COGNITWIN

Cognitive plants through proactive self-learning hybrid digital twins

DT-SPIRE-06-2019 (870130)

Deliverable Report

Deliverable ID	D4.1	**Version**	V1
Deliverable name	Baseline Platform, Sensor and Data Interoperability Toolbox		
Lead beneficiary	SINTEF (SINTEF AS)		
Editor(s)	Arne J. Berre (SINTEF)		
Contributors	Arne J. Berre (SINTEF), Dumitru Roman (SINTEF), Bjørn Marius von Zernichow (SINTEF), Anders Hansen (SINTEF), Ole Kjos (SINTEF), Nenad Stojanovic (NST), Tim Dahmen (DFKI), Nathanael Hania (Scortex) , Michael Jacoby (Fraunhofer), Ljiljana Stojanovic (Fraunhofer), Jan Gunnar Dyrset (CYB), Alexander Morin (SINTEF), Özlem Albayrak (TEKNO), Tilbe Alp (TEKNO), Enso Ikonen (UOULU)		
Reviewers	Perin Unal (TEKNO), Sailesh Abburu(SINTEF)		
Due date	29.02.2020		
Date of final version	29.02.2020		
Dissemination level	PU		
Document approval	Frode Brakstad	29.02.2020	

The COGNITWIN project has received funding from the European Union's Horizon 2020 research and innovation programme under GA No. 870130

PROPRIETARY RIGHTS STATEMENT

Executive Summary

This D4.1 report on COGNITIVE Baseline Platform, Sensor and Data Interoperability Toolbox describes the initial baseline for the COGNITWIN Toolbox in the areas of toolbox architecture, cyber security, data management and data spaces, further with sensors and realtime sensor/data processing.

This report is a documentation of the baseline technologies and methods that are the starting points for the COGNITWIN Baseline Platform, Sensor and Data Interoperability Toolbox and which will be applied in the developments for the COGNITWIN industrial pilots. This will in particular support the COGNITIVE Hybrid AI and Cognitive Twin Toolbox as described further in the accompanying deliverable D5.1.

The information provided in the report will further be useful for aligning the concepts and available tools among the COGNITWIN partners but should also give external readers ideas about new Industry 4.0 possibilities. After the introduction, the report gives in chapter 4 a brief orientation about the 6 industrial pilot cases from Hydro, Elkem, Sidenor, Saarstahl, Noksel and Sumitomo, and the relevant baseline platforms and infrastructure already in operation in these pilot environments.

Chapter 5 describes the relevant reference models for digital platforms from BDVA, AIOTI, IIC and RAMI 4.0 as a context for the different parts of Digital Twin platforms and tools. Chapter 6 on COGNITWIN Platform/Sensor/Data Interoperability Architecture introduces the evolution from Digital Twins to Hybrid Digital Twins and Cognitive Digital Twins, as support for Cognitive Plants.
The overall architecture of the COGNITWIN toolbox is introduced and it is shown how various technology components from the partners fit into the various architecture areas. Chapter 7 provides an overview of the elements of the COGNITWIN Interoperability Toolbox with the various relevant partner technologies that are being considered for the support of the different pilot needs. Chapter 8 describes in particular potential technology elements and components for the support of Digital Twins, Cyber Security and Data Spaces. Chapter 9 describes the analysis of sensor needs and opportunities for the six different COGNITWIN pilots. Chapter 10 present technology possibilities for the support for realtime data handling and sensor data stream processing. Following the conclusions there are 6 Annexes presenting more details on relevant partner technologies from Teknopar, SINTEF Digital, Scortex, Cybernetica, Nissatech and Fraunhofer.

Separate reports on the industrial pilots (D1.1, D2.1, D3.1), the "Baseline Hybrid AI and Cognitive Twin Toolbox" (D5.1) and the Key Performance Indicators (D6.1) and Data Management Plan (D8.1) are issued together with this report and will give a more complete picture about the COGNITWIN challenges and platform elements. In particular the D5.1 report is a logical addition to this D4.1 report and it would be beneficial to see these two reports in close relationship to each other.

Table of Contents

Table of Figures

Acronyms

AIOTI	Alliance for the Internet of Things Innovation
BDVA	Big Data Value Association
CEP	Complex Event Processing
DSS	Decision Support System
MIS	Management Information Systems
ICT	Information and communications technology
IIC	Industrial Internet Consortium
IOT	Internet of Things
HLA	High Level Architecture (from AIOTI)
KPI	Key Performance Indicators
VISPAR	Visual Pattern Recognition
IIoTP	Industrial Internet of Things Platform
IDS (1)	Industrial Data Security (from Teknopar)
IDS (2)	International Data Spaces (from International Data Spaces Association, IDSA)
OPC UA	Open Platform Communications Unified Architecture
MQTT	Message Queuing Telemetry Transport
RAMI 4.0	Reference Architecture Model for Industry 4.0
JSON	JavaScript Object Notation
BDV	Big Data Value
TRL	Technology Readiness Level

1. Introduction to COGNITWIN Baseline Platform, Sensor and Data Interoperability Toolbox

The COGNITWIN projects aims toward supporting the digitalization of European heavy industries. These industries have specific challenges due to very complex processes and in many cases lack of relevant sensor data. By assembling a team of multiple skills within sensor technologies, physics-based modelling, data driven modelling, hybrid modelling and process control COGNITWIN aims to develop cognitive digital twins that can support a significant improvement in industrial operation.

This report deals with the baseline toolbox that will evolve to support "COGNITWIN for Industry Process Excellence". This report thus describes the baseline to be considered for the COGNITWIN Platform/Sensor/Data Interoperability Architecture as the basis for Digital Twins expanding to Hybrid Digital Twins and Cognitive Digital Twins, as support for Cognitive Plants.

The overall architecture of the COGNITWIN toolbox is introduced and it is shown how various technology components from the partners fit into the various architecture areas. This includes technology elements and components from the COGNITWIN partners for the support of Digital Twins, Cyber Security and Data Spaces.
The report also provide an analysis of sensor needs and opportunities for the six different COGNITWIN pilots. Further technology possibilities for the support for realtime data handling and sensor data stream processing is being presented.

Following the conclusions there are 6 Annexes presenting more details on relevant partner technologies from Teknopar, SINTEF Digital, Scortex, Cybernetica, Nissatech and Fraunhofer. In particular the D5.1 report on " Baseline Hybrid AI and Cognitive Twin Toolbox " is a logical addition to this D4.1 report and it is recommended to read these two reports in relationship to each other.

In COGNITWIN we have defined 6 Pilot cases where the objective is to apply the available digital technologies to improve well defined KPIs (Key Performance Index). We further plan to develop toolboxes which may be used in a number of other process industries. In this report we address the "Hybrid AI and cognitive twin toolbox" and the baseline will be related both to general industrial needs and the specific need for our pilot cases. The pilots will be introduced shortly next, more detailed information about the pilot can be found in the COGNITWIN deliverables D1.1, D2.1 and D3.1.

2. State of the practice – COGNITWIN pilots

The following provides a short introduction to the six pilot cases in COGNITWIN to be supported by the technologies from the COGNITWIN Tool boxes. In this D4.1 report we address the initial baseline of the "COGNITWIN Platform, Sensor and Data Interoperability Toolbox". This is the foundation for the "COGNITWIN Hybrid AI and Cognitive Twin Toolbox" where the baseline for this is reported in the accompanying D5.1 deliverable report.

The baseline of the six pilots is introduced next, more detailed information about the pilot can be found in the COGNITWIN deliverable reports D1.1, D2.1 and D3.1.

2.1 Hydro Pilot - Aluminium Production Process

The topic of the pilot is related to Reduced energy consumption in a selected Hydro GTC (Gas Treatment Center).

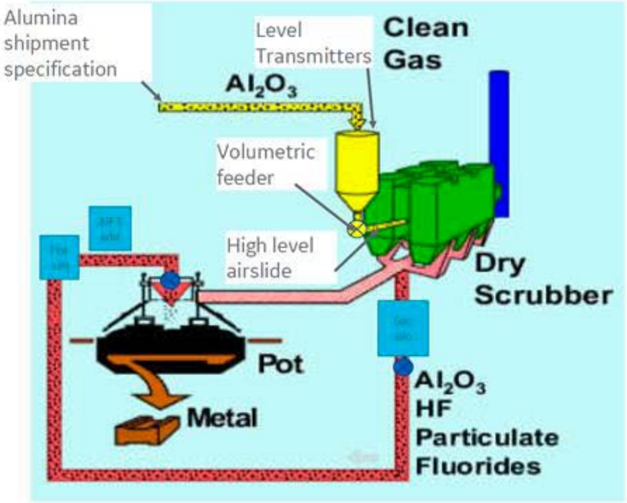


Figure 1 A schematic view of the Hall-Heroult aluminium production process.

Figure 1 shows how the COGNITWIN work is related to the performance of the Gas Treatment Center (top right) and the interactions with the pots (reduction cells) and the inflow of fresh alumina. The ambition is to develop a Digital Twin that allows optimal operation, acceptable emissions of HF (Hydrogen Fluoride) and which can account for the variations in alumina quality from ship load to ship load. The work will increase the overall efficiency with 10% by achieving symbiosis between the actual production (electrolysis) and the cleaning technology (Gas Treatment Centre GTC). Improved environmental impact and optimize energy consumption by maximizing the efficiency of the Gas Treatment Centre. Reduce energy consumption in GTC by 15%. Reduce suction rate overall by 10%, i.e. for the pilot in question, 1500 MWh/y saved fan work, and increased available recovered thermal energy of 13500 MWh/y. Reduced energy consumption and/or replacement by thermal energy (heating) will save CO2 emissions caused by current energy source. The goal is to balance flow distribution to different chambers within ±5% and to decrease process disturbance by preventive maintenance by 5%.

Hydro pilot baseline platform

The Hydro pilot baseline platform includes the collection of data into a DataLake implemented through Microsoft Azure, and a connection to a GE Predix platform. It is an objective of the project to ensure interoperability between these platforms and the COGNITWIN Toolbox.

2.2 Elkem Pilot - Silicon Production Process

The topic is to optimize the post taphole process in an Elkem Silicon plant.

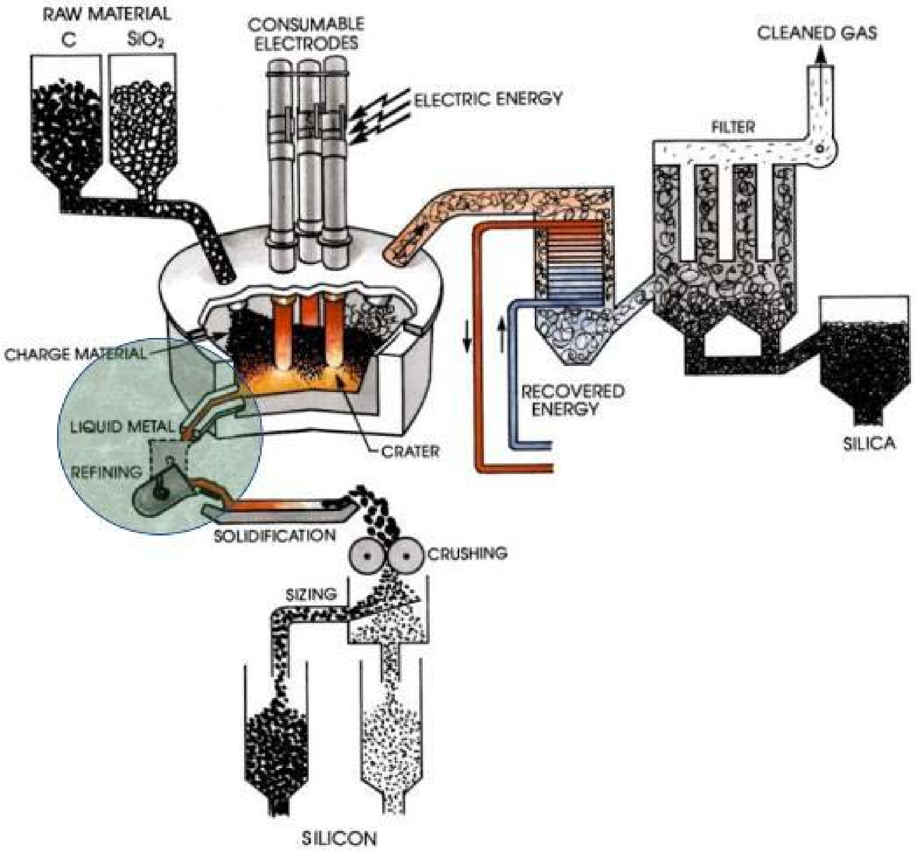


Figure 2 Silicon Production Process

Figure 2 shows the silicon process and where the post tap hole processes in question are found inside the circle (liquid metal / refining). By application of digital technologies to the post tap hole processes (tapping into the ladle, silicon casting into molds) silicon yield can be increased, ladle lifetime can be increased, metal quality can be improved and energy consumption can be reduced. By help of new measurement techniques COGNITWIN will help enabling on-line estimates of the actual silicon flow and its temperatures. Application of new and old data into cognitive hybrid models will be developed to improve the product quality due to more consistent quality and lead to more profitable operation.

Elkem pilot baseline platform

There is no IoT platform in operation for the focus area of this Elkem pilot, and this role will thus be supported by the COGNITWIN Toolbox.

2.3 Sidenor Pilot - Steel Production– Ladle life time improvement

The focus of the Sidenor Pilot is improved ladle lifetime in a Sidenor steel plant.

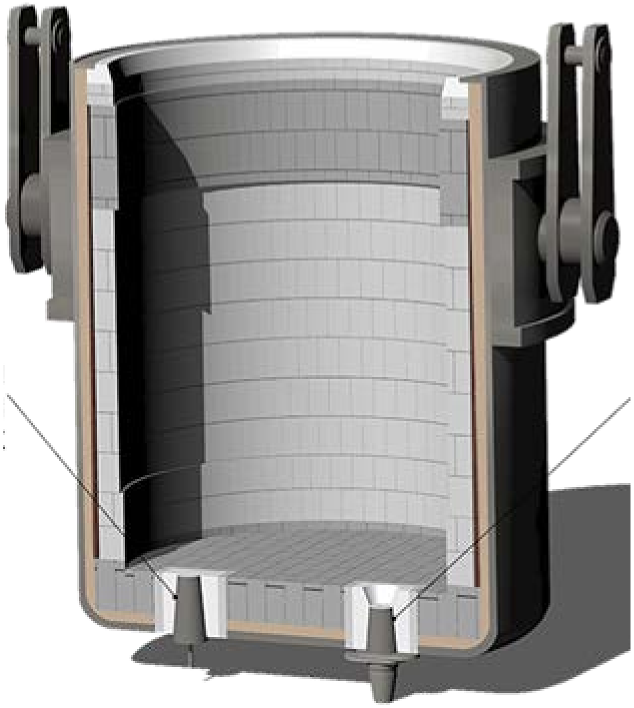


Figure 3 A steel ladle with porous bottom plugs for gas injection.

In Figure 3 it is shown how the belt of dark gray refractory bricks is made of special material which is harder to erode during the operations.

In the steel plant ladles gas injection is applied for refining and stirring. It is observed that ladle lifetime varies a lot and depends on a large number of parameters. The COGNITWIN approach will be to develop a hybrid model that may exploit the large data that already exist. In addition COGNITWIN will apply physics based models that can handle the thermomechanical conditions in the ladle refractory, take advantage of the thermodynamic data for the steel-slag-refractory system, and account for multiphase and multicomponent mass transfer as well as the dynamic temperature variations in the system. Based one available data, new measuring techniques and physics based modeling a Digital Twin for the ladle operation will be developed and used to optimize the ladle lifetime and reduce operational costs.

<u>Sidenor pilot Baseline platform</u>

Currently Sidenor does not have any digitalisation setup specifically for this section of their production process. The conditions of the brick ladles are checked manually and decisions about brick replacement is also done by Sidenor's technicians.

The data management and analytics aspects for this pilot will thus be supported by the COGNITWIN Toolbox.

2.4 Saarstahl Pilot - Tracking system for rolled bars in the rolling mill

This pilot is owned by Saarstahl AG, and where the topic is to enable tracking system for rolled bars in the rolling mill.

Figure 4 A hot, glowing bar in the Saarstahl rolling mill

The rolling mill in Nauweiler is controlled by SAG's Manufacturing Execution System and SAG's Material flow tracking system. These applications flexibly exchange data (sensor and controlling data) to due interoperable data models between the assets and the high level software systems.

Saarstahl pilot Baseline platform

Standards in use includes OPC Unified Architecture, Enterprise Service Bus (Kafka or RabbitMQ), REST-Service Components/services within the platform: MES (Manufacturing Execution System; in-house development), MFT (Material Flow Tracking; in-house development), Interfaces (REST or Message queue).   At present, there is no tracking system in the blooming train. The data management and analytics aspects for this pilot will thus be supported by the COGNITWIN Toolbox.

2.5 Noksel Pilot - Digital Twin Powered Condition Monitoring

This pilot is owned by Noksel. The topic is to apply a cognitive digital twin to power condition monitoring (and control) in the steel pipe manufacturing industry.

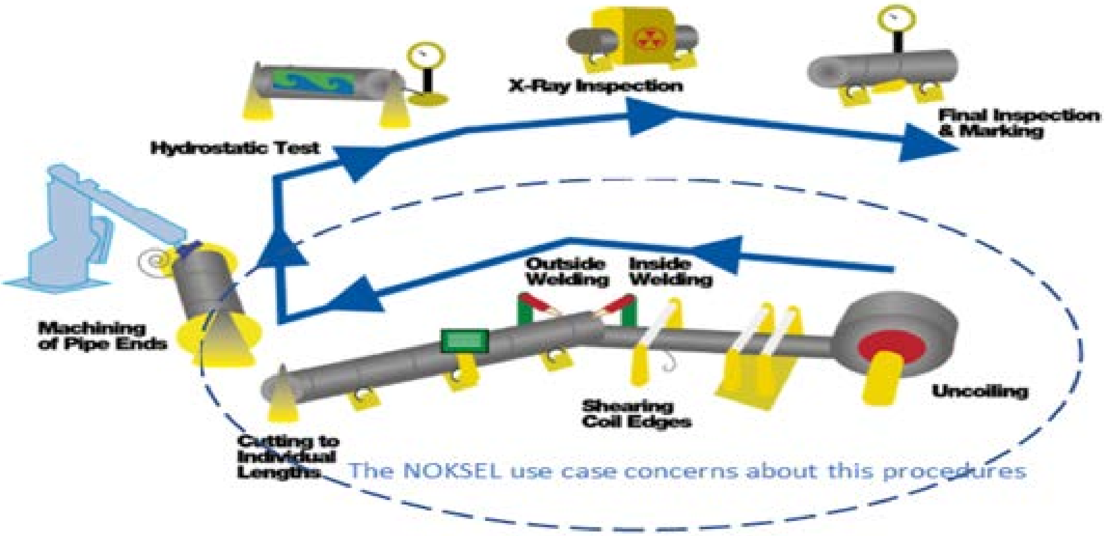


Figure 5 Noksel SWP processes for producing welded steel pipes

Noksel's pilot case is the development of a digital twin for an SWP machine in steel pipe production. The digital twin will collect and analyze multiple sensors' data in real-time, and enable a smart condition monitoring system for predictive maintenance. Real-time data acquisition, communication networks for monitoring, and automated recommendation generation are among the key innovative features of this pilot. Automated recommendations will also be generated.

Smart components that use sensors to gather data about real-time status, working condition, or position will be connected to a cloud-based system that receives and processes all the data the sensors monitor. This input will be analyzed against business and other contextual data through smart visualization systems. The digital twin model will allow joining physical and virtual worlds to create a new networked layer in which intelligent objects interact with each other to virtualize the steel pipe manufacturing process on the SWP machinery. The ambition is to reduce machine downtimes, decrease energy consumption, and increase total equipment performance.

Noksel pilot Baseline platform

Currently the Noksel SWP Machinery is tracking by NOBİS System (Delphi based special software created by NOKSEL and machinery adjustments are made by terminals with installed TEKNOPAR's software. NOBIS is also integrated with the SAP system.
The data management and analytics aspects for this pilot will thus be supported by the COGNITWIN Toolbox with a focus on interoperability with this baseline platform.

2.6 Sumitomo Pilot - Engineering Boiler operations

This pilot is owned by Sumitomo and where a main ambition is allow plants to operate well even if the fuels quality and composition is changing faster than it used to do in the past. This can be made possible through the COGNITWIN cognitive digital twin development.

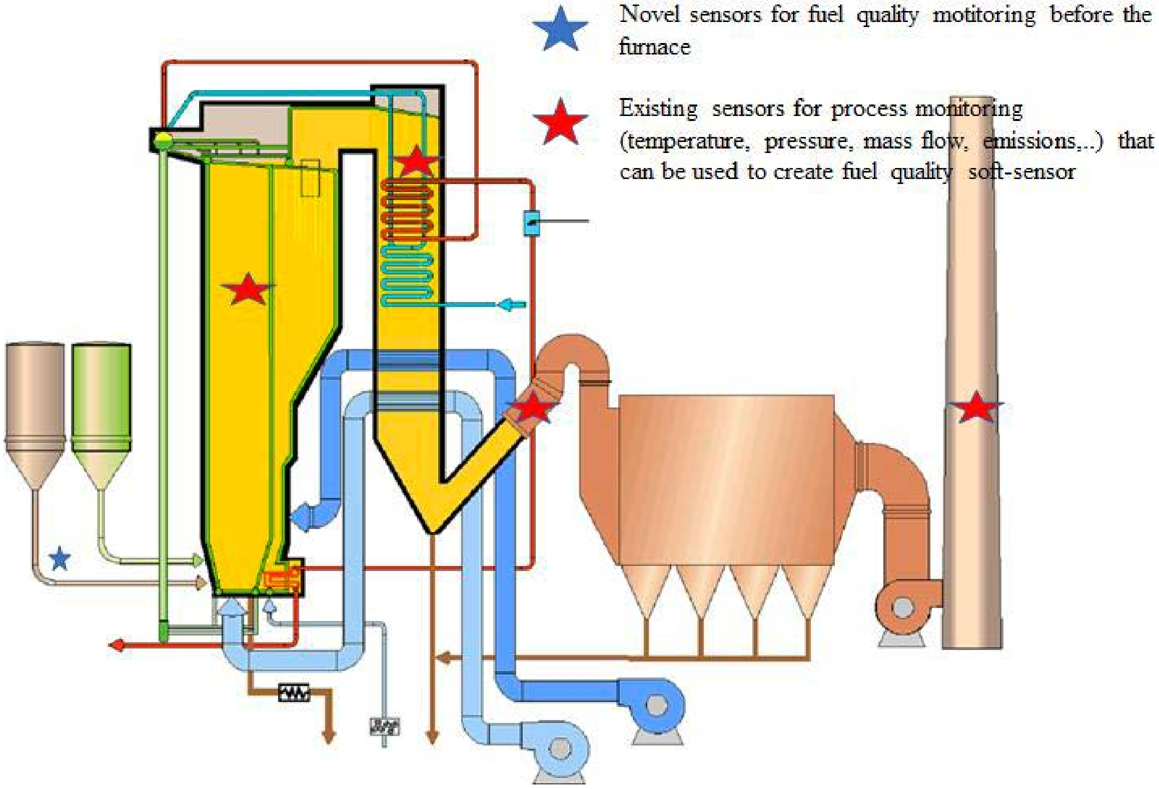

★ Novel sensors for fuel quality motitoring before the furnace
★ Existing sensors for process monitoring (temperature, pressure, mass flow, emissions,..) that can be used to create fuel quality soft-sensor

Figure 6 Overview of the boiler process in a Sumitomo made plant

The innovation and the cognitive element here is to introduce new measuring techniques, combine measured fuel quality data, process data from the power plant and existing physics based models. The developed digital twins should predict how fuel quality changes affect the process, which enables early detection of process disturbances and overall process optimization.

Sumitomo pilot Baseline platform

The current platform that Sumitomo is using for data collection is called SmartBoiler. It is a server-based data collection, storage and analysis system developed by SFW in early 2000's for plant operators and managers to monitor their boiler's operation through a host of plant sensors and analytics. The SmartBoiler remote connection brings the customer's process data available for SFW's process specialists who perform plant performance analyses as well as data deviation and disturbance management. SFW process specialists also provide operation support and remote troubleshooting services and offer regular process performance reporting. The data management and analytics aspects for this pilot will thus be supported by the COGNITWIN Toolbox with an integration and interoperability perspective for SmartBoiler.

3. Reference Models for Digital Platforms

In this chapter we present a number of relevant technical reference models for digital twins and digital platforms, including the following:

- *Big Data Value Association Reference Model (BDVA)*
- *Alliance for the Internet of Things Innovation (AIOTI) High Level Architecture (HLA)*
- *Reference Architecture Model for Industry 4.0 (RAMI 4.0)*
- *Industrial Internet Consortium (IIC) Reference Architecture (IIRA)*

These reference models identify technical areas that will be related to by the COGNITWIN Architecture and Toolbox. This foundation in reference models for digital platforms is also applied in other digital platform projects and have been adapted from these with projects like DataBio (www.databio.eu) and DEMETER (http://h2020-demeter.eu/). The core structure of the COGNITWIN Architecture and Toolbox is based on the layers of the BDVA Reference Model, which again has been related to the other reference models in the following.

3.1 Big Data Value Association (BDVA)

The BDVA Big Data Value Reference Model (from the BDVA SRIA 4.0 document[1] – and developed by COGNITWIN partner SINTEF Digital as BDVA TF6 Task Force Lead) is shown in the figure below.

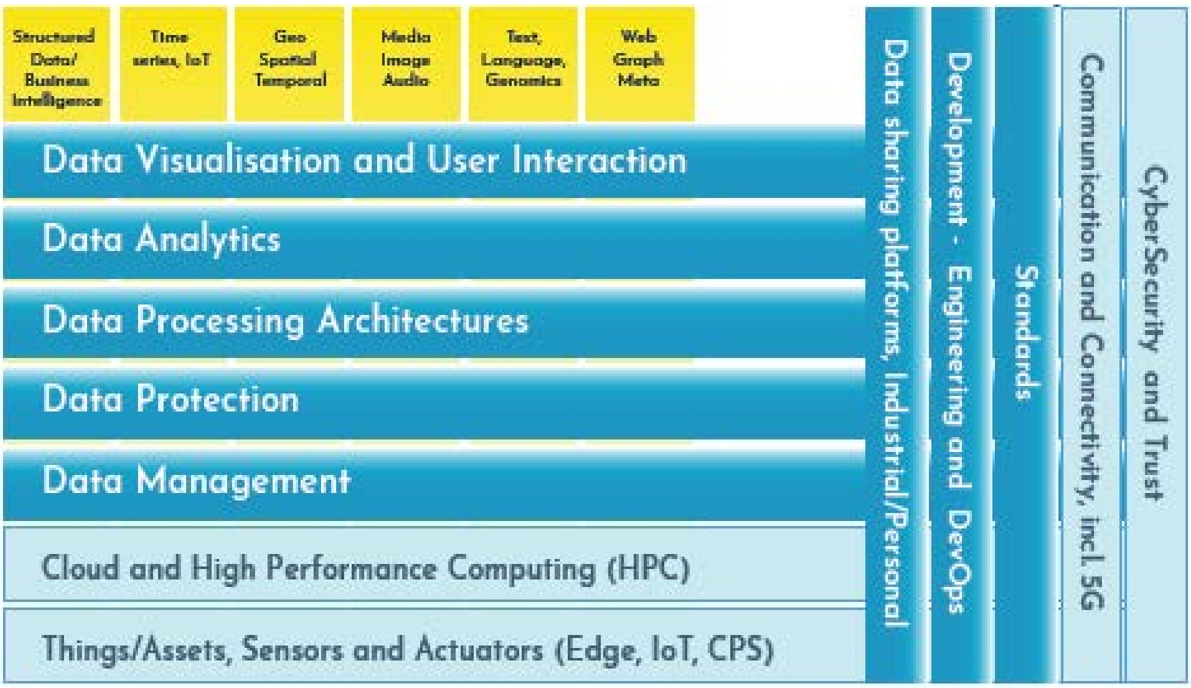

Structured Data/ Business Intelligence
Time series, IoT
Geo Spatial Temporal
Media Image Audio
Text, Language, Genomics
Web Graph Meta
Data Visualisation and User Interaction
Data Analytics
Data Processing Architectures
Data Protection
Data Management
Cloud and High Performance Computing (HPC)
Things/Assets, Sensors and Actuators (Edge, IoT, CPS)
Data sharing platforms, Industrial/Personal
Development - Engineering and DevOps
Standards
Communication and Connectivity, incl. 5G
CyberSecurity and Trust

Figure 7 Big Data Value Association – BDV Reference Model

The BDV Reference Model has been developed by the BDVA TF6 under the lead of SINTEF, taking into account input from technical experts and stakeholders along the whole Big Data Value chain as well as interactions with other related PPPs. An explicit aim of the BDV Reference Model in the SRIA 4.0 document is to also include logical relationships to other areas of a digital platform such as Cloud, High Performance Computing (HPC), IoT, Networks/5G, CyberSecurity etc.

The BDV Reference Model may serve as common reference framework to locate Big Data technologies on the overall IT stack. It addresses the main concerns and aspects to be considered for Big Data Value systems.

The BDV Reference Model is structured into horizontal and vertical concerns.

- **Horizontal concerns** cover specific aspects along the data processing chain, starting with data collection and ingestion, reaching up to data visualization. It should be noted, that the horizontal concerns do not imply a layered architecture. As an example, data visualization may be applied directly to collected data (data management aspect) without the need for data processing and analytics. Further data analytics might take place in the IoT area – i.e. Edge Analytics. This shows logical areas – but they might execute in different physical layers.
- **Vertical concerns** address cross-cutting issues, which may affect all the horizontal concerns. In addition, verticals may also involve non-technical aspects (e.g., standardization as technical concerns, but also non-technical ones).

[1] http://bdva.eu/sites/default/files/BDVA_SRIA_v4_Ed1.1.pdf

Given the purpose of the BDV Reference Model to act as a reference framework to locate Big Data technologies, it is purposefully chosen to be as simple and easy to understand as possible. It thus does not have the ambition to serve as a full technical reference architecture. However, the BDV Reference Model is compatible with such reference architectures, most notably the emerging ISO JTC1 WG9 Big Data Reference Architecture – now being further developed in ISO JTC1 SC42 Artificial Intelligence.
The following technical priorities as expressed in the BDV Reference Model are elaborated in the remainder of this section:

Horizontal concerns:

- **Big Data Applications**: Solutions supporting big data within various domains will often consider the creation of domain specific usages and possible extensions to the various horizontal and vertical areas. This is often related to the usage of various combinations of the identified big data types described in the vertical concerns.
- **Data Visualisation and User Interaction**: Advanced visualization approaches for improved user experience.
- **Data Analytics**: Data analytics to improve data understanding, deep learning, and meaningfulness of data.
- **Data Processing Architectures**: Optimized and scalable architectures for analytics of both data-at-rest and data-in- motion with low latency delivering real-time analytics.
- **Data Protection**: Privacy and anonymisation mechanisms to facilitate data protection. It also has links to trust mechanisms like Blockchain technologies, smart contracts and various forms for encryption. This area is also associated with the area of CyberSecurity, Risk and Trust.
- **Data Management**: Principles and techniques for data management including both data life cycle management and usage of data lakes and data spaces, as well as underlying data storage services.
- **Cloud and High Performance Computing (HPC):** Effective big data processing and data management might imply effective usage of cloud and high performance computing infrastructures. This area is separately elaborated further in collaboration with the Cloud and High Performance Computing (ETP4HPC) communities.
- **IoT, CPS, Edge and Fog Computing**: A main source of big data is sensor data from an IoT context and actuator interaction in Cyber Physical Systems. In order to meet real-time needs it will often be necessary to handle big data aspects at the edge of the system.

Vertical concerns:

- **Big Data Types and semantics**: The following six big data types have been identified – based on the fact that they often lead to the use different techniques and mechanisms in the horizontal concerns, which should be considered, for instance for data analytics and data storage: *1) Structured data; 2) Times series data; 3) GeoSpatial data, 4) Media, Image, Video and Audio data; 5) Text data, including Natural Language Processing data and Genomics representations; 6) Graph data, Network/Web data and Meta data*. In addition, it is important to support both the syntactical and semantic aspects of data for all big data types.
- **Standards**: Standardisation of big data technology areas to facilitate data integration, sharing and interoperability.
- **Communication and Connectivity**: Effective communication and connectivity mechanisms are necessary for providing support for big data. This area is separately elaborated further with various communication communities, such as the 5G community.
- **Cybersecurity**: Big Data often need support to maintain security and trust beyond privacy and anonymisation. The aspect of trust frequently has links to trust mechanisms such as blockchain

technologies, smart contracts and various forms of encryption. The CyberSecurity area is separately elaborated further with the CyberSecurity PPP community.

- **Engineering and DevOps**: for building Big Data Value systems. This area is also elaborated further with the NESSI (Networked European Software and Service Initiative) Software and Service community.
- **Data Platforms**: Marketplaces, IDP/PDP, Ecosystems for Data Sharing and Innovation support. Data Platforms for Data Sharing include in particular Industrial Data Platforms (IDPs) and Personal Data Platforms (PDPs), but also include other data sharing platforms like Research Data Platforms (RDPs) and Urban/City Data Platforms (UDPs). These platforms include efficient usage of a number of the horizontal and vertical big data areas, most notably the areas for data management, data processing, data protection and cybersecurity.
- **AI platforms**: In the context of the relationship between AI and Big Data there is an evolving refinement of the BDV Reference Model – showing how AI platforms typically include support for Machine Learning, Analytics, visualisation, processing etc. in the upper technology areas supported by data platforms – for all of the various big data types.

3.2 Alliance for the Internet of Things Innovation (AIOTI)

The Alliance for the Internet of Things Innovation (AIOTI)[2] has specified a High Level Reference Architecture (HLA) that maps to several other dominant and/or standardised IoT architectural approaches, such as ITU-T[3], oneM2M[4], Industrial Internet Consortium (IIC)[5], RAMI 4.0[6][7][8], Big Data Value Association (BDVA)[9], National Institute of Standards and Technology (NIST)[10], etc.
Based on the IoT-A domain model, they have derived the AIOTI Domain Model depicted in the figure below.

[2] Alliance for the Internet of Things Innovation (AIOTI): https://aioti.eu/

[3] ITU-T FG-DPM, ITU-T Focus Group on Data Processing and Management to support IoT and Smart Cities & Communities, https://www.itu.int/en/ITUT/focusgroups/dpm/Pages/default.aspx

[4] oneM2M, "oneM2M Functional Architecture Baseline Draft", oneM2M-TS-0001, 2014.

[5] Industrial Internet Reference Architecture, http://www.iiconsortium.org/IIRA.htm

[6] VDI/VDE GMA, ZVEI: Status Report - Reference Architecture Model Industrie 4.0 (RAMI 4.0), July 2015, https://www.zvei.org/fileadmin/user_upload/Presse_und_Medien/Publikationen/2016/januar/GMA_Status_Report__Reference_Archtitecture_Model_Industrie_4.0___RAMI_4.0_/GMA-Status-Report-RAMI-40-July-2015.pdf

[7] DIN SPEC 91345:2016-04 – Referenz architektur modell Industrie 4.0 (RAMI 4.0), April 2016, http://www.din.de/de/ueber-normen-und-standards/din-spec/din-specveroeffentlichungen/wdc-beuth:din21:250940128

[8] IEC PAS 63088:2017 Smart manufacturing - Reference architecture model industry 4.0 (RAMI 4.0), March 2017, https://webstore.iec.ch/publication/30082

[9] Big Data Value Association, European Big Data Value Strategic Research and Innovation Agenda, http://bdva.eu/sites/default/files/BDVA_SRIA_v4_Ed1.1.pdf

[10] NIST big data interoperability framework, http://bigdatawg.nist.gov/V1_output_docs.php

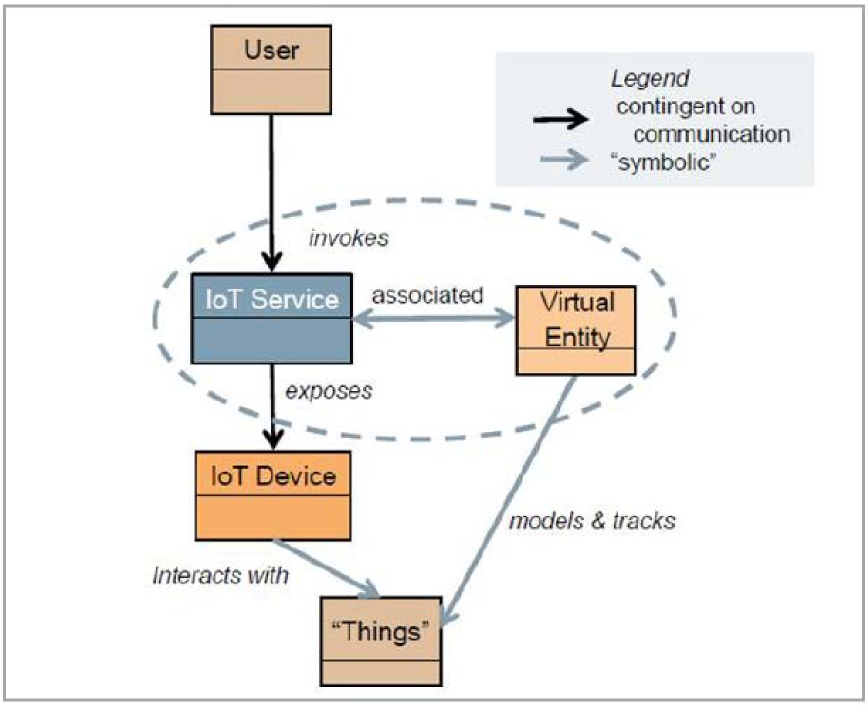


Figure 8. The AIOTI Domain Model

The main concepts and relationships are captured by the domain model at the highest level. Identification and naming of these concepts and relationships make a common dictionary for the domain, moreover they are functional for all other taxonomies and models. There is interaction of the user, which can be human or otherwise, with a physical entity, a Thing. This kind of interaction can be intervened by an IoT Service, which is associated with a Virtual Entity, a physical entity or a digital representation. The Thing and the IoT Service interact with each other, though an IoT Device that exposes the capabilities of the actual physical entity.

As it is depicted in the figure below, the functional model of AIOTI consists of three basic layers. We have to note in this point that the term "*layer*" refers to software architecture logic. So, by saying layer, we mean a group of modules that offer a tight set of services.

The AIOTI functional model describes, as depicted below, interfaces and functions between functions of the IoT system. As it shown, functions are: **1)** *App Entity* is an entity existing in the application layer that implements IoT application logic. **2)** *IoT Entity*, which is an entity with the aim to expose IoT functions to App Entities via the interface 2 or to other IoT entities via interface 5. **3)** *Networks function* that consists of different network technologies (such as PAN, LAN, WAN). It includes different interconnected administrative network domains.

A Device can include an App Entity and a Network interface. For example, it can use an IoT Entity. This example represents a constrained device. But other devices can implement an App Entity, an IoT Entity and a Network interface. The interfaces that are shown in the figure above are:

1. it specifies the structure of the data that are exchanged between App Entities.
2. it enables access to services exposed by an IoT Entity.
3. it enables the exchanging of data across the Networks to other entities.
4. it enables the requesting of network control plane services.

5. it enables the requests and the exposures of services that coming (form)/going (to) IoT Entities.

AIOTI High Level Architecture realizes the digital representation of physical things in the IoT Entities. Those representations help in discovering things of App Entities and enable related services, for example actuation or measurements. In order to realize interoperability, the representation of things contains data and metadata.

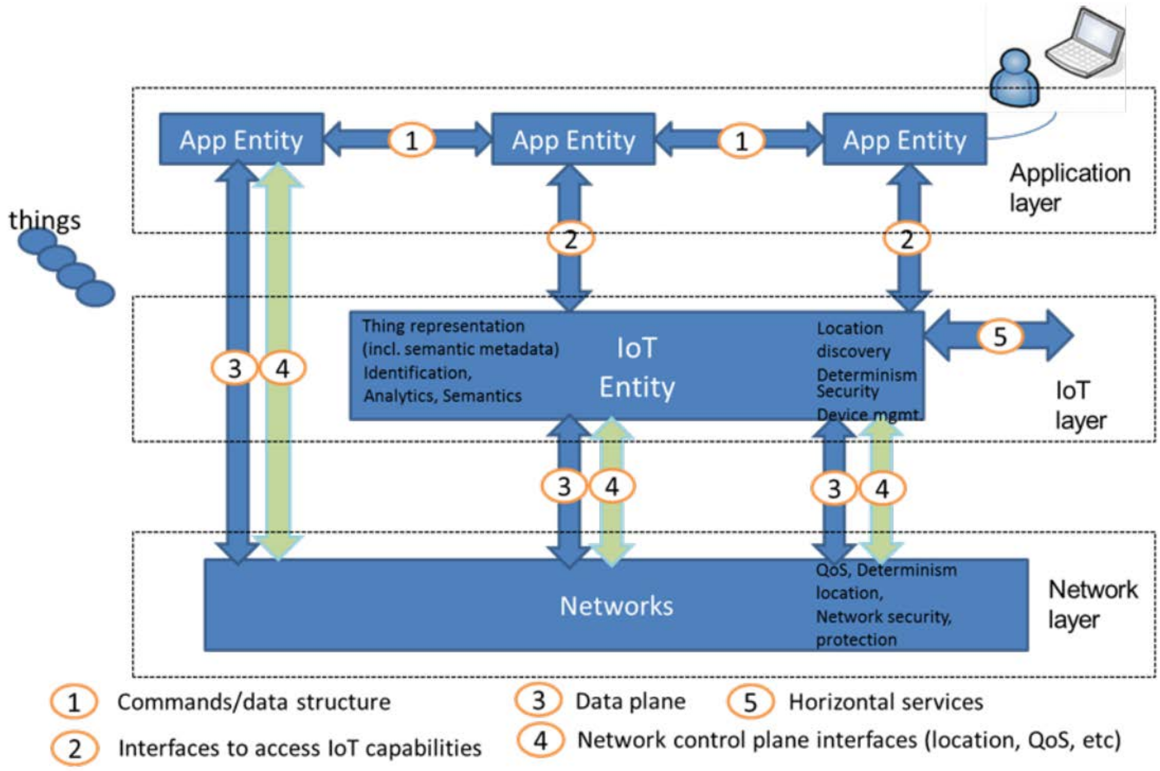

App Entity
1
App Entity
1
App Entity
Application layer
things
2
2
Thing representation (incl. semantic metadata) Identification, Analytics, Semantics
IoT Entity
Location discovery Determinism Security Device mgmt
5
IoT layer
3
4
3
4
3
4
Networks
QoS, Determinism location, Network security, protection
Network layer
1 Commands/data structure
2 Interfaces to access IoT capabilities
3 Data plane
4 Network control plane interfaces (location, QoS, etc)
5 Horizontal services

Figure 9. The AIOTI HLA functional model

A mapping between the AIOTI HLA and the BDV Reference Model is shown in the figure below.

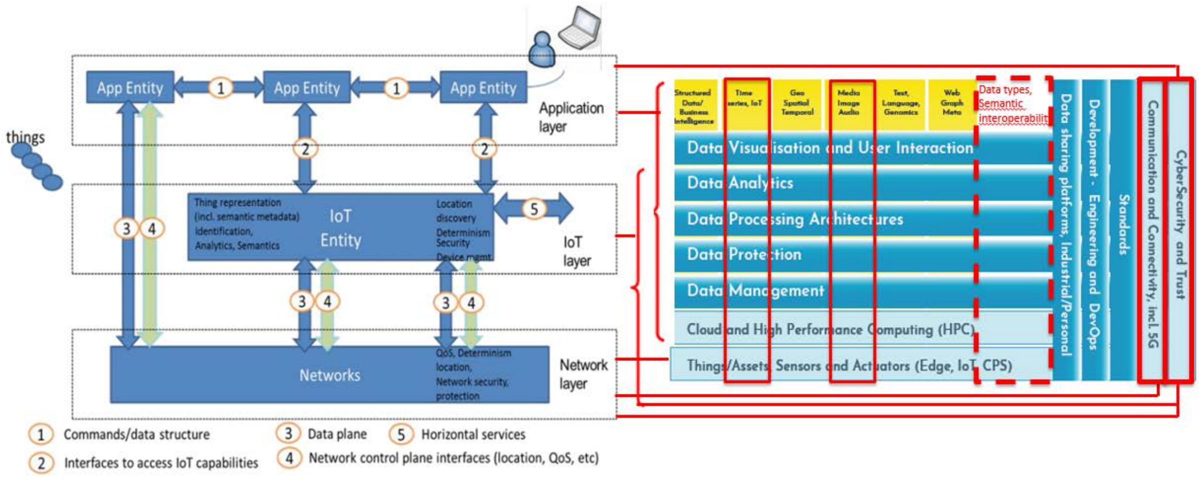

App Entity
1
App Entity
1
App Entity
Application layer
things
2
2
Thing representation (incl. semantic metadata) Identification, Analytics, Semantics
IoT Entity
Location discovery Determinism Security Device mgmt
5
IoT layer
3
4
3
4
3
4
Networks
QoS, Determinism location, Network security, protection
Network layer
1 Commands/data structure
2 Interfaces to access IoT capabilities
3 Data plane
4 Network control plane interfaces (location, QoS, etc)
5 Horizontal services
Structured Data/ Business Intelligence
Time series, IoT
Geo Spatial Temporal
Media Image Audio
Text, Language, Genomics
Web Graph Meta
Data types, Semantic interoperabilit
Data Visualisation and User Interaction
Data Analytics
Data Processing Architectures
Data Protection
Data Management
Cloud and High Performance Computing (HPC)
Things/Assets, Sensors and Actuators (Edge, IoT, CPS)
Data sharing platforms, Industrial/Personal
Development - Engineering and DevOps
Standards
Communication and Connectivity, incl. 5G
CyberSecurity and Trust

Figure 10 BDV Reference Model related into the HLA model

The BDV Reference Model shows related technical areas and capabilities but without an explicit focus on layers. The Time Series/IoT and Media/Image/Audio Data types of the BDV Reference Model is of particular interest in an IoT context and are thus marked red across the various areas. The Semantic Interoperability focus through data types is shown as a dotted line to illustrate that this is a relevant area both in the AIOTI HLA context and in the BDV Reference Model .

The App Entities of HLA might include application logic including data visualisation and user interaction services, data analytics, various kinds of data processing, data protection support and data management logic as well as support for cloud/HPC execution. In addition the App Entities might include support for Cybersecurity and Trust.

The IoT Entities of HLA will include access and management capabilities for sensors and actuators, but might also include support for data analytics (edge analytics), data processing, data protection, and data management. In addition the IoT Entities might include support for Cybersecurity and Trust.

The Networks of HLA is linked to the Communication and connectivity area in the BDV Reference Model. It is in particular supporting short and long range connectivity and data forwarding between entities, and both synchronous and asynchronous communication mechanisms, with appropriate QoS support. In addition the Networks might include support for Cybersecurity and Trust. The Networks also includes special support for IoT communication and connection.

A corresponding mapping of the HLA into the BDV Reference Model is shown in the figure below.

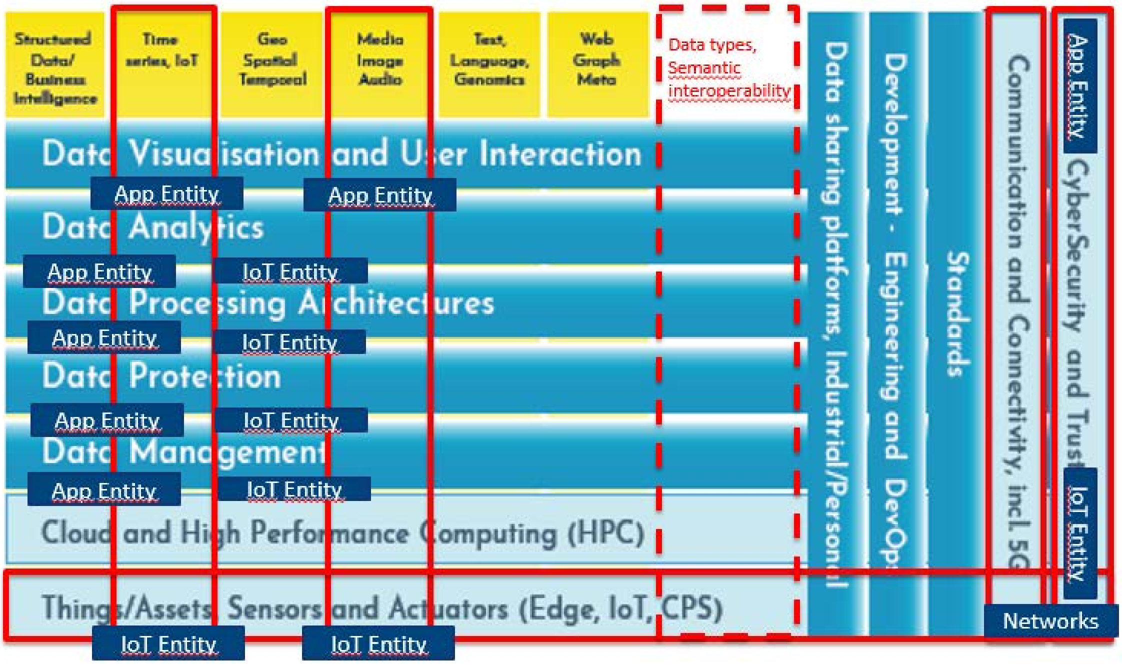

Structured Data/ Business Intelligence
Time series, IoT
Geo Spatial Temporal
Media Image Audio
Text, Language, Genomics
Web Graph Meta
Data types, Semantic interoperability
Data Visualisation and User Interaction
App Entity
App Entity
Data Analytics
App Entity
IoT Entity
Data Processing Architectures
App Entity
IoT Entity
Data Protection
App Entity
IoT Entity
Data Management
App Entity
IoT Entity
Cloud and High Performance Computing (HPC)
Things/Assets, Sensors and Actuators (Edge, IoT, CPS)
IoT Entity
IoT Entity
Data sharing platforms, Industrial/Personal
Development - Engineering and DevOps
Standards
Communication and Connectivity, incl. 5G
CyberSecurity and Trust
App Entity
IoT Entity
Networks

Figure 11 BDV Reference Model - AIOTI HLA mapping

The followings are the core elements of the BDV Reference Model-AIOTI HLA mapping:

- It is acknowledged that HLA is not only focusing on the IoT/Sensor/Actuator capabilities/services but actually spans across the whole BDV Reference Model. In particular, the AIOTI HLA focuses on the full stack for IoT and sensor/actuator related technologies, and in particular on the Big Data types described as Time Series/IoT and Media/Image/Audio data across the stack.

- **The App Entity** of AIOTI HLA is reflecting application logic and is mapped into the technical areas of Data Visualisation, Data analytics and Data processing and Data Protection, and might also include application specific services for data management and cloud/HPC processing. The App Entity is does shown related to different technical areas/capabilities within the BDV Reference Model, to illustrate the fact that an App Entity might include various capabilities including also support for Cybersecurity and trust.
- **The IoT Entity** of AIOTI HLA is reflecting IoT abstractions and is mapped into the technical areas of Things/Assets Sensors and Actuators, Data Management and sensor/actuator access and manipulation. An IoT Entity might also include support for data analytics (edge analytics), as well as data processing, data protection and data management.
- **Networks** in the AIOTI HLA is mapped into the technical area of Communication and Connectivity in the BDV Reference Model. Networks are connected to IoT Entities in particular but also to App Entitites, the Communication and Connectivitiy is a vertical area in the BDV Reference Model
- **Data Types** Time series/IoT and Media/Image/Audio is in particular focused in the IoT context and shown with red lines. The AIOTI HLA focus on semantic interoperability and semantic capabilities is illustrated through the dotted line around the data types where semantic interoperability is enabled through the semantics of data types through ontologies and standard data models. This is an important area both within HLA and the BDV Reference model
- **CyberSecurity and Trust** capabiliites and services are being used by all of App Entities, IoT Entities and Networks.
- Data Sharing platforms might be used for the management of IoT data
- Development/Engineering perspective might be used on the total life cycle of IoT data
- Standards might be identified for all IoT areas.

Note that the different technical areas in the BDV Reference Model are showing capabilities and services without requiring a particular physical or logical layering perspective and the different technical areas and capabilities might physically reside in different clients and servers in different configurations.

3.3 Reference Architecture Model for Industry 4.0 (RAMI 4.0)

Industrie 4.0 covers a highly diverse landscape of industries, stakeholders, processes, technologies and standards. To achieve a common understanding of what standards, use cases, etc. are necessary for Industrie 4.0, a uniform architecture model (the Reference Architecture Model Industrie 4.0 (RAMI 4.0)) was developed by VDI/VDE GMA & ZVEI in Germany, serving as a basis for the discussion of interrelationships and details. RAMI 4.0 has been further defined by DIN as DIN SPEC 91345 [17] and IEC as IEC PAS 63088.

Besides the reference architecture model, RAMI 4.0 defines the I4.0 component which links the assets in the Industrie 4.0 environment like products, production machines or production lines and systems with their virtual presentation in cyber space the so called administration shell.

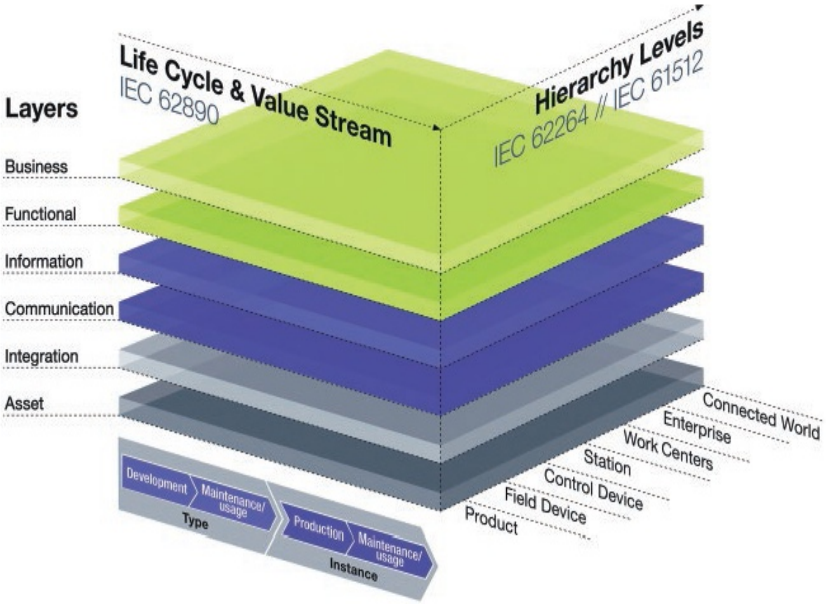


Figure 12 RAMI 4.0 reference architecture

The reference architecture model as shown in Figure 12 structures the Industrie 4.0 space into its fundamental aspects. It expands the hierarchy levels of IEC 62264 by adding the “Field Device” and “Product” or work piece level at the bottom, and the “Connected World” going beyond the boundaries of the individual factory at the top. The left horizontal axis represent the life cycle of systems or products and the value stream of production. It also establishes the distinction between “Type” and “Instance”. Finally, the six vertical layers on the left define various architectural viewpoints on Industrie 4.0 that are relevant from a system design and standardization point of view. The specific characteristics of the reference architecture model are therefore its combination of life cycle and value stream with a hierarchically structured approach a various architectural views.

The Reference Architectural Model Industrie 4.0 (RAMI 4.0) was developed by the Platform 4.0 in 2015 and focuses on the IoT and Cyber-Physical Systems (CPS) in the industrial manufacturing domain. RAMI4.0 is a three-dimensional model, which describes the Industrie 4.0 space and organises the life-cycle/value streams and the manufacturing hierarchy levels across the six layers of the IT representation of Industry 4.0.
Current Industrial Revolution driven by CPS and IoT is expected to have a major impact on the future

of agriculture as well, as there is a natural relation between industry and agriculture. As an extension of Industry 4.0 a new concept can be introduced: Agriculture 4.0. Integration of machines and equipment, increased automation, efficient decisional process represents objectives of an agricultural enterprise facilitating the adaptability to climate and market dynamics and perturbations and allowing for sustainable, ecological and socially beneficial development.

One of the main objectives once adopted is to be able to communicate the scope and design of the system, to further collaboration and integration with other relevant initiatives by framing the developed concepts and technologies in a common model.

The three-dimensional matrix can be used to position standards and describe use-cases. It addresses integration within and between factories, end-to-end engineering and human value-stream orchestration. This model is complemented by the Industrie 4.0 components and both have been described in DIN SPEC 91345. (Reference Architecture Model Industrie 4.0 (RAMI4.0) - DIN SPEC 91345:2016-04, 2016)

In RAMI4.0, each component consists of six layers. Starting with the lowest layer, the structure consists of asset, integration, communication, information, functional and business and represents a layered IT system structure.

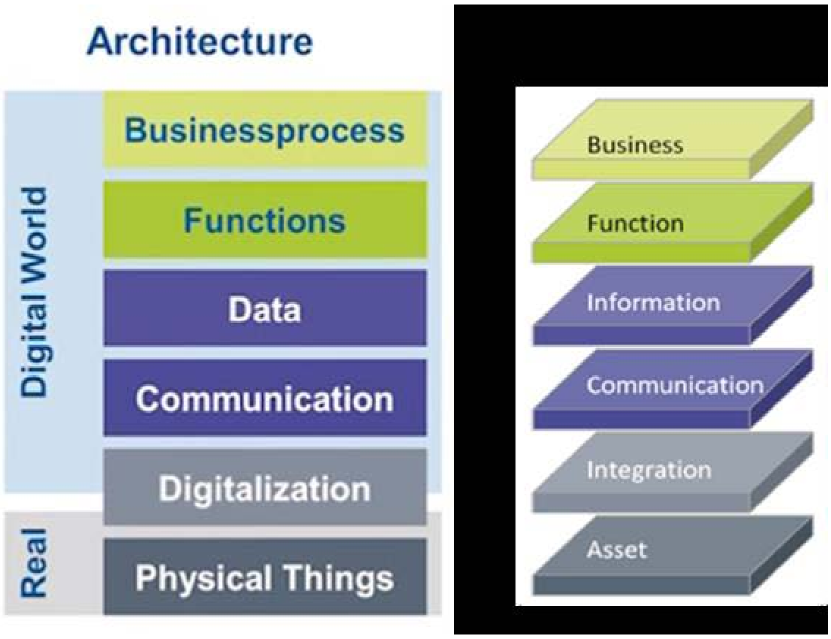


Figure 13. The IT Layers of RAMI 4.0

The function of each layer is:

- The asset layer describes physical components of a system, for example production equipment, product part, sensors, documents, as well as humans. For every asset represented in this layer there must be a virtual representation in the above layers. Among the physical assets, this layer includes the digital interface with humans and the relationship to elements in the integration layer.
- The integration layer deals with easy to process information content and can be considered as a bridge between the real and the IT world. It contains all elements associated with the IT, including field buses, HMIs, necessary to implement a function, as well as the properties and process related functions required to use an asset in the intended way and generates events

based on the acquired information.

- The communication layer is responsible for the standardized communication between integration and information layer. Therefore, it performs transmission of data and files and standardizes the communication from the Integration Layer, providing uniform data formats, protocols and interfaces in the direction of the Information Layer. It also provides services to control the integration layer.
- The information layer holds the necessary data in a structured and integrated form and provides the interfaces to access this structured data from the functional layer. It is responsible for processing, integrating and persisting the data and events, as well as for describe the data related to the technical functionality of an asset. It can be considered the run-time environment for Complex Event Processing (CEP) where rule-based (pre-) processing of events takes place, data APIs and data persistence mechanisms. So, events are received from the communication layer, transformed and forwarded accordingly.
- The functional layer describes the logical and technical functions of an asset providing a digital description of its functions and a platform for horizontal integration of various functions; it also describes the business model mapping, business processes which can be adjusted based on inputs from the functional layer, providing models with runtime data of processes, functions and applications.
- The business layer is in charge to orchestrate the services provided by the functional layer. It maps the services to the business (domain) model and the business process models. It also models the business rules, legal and regulatory constraints of the system. The processes to ensure of the economy are located on this level.

In order to represent the Industry 4.0 or a Process Industry 4.0 environment, the functionalities of IEC62264 have been expanded to include two new levels, at the bottom, the “product” (both the type and the instance, through the entire lifecycle) which are active elements within the production system due to their ability to communicate. They provide information on their individual properties and necessary production steps. At the top there is the “connected world”, which represents its outer networks or the ecosystem, e.g. collaboration with business partners and customers, suppliers or service providers, as well as Internet-based services.
This allows moving from the typical pyramid, with rigid hierarchical structures, to a composite of networked objects and systems as reflected in Figure 14.

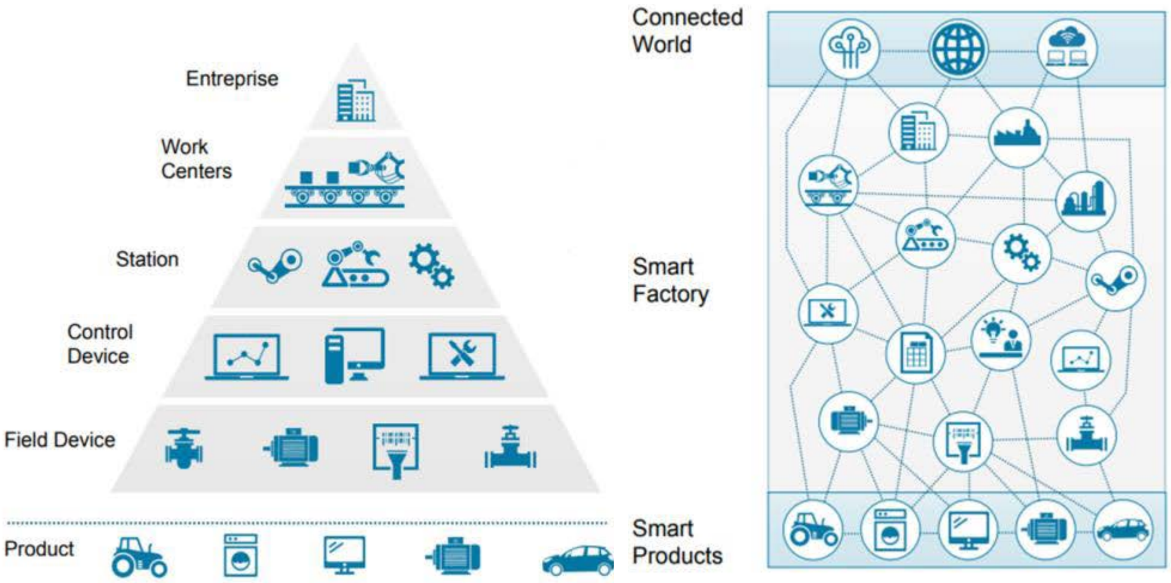


Figure 14. Hierarchy Levels of Industry 3.0 and RAMI 4.0

The mapping of RAMI 4.0 to the AIOTI HLA – functional model - is depicted in the following Figure.

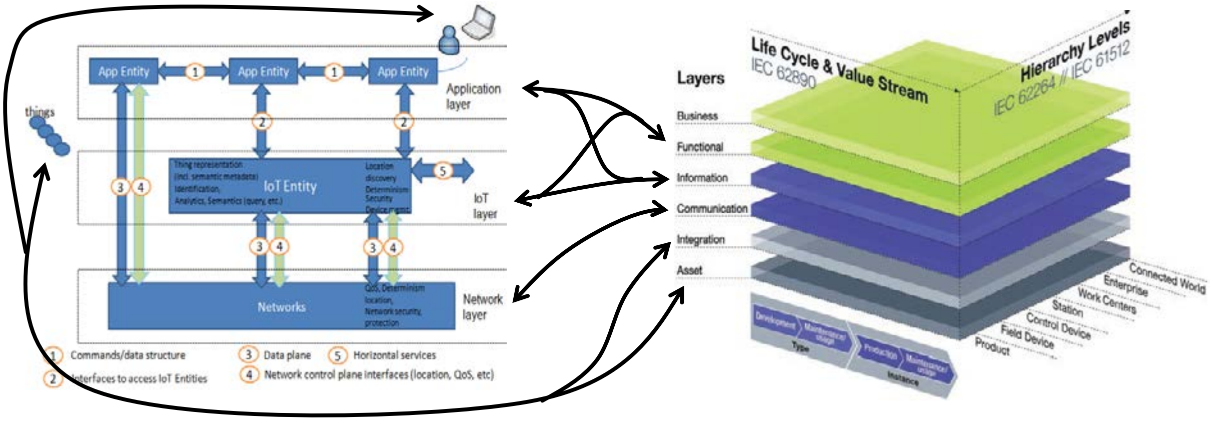


Figure 15 Mapping RAMI 4.0 to AIOTI HLA – functional model

The following explanations can be made as regards Figure 15.

As the AIOTI HLA and RAMI 4.0 have different purposes and approaches only a rough mapping can be performed and a 1 to 1 relation between the components in the two models is not always possible.

- The HLA Network layer represents the IoT communication capabilities and maps to the RAMI 4.0 Communication Layer
- The HLA IoT and App Layer represent functional and information components that map to the RAMI 4.0 Functional and Information layers
- Things, People, HW components map to the RAMI 4.0 Asset and Integration layer
- Note that functions at the network, IoT and App Layer like routers, data storage and processing would appear at the RAMI 4.0 functional layer from an functional point of view and in the physical representation at the asset layer

The mapping of RAMI 4.0 to the AIOTI HLA – domain model - is depicted in the following Figure.

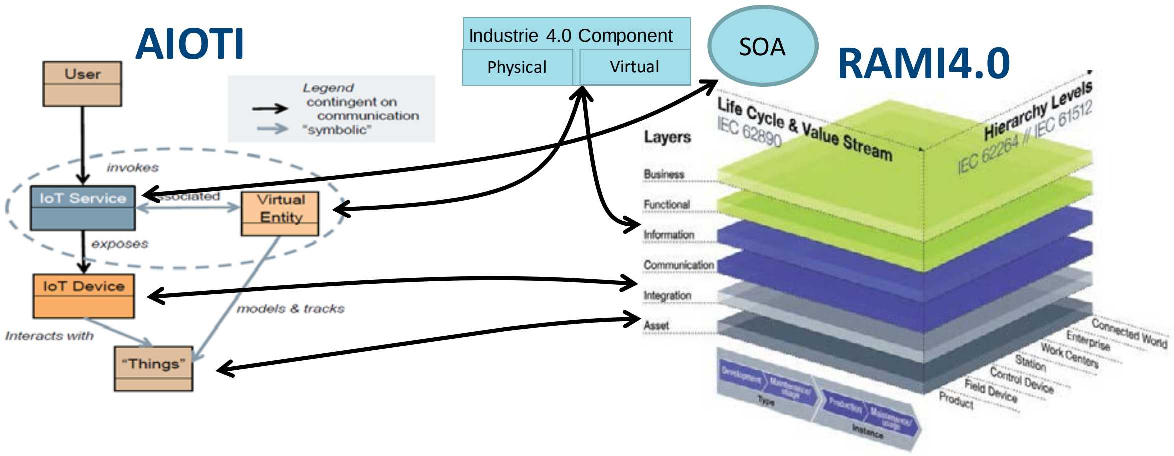


Figure 16 Mapping RAMI 4.0 to AIOTI HLA – domain model

The following explanations can be made as regards Figure 16.

- The Things in HLA are equivalent to the Asset layer of RAMI 4.0. They are the physical part of the I4.0 component and can appear at all hierarchy levels from products to field devices like sensor to whole production lines and even factories.
- In HLA, Things are represented by virtual entities in the digital world. This corresponds to the virtual part of the Industrie 4.0 component of RAMI 4.0
- The HLA IoT Device performs the interaction between the physical things and the digital world. In RAMI 4.0 this is a task of the Integration layer

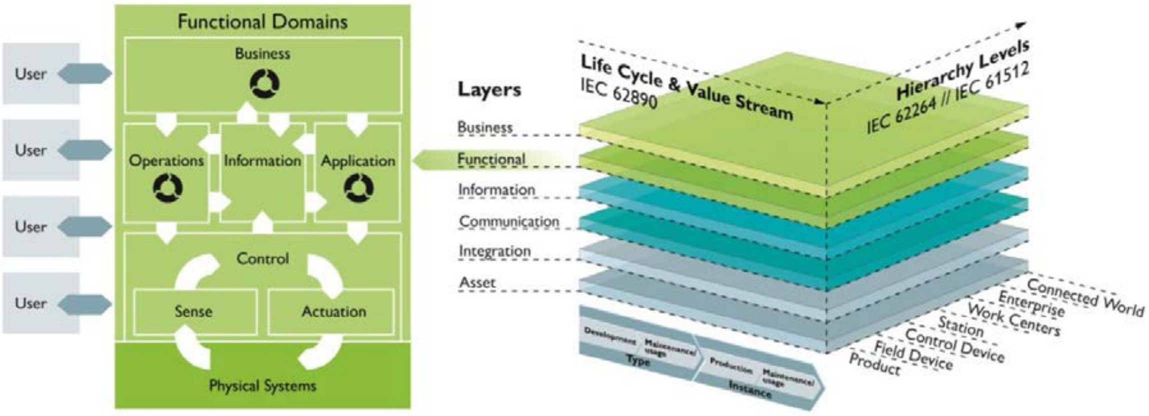
  With the HLA IoT Service the Service Oriented Architecture (SOA) approach of RAMI 4.0 is supported

3.4 Industrial Internet Consortium (IIC)

The Industrial Internet Reference Architecture (IIRA) has been published by the Industrial Internet Consortium (IIC) in the document "The Industrial Internet of Things Volume G1: Reference Architecture" (The Industrial Internet of Things Volume G1: Reference Architecture Version 1.9, 2019) and contains architectural concepts, vocabulary, structures, patterns and a methodology for addressing design concerns. The document identifies the fundamental architecture constructs and specifies design issues, stakeholders, viewpoints, models and conditions of applicability defining a framework by adapting architectural approaches from the ISO/IEC/IEEE 42010-2011 Systems and software engineering—Architecture description standard.

This international standard outlines the requirements regarding a system, software, and enterprise level architecture. The ISO/IEC/IEEE 42010 standard recommends identifying the perspectives of the various different stakeholders that can be: system users, operators, owners, vendors, developers, and the technicians who maintain and service the systems. The aim is to describe system properties as seen from their viewpoint. Such properties include the intended use and suitability of the concept in terms of its implementation, the implementation process itself, potential risks, and the maintainability of the system over the entire lifecycle.

Essentially, the IIRA attempts to identify the most important and common architecture concerns. It then provides an architectural template and methodology that engineers can use to examine and resolve design issues. In addition, the template and methodology suggest ways of addressing the top concerns, allowing designers to glean insights by examining architecture patterns, helping Industrial Internet of Things (IIoT) system designers to avoid missing important architecture considerations and this also helps them to identify design gaps of missing important system functions or components. The core of the IIRA's methodology lies in a set of system conceptualization tools called viewpoints that enable architects and engineers to identify and resolve key design issues. Thus, the IIRA design starts with defining the shapes and forms of an Industrial Internet of Things Architecture by starting with the viewpoints of the stakeholders. These IIRA's viewpoints are arranged in a particular order to reflect the pattern of interactions that occurs between the four elements, because the decisions from a higher-level viewpoint impose requirements on the viewpoints below it. In this sense, the IIRA is a layer model that takes into consideration four different viewpoints (business, usage, functional, and implementation). It focuses on the capabilities from the perspective of the software and their business processes. Each of the four viewpoints outlined in IIRA can be compared with the respective layers on the vertical axis of RAMI 4.0; RAMI 4.0 supplements the model with the axes 'Lifecycle' (with types and instances) and 'Hierarchical Levels.'



Quelle: Plattform Industrie 4.0

Figure 17. The viewpoints of the IIRA related to layers in RAMI 4.0

The IIoT technologies core implemented in IIRA are applicable to the depth and breadth of every small, medium and large enterprise in manufacturing, mining, transportation, energy, agriculture, healthcare, public infrastructure and virtually every other industry. In addition to IIoT system architects, the plain language of IIRA and its emphasis on the value proposition and enablement of converging Operational Technology (OT) and Information Technology (IT) enables business decision-makers, plant managers, and IT managers to better understand how to drive IIoT system development from a business perspective.

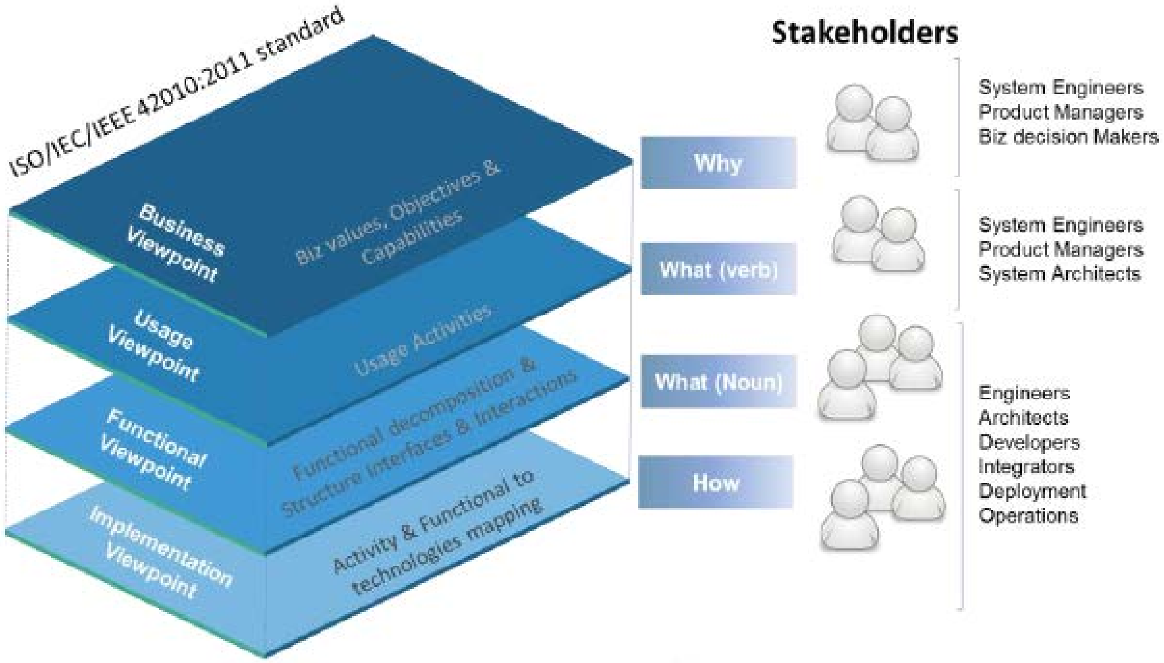


Figure 18. IIRA Architectural Framework

Security Framework (IISF)

Additionally, if the design of the IIoT solution requires considerations within the context of all the viewpoints -crosscutting concerns- as for example security and safety issues, it exists the cross-cutting functions and the system characteristics. The figure below illustrates the relationship between functional domains, cross-cutting functions and system characteristics.

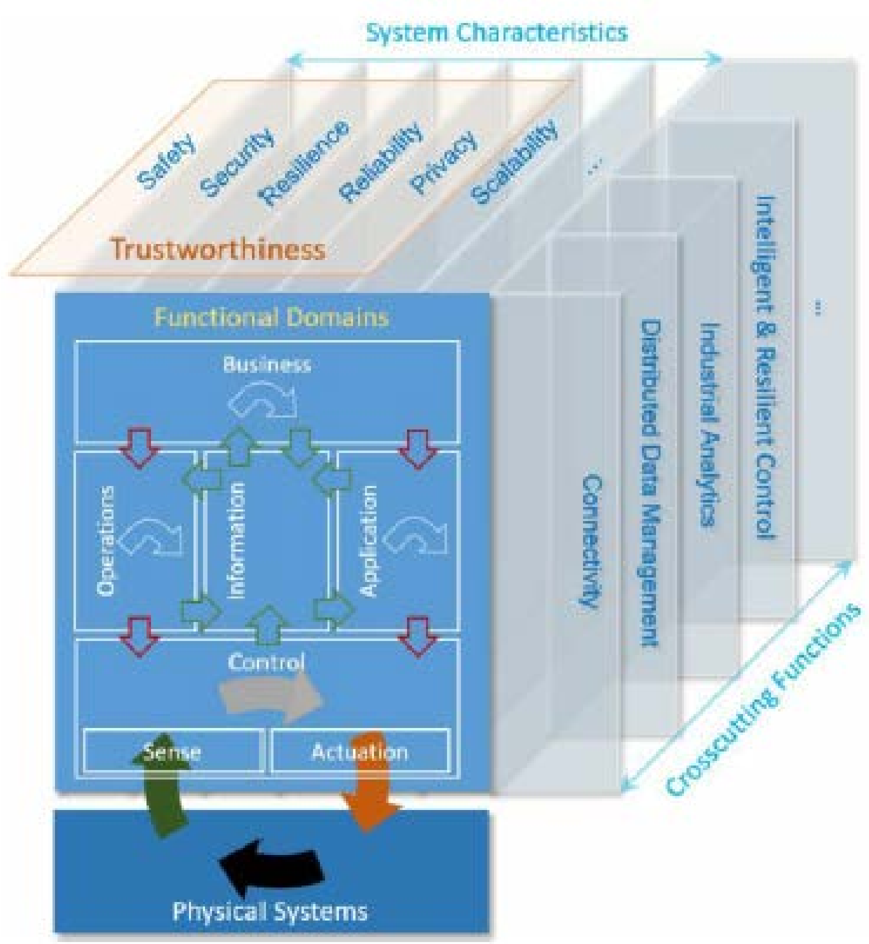


Figure 19. IIRA Functional Domain, crosscutting functions and System Characteristics

IIoT systems are typically systems that interact with the physical world where uncontrolled change can lead to hazardous conditions. This potential risk increases the importance of safety, reliability, privacy and resiliency beyond the levels expected in many traditional IT environments.

The "Industrial Internet of Things Volume G4: Security Framework", (Industrial Internet of Things Volume G4: Security Framework - IIC:PUB:G4:V 1.0:PB:20160926, 2016) published by the Industrial Internet Consortium (IIC), identifies, explains and positions security-related architectures, designs and technologies, as well as identifies procedures relevant to trustworthy Industrial Internet of Things (IIoT) systems. It describes their security characteristics, technologies and techniques that should be applied, methods for addressing security and how to gain assurance that the appropriate mix of issues have been addressed to meet stakeholders' expectations.

4. COGNITWIN Platform/Sensor/Data Interoperability Architecture

4.1 Introduction

The COGNITWIN Architecture as described in the following is expanding on the areas in the previously described reference models in particular with a focus on digital twins, hybrid digital twins and cognitive digital twins. The architectural mappings are done with a foundation in the BDVA Reference Model, and through this it also have mappings to other relevant reference models.

4.2 Concept and methodology

Increased dynamics of the business as well as challenging process environment, force Process industry **to cope with the situations where the numerical models used in simulations are not fine-grained enough and possible drifted** from the real-world assets they represent. Consequently, the Digital twins based on such models cannot deal with all situations which can happen in real-world, **creating a gap (so called "unpredicted uncertainties")** between the digital and real world, as illustrated in Figure 20.

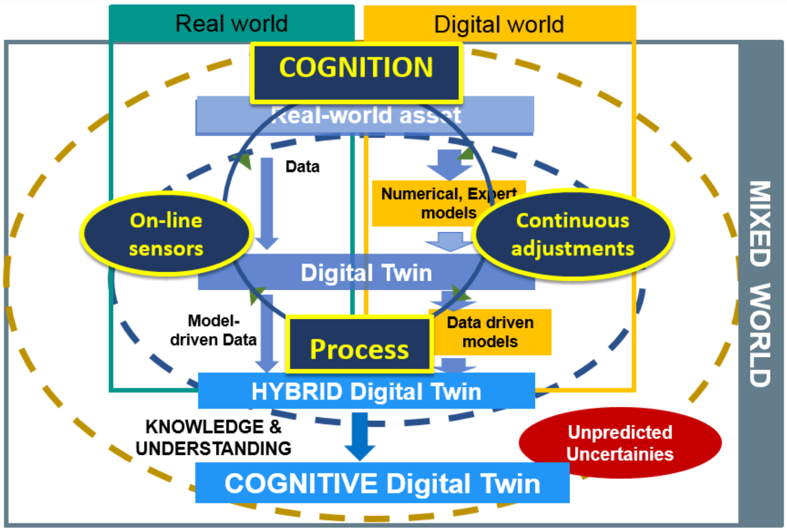


Figure 20: COGNITWIN layered nature

COGNITWIN will resolve this challenge through:

1. Defining **Hybrid Digital Twins** on the top of Digital Twins, by combining data-driven and model-based approaches into **hybrid modelling and analytics**. As illustrated in Figure 20 (cf. inner dotted circle), such a system is not aware of "unpredicted uncertainties" and cannot detect/resolve them. Therefore, such a system cannot be fully automated, i.e. manual human work is required, which is common situation/challenge in the process industry
2. Creating so called "**Cognition-driven automation loop**" which operates on the process data with the goal to keep the system in the **desirable behavior automatically, even in the case of unpredicted events**. This implies a new level of processing where the data cannot be "just" processed (due to unpredictability of the system behaviour) but it **need to be "understood" in the first place, which is the main function of the cognition**. Therefore, we see the cognition as the main process responsible for understanding and resolving unpredictable system behaviour, i.e. the reaction on unpredicted situations. It consists of a blend of Big Data and AI methods that enable:
   - **Cognitive sensing** (cognitive retrofitting of sensors) through automated discovering of the knowledge missing for understanding the situation at hand (answering on how to

understand/discover “unpredicted uncertainties”, since they are neither part of the model nor appeared in past data) and

- **Cognitive control** (cognitive retrofitting of control elements) through automated discovering of the knowledge missing for resolving such a situation (answering on how to resolve “unpredicted uncertainties”, by assuming not having any past experience since they are neither part of the model nor appeared in past data)

3. Introducing the concept of **Cognitive Digital Twins** which are expanding Hybrid Twins with:

- **Self-reflection**, ability to learn “when” (on unknowns) to react on (what is new, previously unknown situation to react on) and
- **Self-adaptivity**, ability to learn “how” to react in unknown situations (adjust own structure and behaviour in reaction to unkno issues/changes)

From the practical point of view, esp. the explosion of IoT has introduced advanced sensing of an industrial asset (product, process, system) that is enriched with the real-time perception of surrounding environment in order to enable real-time control of an asset. As mentioned above, real-world data enables a new quality in interpreting the model. This enables a more efficient operation (run-time) of an asset through increased real world situational awareness.

Physical (real-world) and digital (twin/model) are fully integrated and interlinked with the vision of creating a mixed world where both worlds/paradigms (model, data) are truly interoperable, i.e. the roles are interchangeable/mixed:

- **Model is producing data** (model-driven data);
- **Data is generating a model** (data-driven models).

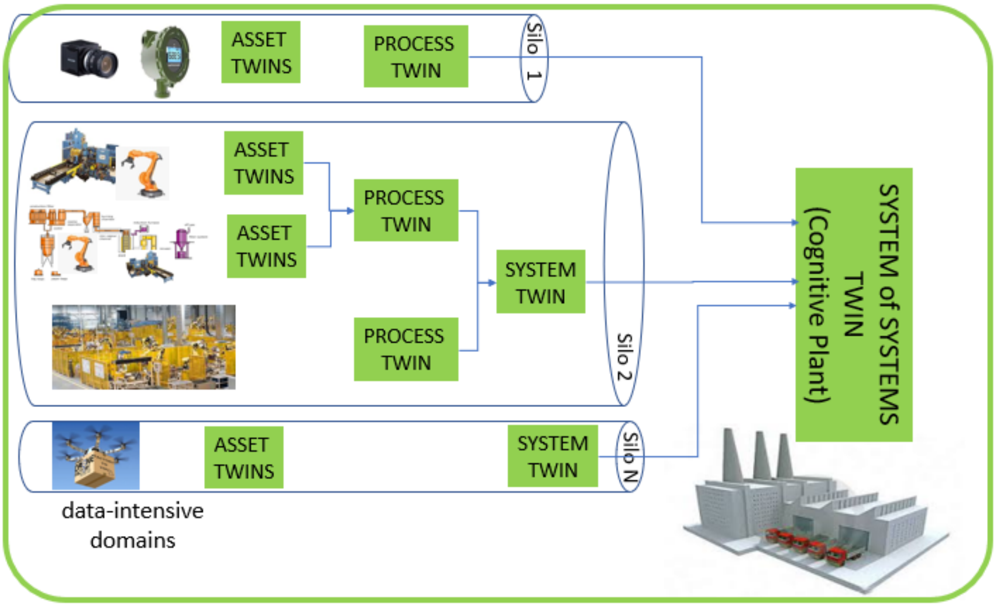


Figure 21 Twins for Cognitive Plants

This will lead to a new type of Digital Twins which are able to reflect own behaviour in both worlds and expose so called self-awareness, realizing the full potential of this ever-emerging technology. Indeed, since self-awareness can be defined as a combination of both “awareness”: from observations (data) to interpretation (models) and from models to data, our hybrid (data/model) approach seems to be very suitable for the realization of self-awareness concept. Moreover, the concept is abstract and can be applied on a hierarchy of twins, creating a hierarchical structure of digital (asset, process, system) twins that sense complex and unpredicted behaviour and reason about dynamic strategies for process optimization, leading to truly cognitive plants that continuously evolve own digital structure as well as collective physical behaviour, as depicted in Figure 21.

From the implementation point of view, the overall concept is based on a multidisciplinary approach to the Cognitive Plants divided in several area of development - Figure 22) that defines the COGNITWIN platform**.** The technology development activity will be achieved gradually and cumulatively through comprehensive analyses, simulator studies and pilot demonstrators where all project members are involved in a collaborative way. The team aims to make use of commercial off-the-shelf components (COTS) and open source capabilities wherever possible and focus development in the areas with low or no maturity, such as safe collision avoidance and situation awareness.

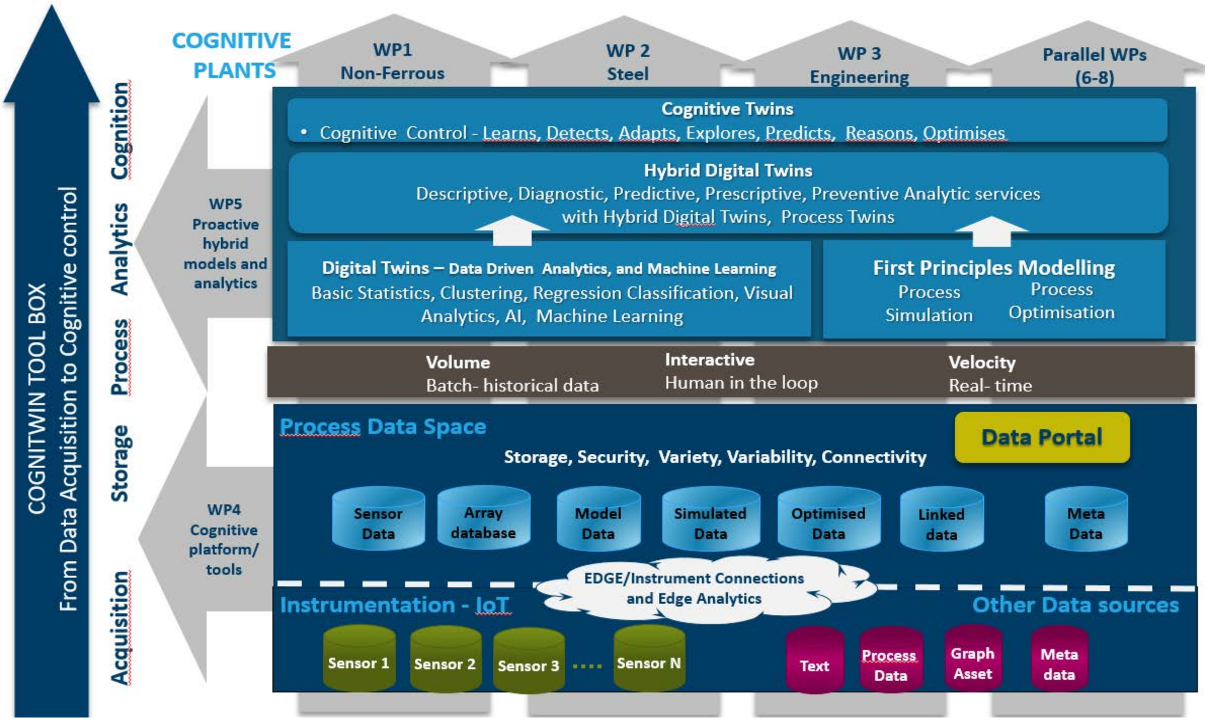

COGNITWIN TOOL BOX
From Data Acquisition to Cognitive control
Cognition
Analytics
Process
Storage
Acquisition
COGNITIVE PLANTS
WP1 Non-Ferrous
WP 2 Steel
WP 3 Engineering
Parallel WPs (6-8)
WP5 Proactive hybrid models and analytics
WP4 Cognitive platform/ tools
Cognitive Twins
• Cognitive Control - Learns, Detects, Adapts, Explores, Predicts, Reasons, Optimises
Hybrid Digital Twins
Descriptive, Diagnostic, Predictive, Prescriptive, Preventive Analytic services with Hybrid Digital Twins, Process Twins
Digital Twins – Data Driven Analytics, and Machine Learning
Basic Statistics, Clustering, Regression Classification, Visual Analytics, AI, Machine Learning
First Principles Modelling
Process Simulation
Process Optimisation
Volume
Batch- historical data
Interactive
Human in the loop
Velocity
Real- time
Process Data Space
Storage, Security, Variety, Variability, Connectivity
Data Portal
Sensor Data
Array database
Model Data
Simulated Data
Optimised Data
Linked data
Meta Data
EDGE/Instrument Connections and Edge Analytics
Instrumentation - IoT
Other Data sources
Sensor 1
Sensor 2
Sensor 3
Sensor N
Text
Process Data
Graph Asset
Meta data

Figure 22: Overall architecture of the COGNITWIN project and the Tool Box

The proposed COGNITWIN platform structure as shown in Figure 22 is made of several building blocks collectively has the potential to enable the Cognitive Plant vision. *The layers and structure of the COGNITWIN Architecture are based on the layers of the BDVA Reference Model as described in the previous chapter.*

From the software engineering point of view, the Platform is offered as a COGNITWIN ToolBox, a set of well-defined programing interfaces which enable building of complex software artifacts for the support of Cognitive Plants.

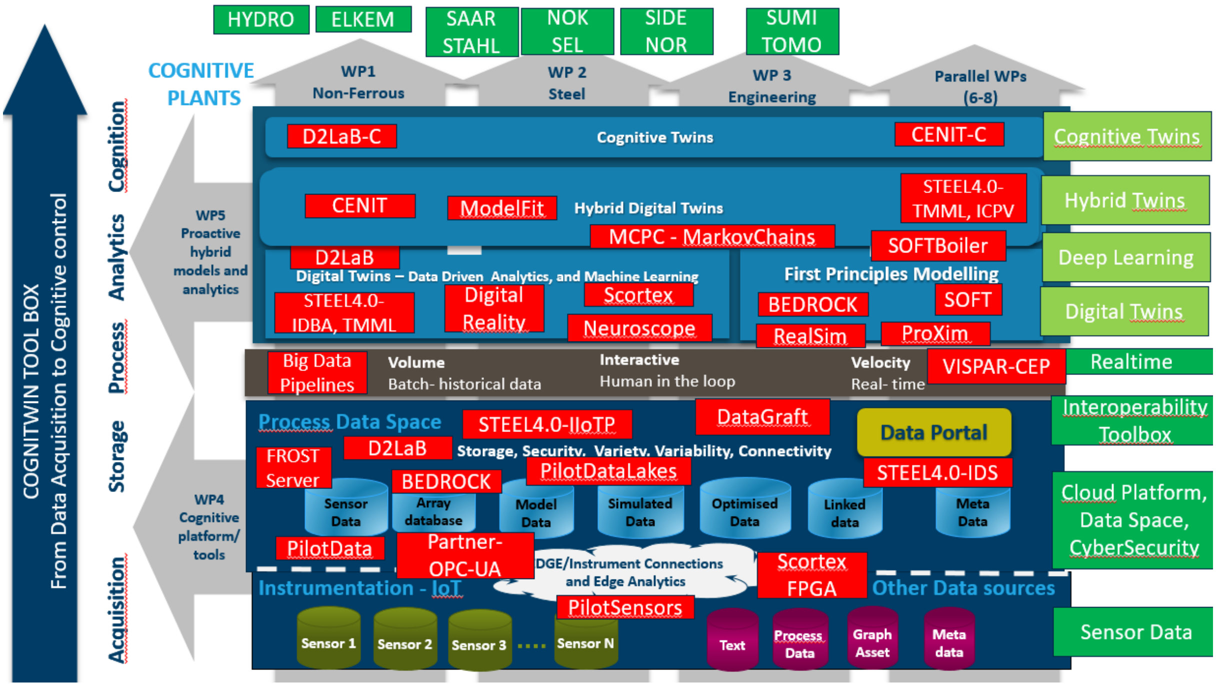


Figure 23 Technology components related to the areas of the COGNITWIN architecture

Figure 23 shows technology components from the COGNITWIN partners mapped into the different areas of the COGNITWIN architecture. In the following part of D4.1 the lower part of this architecture will be described. The upper part is described in the accompanying D5.1 deliverable.

5. COGNITWIN Interoperability Toolbox

5.1 Introduction

This will specify the end-to-end architecture of the COGNITWIN Toolbox. We do not aim at delivering one technically integrated platform for Big Data, IoT & AI/ML technologies, but instead a toolbox. In order to address not only greenfield developments, but also to properly address brownfield integration and interoperability with proprietary solutions and standards, we will take into account different existing platforms (open source as well as commercial). The different pilot partners already have existing IoT platforms in operation. Existing Big Data, IoT and AI platforms will be considered to achieve integration and synergies between the technology stacks and to prepare for access through common interfaces. This will be done with deep involvement of the pilots to ensure that the resulting integration suits the need for commercial use. To ensure flexibility of the COGNITWIN platform, a modular design will be pursued to maximise exploitation opportunities and technology neutrality.

The work will be based on outcomes of the Industrial Internet Consortium (IIC) Task Groups ("Distributed Data Interoperability and Management (DDIM)" co-chaired by Fraunhofer IOSB and "Digital Twin Interoperability") as well as the IoT-EPI projects (Task Force "Platform Interoperability") which some partners are involved in. To realize complex industrial scenarios, Digital Twins (DT) should be capable of capturing characteristics of an asset as specified by the vendor/manufacturer, its state during run time, as well as how an asset interacts with other assets in a complex system. This is usually related to different interoperability problems, which should be resolved using semantic technologies, e.g., semantic models or ontologies.

Main activities in this task are: 1) understand the interoperability requirements, esp. from the SoS viewpoint and self-aware DT context, 2) define modelling and processing activities required and determine their feasibilities, 3) implement Semantic Service and technical testing, and 4) provide a set of guidelines on how this interoperability approach can be applied in similar cases. Main focus will be on semantic models that allow capturing of complex systems in an intuitive fashion, can be written in standardised ontology languages, and come with a wide range of off-the-shelf systems to design, maintain, query, and navigate semantic models. The role of models and modelling languages is critical for DT and hard to overestimate. On the one hand, the models determine how 'close' a DT can mirror its physical counterpart and, on the other hand, models determine what type of analytics can be layered on top of the DT. The outcome is a set of services, ontologies and guidelines that show how the resulting solution can assist in simplifying analytical and machine learning routines for DT.

Next, we will develop basic components for data integration and storage, e.g. generic adaptors, data and model storage, and notifications. Finally, gateways between COGNITWIN services and the relevant open-source platforms and proprietary open commercial solutions will be implemented.
IMPORTANT: the interoperability interfaces to the IIoT platforms used in pilots will be well defined to ensure efficient deployment in pilots. We assume that IIoT platforms have implemented some of the above-mentioned services and provide open interfaces for 3rd party applications. One of the goals of this task will be to provide such an abstraction layer with interoperability interfaces.

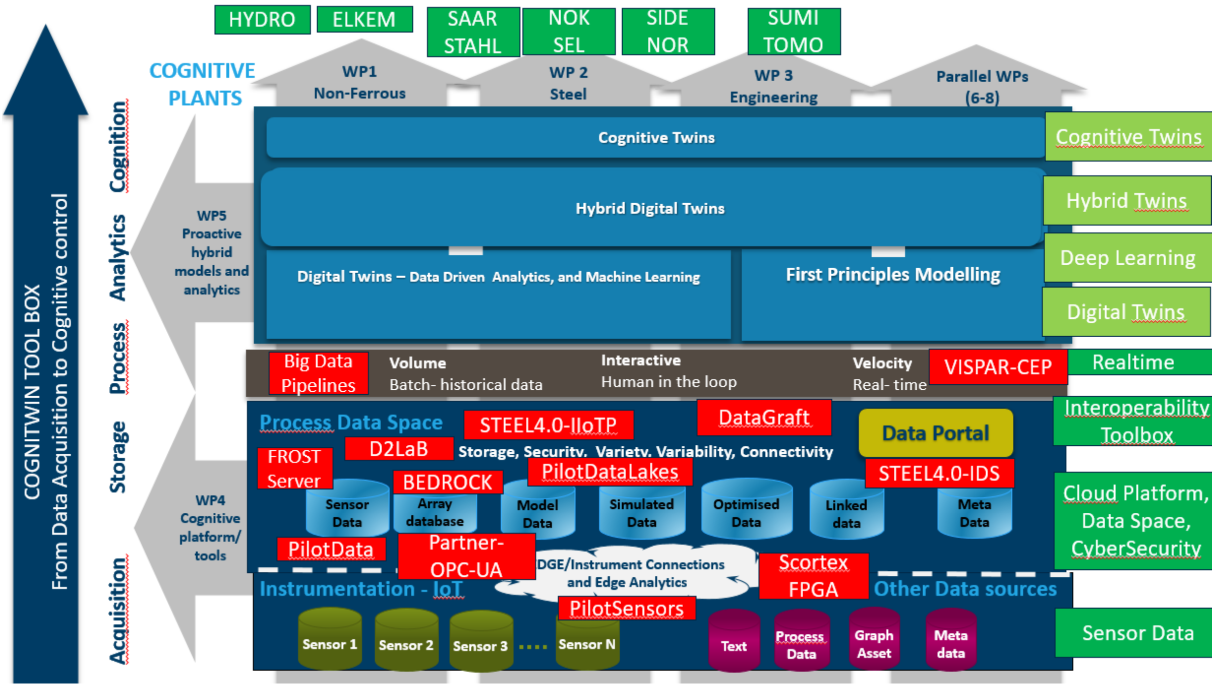


Figure 24 Technology components (in red) related to the COGNITWIN WP4 areas

The following will describe the baseline within the different areas of the WP4 – COGNITWIN – i.e. Interoperability Toolbox, Cloud Platform/Data Space/Cybersecurity, Sensor data management and Realtime data management. – with some of the components shown in red boxes for the different areas above.

The deliverable D5.1 Baseline Hybrid AI and Cognitive Twin Toolbox describes the partner components of the upper part of this figure.

5.2 Partner technologies

5.2.1 Pilot Data, DataLakes and Sensors

The different pilots have already various types pre-existing data collections and data management including data lakes in platforms such as Microsoft Azure. Interoperability with these will be aimed at. Some pre-existing pilot sensors will be used, while analysis and recommendation for new sensors is also supported by COGNITWIN as reported in chapter 9.

5.2.2 Teknopar STEEL 4.0 IIoTP and Industrial Data Security (IDS)

TEKNOPAR will help the development of IIoT Platform as described in Annex 1. TEKNOPAR provides IIoTP component of STEEL4.0 as a generic data acquisition platform with specific functionalities built in for the purpose of integrating data from SWP machinery. It enables both soft real-time stream processing and batch processing. Soft real-time stream processing is conducted at two levels: edge and platform tiers. At the edge, TIA PORTAL is used for PLC programming, and using ProfiNET industrial ethernet protocol, and MQTT protocol, data in JSON format is generated.
TEKNOPAR's IDS can enable data backups for data recovery. Regular and frequent backups will be performed, and data will be replicated to prevent data losses in cases of unwanted situations, such as natural disasters. Data that flow among the STEEL4.0 components might be secured by means of IDS. This is further described in section on Cyber Security in chapter 8, and in Annex 1.

5.2.3 SINTEF DataGraft, Knowledge Graphs and Big Data Pipelines

The SINTEF DataGraft tool provides a foundation for semantic interoperability of data based on Knowledge Graphs and ontologies. There is another component support for the design and execution of Big Data Pipelines through a framework for this. This is further described in Annex 2.

5.2.4 Scortex FPGA Compute platform

Scortex will help the choice of communication to the FPGA compute platform (presented in Annex 3) to bring the data in the right format to the FPGA hardware compute platform for optimised machine learning execution in particular for image analytics.

5.2.5 Cybernetica OPC UA Server

Cybernetica OPC UA Server, described in Annex 4 can be used to collect and distribute real-time data in a standardized way to any application that implement an OPC UA Data Access client interface.
The Cybernetica OPC UA server can easily be extended to collect data from various data sources or proprietary protocols via plugins. OPC stands for “Open Platform Communication” and UA stands for

"Unified Architecture", which is the newest version. It is a standardized way to exchange data. The OPC UA Data Access specification is maintained by the OPC Foundation[11].
More of the Cybernetica tools are described in deliverable D5.1.

5.2.6 SINTEF BEDROCK and SOFT

The SINTEF BEDROCK and SOFT components provides support for process data management and interoperability, and also support for various types of analytics. These are more elaborated in the accompanying deliverable D5.1 Baseline Hybrid AI and Cognitive Twin Toolbox.

5.2.7 Nissatech D2Lab

Nissatech D2Lab provides a stack of technologies from initial data collection through a message broker architecture with associated data ingestion and storage up to data analysis.
This is further described in Annex 5.

5.2.8 Fraunhofer VISPAR-CEP and FROST Server

Fraunhofer VISPAR is a Complex Event Processing (CEP) solution. It is based on the popular CEP Engine Siddhi and uses the IOSB FROST Server with a standardized data model for storage. The standard used for data modelling is the SensorThings API. With this model, different sensors and their data can be modelled. Sensor connections can be supported through OPC UA, HTTP or MQTT. These technologies are further described in Chapter 10 and in Annex 6.

6. Digital Twin Cloud Platform, Data Space and Cyber Security

6.1 Introduction

A cloud platform will be used to gather data from plant assets and will be utilized to arrive at real-time operational insights. In addition to infrastructure services that manage the connectivity, authentication, and the edge devices, the cloud level will provide basic services and business services. The basic services are used to ingest data in the cloud, and to clean, integrate and store data that is received. The business services are intelligent, data-driven services, such as data analytics or asset performance management services (to be developed in WP5) to detect trends and/or create insights about assets and to find the root causes of failures.
Security is vital at all levels and ensures that data/information/services are always managed in a secure way. The goal of this task is not to prescribe one approach for the whole project, but to help selected parties to work with each other, without being hindered by data and model access issues. Existing systems of pilots will be complemented with respect to a security and privacy framework including means to specify data and model access and usage policies including the enforcement of these policies. Access to data and models will be provided as a service controlled by an attribute-based access control mechanism that will employ user specified policies to determine who can access which resources and for what purpose. Based on the results of the International Data Spaces initiative (IDS), we will develop IDS connectors and the methods and tools for usage control by specifying the way in which data and

[11] http://www.opcfoundation.org

models should be handled after access has been granted. The proposed solution will enable tracking and tracing of data/model usage in a transparent manner across different systems.

This task is also responsible for the support of the Digital Twin API (DT API). We will enable creation of DT on different levels: asset, process, system. We will provide a catalogue of micro services (API) for supporting the entire life cycle of DT: creation, validation, update and combination (merging). In addition, supporting services required for the storage, sharing, discovery and tracking (i.e. Digital Thread) etc. will be developed. Several alternatives will be provided for micro services. As an example, a DT can be created manually via UI, semi-automatically via extraction from AML models, JSON files, or automatically by combining/merging DTs.

The DT API will be a harmonised approach related to the DT API interoperability work being done in the Industrial Internet Consortium Digital Twin Interoperability group – to ensure interoperability with DT APIs from different platforms like GE Predix, Microsoft Azure Digital Twin, Cognite, ABB Ability APIs – as a preparation also for future possible standardisation.
Run-time aspects will focus on the behaviour of a DT once it has been configured. We will develop services to monitor physical assets, classify current behaviour and reacting in the case of some issues, through understanding the issue (route cause) and suggesting improvements. A DT will be continuously updated based on real-time data or results of data-driven and knowledge-based analytics models

6.2 Digital Twins, CyberSecurity and Data Spaces

6.2.1 Digital Twin

The Industrial Internet Consortium (IIC) defines digital twin as follows

> A digital *twin is a formal digital representation of some asset, process or system that captures attributes and behaviors of that entity suitable for communication, storage, interpretation or processing within a certain context.*

According to this definition, the central aspect of a digital twin is the formal digital representation of a thing (asset, process or system) including its attributes and behaviours.
Beside that formal digital representation of a digital twin, which we further on address as resource description, we identified two more essential abstract functionalities that a digital twin platform must provide: resource discovery and resource access. Resource discovery describes the fact that to be of any use, these representations or descriptions must be known to other components or systems for them to be aware of their existence. The final step of exploiting a digital twin is to actually access its properties and services (called behaviours in the IIC definition of digital twin). These three criteria define the core set of functionalities a digital twin platform must provide.

The current landscape on digital twin platforms and standards is scattered. There exist a multitude of different, partially overlapping or competitive standards covering one or more of the three core functionalities. According to the Gartner Hype Cycle for Emerging Technologies from August 2018[12], the concept of digital twin reached the "peak of inflated expectations" at this time which cause many SDOs to pursue this topic and release their own standard. Since 2019, it seems that consolidation starts to happen as there is more and more communication and collaboration between the SDOs. In this context, we are currently trying to establish some kind of classification scheme for digital twin standards and platforms to identify their similarities and differences.

As this is still work in progress, we present in the following our thoughts and initial findings on that classification scheme followed by some relevant standards and platforms related to digital twins that we consider to compare based on this classification later in the project.

The classification scheme is currently centred around the three core functionalities resource description, resource discovery and resource access. Additional aspects will be considered, e.g. security & access control, versioning support and existing software landscape. In the following we present details on the three core functionalities.

Resource Description
According to the IIC definition of digital twin, this description must be formal and digital. However, more importantly it must be machine-readable und -understandable. This means it must contain a certain level of semantics. Also, this implies that there must be some kind of language or meta model available to express the resource description in. With a need of defining a language a lot of questions arise that can be used for classification, e.g.

- How to model a resource/asset?
    - support for properties?
    - support for services?
    - support for events?
- What kind of identifiers are used to identify resources?
- What is the underlying type system?
- What features should the language support?
    - Interlinking of resources?
    - Explicit support for geo-spatial data?
    - Explicit support for temporal data?
    - Explicit support for historic data?
- How to support re-use of existing external definitions?
- What serialization formats are supported?

Resource Discovery
Once resources have a digital representation in form of a semantic description, they need to be discovered by software components or systems. Most standards and platform use a (central) repository for this where the resources are registered by sending their resource description. Another way for resource discovery could be to do it in a decentralized manner, e.g. by peer-to-peer

[12] https://www.gartner.com/smarterwithgartner/5-trends-emerge-in-gartner-hype-cycle-for-emerging-technologies-2018/

communication via a special network protocol. For discovering resources via a repository, this repository needs some kind of API to access it. Those APIs are proprietary, i.e. there has not yet been and agreed upon standard.

Regarding resource discovery, digital twin standards and platforms can be classified by the type and capabilities of their API, e.g.

- Type of protocol used (typically HTTP)
- Only listing registered resource vs. searching them
- Query language expressivity
  - Can data and meta data be combined in one query?
  - Support for geo-spatial queries?
  - Support for temporal queries?
  - Support for following links between resources?

Resource access

Once a system identifies a resource of interest it needs to access this resource (read/write/update a property, invoke a service, subscribe to an event) to make any use of it. This means the resource must be accessible through an API. Typically, this API is defined as part of the standard/platform and is the same for all resources. Another possible way is to allow each resource to define or re-use its already existing API and to formally define this API as part of the resource description. The first classification criteria is therefore using a defined API vs. describing a (custom) API. Further classification criteria are, e.g.

- Which communication protocols (e.g. HTTP, WebSocket, AMQP, MQTT, OPC UA) are supported?
- Is the set of communication protocols fixed or extendible?
- Is there support for cross-protocol communication?
- If the standard/platform supports services
  - Is the semantic of service invocation defined?
  - Is there explicit support for error handling, i.e. is the behavior in case of an error defined?

Existing Standards and Platforms

W3C Web of Things Thing Description[13]

This is a W3C Candidate Recommendation standard published in January 2020. It is motivated by the Internet of Things (IoT) but the problems at hand, resource description, resource discovery and resource access, are the same for IoT and digital twin. Although this standard addresses only resource description it is listed here because of its rather unique approach of doing so. Instead of defining an API that all resources must implement, the W3C Web of Things Thing Description (WoT TD) allows resources to keep their already existing APIs and describe them as part of their resource description. This approach allows to easily integrate any kind of (legacy) device without the need to write any code or adapter software.

[13] https://www.w3.org/TR/wot-thing-description/

WoT TD also has a focus on semantics and is based on JSON-LD/RDF. By using this basic technology of the semantic web, it allows to easily re-use most of the existing vocabularies on the web.
The W3C Web of Things Working Group is currently re-chartering and plan to work on a resource repository called ThingDirectory.

The Eclipse Thingweb project provides an open-source implementation of an IoT platform based on the WoT TD. Although it is called an IoT platform, it provides all three basic capabilities of a digital twin platform and can therefore also be classified as a digital twin platform.

Platform Industrie 4.0 Specification: Details of the Asset Administration Shell[14]

The Platform Industrie 4.0 is a network of German industrial and IT companies as well as research organizations in cooperation with the German government. They defined the Asset Administration Shell (AAS) as the central concept for digital twins. The AAS realizes the concept of resource description in a very detailed and formally defined way. It provides many features such as support for different kind of identifiers, semantic definitions and data formats (e.g. JSON, XML, RDF, OPC UA, AutomationML). Currently it does not support resources offering events as well as support for historical values.

Besides the standard document there was an innovation project called BaSys 4.0 funded by the German Federal Ministry of Education and Research (BMBF) with the goal to define a reference architecture for digital twin platforms based on the AAS. The project ended in 2019 but was continued in the follow-up project BaSys 4.2 which will end in mid-2022.
The Eclipse BaSyx project[15] is the official reference implementation of the BaSys architecture and provides a feature-rich digital twin platform based on the AAS concepts.

ETSI Context Information Management: NGSI-LD API[16]

ETSI is an international non-profit standardization organization in the telecommunications industry. The ETSI Context Information Management (CIM) API, also called NGSI-LD API, is a successor to the OMA NGSI 9 and 10 interfaces and the FIWARE NGSIv2 API. Although this standard is not designed explicitly with digital twins in mind, it covers most of the core functionality needed for the realization of a digital twin platform.
Regarding resource description, the NGSI-LD API does not support services or events but provides a strong support for semantic description of resources. Furthermore, it provides a powerful query language for resource discovery including support for geo-spatial, temporal and historical queries. However, no direct resource access is allowed but only through the central API.

There exist two projects currently in the progress of implementing the NGSI-LD, the Orion Context Broker Linked Data Extension and the Scorpio NGSI-LD Broker.

[14] https://www.plattform-i40.de/PI40/Redaktion/EN/Downloads/Publikation/Details-of-the-Asset-Administration-Shell-Part1.pdf?__blob=publicationFile&v=5
[15] https://www.eclipse.org/basyx/
[16] https://www.etsi.org/deliver/etsi_gs/CIM/001_099/009/01.02.01_60/gs_CIM009v010201p.pdf

Eclipse Ditto[17]

Eclipse Ditto is an open-source digital twin platform. It was created in mind-2017 and is primarily driven by industry partners like Siemens. It can be seen as a bottom-up approach to develop a digital twin platform putting running code over feature-completeness and fancy features.

The resource description is based on a meta model that contains only two classes, Thing and Feature, but can be extended by a semantic description based on the Vorto language from the Eclipse Vorto project[18]. It offers a resource repository with and RQL-based query language and support for multiple communication protocols (e.g. HTTP, WebWocket, MQTT) as well as sophisticated security mechanisms.

6.2.2 Cyber Security

Factories are sensible to security about data by 2 angles:

- Data produced have a high value
- Data drive the production

To secure data from end to end it is important to respect some basics security rules.

6.2.2.1 Physical Database Security

Data are stored on server. It is important that the servers are physically secured. Usually we use cloud providers to address that topic. Cloud providers like Amazon, Google, Microsoft, IBM (softlayer) provide the capabilities to choose the server configuration (processor, memory, and so on ...). The service they propose provides machines in a secure area, the servers are cooled properly and managed to ensure the highest availability. The cloud providers provide additional services like a local copy of data in case of disk failure. Data replication over a different geographical region in case of natural disaster of fire in the datacenter. Cloud providers allows the customer to use the operating system and the software they want to use. It means that the software security remains the responsibility of the customer. Depending on the case, cloud providers may also provide software or features to add security.

The topic to consider by using those cloud providers is that they have specific TOS ("Term of Use") that should be carefully considered. It is also important to consider the geographical position of the server ordered since legal rules may change from countries to countries.

6.2.2.2 Backing up the data

It was already briefly presented in the previous section. Problem on data appear frequently. It may be malicious like a hacker that corrupt your data or even a malfunction that causes a loss of data. This is important to perform backups regularly in a way that if data is lost. It is possible to recover it or the whole databases more or less quickly.

[17] https://www.eclipse.org/ditto/

[18] https://www.eclipse.org/vorto/

6.2.2.3 Up to date software

Software are regularly updated thanks to continuous integration and continuous deployments. Those updates may contain additional features or bug fixes. It sometimes contains security breach fixes. It is important to apply those updates to ensure the security of the system. Usually software updates come with a changelog that indicates the modifications, additions, changes, deletions (MACD). By reading those, it is easy to know if an update is mandatory for security or not.

6.2.2.4 Default settings

Software tools came with default settings. Those default settings are well known by hackers and are not made to ensure the highest security. It is important that a trained person configure the tools to ensure the security.

6.2.2.5 Access

Tools usually needs a sign in process to be used. It is important to ensure accesses to member to permit them to work properly. On another hand, it is important to open only needed accesses to avoid human error or attacks. A person that do not need access to specific data should not be able to access the specific data. The permissions are usually based on group rights to facilitate the work of the tool administrators and to avoid managing right permissions user by user that may be time consuming and error prone.

6.2.2.6 Data value

Security has a cost. It is logical that the amount of time and money used to secure data worth it. Then an important pre-work to do is to identify the value of the data and analyze the risk. In case of factories, some data are used into retroactive loops. Corrupting those data will have an impact of the retroactive loop. By clarifying critical and sensitive data, it is possible to then spend an extra effort to ensure the security on those data.

Another solution consists to reduce the value of the data. It may be done by anonymizing the data for example. That way the data looks like a list of values that do not have sense for a person that do not know the structure of it. It is also possible to reduce the value (or risk here) by adding security checking. In the case of corrupted data, adding security checks to avoid weird values may reduce the impact of corrupted data. Those security should send a notification or an alert to inform a human that an issue happen and may need actions.

Anonymization of data is mandatory for non-production usage. Usually, development needs a real look like data to test and ensure the proper working of a tools before deployment. Since those data may be copied and used on not enough secured computer and tool. Anonymisation is requested to avoid this kind of troubles.

6.2.2.7 Data encryption

As we usually do for our personal computers. Encryption is a solution that may be used on data. Encryption is a process that permit to make the data unreadable without a key. This allows, in the case of breach, that the data are unreadable, so it is unusable. This feature is already available on general personal computer and is more important on industrial computers.

6.2.2.8 Database monitoring

One of the best ways to ensure data security consist to monitor the tools that allows to store and read data. By controlling the accesses, it is possible to detect abnormal usage of the data. This abnormal use can be caused to a bad usage of a tool or by a malicious intent.

6.2.3 International Data Spaces Association – IDSA

The International Data Spaces Association (IDSA)[19] is the evolution of IDS (Industrial Data Space) which itself was an initiative lead by Fraunhofer ISST, in cooperation with ATOS, T-Systems, and the idea is promoted by the German Federal Ministry of Education and Research. IDSA is characterized by the focus on information ownership, with the aim of enabling clear and fair exchanges between data providers and consumers. To this end it suggests a reference distributed architecture that accomplishes this goal (IDS Reference Architecture Model Version 3.0[20]).
Broadening the perspective from an individual use case scenario to **interoperability** and a platform landscape view, the IDS Reference Architecture Model positions itself as an **architecture that links different cloud platforms through policies and mechanisms for secure data exchange and trusted data sharing** (through the principle of data sovereignty). Over the IDS Connector, industrial data clouds, individual enterprise clouds, on-premise applications and individual, connected devices can be connected to the International Data Space ecosystem (see Figure 25).

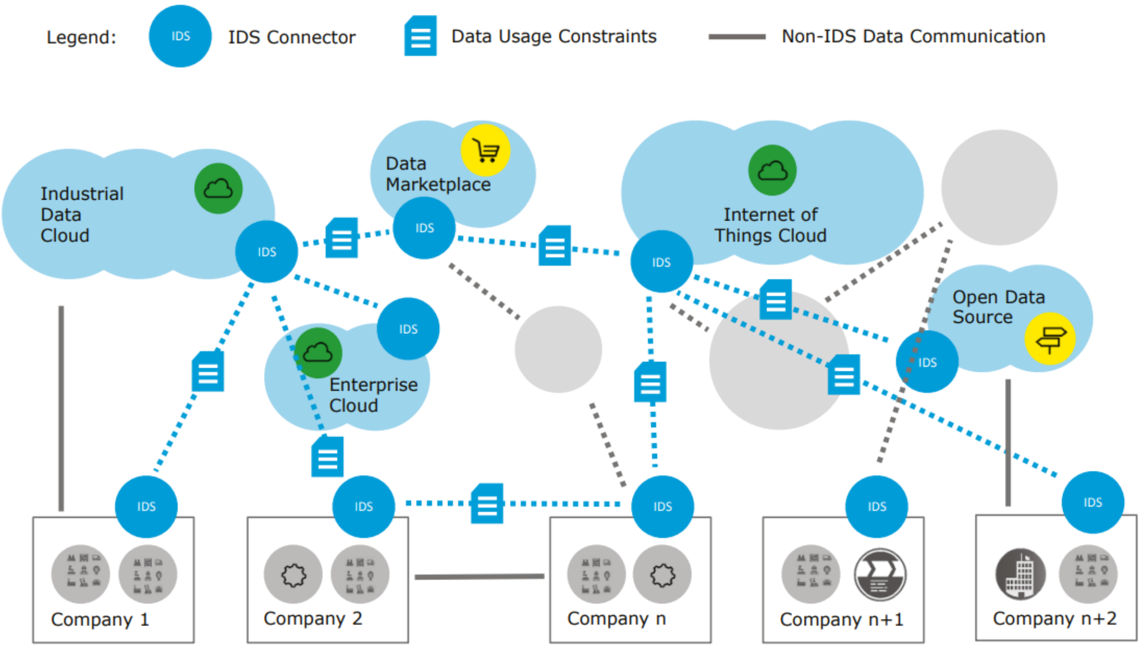


Figure 25. International Data Spaces connecting different platforms

This IDS Reference Architecture Model (IDS-RAM) is described using multiple layers, such as business, functional, process, information and system; between and common to all these layers are transversal functionalities that foster security, certification and governance, as illustrated below.

[19] https://www.internationaldataspaces.org/
[20] https://www.internationaldataspaces.org/wp-content/uploads/2019/03/IDS-Reference-Architecture-Model-3.0.pdf

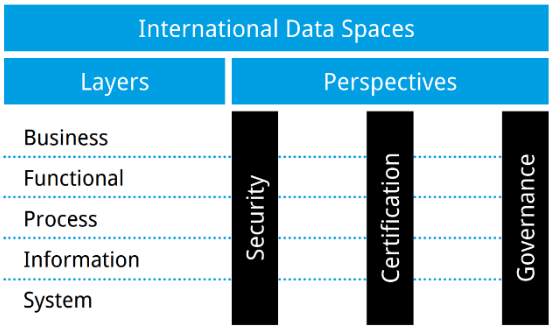


Figure 26 General structure of IDS Reference Architecture Model

The Business Layer specifies and categorizes the different roles which the participants of IDS can assume, and it specifies the main activities and interactions connected with each of these roles. The Functional Layer defines the functional requirements of IDS, plus the concrete features to be derived from these. The Process Layer specifies the interactions taking place between the different components of IDS; using the BPMN notation, it provides a dynamic view of the Reference Architecture Model. The Information Layer defines a conceptual model which makes use of linked-data principles for describing both the static and the dynamic aspects of IDS' constituents. 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. Comparing IDS to IoT-A ARM, the former focuses its specification of the roles for actors within the business layer that would govern the data flows between different domains or data spaces. As such, key participants (actors in the system) would be the Data Owner, Data Provider, Data Consumer, Data User or Broker Service provider. The complete landscape of roles, their functionalities and relationships result in a model depicted in the following Figure 27.

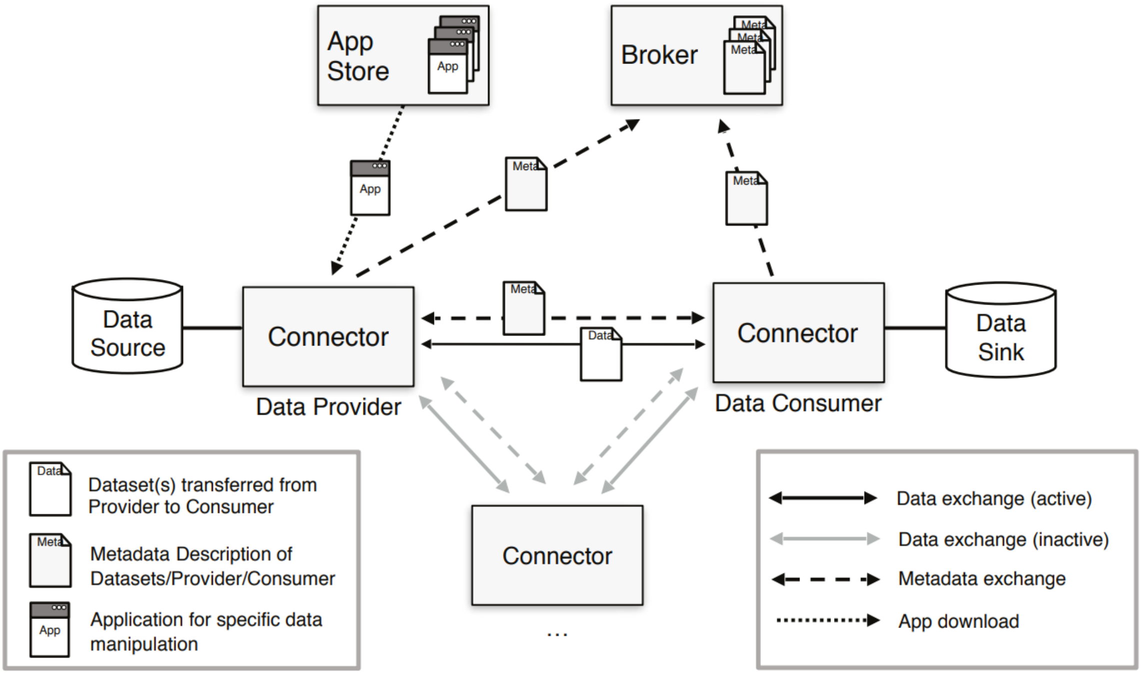


Figure 27. Interaction between technical components of IDS Reference Architecture Model

The **Connector** is the central technological building block of IDS. It is a dedicated software component allowing Participants to exchange, share and process digital content. At the same time, the Connector ensures that the data sovereignty of the Data Owner is always guaranteed. The **Broker Service Provider** is an intermediary that stores and manages information about the data sources available in IDS. The activities of the Broker Service Provider mainly focus on receiving and providing metadata that allow provider and consumer connectors to exchange data. The **App Provider** role is optional in IDS, and its main role is to develop applications that can be used by both data providers and consumers in the data space. Applications are typically downloaded from the remore app store, and run inside the containerized connector.

Establishing **trust for data sharing and data exchange** is a fundamental requirement in IDS. The IDS-RAM defines two basic types of trust: 1) Static Trust, based on the certification of participants and core technical components, and 2) Dynamic Trust, based on active monitoring of participants and core technical components. For data sharing and data exchange in the IDS, some preliminary actions and interactions are required. These are necessary for every participant, and involve a Certification Body, Evaluation Facilities, and the Dynamic Attribute Provisioning Service (DAPS). Figure 28 illustrates the roles and interactions required for issuing a digital identity in IDS, and these interactions are briefly listed here:

1. Certification request: This is a direct interaction between a participant and an evaluation facility to trigger an evaluation process based on IDS certification criteria.

2. Notification of successful certification: The Certification Body notifies the Certification Authority of the successful certification of the participant and the core component. Validity of both certifications must be provided.

3. Generating the IDS-ID: The Certification Authority generates a unique ID for the pair (participant and component) and issues a digital certificate (X.509).

4. Provisioning of X.509 Certificate: The Certification Authority sends a digital certificate (X.509) to the participant in a secure and trustworthy way and notifies the DAPS.

5. Register: After the digital certificate (X.509) is deployed inside the component, the component registers at the DAPS.

6. DTM Interaction: The Dynamic Trust Monitoring (DTM) implements a monitoring function for every IDS Component, and DTM and DAPS then exchange information on the behavior of the component, e.g. about security issues (vulnerabilities) or attempted attacks.

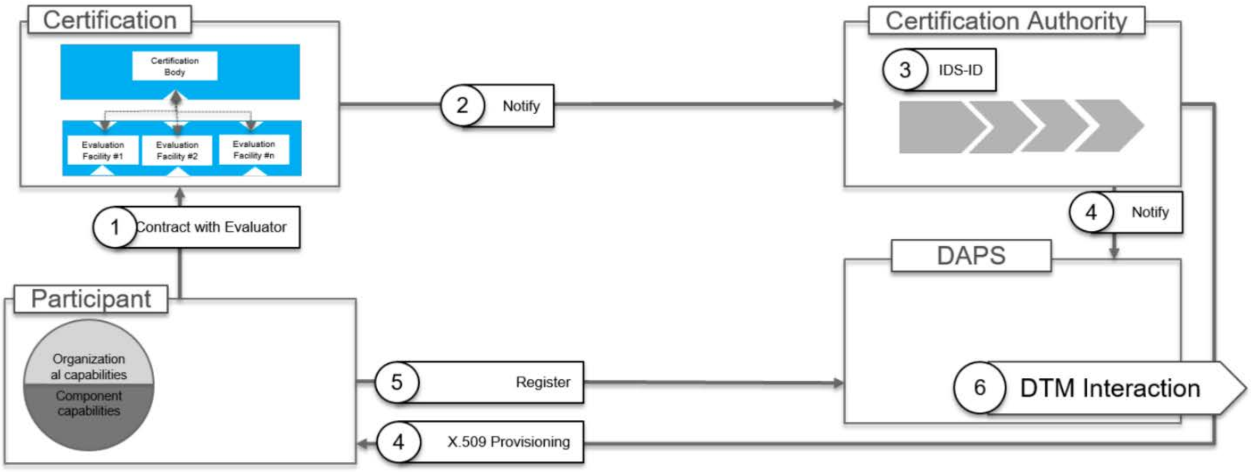


Figure 28. Interactions required for issuing a digital identity in the IDS

The IDS reference architecture contains an internal structure that is strongly supported by the containerization for the development of IDS connectors. It relies on IDS Communication Protocol to enforce security in data exchanges, as it is depicted in the figure below.

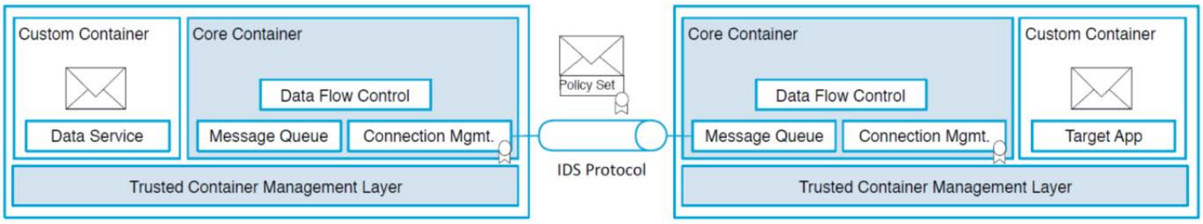


Figure 29. Enforcing security in data exchanges: the IDS Communication Protocol

To sum up, the security implications that guarantee reliable and trusted transfer of information between independent entities in IDS are the following:

- **Secure communication.** The concept of Trusted Connector is introduced as depicted in the figures above.
- **Identity Management** for identification/authentication/authorization enhancing. There is use of certificates issued by a Certificate Authority (CA).
- **Trust Management** that uses Cryptographic methods such as PKI (Public Key Infrastructures).
- **Trusted Platform** for trustworthy data exchange, which defines the minimal requirement for Security Profiles that should be verified by IDS connectors. It also defines the capacity to perform integrity verification of the rest of the involved connectors.
- **Data Access control**. IDS defines authorization criteria based on the previously defined Security Profiles.
- **Data Usage Control**. IDS checks and regulates that data processing is according the intended purposes defined by the original data owner.

7. Sensors, Understanding Sensor Data & Quality Assurance

7.1 Introduction

This will focus on assisting industry in the selection of suitable sensors/instrumentation, accessories, parts and components required for integration/retrofitting and coping with inline measurement environments.

Examples of potential sensors include open-path optical techniques for gas composition, spectroscopy for gas- or particle analysis, x-ray fluorescence for elemental detection, thermal imaging or pyrometry for high-temperature sensing, acoustic sensors for fluid measurements, and 2D/3D imaging for feature identification and tracking. Sensors of various TRL levels need to be considered here as we will likely be addressing new measurement points in the process, therefore, it is not guaranteed that commercial sensors are available. In general, commercial "off-the-shelf" sensors should be prioritized, and development effort should be focused on use-case optimization, data quality and integration.

Next, the task will determine if indirect, "soft" measurements can be used to fill gaps in sensor availability. This is relevant in the cases where direct measurements are too difficult or costly, or where existing sensor data can yield the needed accuracy. Finally, the task may involve retrofitting, adapting or tweaking existing sensors to achieve high data quality.

If necessary, advanced scientific instrumentation/prototypes such as FTIR, MS, GC may be installed in parallel with commercial sensors. These installations will be temporary, to investigate the performance and properties of cheap/commercial sensors, and to develop calibration schemes for permanent installations.

We will also perform edge analytics (local sensor interpretation) so that a first level of processing is performed for individual sensors before pushing data to the process analytics level, where all sensors and data sources are analysed in context.

We will ensure sensor data quality, i.e. identify incomplete, corrupt or suspicious data that needs attention before it can be trusted for use in analytics. It is common for sensor clusters of this size to have frequent data integrity issues. These include gaps in time-series data, sub-optimal sensor performance (drift etc.), or unexpected events, all of which may result in an incomplete or inaccurate cognitive twin. Establishing trust requires both data model assurance and analysis of time series data to see if there are any detectable issues. Planned data integrity investigations will also look for issues with calibration, connectivity, or physical/installation issues. For example, if a sensor is sometimes blocked by a moving crane, this must be understood and accounted for.

Data quality will be assessed on single- or multivariate data, using techniques ranging from simple pattern recognition to multivariate unsupervised anomaly detection (see Task 4.4).

7.2 State of the art: Cognitive Sensing

Where advanced sensor technology is needed in an industrial process, there are often many commercially available systems; however, they are rarely suitable for direct installation in a process. Many advanced sensors are not developed for in-line operation, or they cannot handle the harsh environment in or around the process. Challenges that often prevent direct measurement are difficult access, high temperatures, corrosive environments, scaling and slagging, and unacceptably high system cost or complexity.

The digitalisation trend across process industries is driven by

a) Development of new sensors and sampling technology: miniaturization, improved reliability, increased performance/sensitivity, edge analytics; and
b) Research and development into new data-analytical methods: fast algorithms for real-time data streams, IoT, improved CPU architecture and capabilities, algorithms for multi-collinearity in process data, methods for separating of predictive and orthogonal information, and analytical methods for soft sensing.

7.2.1 Approaches to advanced sensor development

The task will lean on SINTEF's considerable experience with applying state of the art measurement technologies in challenging industrial environments. The common approach is to build bench-scale or pilot systems to collect real data and use experimental analytics to study the effect of various process parameters. Although, such setups provide useful information, they do not consider the context of the overall process. The grand picture of how different parameters interconnect is missing or must be pieced together by local experts. This is where COGNITWIN offers potential by looking at the full process.

Another approach is bringing advanced scientific instrumentation such as FTIR, MS, GC to industrial sites, enabling direct measurements of e.g. chemical composition in order to study reactions and establishing mass/energy balances. Important information can be discovered, ranging from emission estimates for environmental considerations to in-depth process understanding.

In COGNITWIN, we will look at the following types of instrumentation:

1. Research-grade instrumentation that provides detailed data across a wide spectral range, especially when optimum frequency bands and required sensitivity are unknown.
2. Calibrated, high-accuracy, sensors tailored to measure specific parameters in the measurement situation/environment, e.g. quantum cascade laser systems for specific gasses.
3. Lower-cost sensors with IoT-capabilities for distributed parallel sensing. These will often be chosen and configured based on experience from instruments type 1 and 2.

Our approach will consider robustness, capital cost and maintenance costs as critical factors.

7.2.2 Approaches to data quality assurance

Process data often has unknown quality and unpredictable variation. Although the amount of data is large, it is not given that it representative or accurate enough for modelling. Therefore, an important prerequisite for data analytics is to ensure the correct data quality for both on-line and at-line (laboratory) measurements.

Data quality can be affected by sampling procedures (representative sampling), sample processing (laboratory measurements), operator skills, and the quality of the sensor itself. The important thing is that data is relevant to the process being studied. Theory of Sampling and Measurement Systems Analysis are two key subject areas designed to evaluate and ensure good data quality.

COGNITWIN will use exploratory analysis on existing process data to generate new knowledge and new hypotheses and identify the need for better measurements or new sensors. The methods will be based on multivariate hybrid data-analytics, which enables better understanding of the streams of variations in multidimensional data streams through a novel approach of predictive clustering. It combines on-line clustering of the variations in streams and the predictions of anomalies that highlight the need for improved sensing.

7.3 Partner technologies - Sensors

SINTEF will be the technology partner for choosing and evaluating new sensors for all pilots. Quality management of sensor data will be carried out in collaboration with all technology partners.

Scortex will be a technology partner to choose the cameras that may provide data in the proper format to run on the FPGA compute platform (presented in annex 7)

7.4 Pilot requirements - Sensors

The need for sensor data varies significantly between COGNITWIN pilots. Some pilots will use physical data – e.g. temperature, flow rates, chemical analysis – whereas others require machine vision techniques. Some will use both. Pilots also have varying levels of sensor readiness, with some pilots needing significant hardware development and some needing few or no upgrades. The nature of the process for each pilot will not be described here, but can be found in the respective pilot deliverables, D1-1(a-b), D2-1(a-c), D3-1.

The sensors listed below are all on-line. Additionally, a pilot may perform off-line (ground truth) measurements for the purpose of calibration or to provide training data for the cognitive model.

7.4.1 Pilot 1) Hydro – Gas Treatment Center (GTC)

Description of sensors and availability
Sensor data/measurements to be used in pilot
The input to the GTC is • Raw gas • Virgin (primary) alumina • Secondary alumina • Heat all of which require feed rate measurements (gas flow, feeder RPM, heat exchange flow and temperature). The parameters that affect GTC energy efficiency and emission rates, are • Pressure and temperature in chamber • HF content of raw gas • Ambient weather conditions The KPIs to be optimized are • Main fan power input • HF emissions from scrubber (measured at output or at stack) • HF emissions from pot (measured at ceiling of electrolysis hall) Optimal operation will be achieved through cognitive control of the first list based on input data from all three lists.
List of existing sensors and locations
Currently, the following sensors are installed and logging data to the control platform. • Gas pressure sensors; various locations • Feeder rotation counter; primary and secondary alumina infeeds • HF laser monitors; ceiling, raw gas, after scrubber • Weather station; roof • Flow meter; heat exchange pipes
Gap(s) in process data to be covered by new sensors

Currently, gas temperature and flow rate are not measured. Contact measurements of raw gas is challenging due to the corrosive and dusty environment. The large diameter of the main ducts makes ultrasonic (Doppler) flowmeters impractical. Gas temperature is believed to vary across the duct cross-section; hence, thermometers on the duct walls may yield non-representative data. Another unresolved challenge is measurement of HF emissions from individual scrubber chambers. This measurement is needed to achieve independent and optimal control of units. Alternatively, we may measure SO2 emissions from each chamber. SO2 is assumed to be a leading indicator of HF emissions, and may be a better input to the cognitive twin than direct HF testing.
Candidate sensors to fill data gaps
Most of the above data gaps may be filled using existing soft-sensors, or are not critical to cognitive function. However, some new technologies have been considered for testing in COGNITWIN: • Gas flow might be measured using differential pressure sensors installed in bottleneck areas, e.g. valves above pot cells. • Gas temperature could be estimated by the cognitive twin from duct wall temperature combined with ambient (i.e. meteorologica) conditions. Multiple thermometers or a thermal imager may be necessary. Duct elbows may be well-suited for such measurements because they tend to mix air currents. • Chamber emissions could be measured by a “sniffer” assembly that samples air from each test point through a manifold of tubes. An SO2 sniffer may prove easier to implement than a HF sniffer, as SO2 is more inert and less likely to interact with tube walls or resident dust.
Technology Partner(s) for selection, development and test of new sensors
Hydro and GE have experience with gas measurements, and may select and install some sensors themselves. SINTEF will contribute as needed, especially where significant development and/or customization is required.

7.4.2 Pilot 2) Sidenor – Steel ladle

Description of sensors and availability
Sensor data/measurements to be used in pilot
Thermal imaging, laser depth measurements
List of existing sensors and locations
No fixed, on-line sensors; meaurements are performed during repair or demolition.
Gap(s) in process data to be covered by new sensors
Robust and reliable on-line thermal images
Candidate sensors to fill data gaps
Thermal camera
Technology Partner(s) for selection, development and test of new sensors
SINTEF

7.4.3 Pilot 3) Elkem – Silicon process

Description of sensors and availability
Sensor data/measurements to be used in pilot
The environment inside the smelting furnace is too harsh for most sensors to survive. Process information must be derived from measurements up- or downstream. The input to the furnace is raw minerals (charge, various silos) and electrical current. The output is hot metal and slag. Relevant measurements are infeed rates, output mass flow, and temperature and chemical composition of tapped material. Tapping: Ladle weight, ladle fill level, temperature of material Refining: Metal temperature, additions from silo Casting/crushing: Metal temperature, metal weight, coolant flow
List of existing sensors and locations
Load sensors (various locations, including silos and trucks), water flowmeter, pyrometer (crushing only)
Gap(s) in process data to be covered by new sensors
High material temperatures (tapping, in ladle)
Candidate sensors to fill data gaps
Advanced thermal imagers with flame/smoke filters. Radar tranceivers for level measurements.
Technology Partner(s) for selection, development and test of new sensors
SINTEF

7.4.4 Pilot 4) Sumitomo SHI-FW – Boiler operations

Description of sensors and availability
Sensor data/measurements to be used in pilot
Quality of fuel, paired with downstream thermodynamic measurements (T,P,V), emissions and scrubber performance, and direct observation of heat exchangers.
List of existing sensors and locations
Furnace: Pressure and temperature Heat exchangers: Temperature Raw flue gas: Mass flow Smokestack: Gas emissions
Gap(s) in process data to be covered by new sensors
• Direct monitoring of fouling of heat exchange pipes • Chemical quality of fuel: hazardous elements, moisture content, ash content
Candidate sensors to fill data gaps
• Heat exchange monitoring: Machine vision camera or thermal camera • Fuel analysis: XRF or LIBS analyzer

Technology Partner(s) for selection, development and test of new sensors
SINTEF

7.4.5 Pilot 5) Saarstahl AG – Rolled bars in rolling mill

Description of sensors and availability
Sensor data/measurements to be used in pilot
Surveillance cameras, drone-mounted cameras
List of existing sensors and locations
Surveillance cameras in ceilings of assembly hall
Gap(s) in process data to be covered by new sensors
New RGB cameras
Candidate sensors to fill data gaps
RGB cameras – selection and installation in progress
Technology Partner(s) for selection, development and test of new sensors
Scortex and DFKI

7.4.6 Pilot 6) Noksel – Steel pipe manufacturing

Description of sensors and availability
Sensor data/measurements to be used in pilot
Multi-modal sensors will be used as data sources such as temperature, vibration, pressure, electric consumption, oil viscosity. These sensors are going to be used to monitor components such as motors, pumps, electro-mechanical components, conveyors, logistic support components and electrical fuses in the following sub-systems of the SWP machine: Milling, Band Attachment, HGI, V Support, Internal Welding, External Welding, UT Tower, Cut-to Length and External Welding Milling.
List of existing sensors and locations
The existing set of sensors are composed of limit sensors. These sensors are: • Direction limit sensors of carrier cars (forward and backward limits) • Right/Left roll handler limit sensors (in out limits) • Pre delivery up location limit sensors • DP1-DP5 limit sensor list There are five distribution panels at NOKSEL's existing IIoT platform. Data from the existing sensors, i.e. proximity sensors, laser photosensors and pressure switches, are displayed on different distribution panels (DP)s.
Gap(s) in process data to be covered by new sensors
The new sensors will collect data related to the **welding** and **other** sub **processes** of the **whole** SWP production. For the **welding process**, **electrical parameters** of **wire feeding motors will** be **used as process** data. For the **other** sub **processes** of the **whole** SWP production, following **process** data (**parameters**) **exist**:

• **Temperature and** vibration level **of** main **drive DC** Motors, • **temperature and** vibration level **of** edge milling **machine DC** motors, • **temperature and** actual level **of** edge milling **machine DC** motors, • **temperature and** actual level **of** hydraulic oil in the system, • **air pressure of** flux system, • air pressure of plasma cutting machines, • **air pressure of general system**
Candidate sensors to fill data gaps
The Noksel pilot will benefit from several new sensors, including temperature/humidity sensors, oil viscometers, oil detectors (leak detection), vibration sensors, pressure sensors, and electrical sensors (current/voltage). All sensors will be off-the-shelf, with minimal customization for the application. Where possible, multi-modal sensors will be chosen, to reduce local complexity as well as overall complexity of the IoT network.
Technology Partner(s) for selection, development and test of new sensors
Teknopar will assist Noksel with selection of sensors and hardware deployment. SINTEF will assist as needed.

8. Realtime sensor/data processing

8.1 Introduction

This task focuses on developing methods and tools for real-time data analytics, which combine all the sensory data in more meaningful information for making better decisions more efficiently. In the following we introduce the terms real-time and stream processing in detail.

8.1.1 Real-time

The term real-time is frequently used but does not always refer to the same concept as there are multiple slightly different types of real-time. Most commonly it is distinguished between hard, firm and soft real-time.

Hard real-time means that an operation must be finished within a given (constant) deadline no matter what, i.e. missing that deadline means the operation was not successful. This does not imply that the deadline must be as close to the start of the operation as possible, i.e. that the operation is completed as quickly as possible – in only means the operation must be finished before the given deadline. This kind of real-time is most used in scenarios where not reacting in a certain interval of time would cause great loss or even pose a safety thread, i.e. aircrafts, cars, pacemakers but also industrial process controllers.

Firm real-time means that infrequent deadline misses are tolerable but may degrade the system's quality of service. The usefulness of a result received after the deadline is considered zero.

Soft real-time implies that deadline misses are tolerable, but the usefulness of a result degrades with time if it is not received within the deadline.
There is also the term near real-time which does not refer to finishing within a given deadline (no matter how far away in time this deadline may be) but to the delay introduced by processing incoming data. In the context of stream processing, this is the definition of real-time used.

8.1.2 Stream Processing

Processing a continuous stream of data, e.g. sensor readings, in a (near) real-time manner is considered stream processing. Traditional database systems store all the data and you can execute a query against that stored data and receive back the result. Stream processing systems work the other way around. They store the queries and are fed the continuous stream of incoming data. Whenever a query produces a new result, a notification including this result is sent. Therefore, queries are often referred to as continuous queries in stream processing systems. Other terms used to refer to this concept are event stream processing and complex event processing (CEP).

An alternative way to process large amounts of data is batch processing. In batch processing incoming data is collected over a period of time (usually somewhere between few minutes and a day, sometimes even longer) and then processed together. This approach has been around almost since the beginning of computers and is still widely used in data analytics today. The downside of it is that there is quite a delay between incoming sensor data and generation of the result(s). This may be viable for some scenarios, but there also exist scenarios where systems must react more quickly.
An approach for closer to (near) real-time data processing is micro batch processing. This approach is based on traditional batch processing but with much smaller batch sizes (typically in times of seconds or even milliseconds). This allows for smaller delays and is sometimes also as considered stream processing.
“Real” stream processing however is different in the fact that the processing is not triggered by time (or size of data) but by the arrival of new data. This implies that the internal architecture of “real” stream processing systems fundamentally differs from systems that do micro batch processing.

8.2 State of the art and practice

There are lots of existing software frameworks and systems providing stream processing functionality. Almost all of them work in soft (near) real-time. There is at least one system[21] that supports (micro) batch-based stream processing in hard real-time.
However, stream processing with hard real-time can only work under some conditions, e.g. the number, type, frequency, etc of all sensor must be known and constant, all sensors must be connected to the system via hard real-time capable communication hardware and protocols, all continuous queries must be known beforehand and no additional queries may be added or existing one changed.

For soft (near) real-time stream processing there exist multiple commercial and open-source solutions for both approaches, micro-batch and “real” stream processing. Almost every system uses

[21] https://github.com/RTSYork/Real-Time-Stream-Processing

their own proprietary query language. In some aspects these query languages are somewhat consistent, e.g. most of them support the concept of so-called windows to extract some portion of the data on a stream (e.g. data from the last n minutes) yet they differ greatly in supported operators and query complexity. According to Bui22, there are at least three classes of such query languages: data stream query languages, composition-operator-based query languages, production rule query languages. Another approach to classification of stream processing systems was done by Cugola and Margara[23] where they identified further important classification criteria for stream processing systems such as the consumption policy and the capability to load shedding (behavior when the input arrives faster than it can be processed by the system).

In all systems, continuous queries are expressed in a text-based manner. This is hard to impossible for domain experts with little or even no specialized knowledge of the language at hand. Developers who have that knowledge normally lack the domain knowledge. To successfully do stream processing in that cases, domain-experts and developers need to work closely together which is often costly and error-prone. One approach to reduce those costs and risks is to provide a visual editor for the query language to enable domain-experts with little efforts to build the queries themselves. This kind of visual editors are commonly based on the boxes-and-arrows concept where elements such as data sources/stream, operators and data sinks/output stream are represented as boxes and the data flow between them is modelled via arrows connecting them. However, those editors are usually specific to a single query language.

In summary, the technical solutions to stream processing existing today are already quite powerful and sufficient for the scenarios in this project. However, interoperability between and usability of the different technologies is not yet well established.

8.2.1 Initial Overview of Existing Stream Processing Frameworks

In this section we provide an initial overview of existing platforms and frameworks for stream processing. This overview does not discuss platforms/frameworks in detail and does not claim to be exhaustive – it rather provides a first overview of the relevant technologies to be investigated further in this project.

The most prominent platforms/frameworks in this domain are the open-source one, e.g.

- Apache Flink
- Apache Kafka Streams API
- Apache Samza
- Apache Spark
- Apache Storm
- Siddhi (WSO2 Stream Processor/Analytics)
- Esper

[22] Bui, Hai-Lam. "Survey and comparison of event query languages using practical examples." Ludwig Maximilian University of Munich (2009)

[23] Cugola, Gianpaolo, and Alessandro Margara. "Processing flows of information: From data stream to complex event processing." ACM Computing Surveys (CSUR) 44.3 (2012): 1-62.

Besides the open-source solutions, there are also commercial stream processing platforms available such as

- Microsoft StreamInsight
- Amazon MSK (Managed Streaming for Apache Kafka)
- Oracle CEP

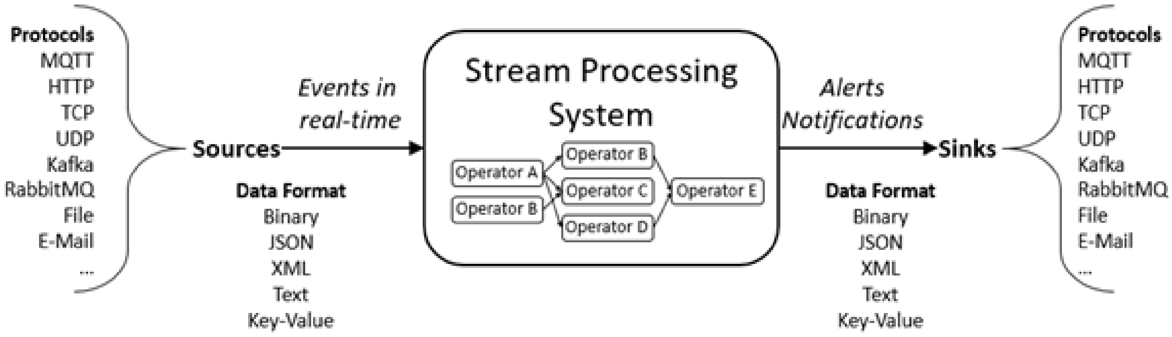
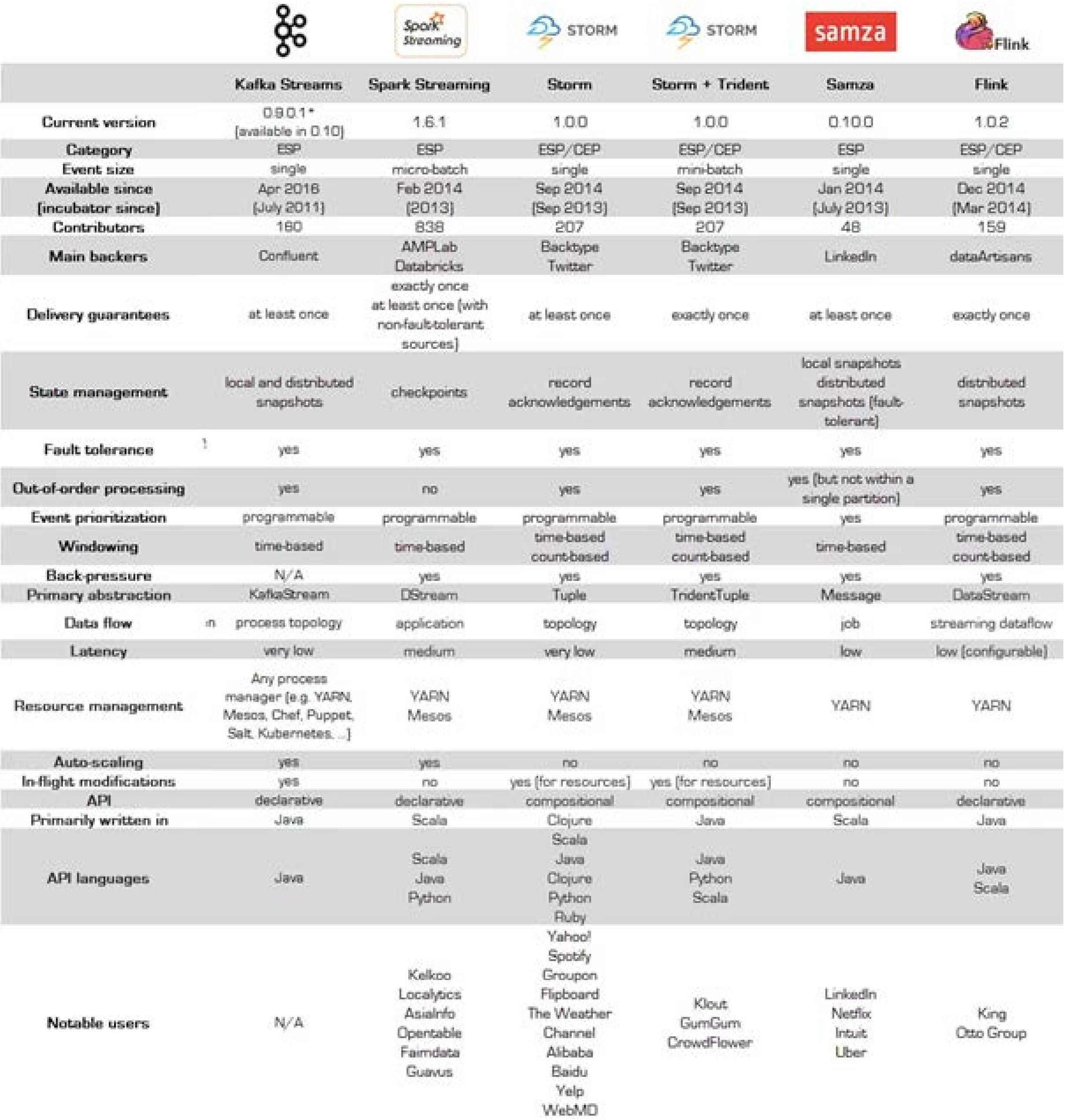


Figure 30: Schematic View of a Stream Processing System.

Figure 30 show a schematic view a stream processing system. From an abstract perspective these systems all work in the same way: they are fed some data through sources, execute some queries and produce some alerts or notifications as output. When looking into the details, the systems can be quite different. Figure 30 shows two major criteria that differentiate stream processing system: their support for different protocols as well as their support for different data formats in the payload going in and out of the system. Figure 31 shows a comparison of some of the systems adding more classification criteria. Beyond these criteria, there are many more, especially related to the expressivity of the used query language such as: Which types of window-operators are support? Are geo-spatial queries supported? Are the language constructs supporting temporal queries (i.e. a sequence of events like A happens before B but within 2 minutes after C)?



	Kafka Streams	Spark Streaming	Storm	Storm + Trident	Samza	Flink
Current version	0.9.0.1* (available in 0.10)	1.6.1	1.0.0	1.0.0	0.10.0	1.0.2
Category	ESP	ESP	ESP/CEP	ESP/CEP	ESP	ESP/CEP
Event size	single	micro-batch	single	mini-batch	single	single
Available since (incubator since)	Apr 2016 (July 2011)	Feb 2014 (2013)	Sep 2014 (Sep 2013)	Sep 2014 (Sep 2013)	Jan 2014 (July 2013)	Dec 2014 (Mar 2014)
Contributors	160	838	207	207	48	159
Main backers	Confluent	AMPLab Databricks	Backtype Twitter	Backtype Twitter	LinkedIn	dataArtisans
Delivery guarantees	at least once	exactly once at least once (with non-fault-tolerant sources)	at least once	exactly once	at least once	exactly once
State management	local and distributed snapshots	checkpoints	record acknowledgements	record acknowledgements	local snapshots distributed snapshots (fault-tolerant)	distributed snapshots
Fault tolerance	yes	yes	yes	yes	yes	yes
Out-of-order processing	yes	no	yes	yes	yes (but not within a single partition)	yes
Event prioritization	programmable	programmable	programmable	programmable	yes	programmable
Windowing	time-based	time-based	time-based count-based	time-based count-based	time-based	time-based count-based
Back-pressure	N/A	yes	yes	yes	yes	yes
Primary abstraction	KafkaStream	DStream	Tuple	TridentTuple	Message	DataStream
Data flow	process topology	application	topology	topology	job	streaming dataflow
Latency	very low	medium	very low	medium	low	low (configurable)
Resource management	Any process manager (e.g. YARN, Mesos, Chef, Puppet, Salt, Kubernetes, ...)	YARN Mesos	YARN Mesos	YARN Mesos	YARN	YARN
Auto-scaling	yes	yes	no	no	no	no
In-flight modifications	yes	no	yes (for resources)	yes (for resources)	no	no
API	declarative	declarative	compositional	compositional	compositional	declarative
Primarily written in	Java	Scala	Clojure	Java	Scala	Java
API languages	Java	Scala Java Python	Scala Java Clojure Python Ruby	Java Python Scala	Java	Java Scala
Notable users	N/A	Kelkoo Localytics AsiaInfo Opentable Faimdata Guavus	Yahoo! Spotify Groupon Flipboard The Weather Channel Alibaba Baidu Yelp WebMD	Klout GumGum CrowdFlower	LinkedIn Netflix Intuit Uber	King Otto Group

Figure 31: Comparison of some open-source technologies for stream processing[24]

8.3 Partner technologies

Fraunhofer IOSB provides the VISPAR framework that offers an environment for complex event processing together with an Android application for query creation designed to be used by non-technical experts. In the app, queries are created via a visual editor in the boxes-and-arrow style. Those queries are later translated into SiddhiQL, the query language of one of the existing CEP frameworks. Data streams are connected via MQTT or HTTP and the open-source FROST Server®[25]

[24] http://www.complexevents.com/2016/06/15/proliferation-of-open-source-technology-for-event-processing/

[25] https://github.com/FraunhoferIOSB/FROST-Server

(an implementation of the OGC SensorThings API). Additionally, sensors can be connected via OPC UA and the IDS (International Data Spaces)[26].

Scortex provides a real time FPGA compute platform that permit to run machine learning algorithm in real time with low latency. Since the machine learning algorithm that worth to improve compute with FPGA, have large inputs like images. It was made to consumes gigE vision input (like industrial cameras). Sensors that produces large amount of data like cameras will be connected directly to the FPGA compute platform and in parallel to a database to store the data. The results will be provided as a stream output. The Scortex solution is not intended to be used in other cases than images or 2D array data.

TEKNOPAR provides IIoTP component of STEEL4.0 as a generic data acquisition platform with specific functionalities built in for the purpose of integrating data from SWP machinery. It enables both soft real-time stream processing and batch processing. Soft real-time stream processing is conducted at two levels: edge and platform tiers. At the edge, TIA PORTAL is used for PLC programming, and using ProfiNET industrial ethernet protocol, and MQTT protocol, data in JSON format is generated. At the platform tier, Kafka Stream API will be used in the operations domain. For batch and real time processing, stream data will be provided to Cassandra over TCP/IP in JSON format. Kafka, Flink, Oracle and Cassandra are used for both real and batch processing at the information domain.

8.4 Pilot requirements

As a first step for designing a generic toolbox suitable for the pilots it is essential to collection detailed information on the requirements of the pilot regarding stream processing. This is especially important as there are six pilots from different domains with potentially quite different requirements. Each of the pilots also have various types of existing infrastructure and IoT platforms that there is a need to have interoperability with.

As for now, the requirements analysis regarding stream processing is still work in progress. First insights show that support for both, hard and soft real-time stream processing may be required. An example for the requirement of hard real-time stream processing might be tracking of moving rolled steel bars in the rolling mill in Pilot 4. For engineering boiler operations in Pilot 6 soft real-time seems to be suitable.

Beside the support for hard/soft real-time, another important aspect of a stream processing system is its supported protocols for data ingestion. so far, we identified the following protocols and data sources relevant for the pilots.

TEKNOPAR: Processed by the controller PLC, multi modal sensor stream data will be displayed on NOKSEL's display panels, and delivered to OPC UA device as well as send back to the control domain, composed of the actuators, sensor and the controller PLC. These data will be provided as input to monitoring and diagnostic functionalities, and stored in database for future analysis. The data will also

[26] https://www.internationaldataspaces.org/

be used in conducting big data analytics and machine learning algorithms related to the KPIs stated in the pilot document.

Protocols	Data Sources
• OPC UA • OPC DA • HTTP • AMQP • MQTT • RTSP (video streams) • TCP/IP	• Kafka • Oracle • RabbitMQ • REST-based services • NOBIS (NOKSEL's system integrated with SAP)

8.5 Initial recommendations/plans

The goal of this support area is to develop methods and tools for real-time data analytics, which combine all the sensory data in more meaningful information for making better decisions more efficiently. The goal is to leverage domain experts who do not need to be data scientists, to quickly create and update real-time analytics out of heterogeneous data streams without the need for deep technical knowledge of underlying data analytics technology. The approach will be based on the stream processing, i.e. complex event processing to process real-time data by applying declarative event-condition-action rules/patterns on a continuous stream of incoming events in order to detect situations with minimal delay. To develop patterns, we will apply a hybrid approach by combining expert knowledge, methods for statistic-based continual pattern improvement and machine-learning methods for pattern evolution based on real-time data or additional data sources (e.g. by extracting from relevant text documents). To overcome the problem of heterogeneous query language we envision moving towards a generic or standardized query language that is supported by existing CEP solutions. As a first step, we propose to define a limited subset of such a language and implement a mechanism to translate queries defined in that language into proprietary languages of existing systems. Something similar has been started in the area of Semantic CEP[27] where they tried to harmonise existing query languages for Semantic CEP. Different systems (e.g. WSO2, Esper, Flink, Storm, etc.) will be evaluated by taking into account criteria like expressiveness of the patterns, scalability, resource-consumption, etc.

Another objective is the integration of ML algorithms into CEP to allow for the easy creation of soft(ware-based) sensors. This functionality will allow to use the algorithms create in WP5 to be integrated in (near) real-time stream processing and enable the creation of cognitive digital twins.

[27] Dell’Aglio, Daniele, et al. "Towards a unified language for RDF stream query processing." European Semantic Web Conference. Springer, Cham, 2015.

9. Conclusions

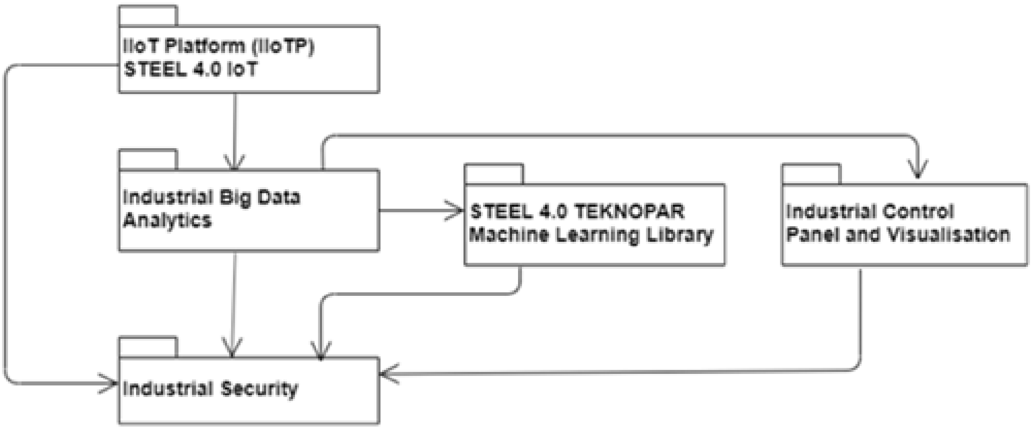
The further focus of the COGNITWIN Toolbox following this report will be on managing the data of the pilots and providing support for the analytics components, based on a further detailed analysis of the requirements emerging from the pilot descriptions in COGNITWIN deliverables D1.1, D12.1 and D3.1.

This D4.1 report on COGNITIVE Baseline Platform, Sensor and Data Interoperability Toolbox has described the initial baseline for the COGNITWIN Toolbox in the areas of toolbox architecture, cyber security, data management and data spaces, further with sensors and realtime sensor/data processing.

These baseline technologies and methods will now be the starting points for the COGNITWIN Platform, Sensor and Data Interoperability Toolbox and will further be applied in the developments for the COGNITWIN industrial pilots. It will in particular support the further development of the COGNITIVE Hybrid AI and Cognitive Twin Toolbox as described further in the accompanying deliverable D5.1.

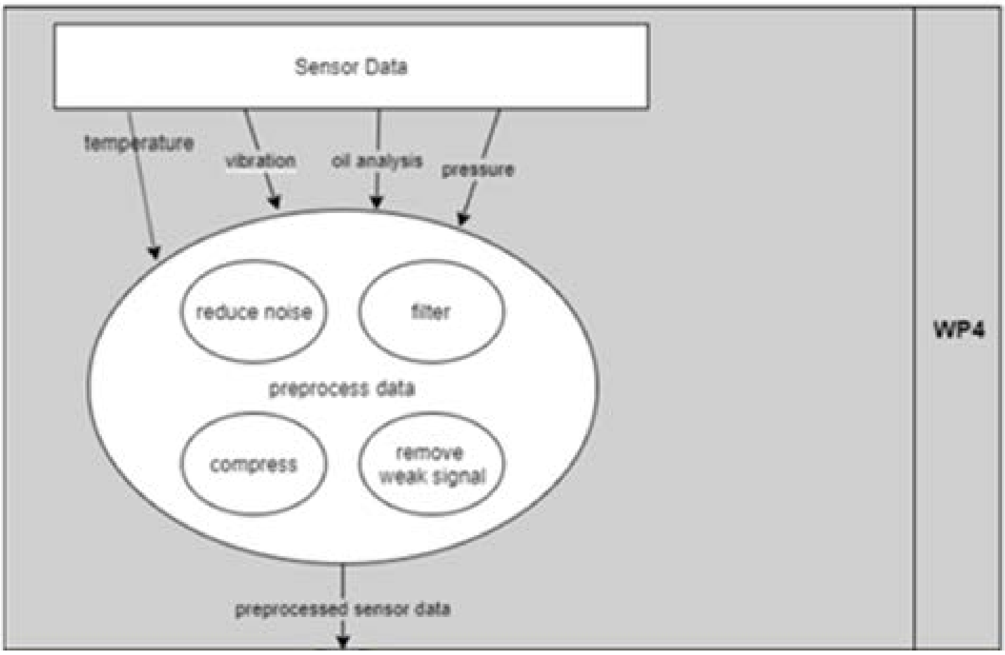
Separate reports on the industrial pilots (D1.1, D2.1, D3.1), the "Baseline Hybrid AI and Cognitive Twin Toolbox" (D5.1) and the Key Performance Indicators (D6.1) and Data Management Plan (D8.1) are issued together with this report. They will now all evolve together for the next milestone of the COGNITWIN project with initial digital twin implementations for all of the six COGNITWIN pilots.

10. Annex 1. TEKNOPAR STEEL4.0 IIoT and IDSecurity (IDS)

Component/Tool description
Component/Tool/Method/Framework/Service Name
COMPONENT: IIoT Platform [IIoTP] (STEEL 4.0 IoT)
Short Description – incl. Purpose
Being one of the TEKNOPAR's components in STEEL 4.0, IIoTP is an IIoT Platform for multi-modal sensorial data acquisition and monitoring in industrial facilities. It is a generic data acquisition platform with specific functionalities built in for the purpose of integrating data from SWP machinery.  IIoT Platform (IIoTP) STEEL 4.0 IoT Industrial Big Data Analytics STEEL 4.0 TEKNOPAR Machine Learning Library Industrial Control Panel and Visualisation Industrial Security UML Package Diagram for TEKNO Components IIoTP provides a condition monitoring system and a network of sensors control process variations to improve productivity and quality. The platform is platform independent and utilizes TCP/IP for wired network connections, including ANT+, Bluetooth, EDGE, GPRS, IrDA, LTE, NFC, RFID, Weightless, WLAN, ZigBee, and Z-Wave. Confirming to FIWARE and OPC-UA standards, the platform is compatible with SWP machine production process in steel industry. Enabling automation, IIoTP shall result in minimum 10% energy consumption reduction and 15% increased efficiency. Multi-sensor data (i.e. temperature, vibration, oil analysis and pressure sensors) will potentially be collected in realtime and massive amount of data will be stored and analyzed. Having robust security and being modular, the platform enables internet connections to the applications and devices, and stored data reviews. IIoTP works on the cloud, with low latency on the scale of seconds, and perform real-time data processing. For changes in size and speed, the platform enables data scalability. In order to retrieve data including all downtimes, production periods, daily working hours, which pipe had been produced from which labelled coils, applied procedures, etc. NOBİS and NOKSEL SAP will be integrated. ((NOBİS is a special software utility used to have full traceability in production process; starting from the labelling of raw material, continuing SWP Machine production, coating, testing till stocking steel pipes in the stockyard area since 2008. For financial purposes, NOBİS is integrated with SAP software.) IIoTP is easily adapted to changing needs.
Function – suitable for which process steps (ICT/Data process)

Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation

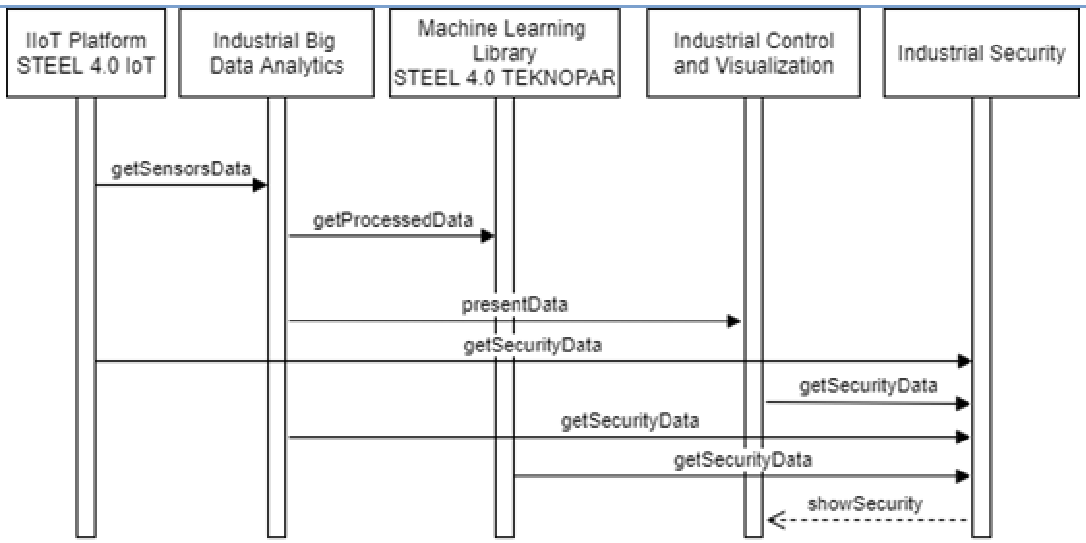
Data collection via HTTP, MQTT, OPC UA, curation, data integration from different sources, data sharing, data access, real time data pre-processing and storage, visualisation



Data Flow Diagram for WP4 TEKNO

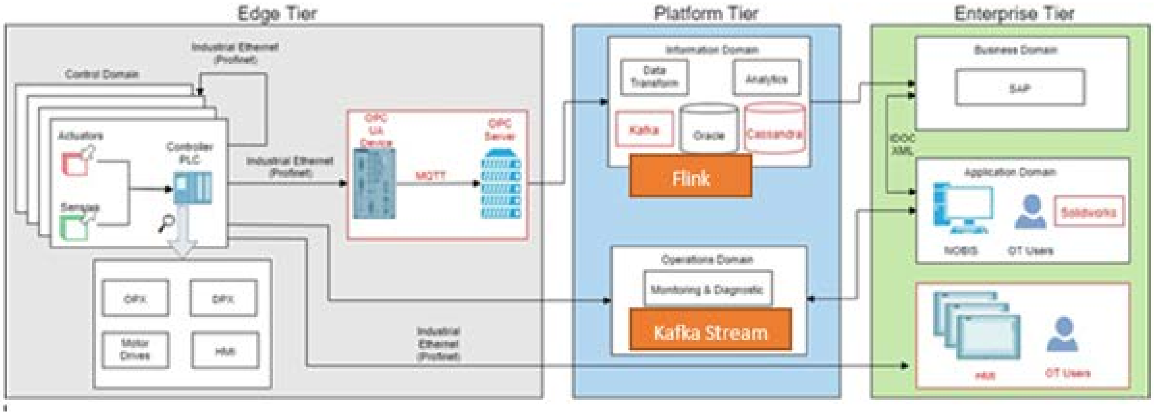
Examples of usage / illustrations

For data acquisition, sensors are connected to PLC automation systems and interfaces are defined for OPC UA and MQTT. Sensor data is stored in Cassandra databases and a PostgreSQL is used for relational data in the monitoring application. As a backend application, FIWARE system is used which supports HTTP and MQTT. A web application is developed for real time monitoring. IIoTP provides sensor data to Industrial Big Data Analytics component of STEEL 4.0. By means of IIoTP, sensor data are turned into preprocessed sensor data, via applications of; noise reduction, filtering, compressing and weak signal removal.

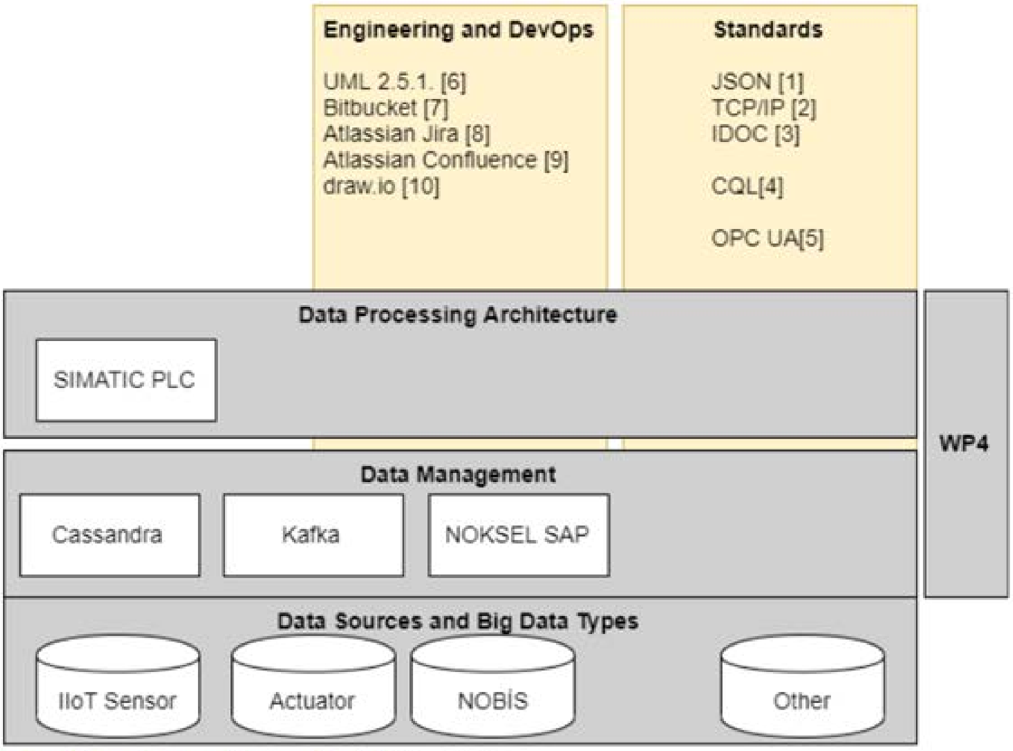

IIoT Platform STEEL 4.0 IoT
Industrial Big Data Analytics
Machine Learning Library STEEL 4.0 TEKNOPAR
Industrial Control and Visualization
Industrial Security
getSensorsData
getProcessedData
presentData
getSecurityData
getSecurityData
getSecurityData
getSecurityData
showSecurity

System Sequence Diagram

Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)

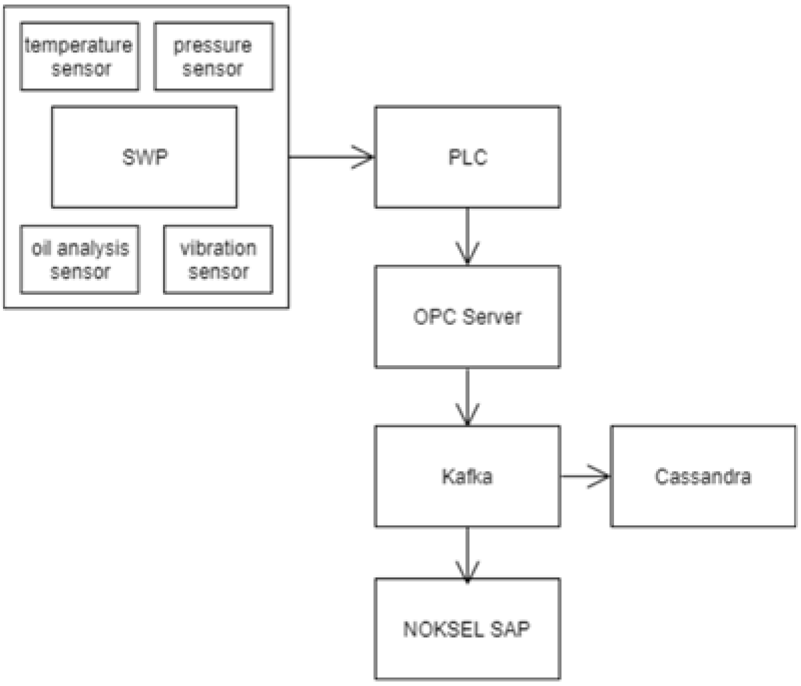


STEEL 4.0 Architecture



1. https://www.json.org/json-en.html, 2 Dec 2019
2. http://www.sis.pitt.edu/mbsclass/standards/viar/TCP-IP.html, 2 Dec 2019
3. https://blogs.sap.com/2012/12/31/idoc-basics-for-functional-consultants/, 2 Dec 2019
4. http://cassandra.apache.org/doc/latest/cql/index.html, 2 Dec 2019
5. https://opcfoundation.org/about/opc-technologies/opc-ua/, 2 Dec 2019
6. https://www.omg.org/spec/UML/, 2 Dec 2019
7. https://bitbucket.org/, 2 Dec, 2019
8. https://www.atlassian.com/software/jira, 2 Dec 2019
9. https://www.atlassian.com/software/confluence, 2 Dec 2019
10. https://www.draw.io/, 2 Dec 2019

BDVA RM and WP4 TEKNO

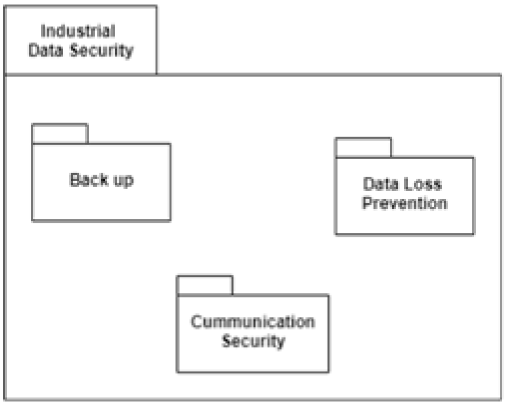


WP4 architecture for TEKNO

Interfaces (in/out) – system/user
IN: sensor data transferred in HTTP, MQTT, OPC UA **OUT:** preprocessed data transferred by OPC UA in JSON format
Subordinates/parts – any platform dependencies
IIoTP utilizes: Cassandra, Kafka,Flink and SIMATIC PLC, NOBİS NOKSEL’s system integrated with SAP
Data (in/out)
IN: Raw sensor data OUT: Stream data to Cassandra over TCP/IP in JSON
Standards (any standards being used)
JSON [1], TCP/IP [2], IDOC [3], CQL[4], OPC UA[5]
Licenses, etc. (free for use in the project)
Apache License 2.0 [11, 12]
TRL for overall component/tool and any parts/subordinates
6
References – incl. web etc.
1. https://www.json.org/json-en.html, 2 Dec 2019 2. http://www.sis.pitt.edu/mbsclass/standards/viar/TCP-IP.html, 2 Dec 2019 3. https://blogs.sap.com/2012/12/31/idoc-basics-for-functional-consultants/, 2 Dec 2019 4. http://cassandra.apache.org/doc/latest/cql/index.html, 2 Dec 2019 5. https://opcfoundation.org/about/opc-technologies/opc-ua/, 2 Dec 2019

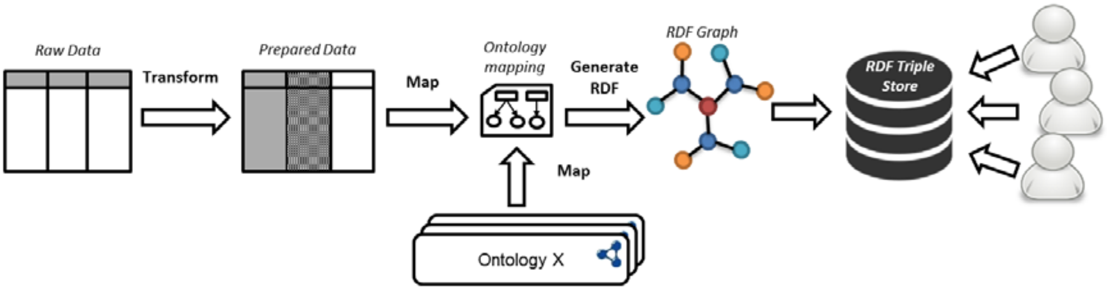
6. https://www.omg.org/spec/UML/, 2 Dec 2019 7. https://bitbucket.org/, 2 Dec, 2019 8. https://www.atlassian.com/software/jira, 2 Dec 2019 9. https://www.atlassian.com/software/confluence, 2 Dec 2019 10. https://www.draw.io/, 2 Dec 2019 11.http://www.apache.org/licenses/, 30 Nov 2019 12. http://www.apache.org/licenses/LICENSE-2.0, 30 Nov 2019
To be considered in particular for the following COGNITWIN pilots
Noksel – (COGNITIVE) DIGITAL TWIN POWERED CONDITION MONITORING (and Control) IN STEEL PIPE MANUFACTURING INDUSTRY

Component/Tool description
Component/Tool/Method/Framework/Service Name
COMPONENT: Industrial Data Security (IDS)
Short Description – incl. Purpose
The purpose of IDS is to provide industrial data security. For security prospects, all parts of the platform is analyzed, and data privacy shall be protected. Data loss will be prevented in cases of electric outages and natural disasters. IDS shall not interfere with data communication, and not affect delay and performance. Data security is achieved by the subsystems of IDS: communication security, data loss precaution, backup**.**
Function – suitable for which process steps (ICT/Data process) **Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation**
IDS indirectly supports data collection, curation, integration, processing, analytics, decision support, control and visualisation functionalities, but its primary function is none of these functionalities. IDS functions are related to security, and hence access and sharing functionalities are part of IDS.
Examples of usage / illustrations
Associated with all of the packages of STEEL 4.0. IDS retrieves security data from IIoTP, big data analytics, machine learning library and industrial control and visualisation components. IDS enables industrial control and visualisation component to display security related data
Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)
Refer to System sequence diagram and UML package diagram for TEKNO components
Interfaces (in/out) – system/user
IN: Security data retrieval related interfaces from other components of STEEL 4.0 **OUT:** Interface with Industrial control and visualization component

Subordinates/parts – any platform dependencies
Associated with all other components of STEEL 4.0, IDS involves: • Communication Security subsystem. • Data Loss Prevention subsystem, and • Backup subsystem  Industrial Data Security Back up Data Loss Prevention Cummunication Security
Data (in/out)
IN: Security data from all other components of STEEL 4.0 **OUT:** Security data to be displayed
Standards (any standards being used)
NA
Licenses, etc. (free for use in the project)
NA
TRL for overall component/tool and any parts/subordinates
TRL 3
References – incl. web etc.
https://teknopar.com.tr/
To be considered in particular for the following COGNITWIN pilots
Noksel – (COGNITIVE) DIGITAL TWIN POWERED CONDITION MONITORING (and Control) IN STEEL PIPE MANUFACTURING INDUSTRY

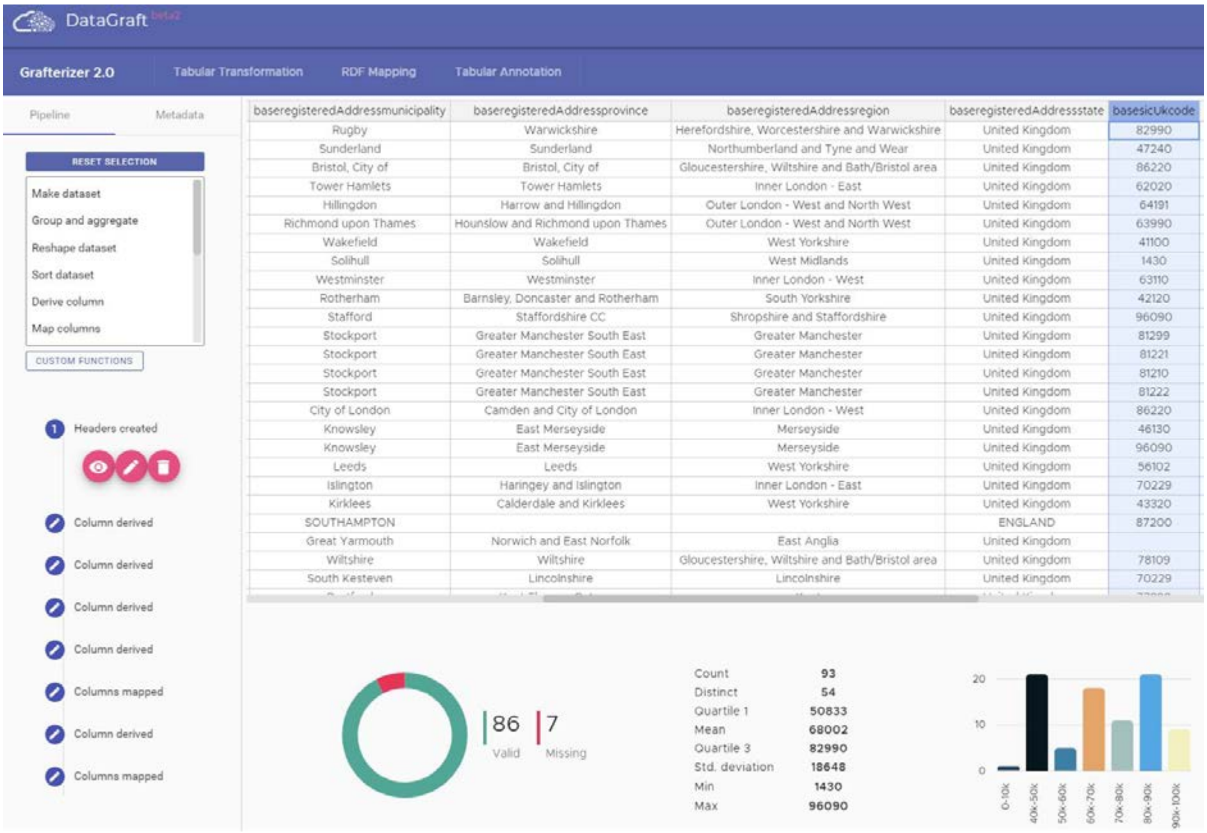
11. Annex 2. SINTEF DataGraft, Knowledge Graphs and Big Data Pipeline Framework

Component/Tool description
Component/Tool/Method/Framework/Service Name
DataGraft
Short Description – incl. Purpose
DataGraft is a general-purpose data management platform focused on supporting the creation and provisioning of Knowledge Graphs. It consists of a set of cloud-based tools and services for data transformation and access. DataGraft aims to offer a complete package for transformation of raw data into meaningful data assets and reliable delivery of data assets by providing a solution that outsources various data operations to the cloud and that eliminates upfront costs on data infrastructure and ongoing investment of time and resources in managing the data infrastructure. DataGraft was developed to allow data workers to manage their data in a simple, effective, and efficient way.
Function – suitable for which process steps (ICT/Data process) ***Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation***
DataGraft implements a set of capabilities for managing data. Key capabilities include: • Interactive design of data transformations, including tabular data cleaning and data enrichment • Repeatable data transformations • Reuse/share data transformations (user-based access) • Cloud-based deployment of data transformations The figure below depicts a typical process supported by DataGraft: from raw data to knowledge graph data.



Examples of usage / illustrations

DataGraft was used in various projects/domains for publishing knowledge graphs. Examples includes the proDataMarket project[28] for publishing geospatial and real estate data, euBusinessGraph[29] for harmonizing and publishing company information, EW-Shopp[30] for enriching digital marketing data. In a typical scenario in this context, data from various sources was collected and harmonized according to specific ontologies. DataGraft provided tool support in the process of data collection, profiling, cleaning, enrichment, storage, and publication. The following figure is a screenshot of DataGraft user interface that shows the functionality for cleaning and transforming tabular data.

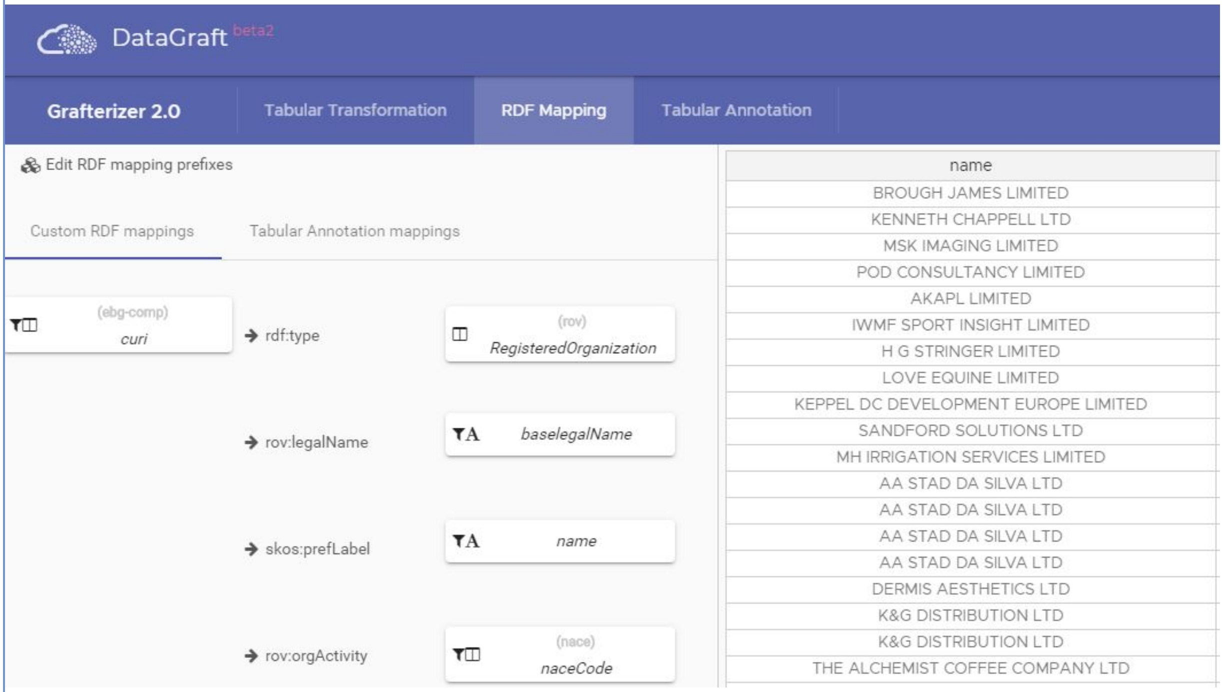


baseregisteredAddressmunicipality	baseregisteredAddressprovince	baseregisteredAddressregion	baseregisteredAddressstate	basesicUkcode
Rugby	Warwickshire	Herefordshire, Worcestershire and Warwickshire	United Kingdom	82990
Sunderland	Sunderland	Northumberland and Tyne and Wear	United Kingdom	47240
Bristol, City of	Bristol, City of	Gloucestershire, Wiltshire and Bath/Bristol area	United Kingdom	86220
Tower Hamlets	Tower Hamlets	Inner London - East	United Kingdom	62020
Hillingdon	Harrow and Hillingdon	Outer London - West and North West	United Kingdom	64191
Richmond upon Thames	Hounslow and Richmond upon Thames	Outer London - West and North West	United Kingdom	63990
Wakefield	Wakefield	West Yorkshire	United Kingdom	41100
Solihull	Solihull	West Midlands	United Kingdom	1430
Westminster	Westminster	Inner London - West	United Kingdom	63110
Rotherham	Barnsley, Doncaster and Rotherham	South Yorkshire	United Kingdom	42120
Stafford	Staffordshire CC	Shropshire and Staffordshire	United Kingdom	96090
Stockport	Greater Manchester South East	Greater Manchester	United Kingdom	81299
Stockport	Greater Manchester South East	Greater Manchester	United Kingdom	81221
Stockport	Greater Manchester South East	Greater Manchester	United Kingdom	81210
Stockport	Greater Manchester South East	Greater Manchester	United Kingdom	81222
City of London	Camden and City of London	Inner London - West	United Kingdom	86220
Knowsley	East Merseyside	Merseyside	United Kingdom	46130
Knowsley	East Merseyside	Merseyside	United Kingdom	96090
Leeds	Leeds	West Yorkshire	United Kingdom	56102
Islington	Haringey and Islington	Inner London - East	United Kingdom	70229
Kirklees	Calderdale and Kirklees	West Yorkshire	United Kingdom	43320
SOUTHAMPTON			ENGLAND	87200
Great Yarmouth	Norwich and East Norfolk	East Anglia	United Kingdom	
Wiltshire	Wiltshire	Gloucestershire, Wiltshire and Bath/Bristol area	United Kingdom	78109
South Kesteven	Lincolnshire	Lincolnshire	United Kingdom	70229

Count	93
Distinct	54
Quartile 1	50833
Mean	68002
Quartile 3	82990
Std. deviation	18648
Min	1430
Max	96090

[28] http://prodatamarket.eu
[29] http://eubusinessgraph.eu
[30] https://www.ew-shopp.eu

The following screenshot depicts the functionality for mapping tabular data to graphs.



DataGraft beta2

Grafterizer 2.0 | Tabular Transformation | RDF Mapping | Tabular Annotation

Edit RDF mapping prefixes

Custom RDF mappings | Tabular Annotation mappings

(ebg-comp) curi → rdf:type (rov) RegisteredOrganization

→ rov:legalName baselegalName

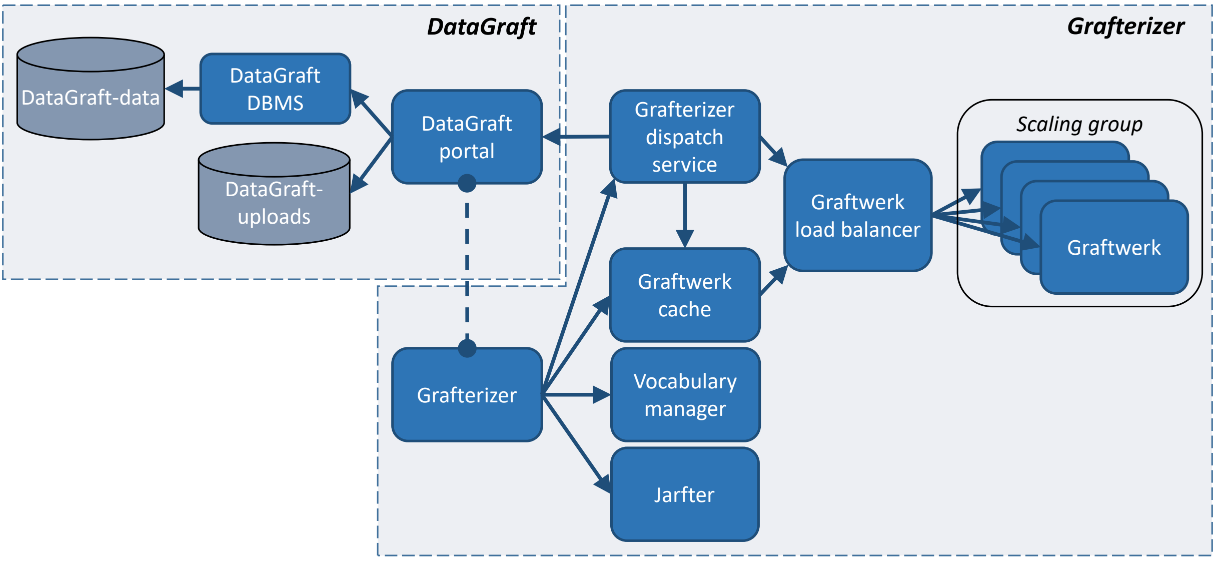
→ skos:prefLabel name

→ rov:orgActivity (nace) naceCode

name
BROUGH JAMES LIMITED
KENNETH CHAPPELL LTD
MSK IMAGING LIMITED
POD CONSULTANCY LIMITED
AKAPL LIMITED
IWMF SPORT INSIGHT LIMITED
H G STRINGER LIMITED
LOVE EQUINE LIMITED
KEPPEL DC DEVELOPMENT EUROPE LIMITED
SANDFORD SOLUTIONS LTD
MH IRRIGATION SERVICES LIMITED
AA STAD DA SILVA LTD
AA STAD DA SILVA LTD
AA STAD DA SILVA LTD
AA STAD DA SILVA LTD
DERMIS AESTHETICS LTD
K&G DISTRIBUTION LTD
K&G DISTRIBUTION LTD
THE ALCHEMIST COFFEE COMPANY LTD

Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)

The DataGraft platform was implemented following a microservice architecture consisting of many subcomponents, each contained in a Docker[31] container. The connected microservices communicate with each other through REST APIs. The individual components are illustrated in the figure below.



DataGraft consists of the following components:

[31] https://www.docker.com/

- DataGraft portal: The portal provides the Web-based frontend that is used by the euBusinessGraph data publishers;
- DataGraft DBMS: This component represents the database management system (PostgreSQL[32]) for the user data and asset catalog. Data is stored in a separate volume (Docker volume or Amazon S3[33] in production).

Grafterizer – the data transformation framework part of DataGraft – includes the following subcomponents:

- Grafterizer frontend: Component that implements the interactive graphical user interface for data cleaning, data transformation, and RDF mapping;
- Grafterizer dispatch service: A server component for the Grafterizer frontend that handles request authentication on its behalf (in order to ensure security) and dispatches requests for input and output across multiple services;
- Graftwerk: A sandboxed server component that executes data cleaning and transformation scripts that are generated by the Grafterizer frontend over a set of input data sent by the dispatch service. Graftwerk uses a proprietary load-balancing component in order to distribute the traffic when a larger number of users use the transformation tool;
- Graftwerk cache: A FIFO cache service for the Grafterizer frontend requests to Graftwerk;
- Vocabulary manager: RDF vocabulary management service for imported vocabularies used by the RDF mapping user interface of Grafterizer. Enables searching through concepts and import of vocabularies;
- Jarfter: A Web service component that compiles executable JARs for transformations generated by the Grafterizer frontend.

Interfaces (in/out) – system/user

DataGraft comes with graphical user interfaces for end users.
DataGraft components expose REST APIs.

Subordinates/parts – any platform dependencies

As described above, DataGraft is based on a microservices architecture with loosely coupled components.

Data (in/out)

Input data is provisioned as tabular data files or accessed via external APIs. Data stored, managed and generated by DataGraft can be downloaded or accessed via APIs.

Standards (any standards being used)

[32] https://www.postgresql.org/

[33] https://aws.amazon.com/s3/

For the construction of Knowledge Graphs, DataGraft relies on W3C standards, in particular Linked Data related standards[34].

Licenses, etc. (free for use in the project)

Eclipse Public License (v1.0)

TRL for overall component/tool and any parts/subordinates

DataGraft and its components are at TRL 4-5.

References – incl. web etc.

DataGraft online service: https://datagraft.io

DataGraft software:

- Installation guide: https://github.com/datagraft/datagraft-portal
- User guide: https://github.com/datagraft/datagraft-reference/blob/master/documentation.md
- API documentation: https://datagraft.github.io/datagraft-API/dist/index.html?url=https://datagraft.github.io/datagraft-API/swagger.yaml
- Source code repository: https://github.com/datagraft/datagraft-portal

Selected publications:

- D. Roman, N. Nikolov, A. Putlier, D. Sukhobok, B. Elvesæter, A. Berre, X. Ye, M. Dimitrov, A. Simov, M. Zarev, R. Moynihan, B. Roberts, I. Berlocher, S. Kim, T. Lee, A. Smith, and T. Heath. DataGraft: One-Stop-Shop for Open Data Management. 2018. Semantic Web Journal. Volume 9, number 5, ISSN 1570-0844. s 393-411. doi: 10.3233/SW-170263.
- S. Sajid, B. M. von Zernichow, A. Soylu and D. Roman. Predictive Data Transformation Suggestions in Grafterizer using Machine Learning. 13th Metadata and Semantics Research Conference, Rome, Italy, October 28th -31st, 2019.
- D. Sukhobok, N. Nikolov, and D. Roman. Tabular data anomaly patterns. In Proceedings of the 3rd International Conference on Big Data Innovations and Applications (Innovate-Data 2017). 2017. DOI: 10.1109/Innovate-Data.2017.10.
- B. M. Zernichow and D. Roman. Usability of visual data profiling in data cleaning and transformation. 2017. Lecture Notes in Computer Science. ISSN 0302-9743. 10574, s 480- 496. doi: 10.1007/978-3-319-69459-7_32.
- B. M. Zernichow and D. Roman. A Visual Data Profiling Tool for Data Preparation. 2017. In Sandjai Bhulai & Dimitris Kardaras (ed.), DATA ANALYTICS 2017: International Conference on Data Analytics, Barcelona, Spain, November 12-16, 2017. ISBN 978-1-61208-603-3.

[34] https://www.w3.org/standards/semanticweb/data

- D. Sukhobok, N. Nikolov, A. Pultier, X. Ye, A.J. Berre, R. Moynihan, B. Roberts, B. Elvesæter, N. Mahasivam, and D. Roman. Tabular data cleaning and linked data generation with grafterizer. 2016. Lecture Notes in Computer Science. ISSN 0302-9743. 9989, s 134- 139. doi: 10.1007/978-3-319-47602-5_27.
- N. Mahasivam, N. Nikolov, D. Sukhobok, and D. Roman. Data preparation as a service based on Apache Spark. 2017. Lecture Notes in Computer Science. ISSN 0302-9743. 10465, s 125- 139 . doi: 10.1007/978-3-319-67262-5_10.

To be considered in particular for the following COGNITWIN pilots

Pilot use cases that require data management, in particular, data cleaning, enrichment, harmonization, integration and interoperability among different systems and services.

Component/Tool description

Component/Tool/Method/Framework/Service Name

Big Data Pipelines Framework – by SINTEF

Short Description – incl. Purpose

A framework to allow high-level design/specification of Big Data processing pipelines and their effective and efficient deployment on the continuum computing infrastructure (heterogenous Cloud/Fog/Edge infrastructure).

The purpose of such a framework is to lower the technological barriers of entry to the incorporation of Big Data pipelines in organizations' business processes, thus making them accessible to a wider set of stakeholders (such as start-ups and SMEs) regardless of the hardware infrastructure. The framework requires new languages, methods, infrastructures, and software for managing Big Data pipelines such that Big Data pipelines can be easily set up in a manner which is trace-able, manageable, analyzable and optimizable and separates the design- from the run-time aspects of their deployment, thus empowering domain experts to take an active part in their definition.

The purpose of such a framework is to allow specification of data processing pipelines by non-IT experts at an abstraction level suitable for pure data processing, in which pipeline specifications are realized using instances of a pre-defined set of scalable and composable software container templates (corresponding to step types in pipelines).

Function – suitable for which process steps (ICT/Data process)
Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation

Effective and efficient processing of large amounts of data on heterogenous computing infrastructure (Cloud/Fog/Edge continuum computing).

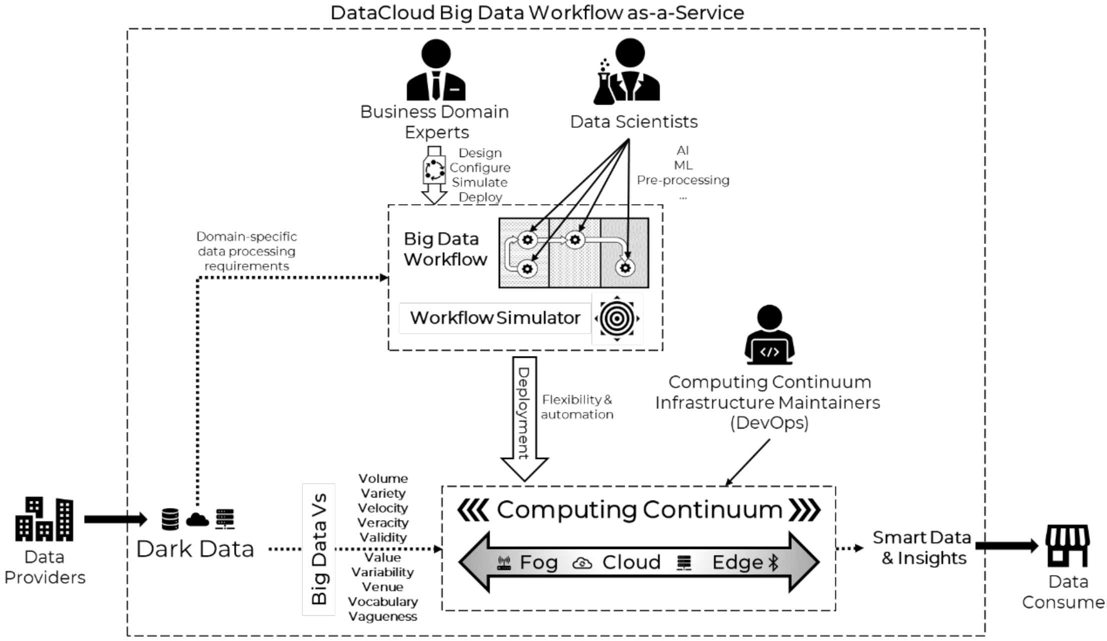
Examples of usage / illustrations

This framework enables various data-intensive applications across multiple systems. The approach is to cope with the management of Big Data and a vast number of heterogeneous systems, from diverse infrastructures and platforms, multi-cloud, computing resources, gateways and devices.

In this context it is required to deploy the Big Data processing pipelines across the entire Cloud/Fog/Edge Continuum that enable carrying out a coordinated distributed process to enable detection and response to various situations.

Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)

A high-level architecture of the framework is depicted below. The framework will enable *business domain experts* to extract domain-specific data processing requirements from the Dark Data, offering them capabilities to structure, design, and simulate Big Data workflows before deploying them. The implementation details, such as specific analytical models and data-specific transformation code will be injected in the data workflow steps by *data scientists*, who possess the technical knowledge to define them using AI an ML approaches. The *Computing Continuum infrastructure maintainers (DevOps)* are engaged in the provisioning of the hardware infrastructure (e.g., Cloud virtual machines, integrated access devices, sensors) and the maintenance of operating systems and resource management software. The data workflows themselves will be exposed as Web services, which can be used by data providers or data consumers to interact with individual steps (e.g., information providers sharing data for a specific analytical task) or the entire Big Data workflows (e.g., data consumers leveraging the result of the analytical process).



Interfaces (in/out) – system/user

The framework will come with UI for design of data workflows and for deployment of workflows.

The data workflows themselves will be exposed as Web services, and various framework components will be exposed using REST APIs.

Subordinates/parts – any platform dependencies

Sub-components will be architected using microservices.

Data (in/out)

Data will be served in batch or real time.

Licenses, etc. (free for use in the project)

Software will be released under a flexible license, e.g., Eclipse Public License (v1.0).

TRL for overall component/tool and any parts/subordinates

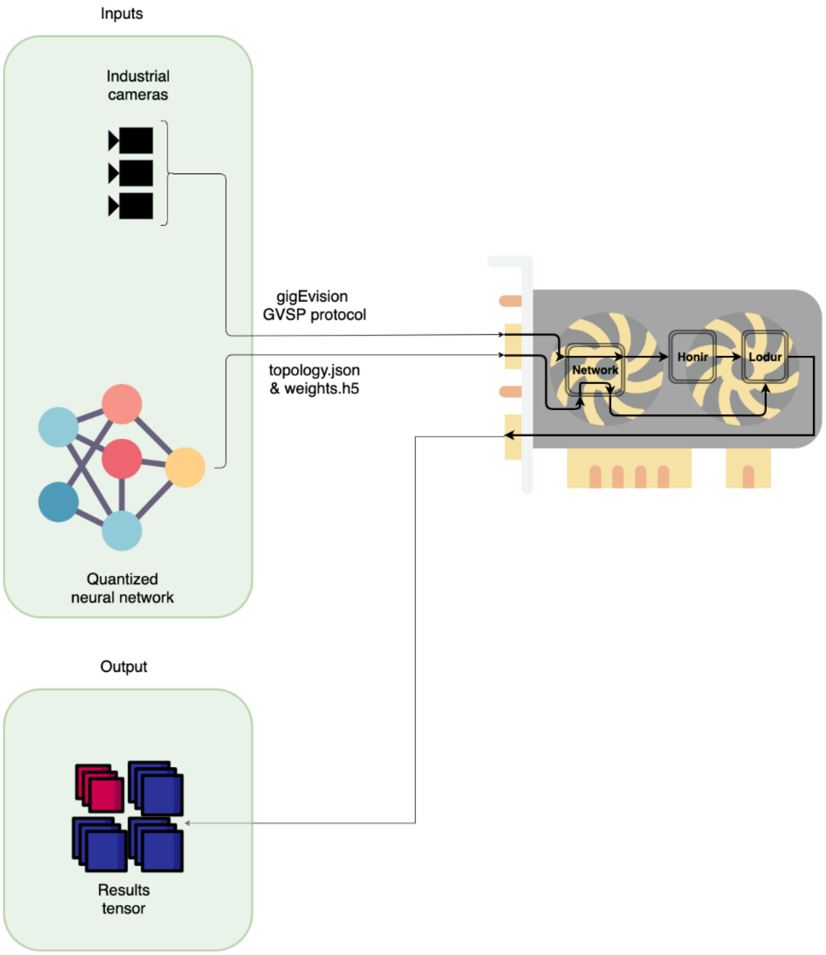
At this stage the framework is under design, thus TRL 2-3.

To be considered in particular for the following COGNITWIN pilots

Pilots that require efficient and effective processing of Big Data on heterogenous infrastructure, to be analysed related to the various pilot infrastructures.

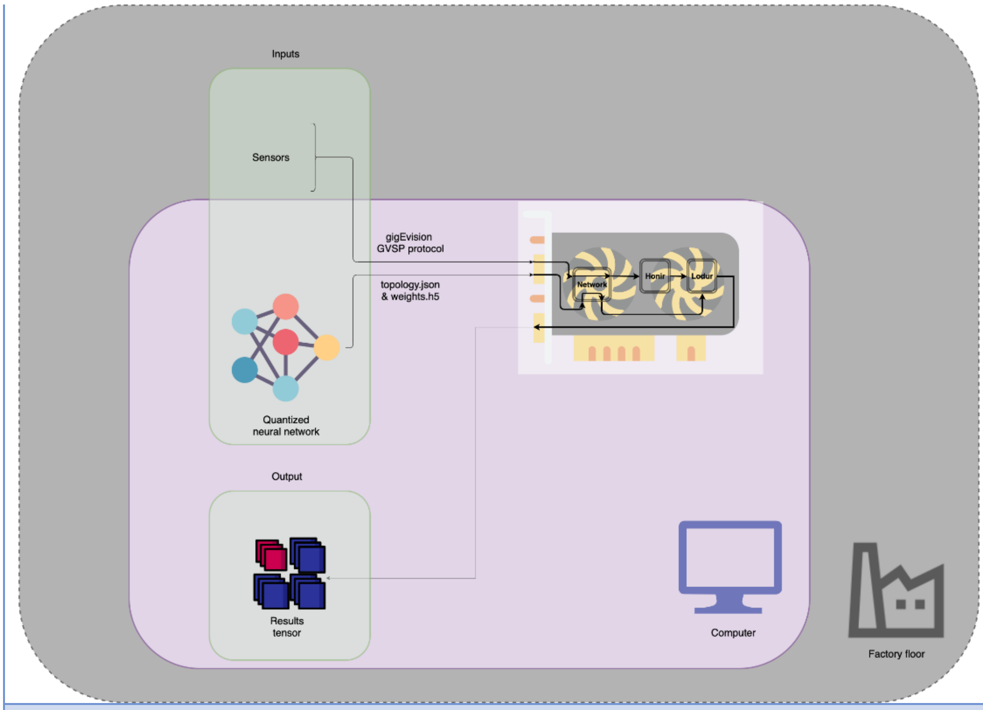
12. Annex 3. SCORTEX FPGA Machine Learning platform

Component/Tool description
Component/Tool/Method/Framework/Service Name
FPGA compute platform for machine learning inference
Short Description – incl. Purpose
This tool is composed of 3 elements: • Machine learning inference engine (codename : Honir) • GigEvision (GVSP) image grabbing (codename : Lodur) • Network management This tool is loaded on a board containing an FPGA (Xilinx VU9P for example) and a QSFP+ network interface. Network management permit configuration of the networking of the FPGA board to send and receive datas properly Lodur is an IP core responsible of grabbing images in gigEvision standard usually supported by industrial cameras. It is also responsible to send the received stream into another stream understandable by the machine learning inference engine. Honir is an IP core responsible of performing machine learning inference. Thanks to a loaded quantized machine learning model, this module consumes images coming from Lodur and produces the tensor results in a stream enhanced with metadatas.
Function – suitable for which process steps (ICT/Data process) ***Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation***
This bundle of tools is here to access data and process them and then share the results
Examples of usage / illustrations



Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)

The Scortex role is to bring an on the edge efficient machine learning compute platform. In that objective, The implantation on site may look like this.

 Inputs Sensors gigEvision GVSP protocol topology.json & weights.h5 Network Honir Lodur Quantized neural network Output Results tensor Computer Factory floor
Interfaces (in/out) – system/user
Inputs Data input stream in gigE vision GVSP standard Topology and weights **Outputs** Inference result : tensor
Subordinates/parts – any platform dependencies
FPGA Hardware
Data (in/out)
In: Vision data, out: Tensor data
Standards (any standards being used)
gigE vision GVSP standard
Licenses, etc. (free for use in the project)
Vivado / Modelsim (not free license)
TRL for overall component/tool and any parts/subordinates
7
References – incl. web etc.
https://scortex.io/
To be considered in particular for the following COGNITWIN pilots
Saarstahl

13. Annex 4. Cybernetica OPC UA Server

Component/Tool description

Component/Tool/Method/Framework/Service Name

Cybernetica OPC UA Server

Short Description – incl. Purpose

The Cybernetica OPC UA Server is a general purpose OPC UA server supporting the Data Access (DA) interface. It can be used as a hub for exchanging real-time data from processes with other clients that support OPC UA.

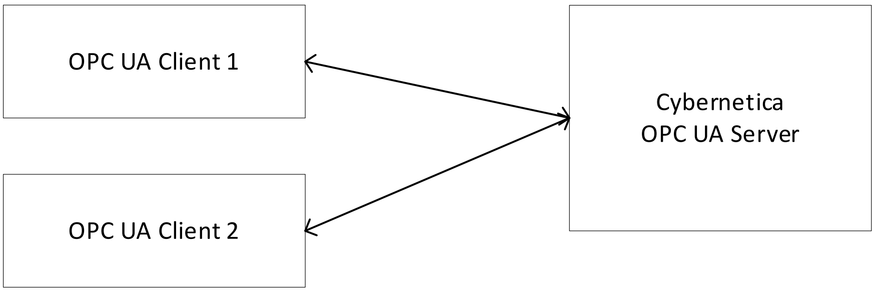
The OPC UA server has a plugin API that allows specialized plugins to be developed. These can be used to collect and distribute data from other data sources (like databases, process control systems or simulators).

Function – suitable for which process steps (ICT/Data process)
Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation

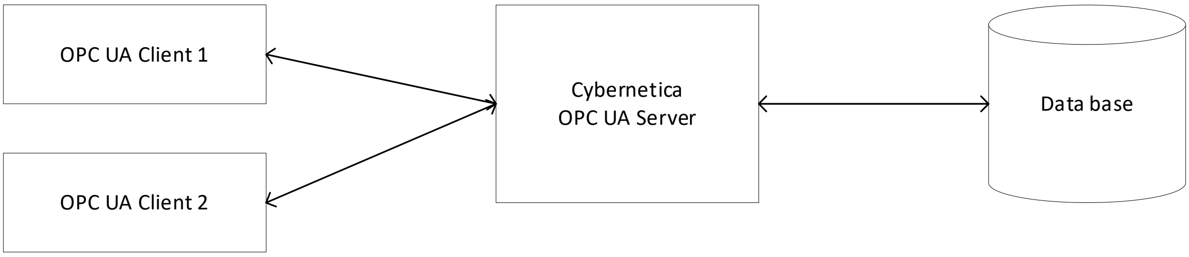
Data collection, integration, sharing.

Examples of usage / illustrations

Example 1: Real-time data exchange



Example 2: Distributing data from a database (or DCS or some other source)



Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)

Interfaces (in/out) – system/user

OPC UA Data Access (DA).

Subordinates/parts – any platform dependencies
Related to other parts of Cybernetica tools, but can be used independent of this.
Data (in/out)
Process data connections – with OPC UA interface out.
Standards (any standards being used)
OPC UA
Licenses, etc. (free for use in the project)
Cybernetica OPC UA Server licenses are provided free of charge for the duration of the COGNITWIN-project for project partners who need such license to execute their work in the project. Should the project result be taken into permanent use after the end of the project, licenses are provided on fair and reasonable terms as stated in the Grant Agreement.
TRL for overall component/tool and any parts/subordinates
8
References – incl. web etc.
http://cybernetica.no/
To be considered in particular for the following COGNITWIN pilots
Hydro, ELKEM and others

14. Annex 5. Nissatech - D2Lab – Big Data Processing and analytics framework

Overview of Technological Components of D2Lab from Nissatech.

D2Lab – Big Data analytics framework

- **Purpose**: lists the software requirements (partly or fully) implemented by this component.
- D2Lab is an advanced data analytics SaaS oriented towards manufacturing data. It uses machine learning techniques to identify anomalies in the production system based on the parameters measured during functional tests.

  D2Lab learns the normal behavior of a system based on historical data. The main innovation is in the combination of the model -based and data-driven approaches for anomaly detection, which enables continuous detection process.

- **Function**: explains the functionality of this component and information about its deployment.
- D2Lab) is a framework (as a Service) for advanced big data processing, which revolutionizes the role of big data for the industry by introducing an active loop between real-time and batch processing. It increases the application potential of big

data approaches from resolving the problems in a process, in the detection of the opportunities for process improvement/optimization. It closes the gap between the industry needs and the available technologies for data-driven, dynamic monitoring of anomalies/unusualities in real-time systems.

- **Example of usage**: Give an illustration/example of usage scenario
- **Overall architecture / pipeline / workflow**: explains how this tool/component works in an application context, interacting with platform and other tools/components in a pipeline.

- D2Lab is a data acquisition, analytics and visualization system. It is meant to provide the means to receive (usually manufacturing) data, adapt it to our unique format and store it, clean it, perform different analysis on it, and offer a portal to visualize both the data and the analysis results.

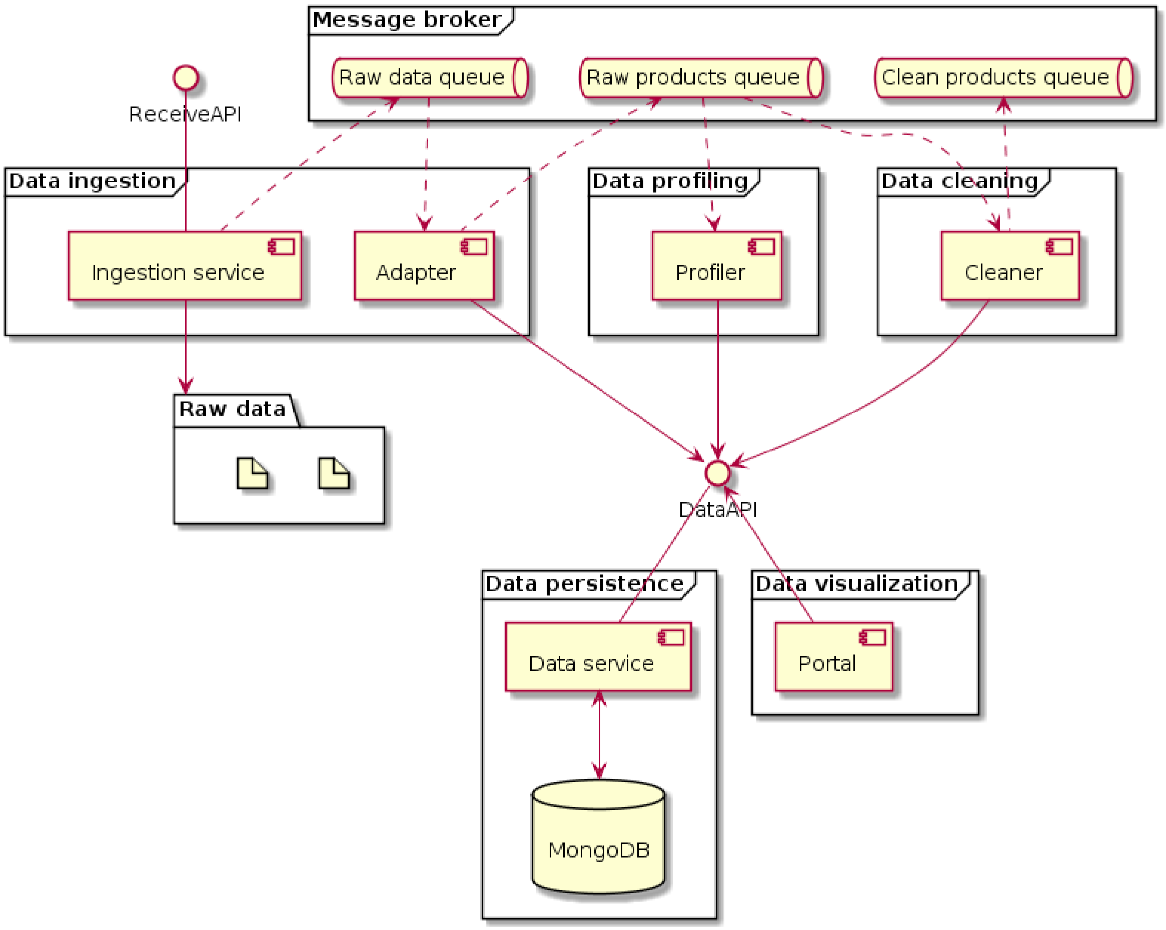


Figure 32 D2Lab v2 architecture (data ingestion and cleaning)

The first part of D2Lab architecture consists of:

- Adapter - adapting data to D2Lab format
- Profiler - statistical profiling
- Cleaner – Using data profile, domain knowledge, and our experience cleaning recipe is made which will be used in data cleaning
- Data Service – REST service which is used to complete CRUD operations on MongoDB
- Portal – is used to visualize data and models and gives the user the possibility to explore data and test hypothesis

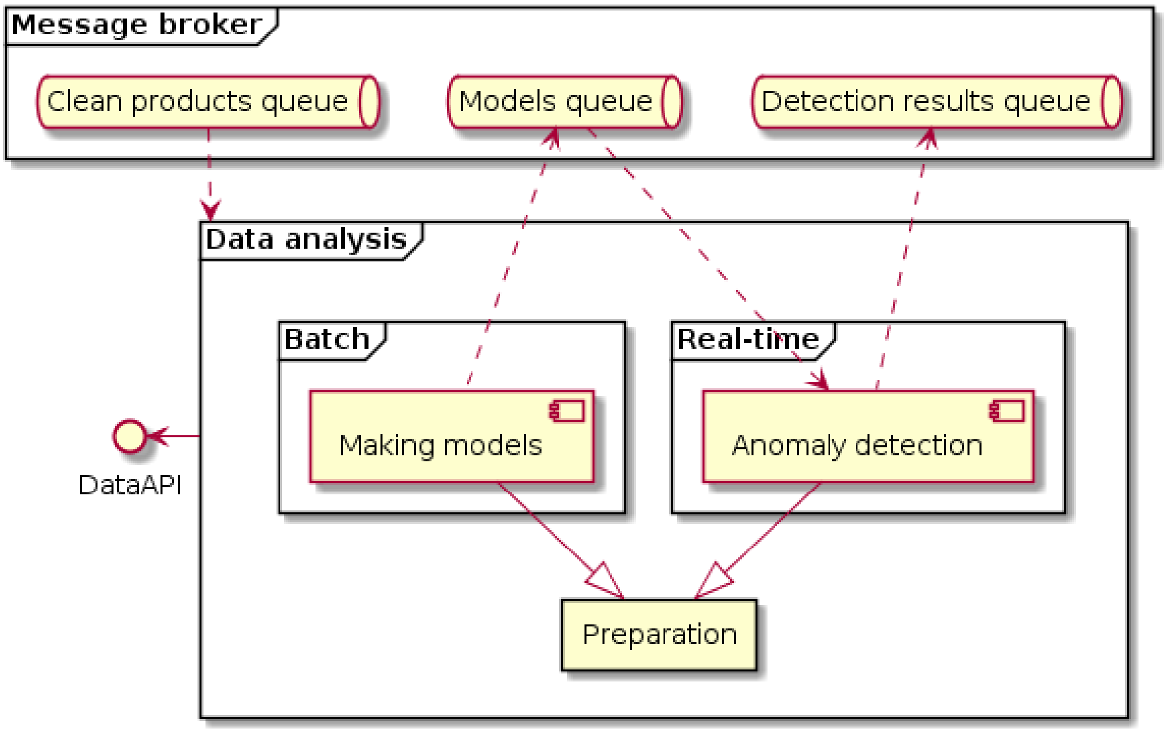


Figure 33 D2Lab v2 architecture (data analysis)

After the data is cleaned, data analysis can start we can distinguish two parts of data analysis:

- Batch processing – which consists of making models and providing initial results for outlier detection based on data-driven models
- Real-time processing – is used in real-time to compare new clean data with the latest model for instant outlier detection

Same way of preparing data is used in both cases so transformed (prepared) data is in the same space

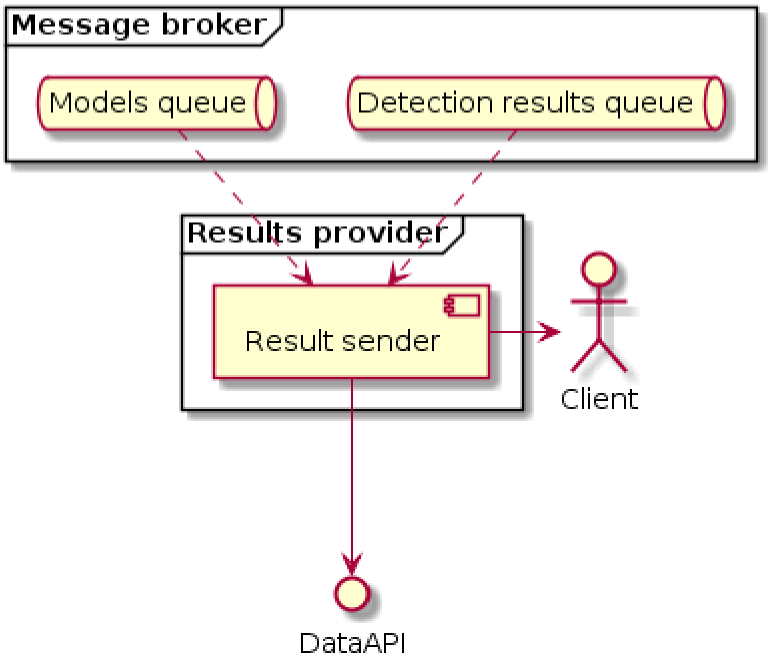


Figure 34 D2Lab v2 architecture (client feedback)

Result sender can be used to trigger alarms, send data for real-time anomaly detection, sending advice, etc.

- **Interfaces**: describes the interfaces provided and consumed by the component.
- **Subordinates**: describes subcomponents of the component (if any).
- **Data**: gives information about the data accessed or processed by the components and related data model.

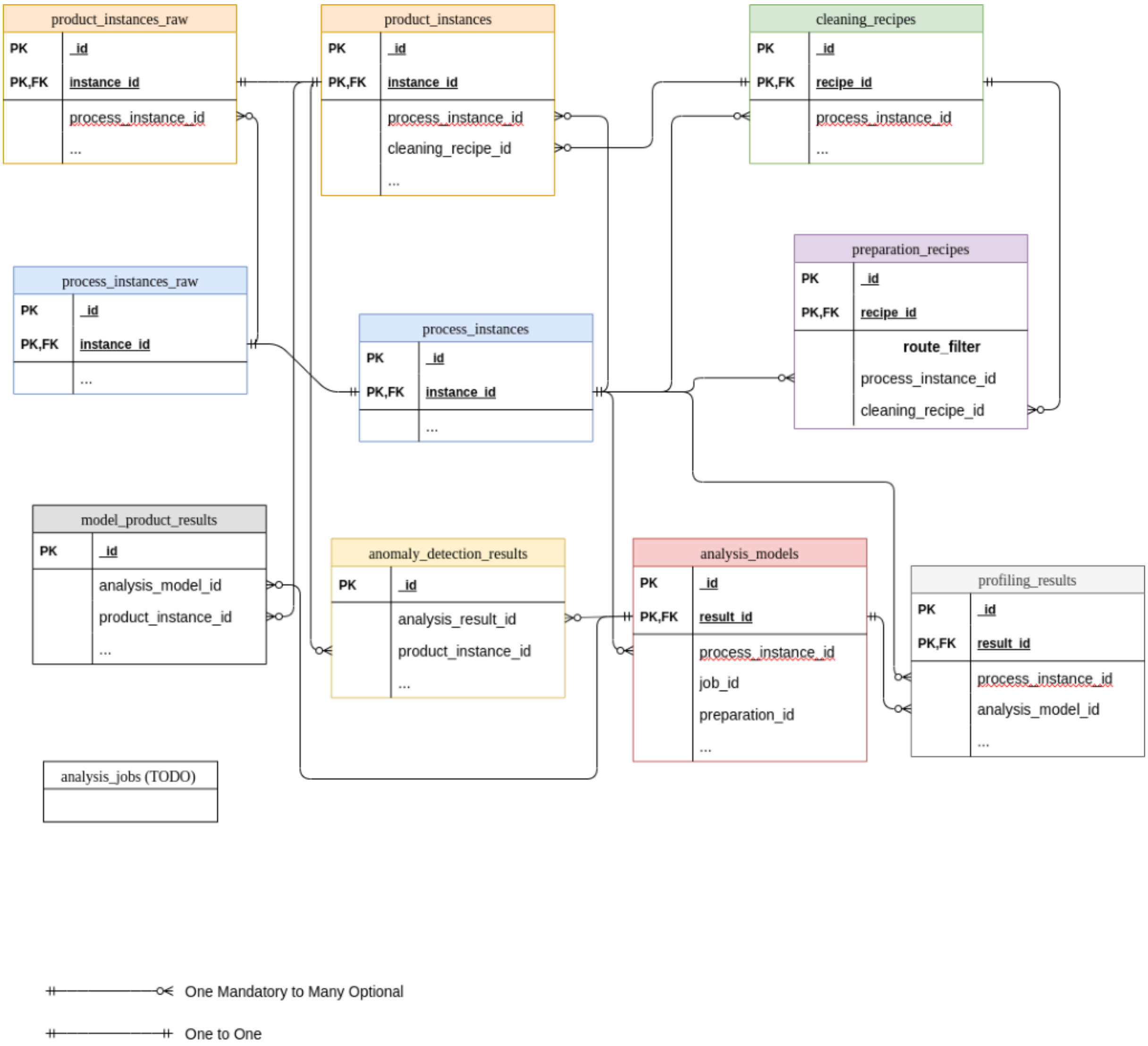


Figure 35 D2Lab Overview on collection relationships

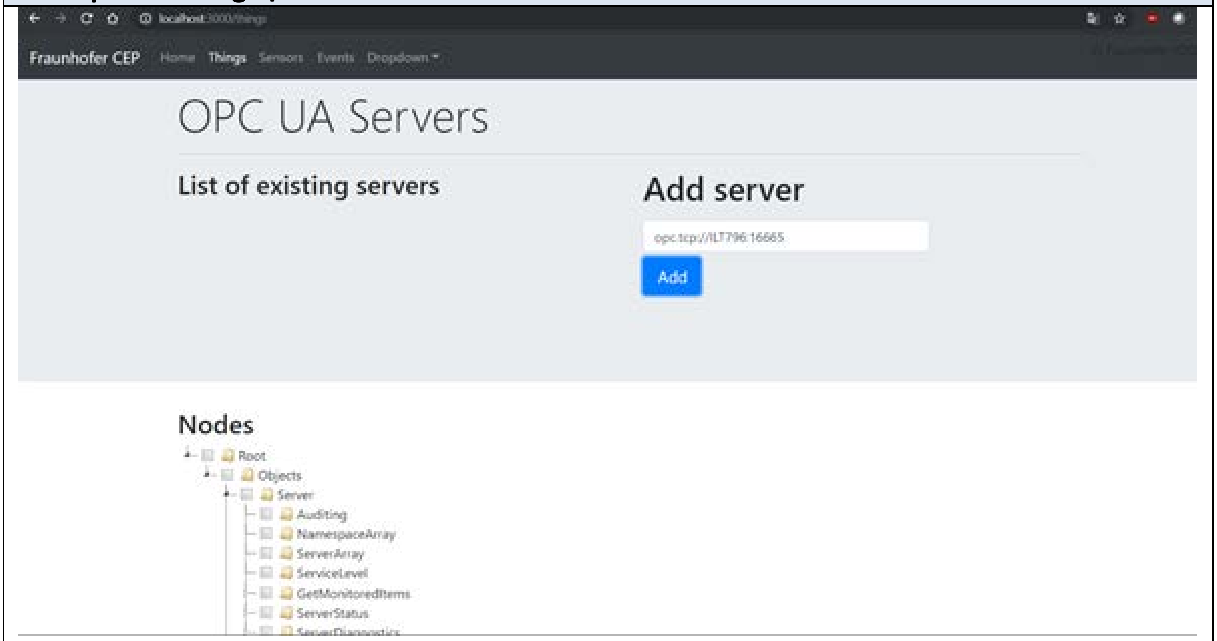
Name	Description	Reference
Data Analytics	The Data Analytics Layer of the BDVA Reference Model addresses required data transformation functionalities and related infrastructure components.	covered
Data Management	The Data Management Layer of the BDVA Reference Model addresses functionalities related to data backup, replication, curation, provenance, registries, indexing/search, metadata and ontologies.	
Data Processing Architecture	The Data Processing Architecture Layer of the BDVA-RM represents the processing frameworks implementing the logic of the	covered

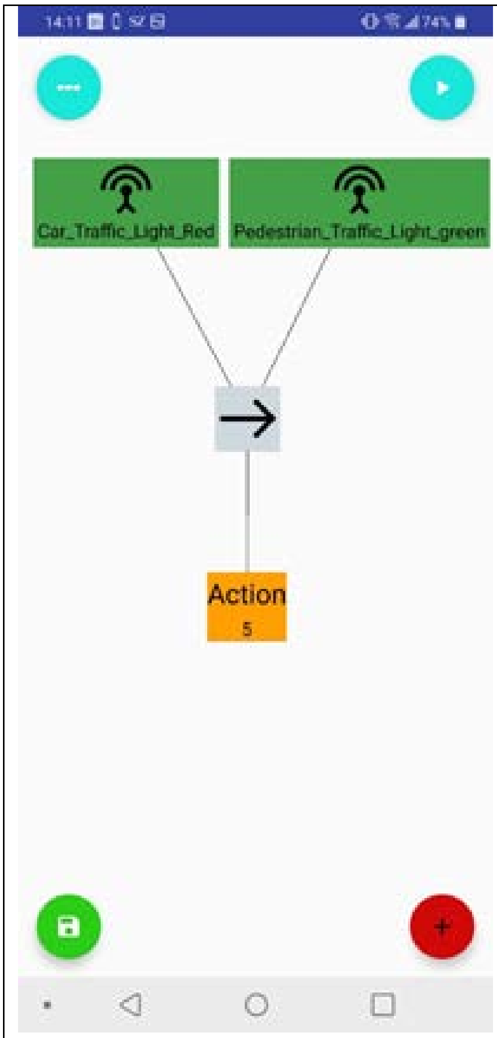
	analytics activities. It typically includes batch, interactive or streaming processing frameworks.	
Data Visualisation and User Interaction	The Data Visualization and User Interaction Layer of the BDVA-RM prepares elements of the processed data and the output of the analytic activity for presentation to the data consumer.	covered
Existing Infrastructure	The Existing Infrastructure Layer of the BDVA-RM represents existing infrastructure components or remote services on which are leveraged by the application components.	

D2Lab related to BDVA Reference Architecture Layers

15. Annex 6. Fraunhofer VISPAR and OPC UA FROST Server

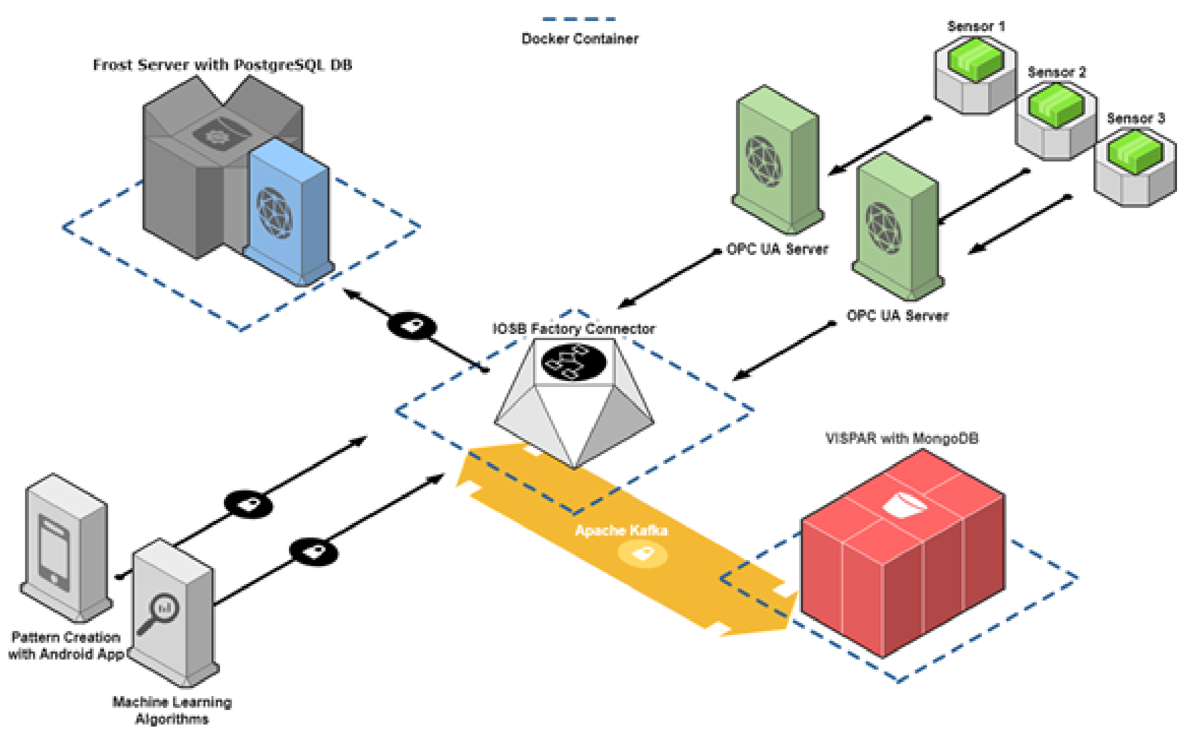
Component/Tool/Method/Framework/Service Name
VISPAR + Android App + Integration into IDS Connector
Short Description – incl. Purpose
VISPAR is a CEP solution developed by Fraunhofer IOSB. It is based on the popular CEP Engine Siddhi and uses the IOSB FROST Server with a standardized data model for storage. The standard used for data modelling is the SensorThings API. With this model, different sensors and their data can be modelled. In a factory, most sensors are connected via OPC UA, HTTP or MQTT. The FROST Server supports HTTP and MQTT. OPC UA integration was added into an IDS Connector. The International Data Space is a data network with focus on data sovereignty. Communication is secured and Usage Restrictions set by Data Owners are respected. We developed a GUI for the connector to add OPC UA servers to the FROST servers as STA sensors and transmit their data. VISPAR further simplifies the creation of CEP patterns by not requiring knowledge about SQL-like Pattern Languages like SiddhiQL. Instead, an Android App with easy-to-use GUI is used to create and model pattern. This graphical representation is then automatically transformed to SiddhiQL and deployed into VISPAR.
Function – suitable for which process steps (ICT/Data process) ***Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation***
Suitable for data collection via HTTP, MQTT, OPC UA Suitable for data curation via SensorThings API standard Suitable for data integration from different sources Suitable for data sharing via secure IDS connector and Usage Control Suitable for data access via more advanced Usage Control (extended access control) Suitable for data processing via CEP Patterns that create more complex and information-rich data Suitable for data analytics via CEP Patterns that analyse and calculate

Suitable for decision support via CEP Patterns that alarm in critical conditions
Examples of usage / illustrations
 opc.tcp://ILT796:16665 localhost:3000/things Fraunhofer CEP Home Things Sensors Events Dropdown OPC UA Servers List of existing servers Add server Add Nodes Root Objects Server Auditing NamespaceArray ServerArray ServiceLevel GetMonitoredItems ServerStatus ServerDiagnostics **Add opc ua servers and sensors**



Create CEP Pattern with sensors

Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)



Component	Function
Sensors	Sensors generate data, which is transmitted via OPC UA.
OPC UA Server	OPC UA Servers aggregate many sensors and production data from the field network.
IOSB FactoryConnector	IDS connector that subscribes to OPC UA nodes, models the data with the SensorThingsAPI and communicates with all other components. The connector can deploy and shut down other Docker Applications.
FROST Server	The Frost Server is a SensorThingsAPI implementation and is used to store data in PostgreSQL for machine learning applications. FROST supports HTTP and MQTT directly.
VISPAR	VISPAR is an IOSB implementation of the complex event processing engine Siddhi.
Apache Kafka	Kafka is a streaming processor and provides data streams instantly to VISPAR
Pattern Android App	Users can create, edit and visualize complex event patterns with an Android App. These patterns are stored by VISPAR in a MongoDB.

Interfaces (in/out) – system/user

HTTP and MQTT to model and store data in FROST server
OPC UA client in IDS connector to transmit to FROST server via HTTP

Data can be accessed securely via IDS connector (HTTP)
Factory can access data directly in FROST via HTTP, MQTT (not accessible from outside the factory)

Subordinates/parts – any platform dependencies
Docker dependency to deploy the different components with one run Java dependency for IDS connector Android dependency for Pattern App
Data (in/out)
In: Sensor data or OPC UA data Out: Complex Event Insights
Standards (any standards being used)
SensorThings API OPC UA, HTTP, MQTT DIN SPEC 27070 (Security Gateway)
Licenses, etc. (free for use in the project)
To be determined, free for research
TRL for overall component/tool and any parts/subordinates
TRL 7 Prototype in use for 2-3 years
References – incl. web etc.
https://www.internationaldataspaces.org/ **https://github.com/siddhi-io/siddhi** **https://www.iosb.fraunhofer.de/visIT/iot/#16**
To be considered in particular for the following COGNITWIN pilots
Relevant for pilots, when in need of real time sensor data stream processing

Component/Tool/Method/Framework/Service Name
FROST®-Server
Short Description – incl. Purpose
The Fraunhofer Open Source SensorThings API (FROST) Server is an open-source (LGPLv3.0) implementation of the OGC SensorThings API standard. It provides a standardized data models and web-based API to manage sensor meta data and measurements. Besides an OData-inspired REST-based API it also provides an MQTT API for subscribing to changes and creation of new elements.
Function – suitable for which process steps (ICT/Data process) ***Data collection, curation, integration, sharing, access, processing, analytics, decision support, control, visualisation***
• Suitable for data collection via HTTP, MQTT • Suitable for data curation via SensorThings API standard • Suitable for data integration from different sources
Examples of usage / illustrations
Basic HTTP operations Base URL: `http://some-server.org/FROST-Server/v1.0` • Read: GET   • `v1.0` → Get collection index   • `v1.0/Collection` → Get all entities in a collection   • `v1.0/Collection(id)` → Get one entity from a collection • Create: POST

  - `v1.0/Collection` → Create a new entity
- Update: PATCH
  - `v1.0/Collection(id)` → Update an entity
- Update: PUT
  - `v1.0/Collection(id)` → Replace an entity
- Delete: DELETE
  - `v1.0/Collection(id)` → Remove an entity

Read single entity (Thing)

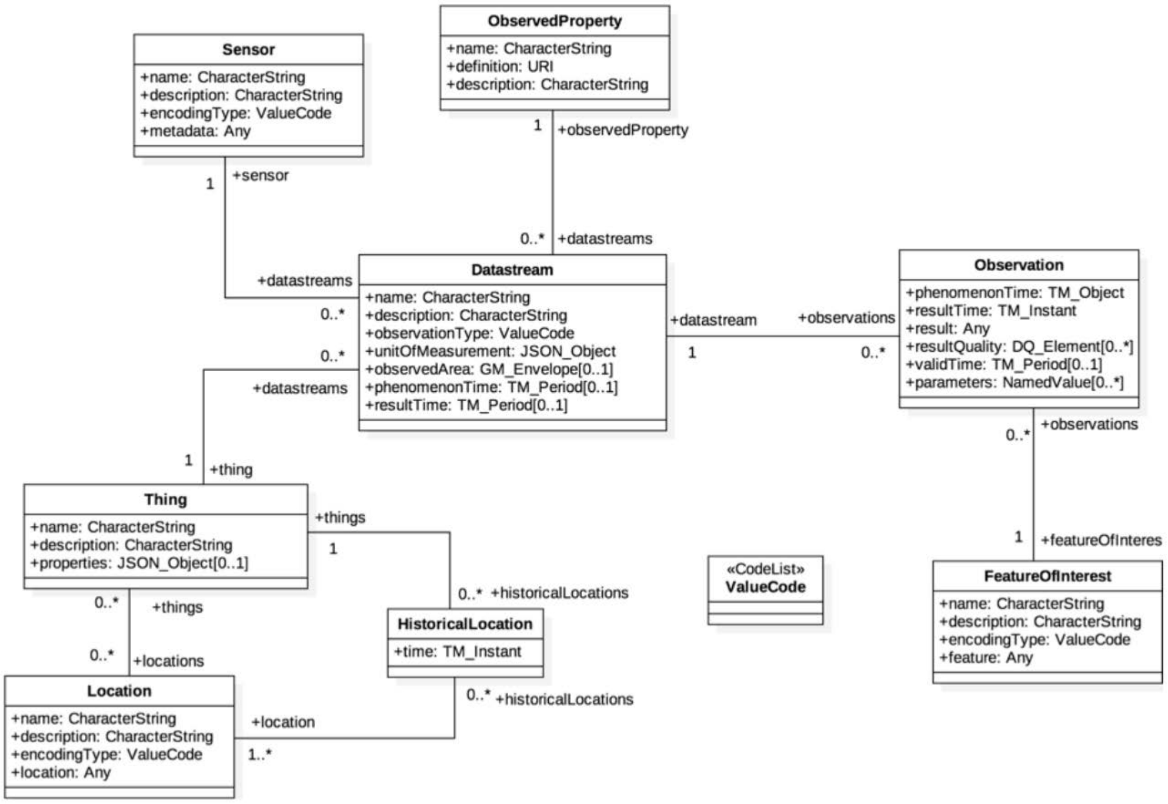
GET `http://some-server.org/FROST-Server/v1.0/Things(1)`

Response

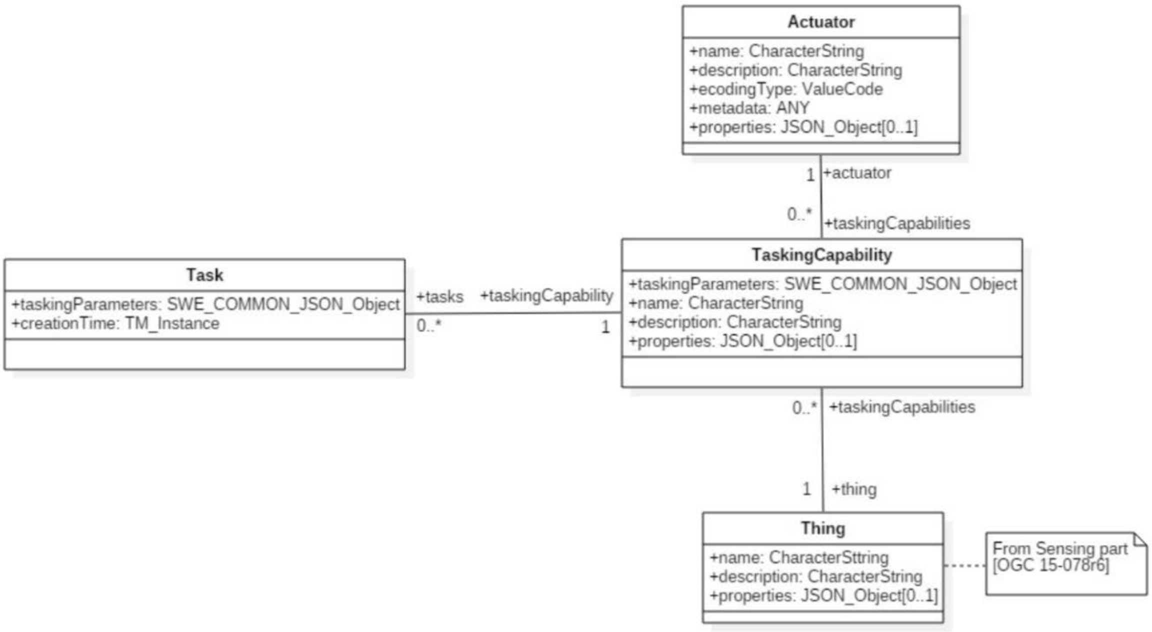
```
{
"name": "My camping lantern",
"description": "camping lantern",
"properties":
{
"property1": "it’s waterproof",
"property2": "it glows in the dark"
},
"Locations@iot.navigationLink": "Things(1)/Locations",
"HistoricalLocations@iot.navigationLink":
"Things(1)/HistoricalLocations",
"Datastreams@iot.navigationLink": "Things(1)/Datastreams",
"@iot.id": 1,
"@iot.selfLink": "/SensorThingsService/v1.0/Things(1)"
}
```

Example for a complex query

```
HTTP GET http://some-server.org/FROST-Server/v1.0/Things?
    $select=id,name,description,properties
    &$top=1000
    &$filter=properties/type eq 'station'
    &$expand=
        Locations,
        Datastreams(
            $select=id,name,unitOfMeasurement
            ;$expand=
                ObservedProperty($select=name),
                Observations(
                    $select=result,phenomenonTime
                    ;$orderby=phenomenonTime desc
                    ;$top=1)
        )
```

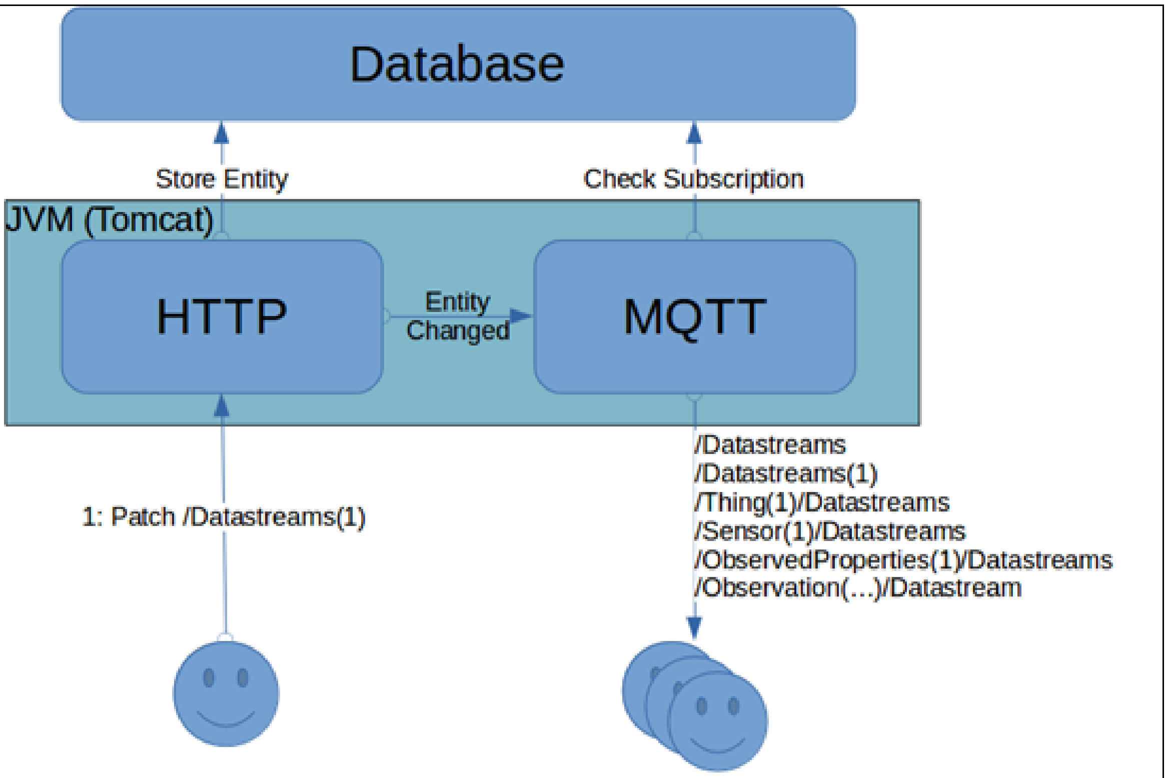


Data model for OGC SensorThings API Part 1: Sensing

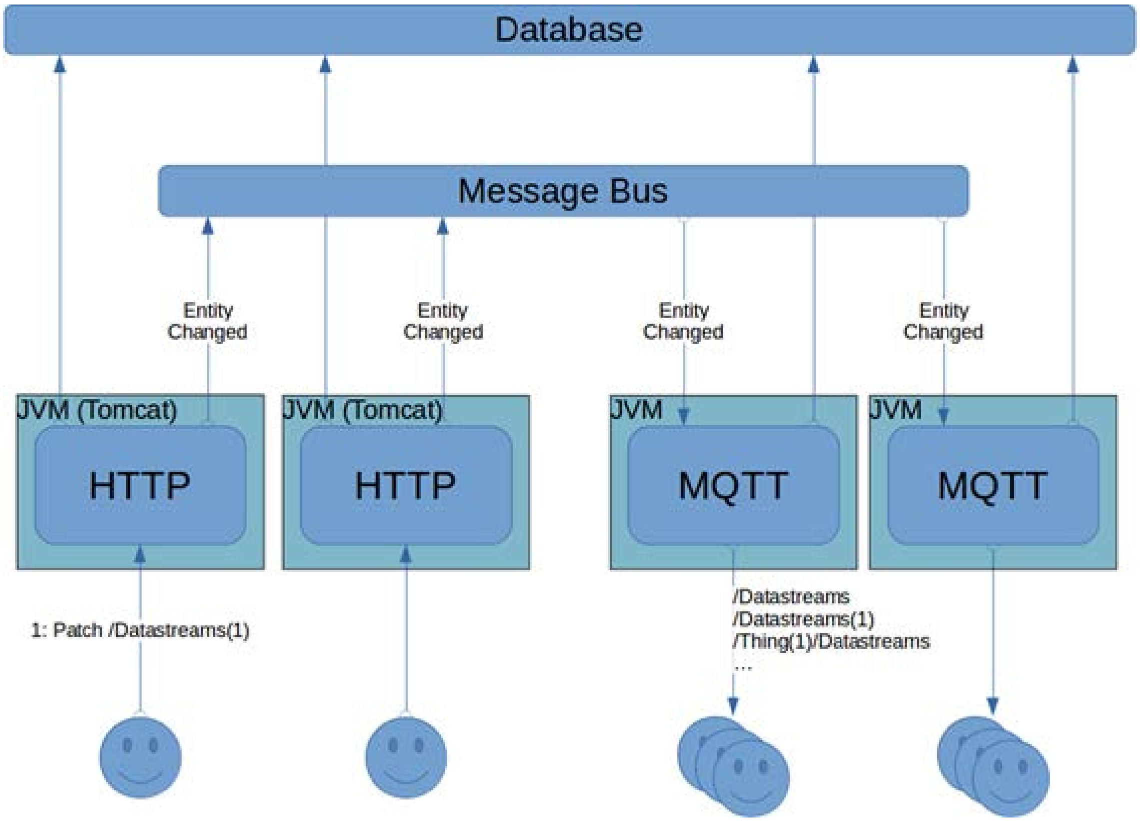

Actuator
+name: CharacterString
+description: CharacterString
+ecodingType: ValueCode
+metadata: ANY
+properties: JSON_Object[0..1]
1 +actuator
0..* +taskingCapabilities
Task
+taskingParameters: SWE_COMMON_JSON_Object
+creationTime: TM_Instance
+tasks
0..*
+taskingCapability
1
TaskingCapability
+taskingParameters: SWE_COMMON_JSON_Object
+name: CharacterString
+description: CharacterString
+properties: JSON_Object[0..1]
0..* +taskingCapabilities
1 +thing
Thing
+name: CharacterSttring
+description: CharacterString
+properties: JSON_Object[0..1]
From Sensing part
[OGC 15-078r6]

Data model for OGC SensorThings API Part 2 – Tasking Core

Overall architecture / pipeline / workflow (incl. figure – elements according to BDVA)



FROST-Server All-in-One Architecture



FROST-Server Separated HTTP & MQTT architecture (for better scalability)

Interfaces (in/out) – system/user

Interfaces are defined by OGC SensorThings API (STA)

For input, HTTP is the main interface. Via MQTT only new Observations can be added.

For output, the STA offers a HTTP interface with a very powerful query language based on the OData query language. Via MQTT subscribing to changes and creation of entities is possible.
Subordinates/parts – any platform dependencies
FROST-Server requires a database to store the data. Currently only implemented option is PostgreSQL with PostGIS, although database access decoupled via an interface, meaning one can easily implement adapter for other data stores. FROST-Server requires JRE to run.
Data (in/out)
In: Sensor meta data and observations Out: Sensor meta data and observations, events about entity creation/changes
Standards (any standards being used)
SensorThings API (inspired but not compatible to OData)
Licenses, etc. (free for use in the project)
LGPLv3.0
TRL for overall component/tool and any parts/subordinates
TRL 7/8 Reference implementation of the OGC SensorThings API Standard, compliance-tested
References – incl. web etc.
https://github.com/FraunhoferIOSB/FROST-Server
To be considered in particular for the following COGNITWIN pilots
Relevant for pilots, when in need of real time sensor data stream processing