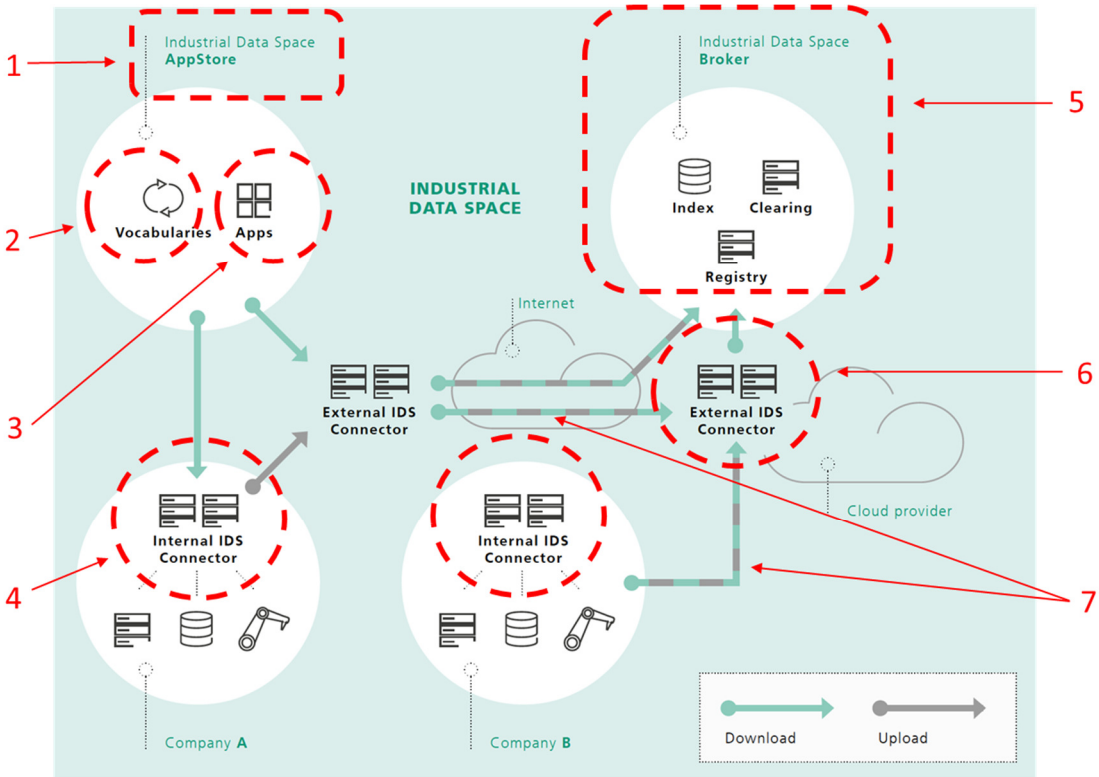


*Figure 12. Materializing the IDS Architecture using FIWARE*

- **1.** Docker-based tools relying on Docker Hub Services enabling automated deployment and configuration of Data Apps.
- **2.** Standard vocabularies are being proposed at https://www.fiware.org/data-models
- **3.** Data Apps map to NGSI adapters or Apps processing context information.
- **4.** Both External and Internal IDS Connectors are implemented using FIWARE Context Broker components.
- **5.** Extended CKAN Data Publication Platform.
- **6.** FIWARE Context Broker components will be used as core component of IDS Connectors. In Z-BRE4K, the Orion Context Broker from FIWARE will be used as the IDS Connector core component, enabling the implementation of IDS into the AUTOWARE RA.
- **7.** Interface between IDS connectors based on FIWARE NGSI.

# 5 Z-BRE4K Component Distribution and Role in Z-BRE4K Architecture

In D1.3, the final list of components that are used in each use case were presented, describing their functional requirements, inputs and outputs, associated strategies, supported APIs and providing a diagram of each of them. The General Architecture needs to support all these components and have the possibility of changing, eliminating or adding new components and modules that add new functionalities to Z-BRE4K.

This section will define the components' roles and services they carry out in the Z-BRE4K predictive maintenance workflow. Once they are defined, they will be placed in the AUTOWARE Service Platform Reference Architecture described in Section 3.1.2.1 and showed in *Figure **2***.

## 5.1 C-11 & C-12 IDS Connectors

The Industrial Data Space foster secure data exchange among its participants, while at the same time ensuring data sovereignty for the participating data owners. It facilitates the exchange of data between participants. The IDS Connector is basically a dedicated communication server for sending and receiving data in compliance with the IDS Connector specification.

As it has been specified in Section 4.2, some FIWARE Generic Enablers will be implemented in the IDS Architecture. This will permit the easy integration of the IDS Connectors into the AUTOWARE-based Z-BRE4K OS.

From a General Architecture point of view, the C-11 IDS Connector and the C-12 Orion Context Broker can be seen as one communication component since one is integrated inside the other. This component will be the communication method used between the system's components, becoming the main communication protocol through the Z-BRE4K General Architecture.

Since the C-12 Orion Context Broker inside the IDS uses NGSI protocol, the real-time machine-control services communications can't be done through this methodology, having to specify a different communication channel and network for the Time Sensitive Network. Due to the fact that these communications take place in the end user's facilities, they will be defined by such end users in every case.

## 5.2 C-01 i-Like Condition Monitoring

Condition monitoring module will be used to visualize data regarding the machine status. Data will be acquired form sensors installed around the machines specialized in gathering the intrinsic and extrinsic machine's key performance parameters, as well as the state of the components in order to visualize the relevant machine information.

Therefore, the main functionalities of the C-01 Condition Monitoring Module are:

- Visualize the machine conditions in order to act in short time: display real time signals from sensors (to be further agreed with end user) related to components health together with a clear identification of the critical threshold.

- Visualization of alarms and warning when the threshold for critical variables is approaching
- Get an overview of the components operative life and of the residual useful life calculated with predictive algorithm to support maintenance planning
- Possibility to retrieve historical data, production related KPIs and trends.

The main **inputs** will be:

- Data from sensors located on/near the industrial machines – if any - (End user)
- Data from PLC (End user)
- Data from legacy systems (End user)
- Data from operators (End user)
- Ontological descriptions of reference model and snapshots correlating machine parameters and system failures (C-03)
- Results from predictive algorithms to estimate critical components RUL (C-07)

And the main **outputs**:

- Machine/component status, alarms, warnings (User Interface)
- Critical components RUL (User Interface)
- Other relevant process information (User Interface)
- Data from sensors/PLC/legacy systems/operators (C07-C08)

The C-01's positioning in the workflow is clear, but if the general architecture is based in AUTOWARE's architecture, where, as said, it separates components by the services they carry out. The C-01 Condition Monitoring Module presented at Figure 13 will be in two different blocks:

- **Control Services:** Since it gathers the information directly from the machine and visualizes sensors' information in real time.
- **Business Management Services:** Since it is able to manage information and show alarms and critical issues regarding the machines' health. It also supports the maintenance planning system.

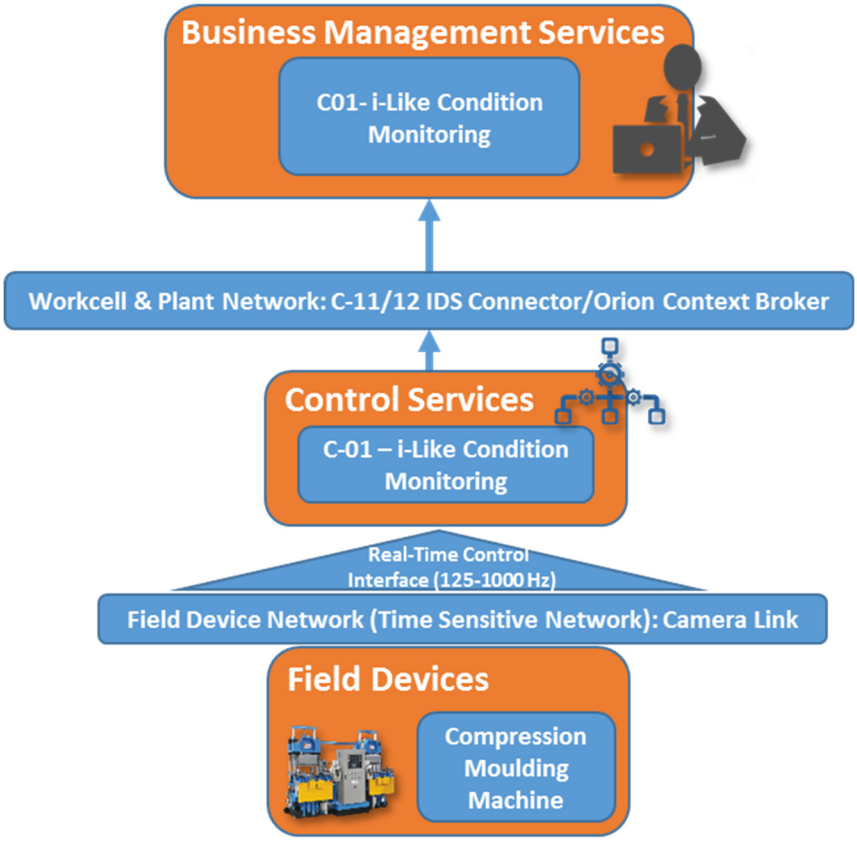


*Figure 13. C-01 i-Like Condition Monitoring*

## 5.3 C-02 Cognitive Embedded Condition Monitoring

The C-02 Cognitive Embedded Condition Monitoring will collect data from the sensors and will use machine learning techniques or expert knowledge-based systems to extract the most important features/measurements from the raw acquired data in order to avoid a bottleneck in the communication and storage systems and to ease the predictive systems to handle high dimensional data.

The Cognitive Embedded Condition Monitoring will be deployed throughout the factory shop floor and will be connected to the sensors/systems to acquire raw data from them. All the components of the CECM will be embedded systems (ARM, FPGA, GPU) that will run machine learning algorithms or expert knowledge-based systems to pre-process the data acquired. The system will upload the data to AUTOWARE IDS both pre-processed and structured in the best way so that the predictive component takes advantage of it.

It is a clear case of data in motion due to the fact that it pre-processes raw sensor data in real-time. It then implies that it must be connected through the Field Device Network (Video streaming in MWIR at 1000 Hz 128x128 pixels) directly to the machines (Gestamp's machine).

In the AUTOWARE block system, C-02 Cognitive Embedded Monitoring presented at Figure 14, is inside the **Control Service** block.

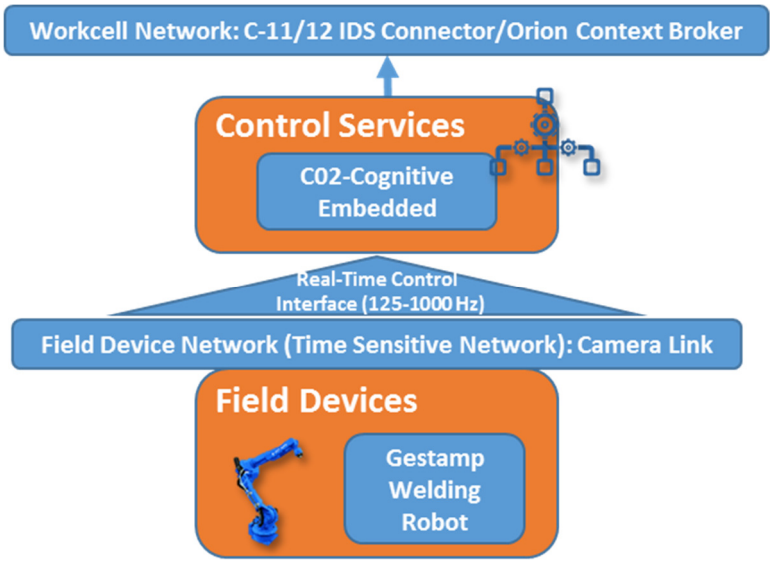


*Figure 14. C-02 Cognitive Embedded Condition Monitoring*

## 5.4 C-03 Semantic Framework

Z-BRE4K semantic model (i.e. Ontology) will be designed to serve as a common reference model for annotation and description of knowledge to represent manufacturing system performance. Mainly, the Semantic Framework will use data from the Z-BRE4K Repository, and the Semantic Data Web Services will provide an interface for the interaction with other components through the SPARQL query engine.

The main functionalities are:

- Serve as a common reference model for annotation and description of knowledge to represent manufacturing system performance.
- Describe the basic entities of Z-BRE4K and model relevant structures of manufacturing systems and processes.
- Meet the requirements of different stakeholders who need to have access to different aspects of machinery and process related information and knowledge.

Due to its main functionalities and purpose, the C-03 Semantic Framework presented on Figure 15, will be inside the **Learning Services** AUTOWARE Block, but to position it in the workflow, the inputs and outputs must be clear:

**Inputs:**

- Data from Z-BRE4K repository (data concerning machines, processes, production data logs, actors, and activities etc.), e.g. in XML.

**Outputs:**

- Semantic enrichment of shop-floor data for modelling not only the actors and procedures at the shop floor, but also machinery and their critical components, their

failure modes and their criticality, their signatures of healthy and deteriorated conditions, etc. e.g. as RDF Triplets (to be used by Z-BRE4K Knowledge Base System in T3.2).
- Semantic rules (e.g. SPARQL-based rules) for data browsing/analysis.

The C-03 Semantic Framework must receive the data from the Context Broker and send it to the different components to continue with the data treatment. This implies that that Learning Services Block must be able to receive data from the Context Broker and send it back to it so the Context Broker spreads the data with the new semantic to the components that require it.

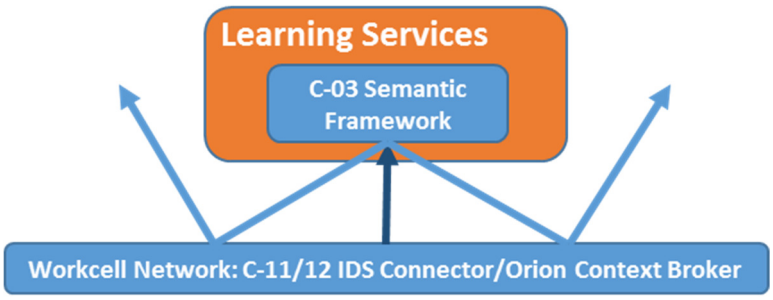


*Figure 15. C-03 Semantic Framework*

## 5.5 C-04 DSS

Z-BRE4K will use the DSS component presented on Figure 16, to assess the machines' performance and accurately diagnose and predict failures, deciding on actions to prevent their occurrence, suggesting strategies to activate at each time; and recommending actions to improve maintainability and operational efficiency at the shop floor, while increasing the remaining useful life of production assets and preventing unexpected BRE4Kdowns.

The following capabilities will be available through the DSS:

- Decision support towards increased maintainability and operating life.
- Incorporation / triggering of Strategies.
- Machine-learning based recommendations.
- Maintenance scheduler.
- 2nd level alarm filtering.
- Schedule updates communicated to MES, leveraging available Z-BRE4K system communication interfaces and protocols.
- Visualization of results.

Since the DSS has such a high number of different software applications, it may offer more than one service. The main role are the decision support services that are inside the **Business Management Services**, but it also has several rule-based cognitive computing and data acquisition management and processing capabilities, which includes it inside the **Learning** and **Cognitive Computing Services** Blocks in the factory level.

The information flow will be directly related with the Orion Context Broker which will be in charge of sending the inputs needed from the control services and other factory-level components and publishing the outputs so the control services and the user interfaces can recollect them and take the necessary actions with them.

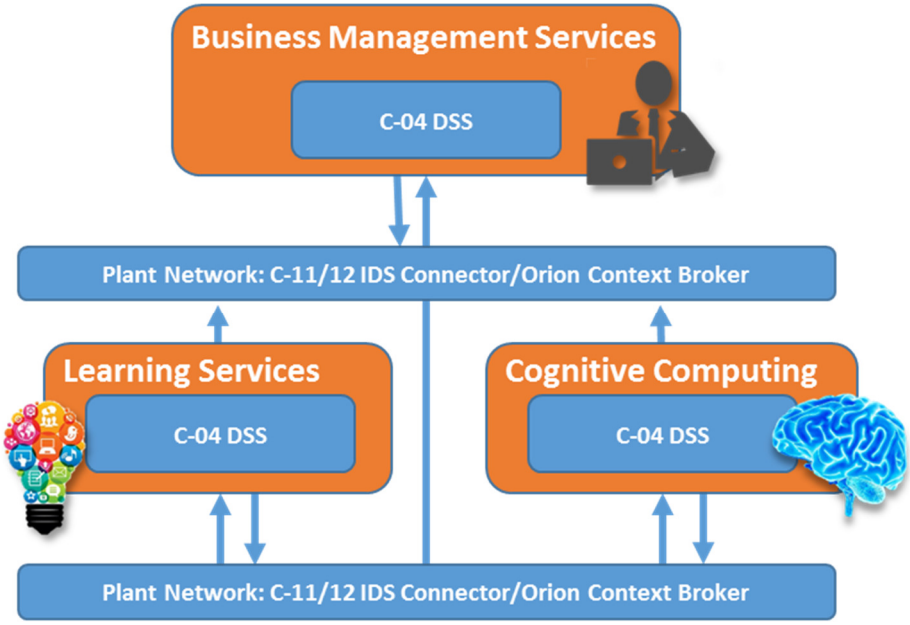


*Figure 16. C-04 DSS*

## 5.6 C-05 FMECA

Failure Modes and Effect Analysis (FMEA) is a systematic procedure for the analysis of a system to identify the potential failure modes, their causes and effects on system performance. FMEA generally deals with individual failure modes and the effect of these failure modes on the system. Each failure mode (FM) is treated as independent.

**FMECA (Failure Modes, Effects and Criticality Analysis)** is an extension to the FMEA to include a means of ranking the severity of the FMs to allow prioritization of countermeasures. This is done by combining the severity measure and frequency of occurrence to produce a metric called criticality (i.e. considering criticality combined with severity as a measure of risk).

Main inputs:

- Machinery systems (submachines, replaceable units, individual parts, etc.).
- FMs per machinery system.
- Severity and effects per FM.
- Failure data (i.e. in order tocalculate frequency of FMs).
- Time of component/system operation per FM (i.e. in order to calculate criticality number).
- KRIs, incl. secondary and tertiary factors.
- Tolerance level per KRI (min/max thresholds).

Main Outputs:

- Risks with calculated values.
- RPNs.
- Criticality numbers per FM.
- Criticality Matrix.
- Alerts (i.e. when a KRI breaches the min/max tolerance level).

Main functionalities:

- Risk (R) & Probability (P) of a FM occurrence builder.
- RPN & Severity & builder.
- Effects builder:
    - Identify those failures which have unwanted effects on system operation, e.g. preclude or significantly degrade operation or affect the safety of the user, and the sequences of events brought about by each identified item failure mode, from whatever cause, at various levels of the system's functional hierarchy.
    - A classification of identified failure modes according to relevant characteristics, including their ease of detection, capability to be diagnosed, testability, compensating and operating provisions (repair, maintenance, logistics, etc.);
- Potential causes of failure mode builder.
- Alerter (issues and dispatches alerts when a KRI breaches the min/max tolerance level).
- Criticality Matrix builder.

The main conclusion that can be done through a thorough analysis of the component is that it must be placed inside the Cognitive Computing Block due to its rule-processing nature. Many of the inputs don't come directly from the data gathered by the sensors, but from the machines' failure modes that have been analyzed and defined by the end users. Nevertheless, the DSS must specify the Failure Mode that will occur and publish it in the Context Broker so FMECA can receive it and know which information (Criticality, Risk…) it must send. The outputs are well defined and it's clear the use it's given to the FMECA component. Therefore, in the general architecture, the FMECA has no inputs since the information flow isn't considered for them.

The FMECA (Figure 17) will send information regarding risks and critical issues to the business management systems for the correct severity analysis and future action and decision support mechanisms.

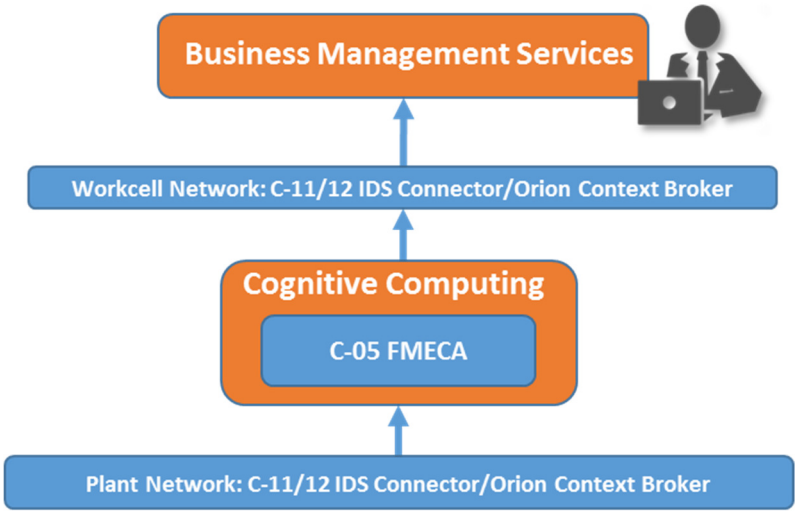


*Figure 17. C-05 FMECA*

## 5.7 C-06 VRfx

VRfx presented at Figure 18, is a tool for rapid creation of high quality VR visualizations. Based on geometric models from various data sources like CAD or animation systems, users can compose scenes using an easy-to-use graphical user interface. Materials and Shaders can be applied to any node in the loaded geometric models.

It will be used to support humans in understanding complex issues and context, visualization is seen as a suitable mean to tackle complexity and to generate an overview. Especially in technical areas, a promising approach is 3D-based visualization under application of Virtual Reality (VR). Thereby, immersion facilitates another way of access to data models and their understanding in an intuitive manner – so, to enable humans to concentrate on their technical task and be relieved of the challenge to understand and interpret the data itself.

Main inputs:

- Geometric data of the considered manufacturing equipment to be visualized.
- Information about the current state and related changes of the considered equipment.

Main outputs:

- Stereoscopic, immersive visualization to advance human understanding.
- Visualization of simulation results.

Placing this component in the AUTOWARE RA is not an easy task due to the high number of different services it can offer a high-tech visualization application. Since it is more an interface for visualization of results than a service for automatization it doesn't fit perfectly in any block.

The most interesting services it can be attached to are the **Perception Services**, due to the fact that it gathers visual information from the machines and it can be really useful for dual reality simulations visualization at the highest possible layer (**CPPS & Dual Reality Modelling Services**

Block), receiving the different models and visualizing them in a complete virtual reality scenario for different purposes.

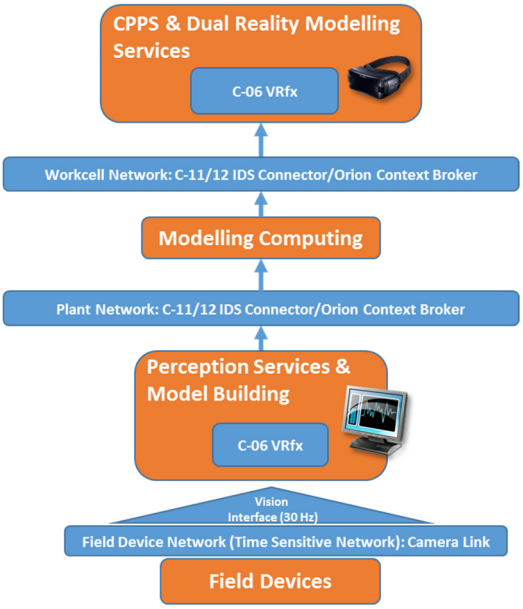


*Figure 18. C-06 VRfx*

## 5.8 C-07 Predictive Maintenance

The component Predictive Maintenance (Figure 19) is an event analysis method to find the causes of failure/BRE4Kdown, evaluate the impact and eliminate the potential causes of sudden BRE4Kdowns by understanding the events that lead to BRE4Kdowns. By detecting the chain actions that will cause failure, finding solutions to avoid or alert operations for remedial actions. It provides a predicted failure mode, meantime between failure, Type of failure, and possible cause.

Main inputs:

- Historical information regarding critical component reliability, MTBF, and any other Maintenance information (Source: Maintenance Log Books OEM and operator).
- Detail reports on the procedures and actions taken with regard to Fault Detection, Fault Diagnosis, Prognosis and implementation (Source: Operator and OEM).
- Type of BRE4Kdowns, severity, risk, service time. (Source: OEM, Z-Prevent, Z-Manage).
- Current preventative maintenance plans (Source OEM and Operator).
- Reports on causal relationships between component states and type of failure (Source: OEM and Operator).

Main outputs:

- Predicted Meantime between failure.
- Predicted Service Time.
- Prognosis.

Predictive Maintenance carries out two different services to elaborate the predictions: on one hand it has a number of rules and bases installed by the end users (operators and OEMs) which define the different Machine Faults and Failures, historical data and maintenance plans, placing it in the **Cognitive Computing** Block, and on the other hand, the huge amount of data coming from the field devices (which has been gathered by the control services components and homogenized by the Semantic Framework) and from the simulations carried out by the Simulation Services Block to carry out the predictions, placing it also in the **Learning Services** Block.

The Information Flow will be simple, receiving pre-processed data from the Work-cell layer through the Work-cell Network (Orion Context Broker) and from the Simulation Services, also through the Orion Context Broker and sending the outputs through the Plant Network (also the Orion Context Broker) to the Business Management Services for their analysis, action and decision taking processing.

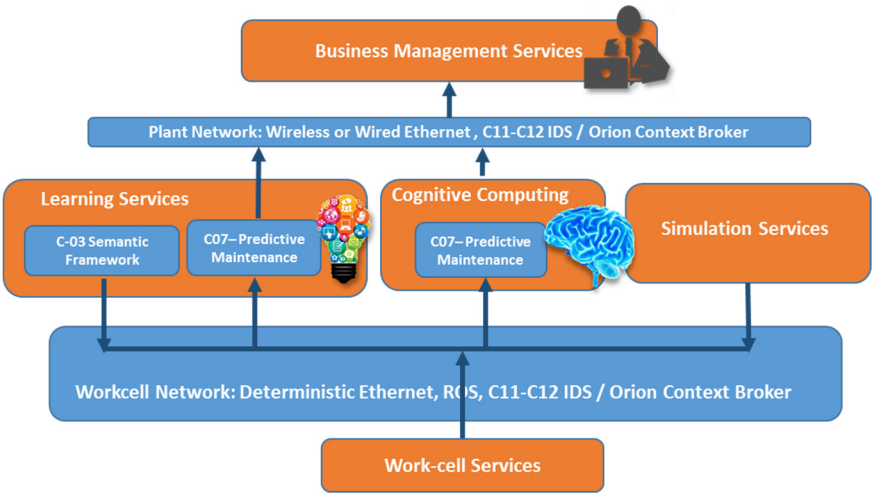


*Figure 19. C-07 Predictive Maintenance*

## 5.9 C-08 Machine Simulators

The machine simulator works side by side with the C-06 Predictive Maintenance component (Figure 20), providing visual and the necessary illustration of machine state. Using mathematical models and logics the function is to display the current state of the machine with respect to potential failures (alarms) and causes of failures.

As the name will suggest, the C-08 Machine Simulators will be placed in the **Simulation Service** Block from the AUTOWARE Reference Architecture.

To define the Information Flow, its outputs are used by the C-06 Predictive Maintenance to elaborate the failure prediction capabilities explained in the previous point. At the same time, its outputs are sent to the Business Management Services for their analysis and decision making features, the same way the Predictive Maintenance does. All these will be done through FIWARE's Orion Context Broker in the different use cases.

The inputs are:

- Detail CAD/CAE details of the machines and the critical components (source: OEM Repository).
- Detailed engineering electro/mechanical and electronics properties and specifications of the machines (Repository: OEM and Machine Operators e.g. users).
- Existing embedded (sensors/actuation/controllers) Data Acquisition systems for condition monitoring of the machines. If not existing suggestions for mounting such equipment (sensors mainly) on the machines. (Source: OEM).
- Control Area Network specifications (e.g. Profibus, Modbus, devicenet…) (Source Operator and OEM).
- The specifications of the Machine Network SCADA systems (Source: Operator and OEM).
- Any existing Machine and Production Process Simulation files and details. (Source: OEM).
- Overall specifications and technical set up of the real-time data management systems (Source: OEM).
- Production line data (Network of machines in action) (Source Operator).

This input data in this case will come from the work-cell layer services, passing previously through the C-03 Semantic Framework for its semantic transformation if necessary and desired, through the Orion Context Broker.

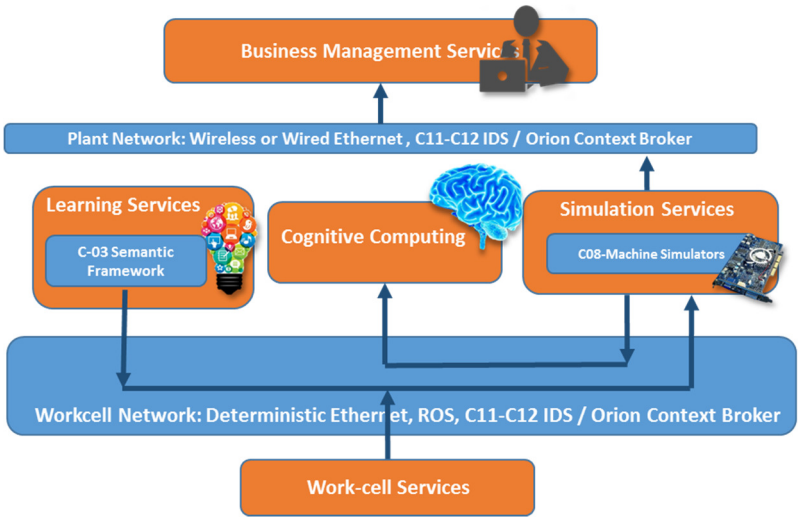


*Figure 20. C-08 Machine Simulators*

## 5.10 C-09 M3 Gage

M3 Gage (Figure 21) is a dimensional inspection system specifically designed for in-line inspection. It digitizes parts and obtains a virtual part with high accuracy from which different data that can later be obtained (geometries, CAD comparison, dimensional deviations…). M3 gage brings a complete integration, connectivity and traceability in just one workflow.

The M3 Gage is a **Field Device** which will scan the object and will send an XYZ cloud of points for its analysis. The information flow will be in real time to the C-10 M3 Software where the model building will take place.

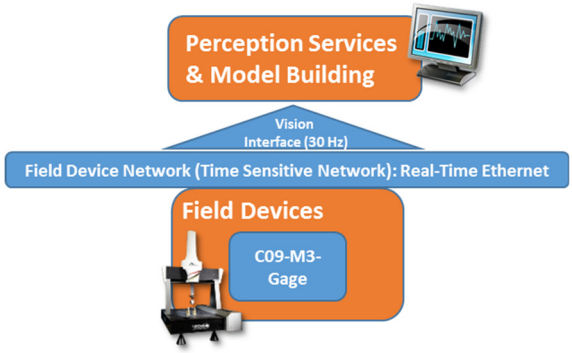


*Figure 21. C-09 M3 Gage*

## 5.11 C-10 M3 Software

M3 is the name given to the platform that will serve to integrate different data sources, mainly from M3 gage systems (in Z-BRE4K in line dimensional inspection systems). The M3 platform at Figure 22, serves as a data repository as well as allows the interconnection of information from various sources using a format that can be used by the M3 Analytics module. The platform will also provide computing mechanisms to process the data it contains and generate additional information.

M3 Analytics module (M3 Dashboard, M3 Statistics, M3 Reports) is a powerful tool that enables the visualization, the statistics analysis and the reporting operations related to all the data stored in the cloud by means of several algorithms and computational components. As this tool makes use of the memory and computational resources available in the cloud, it is possible to use it anytime and anywhere and by means of a simple computer or tablet with low computational capacities.

The main inputs will be the XYZ cloud of points generated by the C-09 M3 Gage and the main outputs will be:

- 3D representations
- Recommendations.
- Notifications.
- Reports.
- Histograms

Since it generates a 3D model from the M3 Gage cloud of points and it's connected to it in real time through the Time Sensitive Network (Real-Time Ethernet), the M3 Software must be placed in **the Perception Services & Model Building** Block. At the same time, it does several cloud/fog computing and business recommendations so it will also appear in the Business Management

Services Block. Finally, the CAD and CAM models built by this software will be placed in the CPPS & Dual Reality Modelling Services.

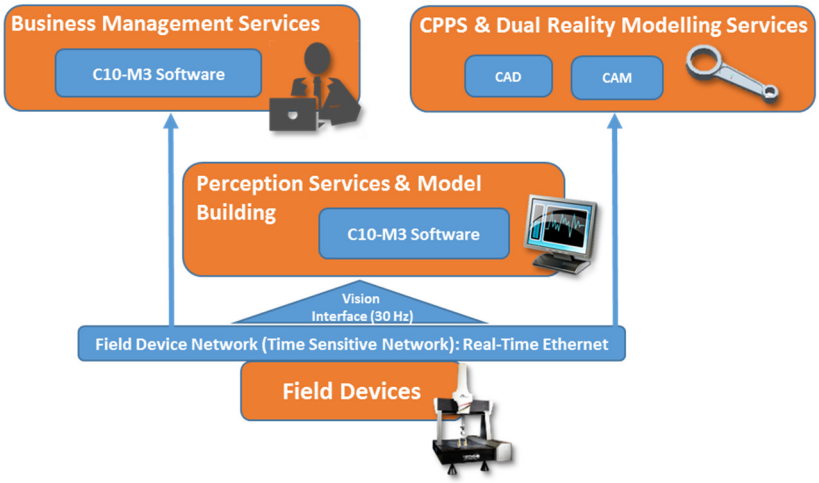


*Figure 22. C-10 M3 Software*

# 6 Z-BRE4K General Architecture

## 6.1 Z-BRE4K General Architecture Structure

The Z-BRE4K General Architecture will be a combination of the AUTOWARE RA from *Figure **1*** with a vertical separation definition included in the Digital Shopfloor Alliance Reference Architecture from *Figure **10*** and the integration of the IDS General Architecture from *Figure **11*** by using FIWARE Generic Enablers as IDS core components.

The main result is shown in *Figure **23***, where the Z-BRE4K Automation, Z-BRE4K Analytics and Z-BRE4K Simulation are presented following the Far-Edge Architecture principles envisioned in the DSA Reference Architecture.

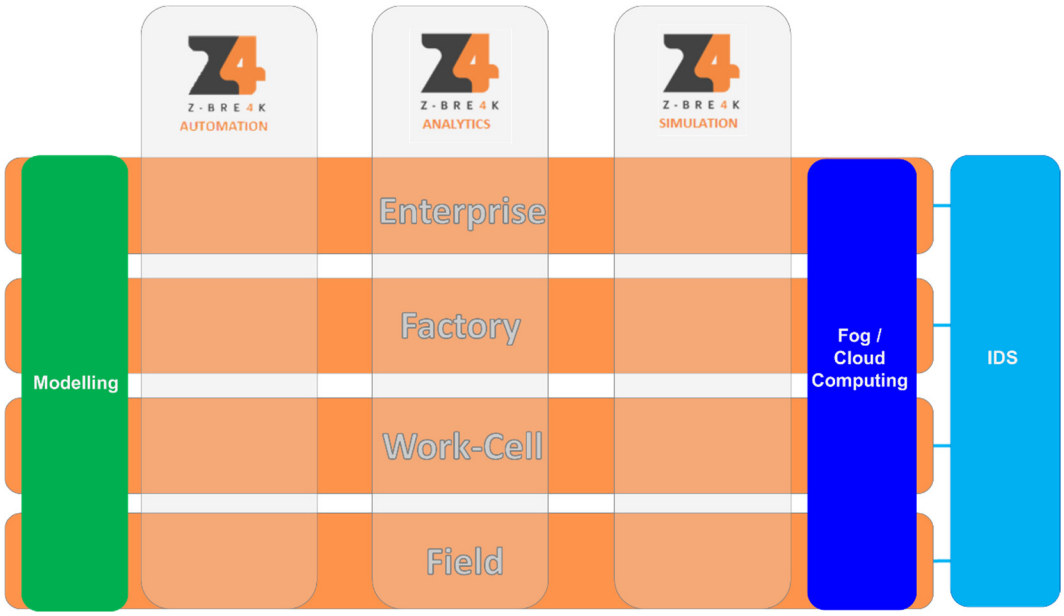


*Figure 23. Z-BRE4K General Architecture Structure*

## 6.2 Z-BRE4K General Architecture Information Workflow

Since the predictive maintenance Z-BRE4K is aiming at has been envisioned as a service, the General Architecture will adapt AUTOWARE Service Platform Reference Architecture to the Z-BRE4K structure as shown in *Figure **24***.

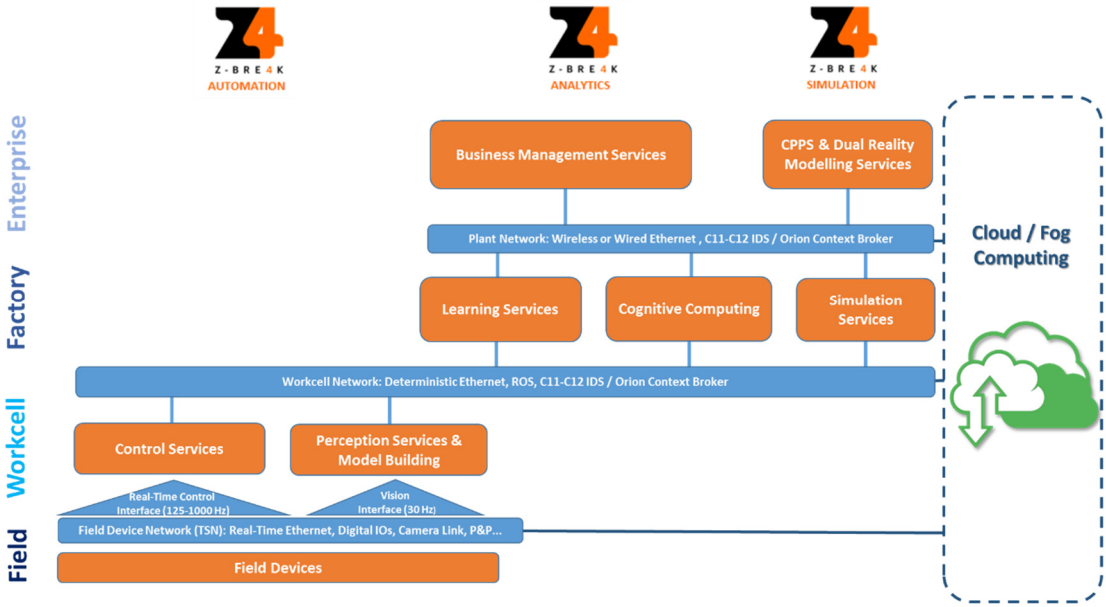


*Figure 24. Z-BRE4K General Architecture Connections*

*Figure* ***24*** shows the different services divided into the AUTOWARE different blocks and layers, all of them interconnection through the blue Networks. The main work cell and plant network will be done through the IDS Connector and FIWARE Orion Context Broker principally, but not necessary, so other communication methodologies are proposed, to be able to adapt the architecture to any future use case implementation.

The Fog/Cloud interconnection is always available through the fog nodes described in Section 3.1.2.3. This will permit the use of storage, HPC and Deep Learning FIWARE Generic Enablers for better computing and calculating processes.

The principal information workflow will be as shown in *Figure* ***25*** where it's clearly shown that the Orion Context Broker is in charge of all the communications and data spreading throughout the higher computing layers.

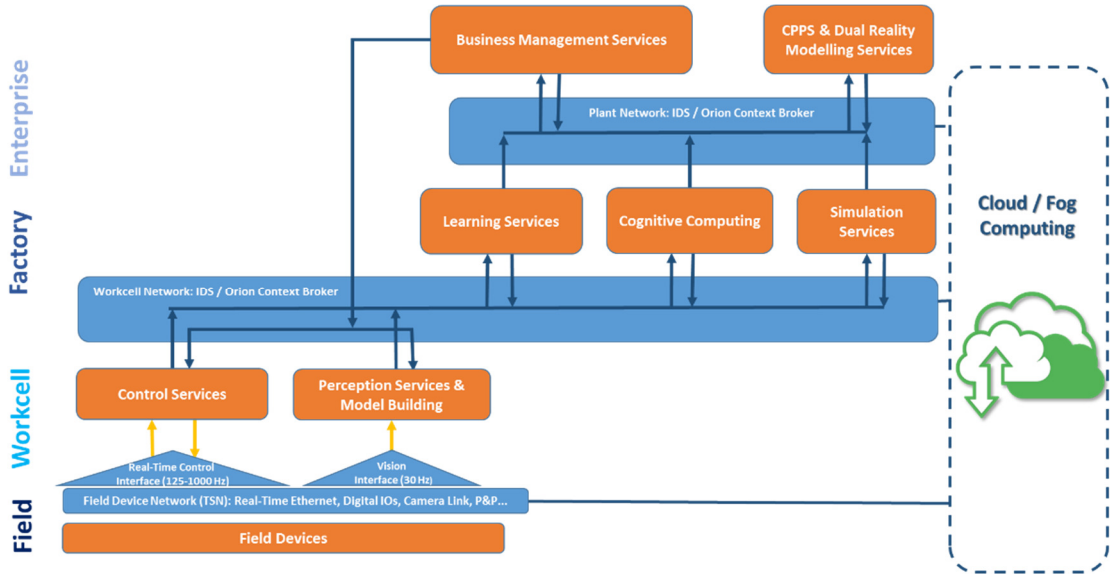


*Figure 25. Z-BRE4K Principal Information Workflow*

The information captured by the Field Devices (sensors, machines…) is sent through the Time Sensitive Network (TSN) located in the end users facilities to the Control Services and Perception Services & Model Building components in Real-Time.

The next step is, through the IDS Connectors connected to the Workcell layer components, the data (normally preprocessed by the Workcell components) is sent to (published) the Orion Context Broker. The different components from the factory layer that are subscribed to each data set will receive it for their analysis and processing. The factory services components, which are divided into Learning, Simulating and Cognitive Computing Services, may require processed data from another factory layer service. The outputs from factory layer components that are required as inputs by other factory layer components will be published once again in the Orion Context Broker in the workcell. The factory layer components that need those outputs as inputs will be subscribed to that data and will receive it. That's how the communication and information flow will be carried out through the different hierarchical levels.

The Learning, Simulating and Cognitive Computing Services will end up creating valuable information as outputs that will be published in the Plant Network's Orion Context Broker. The different Business Management Services will recollect the information required as inputs for their processing and will elaborate reports, actions, alerts, decision support actions… Dual Reality and Modelling Services will also gather information and will process it to give extra support information for Business Management decision making and user interfaces by publishing it back in the Plant's Orion Context Broker.

The Business Management Services will be able to send information to the Control Services for user interface issues or optimization actions if necessary.

## 6.3 Z-BRE4K General Architecture Component Distribution

Following the Z-BRE4K General Architecture Service Block division from *Figure* ***24*** and the component distribution defined in Section 5, the final Z-BRE4K General OS will be as shown in *Figure* ***26***.

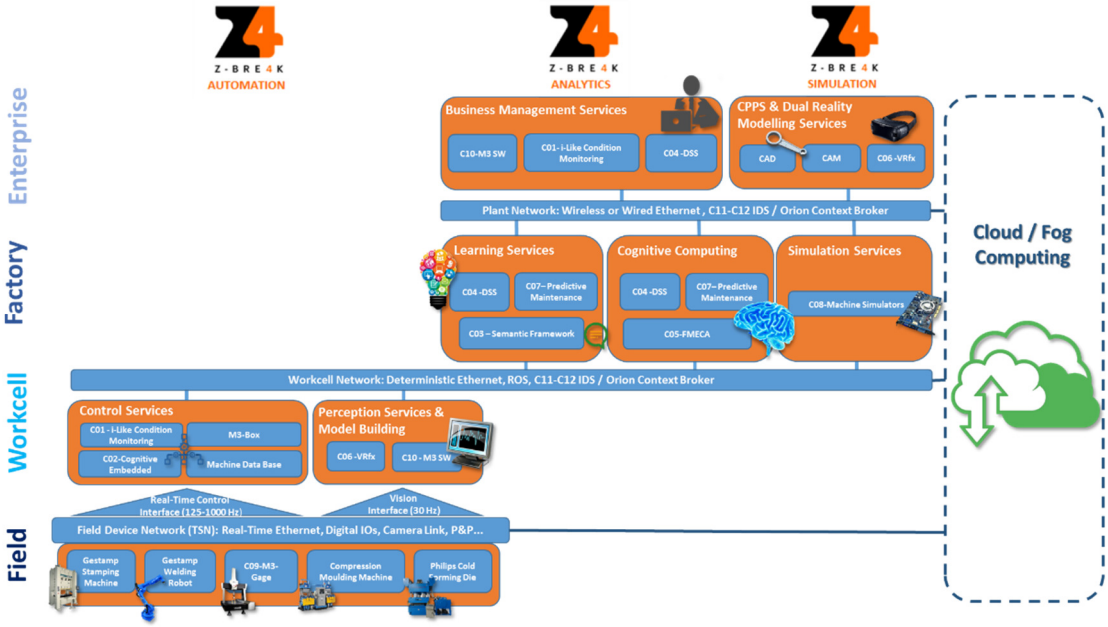


*Figure 26. Z-BRE4K General OS*

This Reference Architecture englobes all the components that are being developed under the Z-BRE4K project. Due to the different use case scenarios, not all the components will be used in each use case. The main purpose of this Reference Architecture is to be valid for any use case, from the three main use cases under the project's scope to future use cases of any industrial scenario.

To be able to apply this Reference Architecture, a series of steps must be taken by any use case:

1. Define the Subsystem/Component that we want to monitor.
2. Define the Failure mode(s) that we want to monitor.
3. Define the Signals (measurements) which could be useful to monitor/predict the specific Failure modes(s).
4. Define the Sensors which are going to provide the specific signals.
5. Define the Blocks (orange boxes) from the general architecture required on each use case.
6. Define the Components (blue boxes) inside each Block that will be used in each use case.
7. Define if needed any extra Component of your own to add to for the use case.
8. Define the communication protocol that will be used in each step.

The first four steps must be carried out by the end users since they are the ones with the machine and process knowledge to correctly define such applications. The other four steps are

more software and hardware definitions regarding analysis and processing components and communication protocols, which depending on the desired predictive maintenance may vary.

## 6.4 Use Cases Analytics and Simulation Component Information Workflow

The three use cases have different ways of gathering the information from the different machines they will use, including some pre-processing capabilities from the different C-01 i-Like Condition Monitoring and C-02 Cognitive Embedded Condition Monitoring, but once the data has been published in the Orion Context Broker, the idea of how to achieve the predictive maintenance by using the components developed under the project's scope is unique.

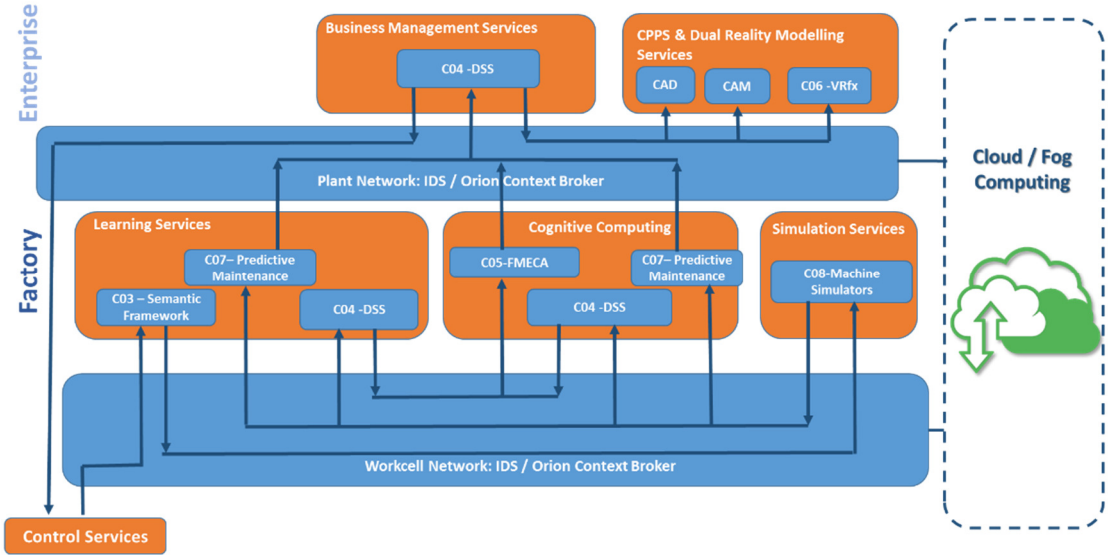


*Figure 27. Use Cases Particular Information Workflow*

The information in the particular use cases presented on Figure 27, for the predictive maintenance will go as follows:

1. The information is gathered by the field devices, pre-processed if necessary by the control and model building services and published in the Orion Context Broker through each use cases' IDS Connector.
2. The data is collected by subscription by the C-03 Semantic Framework, where it is given the semantic structure and stored in a DB (fog/cloud computing most probable). Then it's published again in the Context Broker.
3. Data is used to feed the C-08 Machine Simulators.
4. Prediction algorithms (from the C-07 Predictive Maintenance) are run through the C-08 outputs.
5. The C-04 DSS gathers information from the C-07 Predictive Maintenance and analyses it, giving as an output the failure mode.
6. The C-05 FMECA gets the failure mode from the DSS through the context broker.

7. FMECA returns Criticality, Risk, Redundancy… for the specific Failure Mode to the DSS through the Context Broker.
8. The DSS, based on the Rules set, provides Recommendations to the Technicians through a common User Interface and control services.
9. The Technicians can use the C-06 VRfx for the better understanding of the information.
10. The Technicians take Actions on the assets through the control services based on the recommendations given.
11. The Technicians provide Feedback on the accuracy of the Recommendations given by the DSS.
12. The DSS improves its Rules and Recommendations based on the Feedback received.

# 7 CONCLUSIONS

The Deliverable 2.1 describes the main results of task T2.1, which has been the definition of the General Architecture and Operative System for Z-BRE4K Prediction Maintenance. It has been demonstrated that it's possible to develop a reference architecture which fulfils the modular, high interoperability and open source API-based structure the predictive maintenance strategies envisioned under the Z-BRE4K project require and within SMEs' reach.

The combination of the AUTOWARE RA, the Digital Shopfloor Alliance Reference Architecture and the IDS communication architecture ensure an easy implementation to different industrial scenarios and use cases. With this, the issue of creating such a holistic yet focused and usable solution has been solved, leading the way to the following tasks in the project.

# 8 REFERENCES


- [AUTOWARE] AUTOWARE D1.3 (ARCHITECTURE)] http://www.AUTOWARE-eu.org/
- [ISO/IEC/IEEE] ISO/IEC/IEEE 42010:2011 https://www.iso.org/standard/50508.html
- [FIWARE Technologies enabling Industry 4.0] http://www.fiware4industry.com/?page_id=292
- [Z-BRE4K] Z-BRE4K D1.3 (SYSTEM ARCHITECTURE)] https://www.z-bre4k.eu/
- [MIDIH] MIDIH D4.2 (MIDIH Reference Architecture) http://midih.eu/project.php
- [SOFTWARE ARCHITECTURE PATTERNS, *MARK RICHARDS*], https://www.oreilly.com/programming/free/files/software-architecture-patterns.pdf
- [I4MS PROGRAMME], www.i4ms.eu
- [OPENFOG CONSOTIUM]; https://www.openfogconsortium.org/#consortium
- [DIGITAL SHOPFLOOR ALLIANCE], "A position paper towards the establishment of a European Digital Shopfloor Alliance"; https://digitalshopflooralliance.eu/