Z - B R E 4 K

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

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

Z-BRE4K

Deliverable D2.2

IDS, connectors and containers ready for integration with AUTOWARE

Work Package 2

Operating system and networked machine simulators

Document type	:	Report
Version	:	V02
Date of issue	:	03/07/2018
Dissemination level	:	*PUBLIC*
Lead beneficiary	:	INOVA+ (P11)

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

EXECUTIVE SUMMARY

Abstract	Industrial Data Spaces (IDS) are being more and more used and accepted by many industrial sectors as a common ground for reliable and trustful data exchange across complex networks. Z-BRE4K architecture relies on IDS Connectors to link all its components, thus creating a true Z-BRE4K environment. Herewith, the architecture of the dedicated IDS Connectors is designed and described for reference at the implementation stage. In specific, the internal modules to properly send/receive data in agreed formats and to relevant endpoints of the system are defined and its configuration proposed. Moreover, specific architectures for the System Adapters at the shopfloor/edge level are devised.
Keywords	Industry Data Spaces; IDS; IDS Connector; FIWARE; Context Broker; NGSI; System Architecture; …

REVISION HISTORY

Version	Author(s)	Changes	Date
V01	Vitor Sousa (INOVA+); Hugo Faria (INOVA+); Alice Reina (HOLONIX); Weian Xu (INOVA+); Paloma Taboada (TRIMEK);	n/a	30-06-2018
V01.1	Daniel Gesto	Format changes	30-06-2018
V01.2	Jovana Milenkovic	Review and minor corrections.	02-07-2018
V01.3	Paloma Taboada (TRIMEK)	Review of C09+C10 associated System Adapter definition.	02-07-2018
V02	Hugo Faria (INOVA+)	Technical review of Use Cases. Introduction of List of Acronyms & Abbreviations; List of Figures and Tables. Minor corrections.	03-07-2018

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS/ABBREVIATIONS

API - Application Programming Interface;
AUTOWARE – Wireless Autonomous, Reliable and Resilient Production Operation Architecture for Cognitive Manufacturing;
CKAN – Comprehensive Knowledge Archive Network;
CMM - Coordinate Measuring Machine;
CPU - Central Processing Unit;
EARTO - European Association of Research and Technology Organisations;
FIWARE – Future Internet (Soft)Ware;
GUI – Graphics User Interface;
HDFS - Hadoop Distributed File System;
HW – Hardware;
IDAS - Intelligence Data Advanced Solution;
IDS – Industrial Data Space;
IIoT – Industrial Internet of Things;
IoT – Internet of Things;
NGSI - Next Generation Service Interface;
PDP - Policy Decision Point;
PEP – Policy Enforcement Point;
PLC - Power Line Communication;
QIF - Quality Information Framework;
ROI – Return on Investment;
SCM - Software Configuration Management;
SFDC – ShopFloor Data Collection;
SME – Small and Medium Enterprises;
STH – Short Time History;
SW – Software;
TCO – Total Cost of Ownership;
VCM – Version Control Management;
W3C – World Wide Web Consortium;

1 DEFINITION OF IDS

Industrial Data Spaces (IDS) is a virtual data space, leveraging existing standards and technologies to facilitate the secure and standardized exchange and easy linkage of data in a trusted business ecosystem. It proposes a Reference Architecture Model that ultimately envisages the capability of a person or corporation to be entirely self-determined regarding its data[1]. Data sovereignty is, thus, a central aspect for IDS. The standards materialize in the **Reference Architecture Model** itself and defined methods for secure data exchange between the various **Industrial Data Space connectors**.

Several strategic requirements are sought with IDS[1]:

- TRUST: Trust is the basis of the Industrial Data Space. It is supported by a comprehensive identity management focusing on the identification of participants and providing information about the participant based on the organizational evaluation and certification of all participants.
- SECURITY AND DATA SOVEREIGNTY: Components of the Industrial Data Space rely on current security measures. Next to architectural specifications, this is realized by the evaluation and certification of the components. In line with the central aspect of ensuring data sovereignty, a data owner in the Industrial Data Space attaches usage restriction information to its data before it is transferred to a data consumer. The data consumer may use this data only if it fully accepts the data owner's usage policy.
- ECOSYSTEM OF DATA: The architecture of the Industrial Data Space does not require central data storage capabilities. Instead, it pursues the idea of decentralization of data storage, which means that data physically remains with the respective data owner until it is transferred to a trusted party. This approach requires a holistic description of the data source and data as an asset combined with the ability to integrate domain-specific vocabularies for data. Brokers in the ecosystem enable comprehensive real-time search for data.
- STANDARDIZED INTEROPERABILITY: The Industrial Data Space Connector, being a central component of the architecture, is implemented in different variants and from different vendors. Nevertheless, each connector is able to communicate with every other connector or component in the ecosystem of the Industrial Data Space.
- NETWORK OF PLATFORMS AND SERVICES[2]: Providers of data can be individual enterprises, but also »things« (i.e. single entities within the »internet of things«, such as cars, machines, or operating resources) or individuals. Other Data Providers may be data platforms or data market-places currently being established in various industries. Furthermore, data services of various providers are made available via an AppStore.

[1] Boris Otto, Steffen Lohmann, Sebastian Steinbuβ, Andreas Teushcer, *et al.*, *IDS Reference Architecture Model – Industrial Data Space*, Version 2.0, Dortmund, 2018.
[2] Boris Otto *et al.*, Industrial Data Space – Digital Sovereignty Over Data, White Paper, Fraunhofer-Gesellschaft, München 2016.

- VALUE ADDING APPS: The Industrial Data Space enables app injection to connectors to add services on top of the pure data exchange. This includes services for data processing as well as the alignment of data formats and data exchange protocols, but also enables analytics on data by the remote execution of algorithms.
- DATA MARKETS: The Industrial Data Space enables the creation of novel, data-driven services that make use of data apps. It also fosters new business models for those. Being the central deliverable of the research project, the Reference Architecture Model of the Industrial Data Space (IDS-RAM) constitutes the basis for a variety of software implementations, and thus for a variety of commercial software and service offerings.

Different Architectural Layers are typically considered and treated at different governance perspectives:

- BUSINESS LAYER: The Business Layer facilitates the development and use of new, digital business models to be applied by the participants in the Industrial Data Space. It is thereby directly related to the Governance Perspective by considering the business point of view regarding data ownership, data provision, and data consumption, and by describing core service concepts such as data brokerage.
- FUNCTIONAL LAYER: The Functional Layer defines the functional requirements of the Industrial Data Space, and the concrete features resulting from them, in a technology-independent way. Beside the Clearing House and the Identity Provider, which are entities for which the relation to governance is obvious, the functionality of certain technical core components (e.g., the App Store or the Connector) relates to this Governance Perspective.
- PROCESS LAYER: Providing a dynamic view of the architecture, the Process Layer describes the interactions taking place between the different components of the Industrial Data Space. The three major processes are providing data, exchanging data, and publishing and using Data Apps.
- INFORMATION LAYER: The Information Layer specifies the Information Model, which provides a common vocabulary for the participants to express their concepts. It thereby defines a framework for standardized collaboration. The vocabulary plays a key role in the Governance Perspective because of its relevance for describing data by metadata in the Industrial Data Space.
- SYSTEM LAYER: The System Layer relates to the Governance Perspective due to its technical implementation of different security levels for data exchange between the Data Endpoints in the Industrial Data Space.

The main **benefits and strengths** of using IDS in many fields and use cases have been identified as follows[3]:

- Linking data of **multiple data sources**;

[3] Jan Jürjens, The Industrial Data Space: Digital Industrial Platform Across Value Chains in All Sectors of the Economy, presented at na EC Futurium Event.

- Integration of **different data types** (e.g. product data and environment data of production);
- Involvement of at least **two companies**;
- Integration of more than two **levels of the enterprise architecture** (e.g. »shopfloor« and »officefloor«);
- Foundation for »**Smart Services**«.

Looking specifically to the main functionality to be attained here in Z-BRE4K, the **unique linking and connection between different entities and components** of the platform, the definitions of IDS Connectors, either internal or external to a certain "limited" domain, bring relevant resources to the developers of Z-BRE4K. The participants/entities shall use a certified **Internal IDS Connector** to offer or import safely containerized views of their data, with well-defined usage controls. In larger networks, **specialized IDS Brokers** can offer services for searching, negotiating and monitoring service and data contracts. **Other service providers** can bring in data cleaning or analytics services through their **App Stores**, again connected to the system by IDS Connectors. For confidentiality as well as for performance reasons, service execution should have a choice between policy-based uploading and downloading data, or distribution of the service algorithms themselves which may require additional **External IDS Connectors**[4]. Figure 2 schematically depicts these interactions.

[4] Matthias Jarke and Christoph Quix, On Wharehouses, Lakes, and Spaces: The Changing Role of Conceptual Modeling for Data Integration, Chapter 16, *in* Conceptual Modelling Perspectives, Springer International Publishing AG, 2017, Cham – Switzerland.

2 UPDATED REVIEW OF THE STATE OF THE ART

2.1 Background

Founded in Germany, the activities of the Industrial Data Space are closely aligned with Plattform Industrie 4.0. It is important to note that Plattform Industrie 4.0 addresses all relevant architectural layers, whereas the Industrial Data Space initiative focuses on the data layer and economy.

Cross-company data exchange and inter-organizational information systems are not a new topic but have been around for decades. With the proliferation of Electronic Data Interchange (EDI) in the 1980s, many different data exchange scenarios emerged over time being accompanied by the development of respective standards.

Figure 1 shows the evolution of different classes of data exchange standards and identifies a need for standardization. Data sovereignty materializes in "terms and conditions" that are linked to the data upon its exchange and sharing. However, these terms and conditions (such as time to live, forwarding rights, price information etc.) have not been standardized yet. In order to foster the emergence of data sovereignty in the exchange of data within ecosystems, standardization activities are needed (for this purpose, Z-BRE4K conducts its own standardization initiatives in T7.1).

This does not mean that existing standards will become obsolete. Contrary to that, the overall set of standards companies need to comply with when exchanging and sharing data is extended.

The growing number of industrial cloud platforms is also driving the need towards a standard for data sovereignty. With the large number of different platforms emerging – driven by technology providers, software companies, system integrators, but also existing intermediaries – it is very much likely that the platform landscape will be heterogeneous – at least for a significant amount of time. Platform providers will increasingly have to provide capabilities for secure and trusted data exchange and sharing between their own platform and other platforms in the ecosystem.

Data owners and data providers will choose the platform depending on the business criticality and the economic value of the data goods they want to exchange and share via the respective platform. As the entire data resource of a company consists of data of different criticality and value, many companies will use different platforms for different needs.

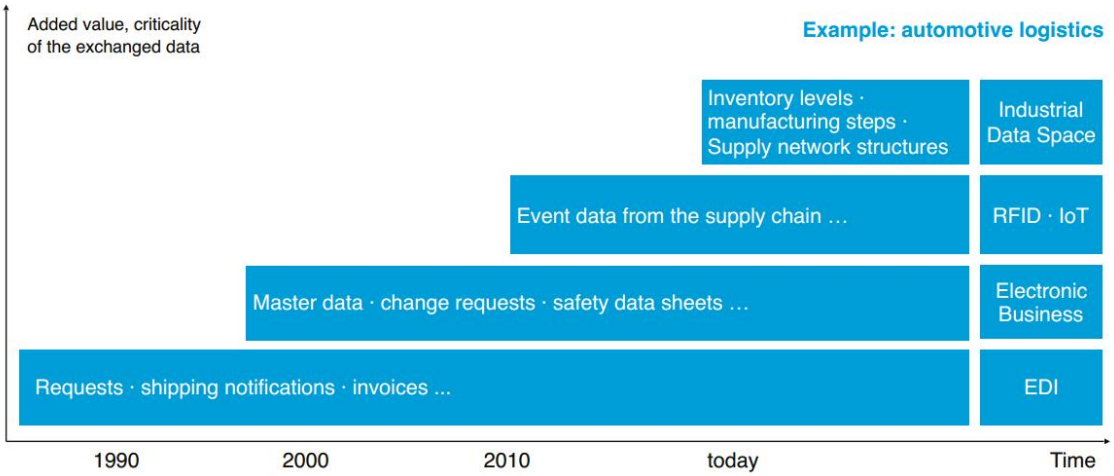

Figure 1. Data Exchange Standards

Currently, important efforts are being driven by the following guidelines (and objectives)[1]:

- OPEN DEVELOPMENT PROCESS: The International Data Spaces Association is a non-profit organization institutionalized under the German law of associations. Every organization is invited to participate, as long as it adheres to the common principles of work.
- RE-USE OF EXISTING TECHNOLOGIES: Inter-organizational information systems, data interoperability, and information security are well-established fields of research and development, with plenty of technologies available in the market. The work of the Industrial Data Space initiative is guided by the idea not to "reinvent the wheel", but to use existing technologies (e.g., from the opensource domain) and standards (e.g., semantic standards of the W3C) to the extent possible.
- CONTRIBUTION TO STANDARDIZATION: Aiming at establishing an international standard itself, the Industrial Data Space initiative supports the idea of standardized architecture stacks (in this particular aspect, Z-BRE4K has specific agenda for this, through the T7.1 – Standardisation Activities.

The data and service architecture constitute the functional core of the Industrial Data Space concept. It specifies the functions to be implemented in the pilot applications. The data and service architecture does not however make decisions on the use of certain technologies or applications[2]. The functions are arranged in several **predefined blocks**, which in turn are assigned to one of the following **functional components**: AppStore, Broker, Connector.

Table 1. Data and service architecture

Industrial Data Space AppStore	Basic Data Services Provisioning	Data Service Management and Use	Vocabulary Management	Software Curation
	Data Provenance Reporting Data Transformation Data Curation Data Anonymization	Data Service Publication Data Service Search Data Service Request Data Service Subscription	Vocabulary Creation Collaborative Vocabulary Maintenance Vocabulary/Schema Matching Knowledge Database Management	Software Quality and Security Testing
Industrial Data Space Broker	Data Source Management	Data Source Search	Data Exchange Agreement	Data Exchange Monitoring
	Data Source Publication Data Source Maintenance Version Controlling	Key Word Search Taxonomy Search Multi-criteria Search	»One Click« Agreement Data Source Subscription	Transaction Accounting Data Exchange Cleaning Data Usage Reporting
Industrial Data Space Connector	Data Exchange Execution	Data Preprocessing Software Injection	Remote Software Execution	
	Data Request from Certified Endpoint Usage Information Maintenance (Expiration etc.) Data Mapping (from Source to Target Schema) Secure Data Transmission between Trusted Endpoints	Preprocessing Software Deployment and Execution at Trusted Endpoint	Data Compliance Monitoring (Usage Restriction etc.) Remote Attestation Endpoint Authentication	

The central functional entity of the Industrial Data Space is the Connector (IDS Connector). It facilitates the exchange of data between participants. The Connector is basically a dedicated communication server for sending and receiving data in compliance with the Connector specification. Participants should be able to run the Connector software in their own IT environment. Alternatively, they may run a Connector on mobile or embedded devices. The Connector must receive data from an enterprise backend system, either through a **push mechanism or a pull mechanism**. The data can be provided via an interface or pushed directly to other participants. A **data processing app** (subtype of a Data App) should provide a single, clearly defined processing functionality to be applied on input data for **producing an expected output**.

The Connector connects the Data Provider and the Data Consumer to facilitate the exchange of data. Figure 2 illustrates (with two alternative schemes) how data usage control is integrated into the functionality of the Connector. The data flow can be controlled in the respective **Message Router** of the Connector. Connectors implemented as prototypes so far use Apache Camel for data usage control across different systems and applications. Apache Camel allows integration of interceptors being executed each time before and after a processor is executed[1].

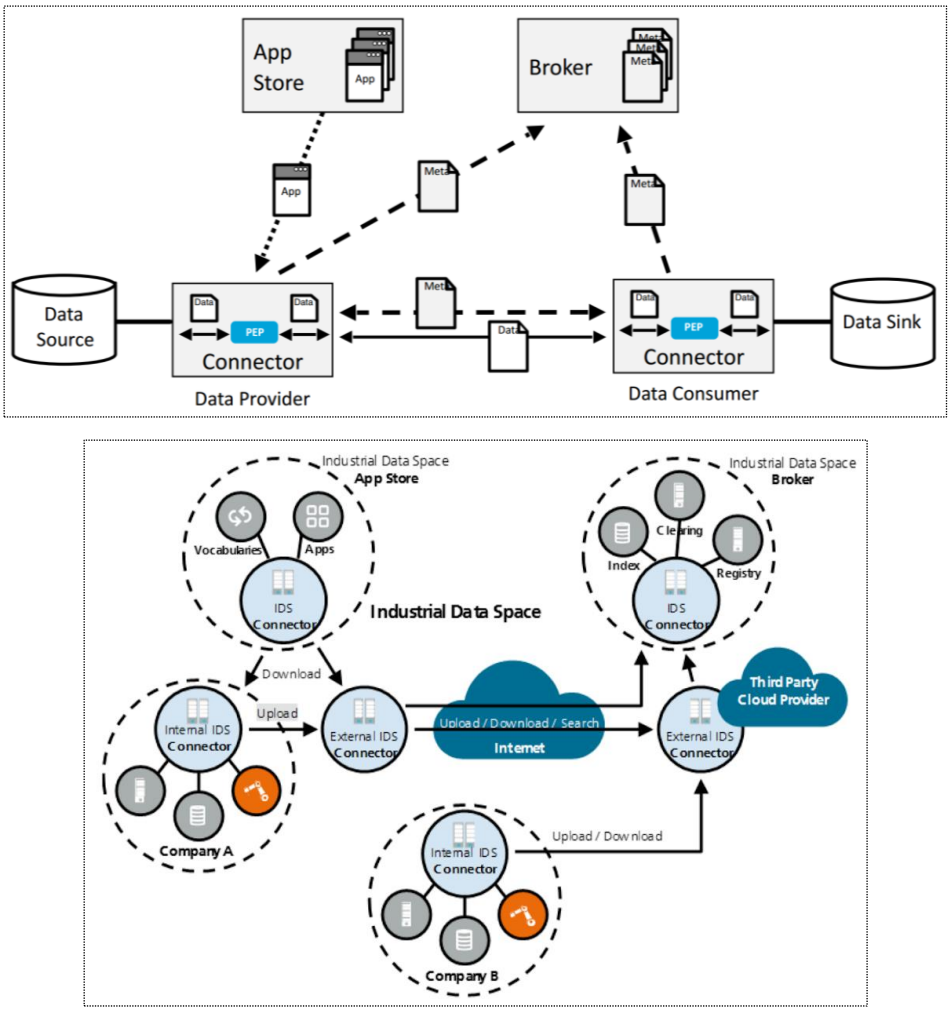

Figure 2. Typical Usage Control for the Data Exchange in IDS environment. Non-contextualized schematics (upper) and contextualized schematics (lower).

Because of the correlation between good data quality and maximizing the value of data as an economic good, the Industrial Data Space explicitly addresses the aspect of data quality. Due to this premise, the Industrial Data Space enables its participants to assess the quality of data sources by means of publicly available information and the transparency it provides with regard to the brokerage functionality it offers. Especially in competitive environments, this transparency may force Data Providers to take data maintenance more seriously. By extending the functionality of the Connector with self-implemented Data Apps, the Industrial Data Space lays the foundation for automated data (quality) management.

The technical elaboration of these and other aspects of the IDS reference architecture is still in an early stage. Many of the mapping and integration approaches for heterogeneous data must be augmented by conceptual models of the players and their desires and permissions, as well as of the physical architecture which explicitly addresses data sharing (or not-sharing) relationships and various optimization strategies around the cloud computing and edge computing literatures. If conceptual modelling is manually done at all in such a setting, it must occur across

organizational boundaries and in almost real-time, such that near-realtime collaborative modelling is expected to play a major role in such a setting[4].

Z-BRE4K platform will not only develop its own adapted architecture (based on conceptual definitions, compliant with standards and architectures of AUTOWARE® and FIWARE®), as it will also contribute to such collaborative modelling initiatives through its effective results and dissemination activities. Within the scope of predictive maintenance, Z-BRE4K aims at establishing a reference architecture. Particularly, specific architecture for **dedicated IDS Connectors will be designed and developed**, for which detailed descriptions and rationale are provided herewith.

2.2 Participants, Entities and Roles

In general IDS implementations several roles are commonly defined and accepted which build the proper sharing of tasks and responsibilities within each platform. Most roles require certification of the organization, entity or device that wants to assume that role, including certification of the technical, physical, and organizational security mechanisms it implies. This is considered a measure to establish **trust** among all "participants" (especially regarding roles that are crucial for the functioning of the Industrial Data Space, such as the Broker Service Provider, the App Store, the Identity Provider, or the Clearing House).

The main ones may be considered as follows:

- DATA OWNER: The Data Owner holds all legal rights of, and has complete control over, its data. Usually, a participant acting as a Data Owner automatically assumes the role of the Data Provider as well. However, there may be cases in which the Data Provider is not the Data Owner (e.g., if the data is technically managed by a different entity than the Data Owner, such as in the case of a company using an external IT service provider for data management).
- DATA PROVIDER: The Data Provider makes data available for being exchanged between a Data Owner and a Data Consumer. To submit metadata to a Broker, or exchange data with a Data Consumer, the Data Provider uses software components that are compliant with the Reference Architecture Model of the Industrial Data Space. Providing a Data Consumer with data from a Data Owner is the main activity of the Data Provider. To facilitate a data request from a Data Consumer, the Data Provider should provide a Broker Service Provider with proper metadata about the data. However, a Broker Service Provider is not necessarily required for a Data Consumer and a Data Provider to establish a connection. Exchanging data with a Data Consumer needs not necessarily be the only activity of the Data Provider. At the end of a data exchange transaction completely or partially executed, for example, the Data Provider may log the details of the successful (or unsuccessful) completion of the transaction at a Clearing House to facilitate billing or resolve a conflict. Furthermore, the Data Provider can use Data Apps to enrich or transform the data in some way, or to improve its quality (Data Apps are specific applications that can be integrated into the data exchange workflow between two or more participants in the Industrial Data Space). If the technical infrastructure for

participating in the Industrial Data Space is not deployed by the Data Consumer, a Data Provider may use a Service Provider to connect to the Industrial Data Space.

- DATA CONSUMER: The Data Consumer receives data from a Data Provider. From a business process modeling perspective, the Data Consumer is the mirror entity of the Data Provider; the activities performed by the Data Consumer are therefore similar to the activities performed by the Data Provider. Before the connection to a Data Provider can be established, the Data Consumer can search for existing datasets by making an inquiry at a Broker Service Provider. The Broker Service Provider then provides the required metadata for the Data Consumer to connect to a Data Provider. Alternatively, the Data Consumer can establish a connection with a Data Provider directly (i.e., without involving a Broker Service Provider). In cases in which the information to connect with the Data Provider is already known to the Data Consumer, the Data Consumer may request the data (and the corresponding metadata) directly from the Data Provider. Like a Data Provider, the Data Consumer may log the details of a successful (or unsuccessful) data exchange transaction at a Clearing House, use Data Apps to enrich, transform, etc. the data received, or use a Service Provider to connect to the Industrial Data Space (if it does not deploy the technical infrastructure for participation itself).
- DATA USER: Similar to the Data Owner being the legal entity that has the legal control over its data, the Data User is the legal entity that has the legal right to use the data of a Data Owner as specified by the usage policy.
- BROKER SERVICE PROVIDER: The Broker Service Provider is an intermediary that stores and manages information about the data sources available in the Industrial Data Space. As the role of the Broker Service Provider is central but non-exclusive, multiple Broker Service Providers may be around at the same time (e.g., for different application domains or even different locations).
- IDENTITY PROVIDER: The Identity Provider should offer a service to create, maintain, manage and validate identity information of and for participants in the Industrial Data Space.
- APP STORE: The App Store provides Data Apps, i.e., applications that can be deployed in the Industrial Data Space to facilitate data processing workflows. Data Apps might be certified by a Certification Body.
- APP PROVIDER: App Providers develop Data Apps to be used in the Industrial Data Space. To be deployable, a Data App has to be compliant with the system architecture of the Industrial Data Space. Each Data App must be published in the App Store for being accessed and used by Data Consumers and Data Providers. App Providers should describe each Data App using metadata (in compliance with a metadata model) with regard to its semantics, functionality, interfaces, etc.).
- VOCABULARY PROVIDER: The Vocabulary Provider manages and offers vocabularies (i.e., ontologies, reference data models, or metadata elements) that can be used to annotate and describe datasets. In particular, the Vocabulary Provider provides the Information Model of the Industrial Data Space, which is the basis for the description of data sources. In addition, other domain specific vocabularies can be provided.

- SOFTWARE PROVIDER: A Software Provider provides software for implementing the functionality required by the Industrial Data Space (i.e., through software components). Unlike Data Apps, software is not provided by the App Store, but delivered over the Software Providers' usual distribution channels, and used on the basis of individual agreements between the Software Provider and the user (e.g., a Data Consumer, a Data Provider, or a Broker Service Provider).
- CERTIFICATION BODY AND EVALUATION FACILITY: The Certification Body and the Evaluation Facility are in charge of the certification of the participants and the technical core components in the Industrial Data Space.

In a simplified view, one may consider the following distinction of Four Basic Roles in IDS[3]:

Table 2. Industrial Data Space: Four Basic Roles

Data Owner	Data User	Broker	Certification Authority
Provides data Operates endpoint in Industrial Data Space (or access via service provider). Defines terms of use and fees of data.	**Uses data** for providing **services** or for **internal purposes.** Satisfies terms for use of data.	**Brings** data owner and data user **together.** Operates »data directory«. Undertakes monitoring and clearing tasks.	**Certifies participants** to standards of Industrial Data Space (e.g. security, terms of use, use of standards)

2.3 Applications / Use Cases

Envisaging a true European Data Space, several initiatives were and are being conducted, accounting for the participation of a large number of European organizations (from academy to industry), specialists and stakeholders. Networks and "task forces" like the EARTO Working Group – "Towards a European Data Space", Smart Industry Platform (NL), European Open Science Cloud, FIWARE and Industrie 4.0 (DE), are pushing Industry Data Spaces into a central role in European IT projects and enforcing the internationalization of IDS.

Private consortia of well-known European companies like ATOS, PwC, Santander, Siemens, Telefonica, and projects like I4MS and EOSCpilot are improving the wide perception of IDS as a reliable, useful and integrative tool for European and International platforms dealing with critical and analytical management of data.

In the recent past, a few pilot applications have been implemented, in fields like (i) High Performance Supply Chains, (ii) Life Sciences and (iii) Traffic Management, respectively in use

cases of (i) controlling trucks in inbound logistics, (ii) developing medical and pharmaceutical products and (iii) complete transport monitoring[5].

However, only a few use cases implemented so far truly relate with Z-BRE4K thematic. These are[6]:

- DATATRONiQ – Predictive Maintenance and Process-Accompanying Quality Assurance

DATATRONiQ GmbH developed DATATRON, an intelligent device for data recording that collects and evaluates telemetry data from production machines and transfers them into cloud services. Operability is to be extended, depending on each respective purpose, so that the data to be transferred are supplemented by user profiles and authorisations and the access management that goes along with them.

In this application scenario, an IDS Provider Connector is deployed as an intermediary between DATATRON, as the data source, and the consumers of data. The consumers can be external users or can be within the same organisation.

As part of the use case, a number of IDS connectors were implemented. These are particularly suitable for telemetry data as they occur, for example, in the field of industrial series production. Particular attention was paid to a simple and flexible way of configuring the usage rights linked to the data – depending on the respective source and the planned user.

After acquiring machine, operation and process data (e.g. currents, moments, vibrations, fault messages and status information) DATATRON transforms and compresses them as required before they are forwarded to the IDS Provider Connector. This in turn ensures that the data are automatically and continuously forwarded to external users. Always provided that the latter produce the necessary authorisations and follow the rules.

- FIWARE – Predictive Maintenance of Fleets

Maintaining vehicles, not at fixed intervals, but according to predictive necessity – this is what the topic “Predictive Maintenance” is about. In this use case, vehicle parameters are used to avoid technical breakdowns and therefore to increase availability. The IDS FIWARE architecture enables secure data exchange between vehicles and fleet operators to improve predictive maintenance for vehicles. The hardware component from Stratio Automotive, which is installed in the vehicles, transmits information such as the actual engine temperature or the battery charge status to the server. This is where the true intelligence of the system is. The server matches the current data with recorded data and complements them with further data about weather conditions, fine dust pollution or traffic situations. By matching these data, the system generates new information. In this way, problems are identified before they occur.

[5] International Data Spaces – International Data Spaces Association (https://www.internationaldataspaces.org/en/industrial-data-space/#anwendungen)
[6] International Data Spaces – Use Case Overview (Brochure), Industrial Data Spaces Association, Dortmund.

If the system recognises a potential mechanical error, it issues information in real time. The fleet management and the service station know which vehicle requires repairing at which location even before the driver arrives at the depot. The merging and controlled exchange of data from different areas are guaranteed via the IDS Connector. Thanks to IDS, participants can only use the confidential information if they are authorised to do so.

All operators, and even users of vehicle fleets, can benefit from the FIWARE use case "Predictive Maintenance of Fleets". In this way, for example, passengers using a transport company can use apps to track the exact position of their bus.

- FIWARE – Zero Break-downs/Defect Manufacturing

As part of Industry 4.0, big data analytics are useful in predictive manufacturing and are a major theme for industrial technology development. To assist manufacturers in maintaining a competitive edge in operational management control and in improving their production efficiency and yield rates, the FIWARE Foundation, in cooperation with the Industrial Data Association, is developing a distributed data management solution with integrated learning capability.

The Zero Break-downs/Defect Manufacturing Use Case has been developed by selected members of the FIWARE Foundation (MARTEL, Atos, NEC and UPM) in collaboration with the Swiss mechanical engineering company Georg Fischer AG and the Innovalia Group. It shows how the maintenance of milling and coordinate measuring machines (CMMs) can be improved by exchanging plant data in a confidentiality-preserving way. Each of the two machines can improve configuration and maintenance by using the data produced by the other machine. The contextual data is exported directly from the installation via the FIWARE platform. Thus, both machines work synchronously and adapt to each other for the best possible result of the manufactured end product. Maintaining vehicles, not at fixed intervals, but according to predictive necessity – this

  - An advanced machine learning algorithm analyzes process data collected from production systems to provide early warning for anomalies and system failures and to predict product quality.
  - It improves predictive maintenance tasks, reduces failures.
  - It improves the quality management of the components.
  - Data analysis is based on the manufacturing process, overriding information silos.
  - The Zero Break-downs/Defect Manufacturing solution ensures secure and certified data exchange between companies in the supply chain without the data owners losing sovereignty – control – over their own data. It therefore helps to utilise and spread smart service concepts.
  - It ensures that only specific data are exchanged and only for specific purposes.
  - The solution, based on FIWARE, is open source, easily customized, scalable and interoperable, reducing the ROI and TCO, making it affordable for SMEs.

- SETLOG – Predicting Lead Times

SETLOG, together with OSCA®, develops tailor-made SCM and VCM software. Many of the software developer's customers come from the textile and consumer goods industry: companies that buy and produce goods globally. Because of individual customer requirements, reduced product life cycles and increasingly volatile demand, companies are being put under more and more pressure. When delivering their products, however, they still rely on planning data as no real-time data are available. This complicates the control of the processes that are part of procurement and distribution logistics. It is difficult to predict when goods will actually arrive at their destination. That leaves little scope for planning further steps, such as door and warehouse planning or (pre-) order picking. The resulting planning uncertainties lead to increased buffer times in the supply chain. The consequence is a lack of transparency influencing further processes and therefore invalid planning for goods deliveries. Delayed deliveries cost time, money and resources.

The application scenario "Predicting Lead Times" aims to plan supply chains in an intelligent and cost- and process-optimised way. Transport data from the past are combined with planning data from the enterprise resource planning system, actual data from OSCA® and public data in order to obtain an exact statement of transport time and delivery time. INTERNATIONAL DATA SPACES helps the companies involved in the supply chain to connect to each other.

The data gathered for the companies are anonymised in the process so that they do not disclose any company secrets but nevertheless offer added value for other companies. The participating companies and their data are protected by the IDS architecture.

- nicos AG – Identity Provider in the Environment of IDS

nicos AG connects globally operating companies to their international locations and production sites via secure global data networks. In addition to strategic planning, network design and the provision and setup of all network components, nicos takes care of the reliable operation of customer networks.

For INTERNATIONAL DATA SPACES the company provides one of the key components of the entire IDS architecture: the IDS Identity Provider. Only because of this crucial component are the participating connectors able to authorise and authenticate themselves. The user has to prove their identity to the system. Only then can it communicate with other users. The data provider has information that it makes available to other participants in INTERNATIONAL DATA SPACES. Thanks to the IDS Identity Provider, these data are only exchanged if certified partners request them. All participants retain sovereignty over their own data at all times.

- DATA AHEAD – Renewable Energy Data Management – Readiness for Multi Stakeholding

Data Ahead developed out of a system company for measurement, control and automation technology to become a specific provider of industrial mass data logistics. The company provides system integrators and application companies with specifically configured gateways, edge-computing and high-speed architectures. This logistics company for industrial data provides an

innovative access architecture for the data management of renewable energy in the application scenario for “Renewable Energy Data Management – Readiness for Multi Stakeholding”: a search engine for mass raw data that simultaneously guarantees variety, volatility, volume, speed and ubiquity. The architecture includes more than 2,500 edge devices and ensures users access to raw data in original granularity without compression or archiving – in less than a second. This architecture is built so that anybody who wants to do something with the data at a later point in time can correlate them completely freely. For example, network agencies that want to decide automatically and within seconds which field is to be fed by a local battery or wide area network. In future, this topic could be relevant for micro service providers and such providers in the Internet who build their business models on freely accessible raw data.

In the Industrial Internet of Things (IIoT) no company can depict the entire value-added chain on its own. This is only possible in collaboration with the best-in-class players that are part of INTERNATIONAL DATA SPACES.

3 INTEGRATION INTO Z-BRE4K ARCHITECTURE

The IDS core entities that will be used in Z-BRE4K are the Connector and the Broker. However, due to the selected decentralized architecture scheme, the Brokers will be embedded within the Connectors and therefore only Connectors will be defined and developed.

In section 3.1 the relationship and integration of the dedicated IDS Connectors within the overall Z-BRE4K platform is defined and described. As a general starting point, one may define the Z-BRE4K IDS Connectors as the "gates" and "corridors" that allow communication of data between all components of the platform. Figure 3 depicts a simplistic analogy. Therefore, several IDS Connectors will need to be populated across the Z-BRE4K architecture.

In section 3.2 the specificities of the inputs and outputs of each Connector type are addressed.

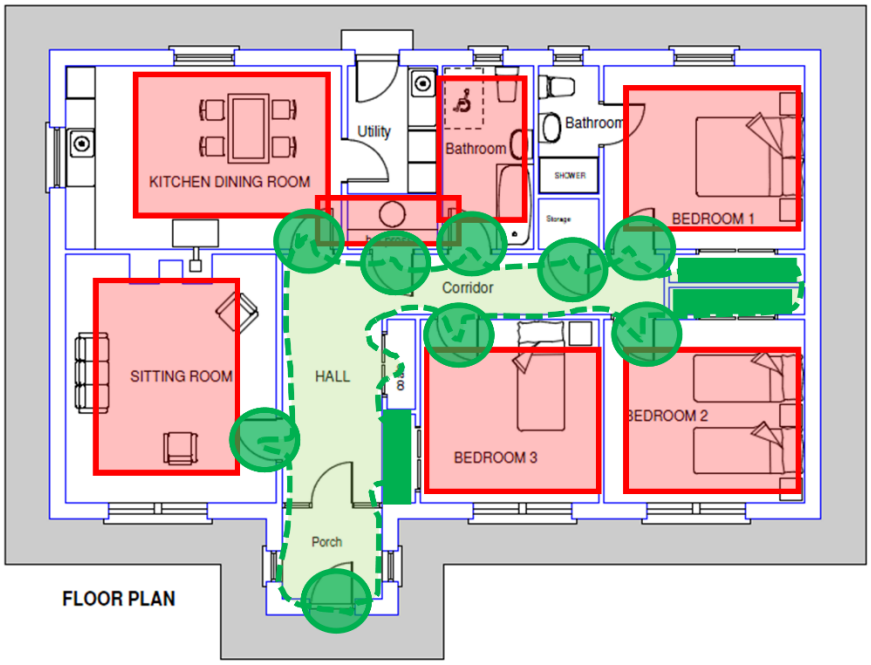

Figure 3. Analogy with a typical house floor plan. Rooms (representing the Z-BRE4K components) are connected by Doors and Corridors (representing the Z-BRE4K IDS Connectors).

3.1 Z-Bre4k Scheme

The global architecture of Z-BRE4K (D1.3) defines several components of the system working at potentially 4 different levels: Work Space, Edge, Fog and Cloud. Operational devices (production machines, workcells, quality check/inspection equipment, etc) operate at the shop floor level, whereas the analytical components (logics ruling, machine learning, modelling, knowledge extraction, prediction, decision support, big data computing, etc) act at cloud level.

As a principle, all end points of communication segments within Z-BRE4K environment shall have an IDS Connector coupled to it. Communication is granted exclusively through IDS Connectors. This strategy allows one to define a general architecture for the Connectors that concentrate all the needs for identification, pre-processing, redirecting and compliance of data under exchange

process, under a FIWARE compliance scheme and, more importantly, using the common selected standards for data configuration and formats for entire Z-BRE4K platform (e.g. NGSI). In Figure 4 one may observe this.

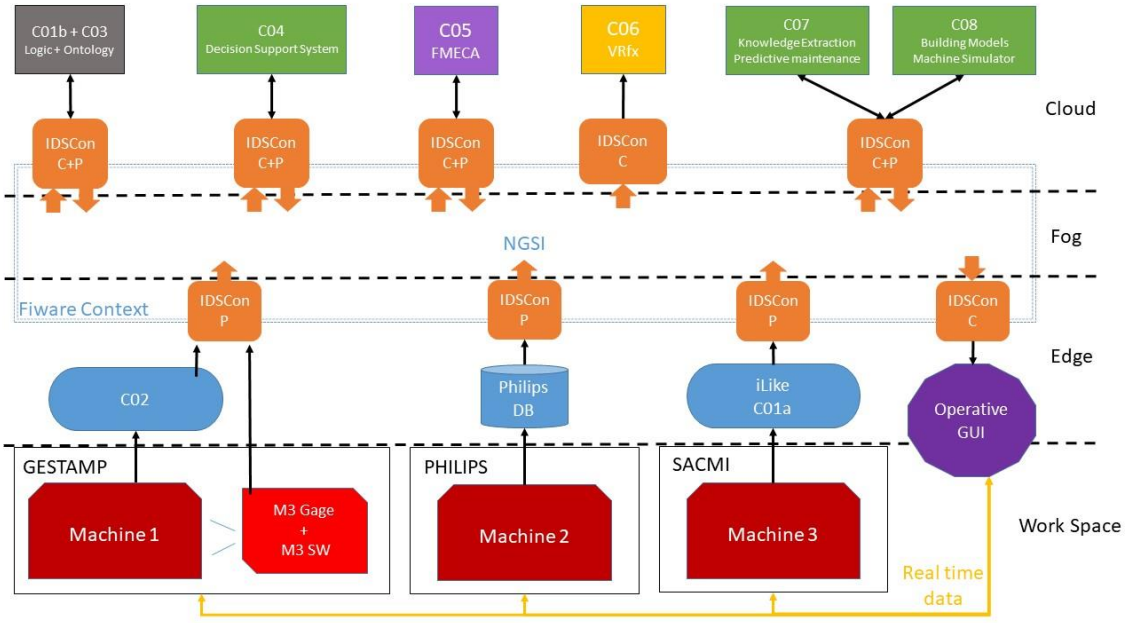

Figure 4. Schematics of the incorporation of the IDS Connectors into the global Z-BRE4K architecture

Since the 3 machines (corresponding to the 3 use cases under development – GESTAMP, PHILIPS and SACMI) have dedicated systems for collection of relevant data (Component C02 for GESTAMP, PHILIPS DataBase for PHILIPS and iLike Machine-based Component C01a for SACMI), the first entry points into the Z-BRE4K exclusive communication routes are at the output levels of these components (C02, Philips DB and C01a). These components are not intended to receive/consume data from the other Z-BRE4K components, and thus their associated **IDS Connectors are of type P** (only provider).

Then, all other components of the Z-BRE4K architecture (all in the cloud level) have **two-ways IDS Connectors (C+P)**. Exception is observed in the VRfx component, whose output is expected to work at a dedicated SW/HW interface (external to Z-BRE4K development).

At the Work Space and Edge levels there might be additional data flow routes, linking the machines (or first level data collectors) to the Operative Graphics User Interface (GUI), in order to have real-time or near-realtime system checkup and feedback for control assistance. However, these are not defined yet and are out of scope of the Z-BRE4K platform. In fact, these would, at most, be parallel sub-systems developed and owned by the machine owners (end users).

Three types of IDS Connectors have been defined: (i) Provider, (ii) Consumer and (iii) Provider + Consumer. Their rationale, definitions and detailed architecture are addressed in subsequent sections.

Figure 5. Three types of IDS Connectors defined for Z-BRE4K: for Data Consumer (left), Data Provider (middle) and Data Provider + Consumer (right)

The arrows in the diagram represent the expected flow of data, despite the logical information flow and evolution that Z-BRE4K aims at providing from a modelling and support perspective. This means that all entities are virtually able to communicate (exchange data) with any other entity in the system. Limitation to this will be provided by the Context broker definitions itself.

This design has three main justifications:

1. It is adjusted to the effective Z-BRE4K architecture and prescribed functionalities for the current 3 use cases under study and implementation;
2. It is scalable to any structure size;
3. It is easy to upgrade and adapt in the future (should there be any need for that).

3.2 Input/Output (I/O)

The several components of the Z-BRE4K architecture have different functions within the system and treat data and information in different ways and with different objectives. Hence, the quantity, type, frequency, uniqueness, organization and format of the data being exchanged in different segments of the connection routes within the Z-BRE4K platform may differ significantly. Also, its relevance and criticality to each and every component is different, at different stages of its processing. So, the **heterogeneity of the data** content is a characteristic that Z-BRE4K needs to account with. One of the aims of using Fiware-compliant IDS Connectors is precisely to allow for the possibility of **adapting data**, while sending (providing) or receiving (consuming), to a common format that is well understood by all components that communicate it, despite different amounts, configurations and even sub-formats of such data. The common ground is the **NGSI standard** (a modified JSON) which allows the packing of very different types of data (single values, arrays, multi-indexed, alpha-numeric, etc) within a simple file organization scheme, accounting for additional simple meta-data to be associated with. In sort of saying, it packs heterogenous types of data into similar packages that can follow same routes and queues at the several brokers they pass by.

At this moment (end of T2.2) only the **raw data from the shop floor level** (outputs from machines and/or associated devices) are well known and their push/pull mechanisms into the Z-BRE4K environment are drawn. All other components at the cloud level are not yet fully defined, and therefore only vague definitions of data requirements (organization of data, push/pull definitions, data aggregation, truncation and pre-processing, etc) exist. Nevertheless,

all shall be exchanged in NGSI format, and thus IDS Connectors are already conceptualized and designed to attain such scope.

At the shop floor level, the three use cases will provide data in **different formats** and with quite **different levels of complexity**. The machine data collecting systems (components C02 + C09 + C10 for the GESTAMP use case, Philips DataBase for the PHILIPS use case and C01a for the SACMI use case), will collect and pre-process these data, in compliance with the data owners permissions and then provide them into Z-BRE4K through the **first level IDS Connectors**.

The specific sources and contents of these data to retrieve directly from the production unit cells by dedicated components developed by Z-BRE4K technical partners are known but will not be released in detail herewith, due to confidentiality and proprietary reasons. Nevertheless, one may describe it in general terms:

- Data will be gathered from sensors installed in the machines, that allow for precise and accurate assessment of pre-determined machine parameters;
- Data will be gathered from machine integrated CPU or PLC, that allow for history tracking of the main operation state of the equipment itself;
- Data will be gathered from operators input, allowing for tracking main maintenance operations, conditions, timings and results.

Real time data streams, CSV, TXT, XML and JSON files will be collected. These comprise different “sizes”, collection rates, internal organization of data and pre-processed conditions. Table 3 indicates the input/output types of data as well as the expected data flows for all the Z-BRE4K components.

Table 3. Inputs and outputs for each component of the Z-BRE4K architecture

		INPUTS			IDS CONNECTOR			OUTPUTS			IDS CONNECTOR	
COMPONENT		**INPUT FROM WHERE**	**INPUT TYPE**	**REAL TIME**	**DEDICATED IDS CONNECTOR**	**DATA APP**		**OUTPUT TO WHERE**	**OUTPUT TYPE**	**REAL TIME**	**DEDICATED IDS CONNECTOR**	**SYSTEM ADAPTER**
M1 GESTAMP		N/A	N/A	N/A	N	N/A		C02	N/A	Y	Y – Provider	N/A
M2 PHILIPS		N/A	N/A	N/A	N	N/A		PHILIPS DATABASE	N/A	Y	N	N/A (proprietary)
PHILIPS DATABASE		M2 PHILIPS	TXT + JSON FILES	Y	N	N/A (proprietary)		C01b; (C03?); C04; C05; C06; C07; C08; C11;	NGSI FILES		Y – Provider	Converts ONE type of TXT file and 7 types of JSON files into the NGSI single format
M3 SACMI		N/A	N/A	N/A	N	N/A		C01a	JSON + TXT	Y	N	N/A (iLike embedded)
C01a		M3 SACMI	JSON + TXT	Y	N (iLike embedded)	N/A (iLike embedded)		C01b; C03; C04; C05; C06; C07; C08; C11;	NGSI	Y	Y – Provider	Converts json/txt into NGSI format
C01b		C02; PHILIPS DATABASE; C01a;	NGSI	Y/N	Y – Consumer	Storage		C08; C07; C11; (C04?)	NGSI	N	Y - Provider	Enables data stored in C01b to be transferred in NGSI format
C02		M1 GESTAMP	video & real time data streams, TXT, XML, FAGOR press specific	Y	Y - Consumer	N/A		C01b; (C03?); C04; C05; C06; C07; C08; C11;	NGSI	Y	Y - Provider	Conversion into NGSI format
C03		N/A	N/A	N/A	N (C03 is added on C01b)	N (C03 is added on C01b)		C01	N/A		N (C03 is added on C01b)	N (C03 is added on C01b)
C04		Users;C05;C03	JSON	Y	Y – Consumer	Internal storage			JSON/NGSI		Y – Provider	

		INPUTS			IDS CONNECTOR			OUTPUTS			IDS CONNECTOR	
COMPONENT		**INPUT FROM WHERE**	**INPUT TYPE**	**REAL TIME**	**DEDICATED IDS CONNECTOR**	**DATA APP**		**OUTPUT TO WHERE**	**OUTPUT TYPE**	**REAL TIME**	**DEDICATED IDS CONNECTOR**	**SYSTEM ADAPTER**
C05		Users; C01; C02; C08	JSON		Y – Consumer	Internal storage			JSON/NGSI		Y – Provider	
C06					Y – Consumer							
C07		C02; Philips Database; C09+C10; C01a;	CSV, JSON, Reports	Y	Y – Consumer	Storage		C04; C06	SIGNAL	Y	Y – Provider	N/A
C08		C02; Philips Database; C09+C10; C01a;	CSV, JSON, Reports	Y	Y – Consumer	Storage		C07; C06	SIGNAL	Y	Y – Provider	N/A
C09		Gestamp product	Physical object	N/A	N	N/A		C10	TXT FILES	Y	N	N/A
C10		TRIMEK CMM C09	Pointclouds – TXT	Y/N	N	Processing		C01b; (C03?); C11; C12; C10	NGSI	N	Y – Provider and consumer	Enable of QIF standard files (XML) to be transferred in NGSI format
C11					Y - Consumer						N/A (proprietary)	
C12		ALL COMPNENTS	ALL TYPES	N	N/A	N/A		ALL COMPNENTS	ALL TYPES	N	N/A	N/A

One may see that the data formats that the production machines and/or their sensors installed output are well-known.

The challenge is, therefore, to ensure that **all data is exchanged into the Z-BRE4K environment in a format compatible with the Fiware context** (since modules may be used now and/or in the future to add/replace functionalities (see section 4)) **and NGSI standard** (since this was pre-selected for Z-BRE4K). For this it is proposed the implementation of IDS Connectors that must contain the necessary **modules (functions) for the conversion/transformation** of the data received into the required compatible format and forward it to other entities (consumers) that must also be able to receive and handle information under same context and compliances.

The first conversion of these will actually occur at the first IDS Connector they pass by. **All files and data streams will be converted into NGSI compliant files** in the IDS Connectors with sole Provider (P) profile at the Edge Level. This conversion will be processed at the **System Adapters**, which are specific functional modules within the IDS Connectors. These will be described in detail in the next section.

Several FIWARE available modules were identified that can be used and implemented according to the type of requirements of each entity. The **Orion Context Broker** is the selected Broker entity to handle the management of all communication between entities. The **Cygnus** app is selected for the handling of received data as well as partial/full storage of data into predefined databases and locations.

The identification of the destination of the data being outputted from each component (at whatever level of the platform) will allow the specific Context Broker to be defined/programmed accordingly. This will also be further explained in the next section.

In section 4 a specific Z-BRE4K-dedicated architecture for each of these IDS Connectors is designed and described in detail, as well as several FIWARE modules (openly available) that can be used in its implementation are identified.

4 IDS CONNECTOR DETAILED/SPECIFIC ARCHITECTURE

IDS Connector facilitates the exchange of data between participants and is basically a dedicated communication server for sending and receiving data in compliance with the IDS and FIWARE specifications.

The IDS Connector must receive data from an enterprise backend system, either through a **push mechanism or a pull mechanism**. The data can be provided via an interface or pushed directly to other participants. The Connector must be able to receive information, for example directly from the shop floor, to process such data and send it in such a way as to be accepted in a Fiware environment.

In the most generic possible architecture, **each IDS Connector embeds three levels of functionalities within it**. A level of functionalities related to the modules where it is able to either pre- or post-process data. These are the so-called **System Adapters or Data Apps**, respectively. Another level of functionalities deals with all ID recognition/certification, history logging, redirection and handling of data. These integrate a **Data Router**. Finally, a third level of functionalities are the Broker itself, which communicates only with its counter-parts placed in the IDS Connectors of the other end point(s) of the communication segment(s). This is the **Context Broker**.

Two elementary architectures for the IDS Connectors are, then, immediately sought:

- **IDS Connector for Data Providers**:

These are expected to include data collection mechanisms – **System Adapters** – capable of collecting data from different types of sources and in different formats and complexity and then converting (or just re-adjusting) it into NGSI file format. The push or pull mechanisms will depend on the prescribed relationships with the data sources (upstream), the remaining modules of the IDS Connector (e.g. the Context Broker). Through either publishing or subscription relationships, these capabilities are enforced into the intrinsic behaviour of the System Adapters (and the Connector itself). A **Data Router** module shall include functions related to security (ID verification and/or attributes), routing (path and destination selection), registry (history logs for operations and exchanges) and management (e.g., any additional splitting or stop-over for storage or certification elsewhere).

Finally, a **Context Broker** is required to link directly to the other end point. It is only responsible for the "traffic" management, thus managing the message queue through associating each file ID (or specific meta-data) with specific prescribed subscription or publishing rule.

- **IDS Connector for Data Consumers**:

Being at the other endpoint of the segment, one may see these as "symmetric" connectors (compared to the Providers ones), also in terms of their architecture. Data flow occurs in the opposite way, throughout the three levels of functionalities.

A **Context Broker** acts as the first entry point, since it must be directly linked to its counter-part in the Data sender (provider) side). It is only responsible for the "traffic" management, thus managing the message queue through associating each file ID (or specific meta-data) with specific prescribed subscription or publishing rule.

A **Data Router** module shall include functions related to security (ID verification and/or attributes), routing (path and destination selection), registry (history logs for operations and exchanges) and management (e.g., any additional splitting or stop-over for storage or certification elsewhere).

Finally, data processing mechanisms – **Data Apps** – including their own mechanisms for local/remote storing received data in a database, converting it for specific required formats or file batches or just forwarding them directly for data analysis or data processing applications/services. These shall be able to introduce the received data into other major components who need it and justify the dedicated IDS Connector. Again, the push or pull mechanisms will depend on the prescribed relationships with the data receivers (downstream), the remaining modules of the IDS Connector (e.g. the Context Broker). Through either publishing or subscription relationships, these capabilities are enforced into the intrinsic behaviour of the Data Apps (and the Connector itself).

Any of these functionalities (at each of the three levels mentioned) may be unique or shared among several IDS Connectors within the network. This mostly depends on the positioning of the IDS Connector within the ecosystem. If at first entry levels, conversion and security become mandatory, whereas at intermediate levels of communication within the network, such functionalities may not be necessary.

Figure 6 depicts these two generic architectures and their relation for an end-to-end data flow.

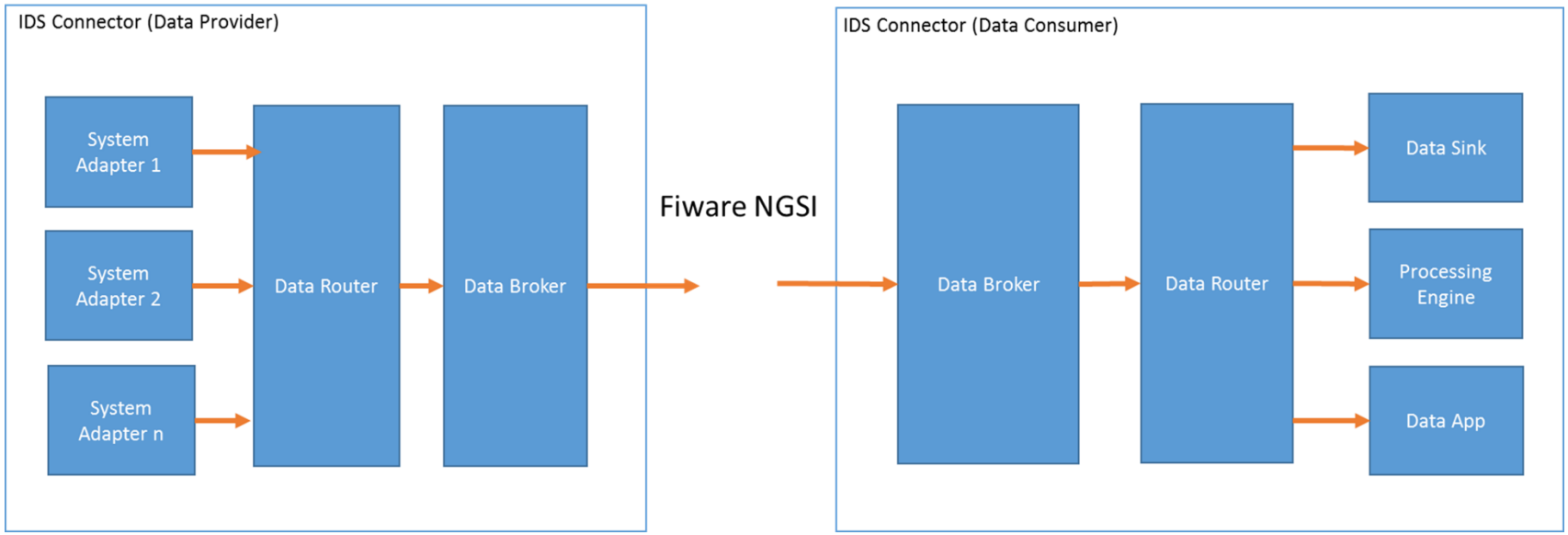

Figure 6 – End-to-end data communication flow between the two generic architectures for IDS Connectors

Proposed Fiware components for implementing the IDS Connector

In order to implement the abovementioned three levels functionalities within the IDS Connectors (using the referred architecture), one has several modules available from FIWARE.org that may be used (Fig. 7). Herewith a selection of specific modules is proposed for generic implementations of this IDS Connectors architecture.

Each module has its own purposes as briefly described (and justified) here below:

- **Publish/Subscribe Context Broker** - Orion Context Broker is an implementation of the Publish/Subscribe Context Broker GE, providing the NGSI9 and NGSI10 interfaces.
- **Cygnus** - implements a connector for context data coming from Orion Context Broker and aimed to be stored in a specific persistent storage, such as HDFS, CKAN or MySQL.
- **Short Time History -** in charge of managing (storing and retrieving) historical raw and aggregated time series information about the evolution in time of context data (i.e., entity attribute values) registered in an Orion Context Broker instance**.**
- **Backend Device Management (IDAS)**
- **PEP Proxy Wilma -** Thanks to this component and together with Identity Management and Authorization PDP GEs, you will add authentication and authorization security to your backend applications. Thus, only FIWARE users will be able to access your GEs or REST services. But you will be able also to manage specific permissions and policies to your resources allowing different access levels to your users. It is thought to work with OAuth2 and XACML protocols, the standards for authentication and authorization chosen in FIWARE.
- **IDAS OPC-UA Agent -** is a component to connect in bidirectional way, the IoT Devices which implements the OPC-UAstandard connection technology, with a NGSI Publish/Subscribe Context Broker as Fiware Orion.
- **Shopfloor Data Collection (SFDC) -** is a collection of modules i.e. Fitman Tag &Trace (FitmanT&T) and Fitman Sensor Networks (FitmanSN) -which can be deployed and used independently; for data acquisition from the shop floor making use of smart objects, in the scope of Internet of Things-oriented Manufacturing Ecosystems.

However, under certain assumptions and controlled definitions, the IDS Connectors may be even simplified, for sake of more efficient implementation and, more importantly, greater real-time data exchange requisites compliance for most use cases. Hereafter the dedicated architecture designed for the specificities of Z-BRE4K is presented.

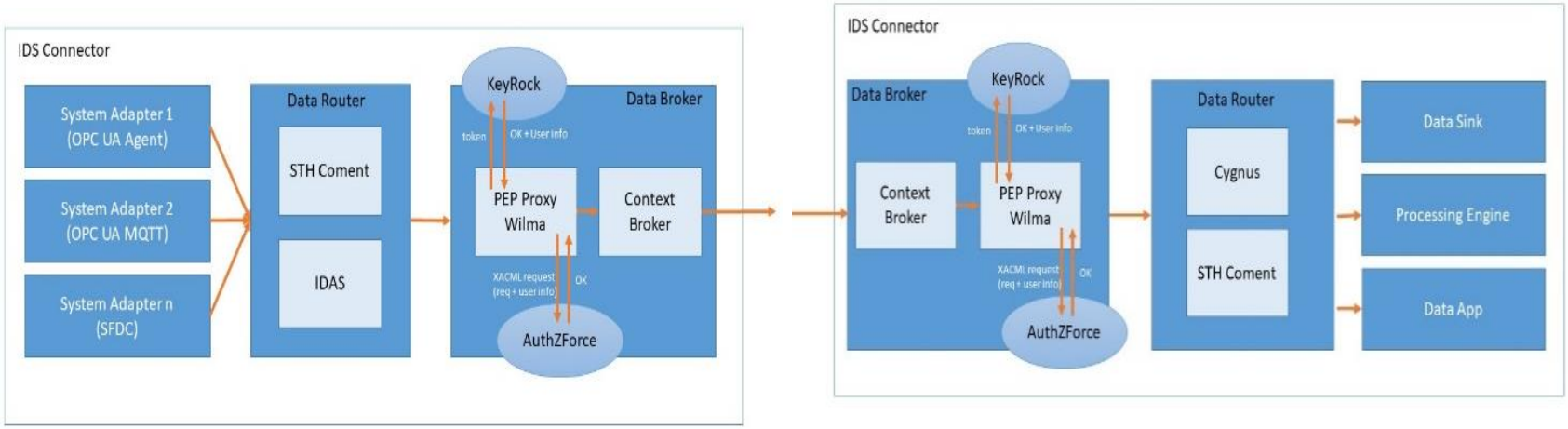

Figure 7. Proposed FIWARE modules for each relevant functionality within the two elementary architectures for the dedicated IDS Connectors: Provider (left) and Consumer (right)

Dedicated architecture for Z-BRE4K specific needs and use cases

The Z-BRE4K context is controlled and limited to specific use cases that are designed, prescribed and prepared each time. This means that one may **specify and tailor IDS Connectors for each well-known location** within the system. Upon full definition of the overall system architecture for Z-BRE4K (D1.3), this specification is straightforward. Moreover, several partners of the Z-BRE4K consortium have know-how and heritage in using specific languages, FIWARE compliant modules and standards, which further facilitate the common understanding.

For this specific context, **three types of IDS Connectors** have been identified with the same overall purpose as described above in this section and depicted in Figure 4 and Figure 5:

- IDS Connector for Data Providers **(IDScon P)**;
- IDS Connector for Data Consumers **(IDScon C)**;
- IDS Connector for Data Providers + Consumenrs **(IDScon P+C)**.

The P+C Connector brings together both the functionalities of the Provider and Consumer sides, namely through embedding **System Adapters** (provider side) and **Data Apps** (Consumer side) simultaneously. The input/output communication may occur simultaneously or not, but the architecture allows for any scenario.

All the three types of dedicated IDS Connectors contain a **Data Broker** module that is responsible for the management of all communication verified in the system. Data Broker inserts itself in the FIWARE context with the **Orion Context Broker** Components to receive and send information in **NGSI**. It is an implementation of the Publish / Subscribe Context Broker GE, providing the NGSI9 and NGSI10 interfaces, and the **PEP Proxy Wilma** component for authentication and authorization of all requests made through the IDS Connector.

However, in these dedicated architectures, the **Data Router level is dismissed**. This is due to the fact that the data ID, certification, temporary storage and destination are well known and are provided by other components of the Z-BRE4K system. Thus, the **Z-BRE4K IDS Connectors architecture becomes simpler and more efficient**.

The **IDS Connector Provider (IDScon P)**, represented in Figure 8, is responsible for sending compatible data sets into the Z-BRE4K environment (in this case also compatible with FIWARE environment). All data streams and/or files inputs are converted into NGSI compliant files that all corresponding consumers are able to read. It the Data Broker module for the management of all communication under a prescribed publishing/subscription methodology linking to any other system component. The IDScon P will be implemented with the System Adapter module that is responsible for receiving and handling the raw data received from the Data Provider (e.g. the shopfloor sensors and/or devices or the data collectors implemented at the edge level) and translating or converting them into the NGSI format that is compatible with the Orion Context Broker and thus the Z-BRE4K context itself.

In Figure 8, the **IDAS module** is used as a general proposal for **System Adapters at the shopfloor**. It is an implementation of the Backend Device Management GE, according to the FIWARE

reference architecture. It allows to connect **IoT devices / gateways** to FIWARE-based ecosystems. IoT Agents translate IoT-specific protocols into the NGSI context information protocol, which is the FIWARE standard data exchange model. In short, IoT Agent converts the physical devices to an Orion Context Broker entity, and in the same way, through IoT Agent the execution or writing of operations is possible. The IDAS module is not needed if the shopfloor devices or gateways natively support the NGSI API, or if other dedicated sub-systems will take care of such functionality. This is the case of GESTAMP and SACMI use cases, in which dedicated SW/HW interfaces will be developed by Z-BRE4K technical partners, thus replacing the need for implementing the System Adapters through IDAS scheme. The System Adapters, in these cases, will be designed, developed and linked directly into the production unit cells, and they will provide the entry point to the Z-BRE4K environment via first level IDS Connectors P. In the PHILIPS use case, no IDAS modules will also be needed, due to the intermediation of a proprietary database.

IDScon P can be deployed with **multiple System Adapters** for different types of information, data type or different systems, as long as they can handle the information received and transform it into an **Orion Context Broker compatible format** so that it can be sent to other entities. Also note that systems that already provide the information in NGSI, format compatible with FIWARE, may not need to use any System Adapter and the information goes directly to the Orion Context Broker to be sent to another entity.

The Z-BRE4K dedicated System Adapters may be observed as simple “converters” from the native formats into NGSI files. Therefore, they will be developed for each specific use case, and are the unique feature that is entirely tailored to each application. These will be described in section 5.

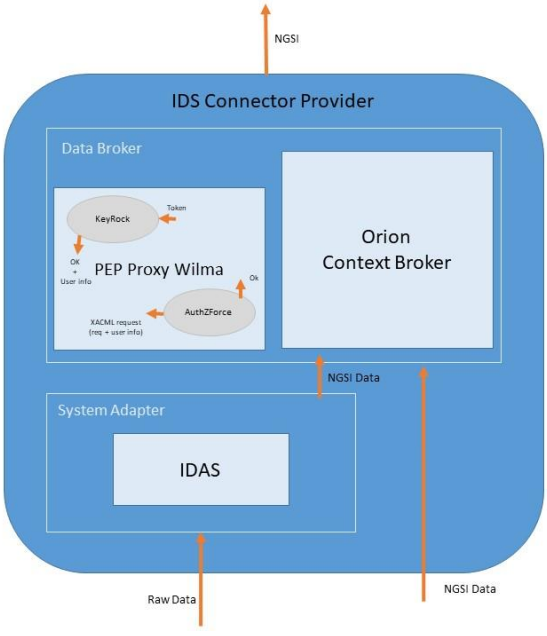

Figure 8. Dedicated architecture of the IDS Connector type Provider (P) for the Z-BRE4K environment

The **IDS Connector Consumer (IDScon C)** receives the data through the **Orion Context Broker**, already in **NGSI** format respecting the FIWARE environment, and returns that information, with no translation or modification whatsoever, to the **Data App** module present in IDScon C. The Data App is then able to deal with registry, storage or post-processing if needed into the Data Consumer component associated with this connector. In Figure 9, the FIWARE module called **Cygnus** is selected to implement the connector Data App for context data coming from Orion Context Broker and intended to be stored in a specific persistent storage, such as HDFS, CKAN or MySQL. In this example one has the possibility, to save the information received in NGSI in an external database. However, it is likely that the IDScon C may contain more than one Data App developed only to handle the information received and forwards it to another place as an external application or an API. This will, again, depend on each specific component need across the Z-BRE4K platform.

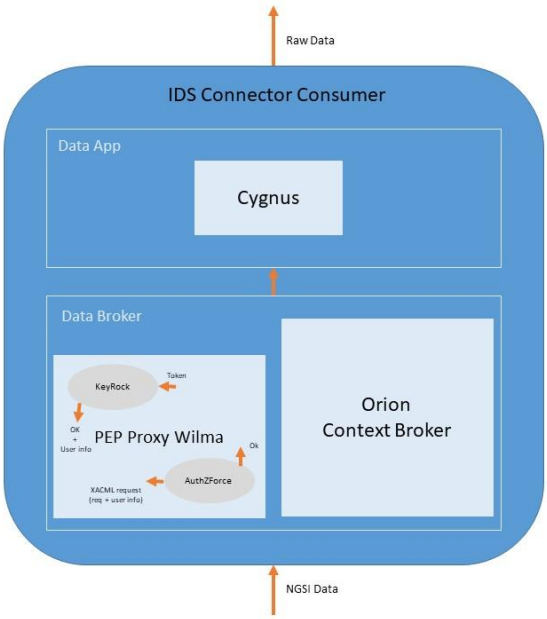

Figure 9. Dedicated architecture of the IDS Connector type Consumer (C) for the Z-BRE4K environment

Finally, one has the **IDS Connector Provider + Consumer (IDScon P+C)** for entities that require bidirectional communication and act as both Data Provider (sender) and Data Consumer (receiver). This Connector is a combination of the previous two types of connectors (Providers – P, and Consumers, C) containing the **Data Broker** module such as the **Data App** and the **System Adapter** modules. As said before, these may operate simultaneously or intercalated, according to the specifications and needs of the associated component. The management of the publishing and/or subscriptions to other Connectors is fully defined by the Orion Context Broker setup.

This Connector follows the same features and functionalities as the "one-directional" Connectors and is able to **collect data from the associated component** (e.g. Z-BRE4K components at the cloud level), convert it into a format compatible with the Orion Context

Broker and **send it to another entity** in the Fiware/Z-BRE4K context. Similarly, it is able to **receive data** from other Z-BRE4K entity, treat it and **save or post-process it** through the Data App in a database or external applications.

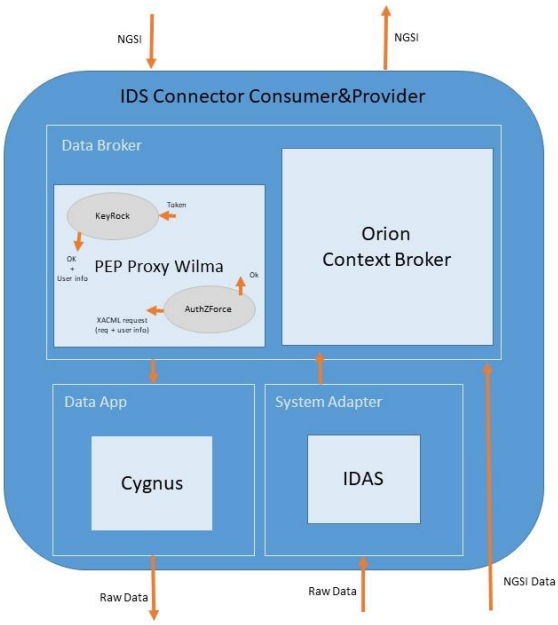

Figure 10. Dedicated architecture of the IDS Connector type Provider + Consumer (P+C) for the Z-BRE4K environment

The FIWARE Components (here called modules) presented and described herein may be replaced by others provided they meet the same requirements and functions. In fact, several modules will be designed, developed and implemented with tailored sub-architectures and purposes, truly dedicated to the current use cases. These will be explained in the next section.

5 SYSTEM ADAPTERS & DATA APPS

The System Adapters and Data Apps are the features that may be unique for each IDS Connectors within Z-BRE4K. They deal with data conversion, pre- or post-processing and push/pull from/into the corresponding Context Broker. They are unique interfaces with the specific components of the overall architecture.

The System Adapters and Data Apps for the Cloud Level Z-BRE4K Components are not yet fully defined, as these depend on the detailed specifications of each of those components, regarding WHAT information is exchanged, WHAT results are expected from their processing, WHEN are such results outputted and WHERE to are they being sent. Although a general information flow is already agreed, the detailed specifications on the data formats, priorities and associated information are not yet closed. Therefore, no Data Apps will be fully described herewith.

However, things are already well defined at the shopfloor and thus the Data Collection mechanisms and the first level outputs are already studied and defined. System adapters for the three IDS Connectors associated with GESTAMP, PHILIPS and SACMI machines (presses) are generally defined and are described hereafter.

5.1 GESTAMP

In the GESTAMP use case two sources of data exist at the shopfloor level. One source is the Welding Robotic Cell itself (the production machine) and another source is the Coordinate Measuring Machine (CMM) which will measure parts manufactured by the former (physical quality control). Both equipment will provide data in a non-synchronized manner. Data collected directly at the machine sensors, computer systems and operator inputs, will be collected by Z-BRE4K Component C02, developed by AIMEN. Therefore, the corresponding IDS Connector (type Provider) will be coupled to the C02 output. In addition, “offline” information from the CMM will be transferred into the same IDS Connector. The CMM-related data will be collected, pre-processed and sent to the Z-BRE4K environment by a TRIMEK’s proprietary Components C09 and C10 (M3 Gage and M3 SW, respectively). A dedicated System Adapter will, then, be developed and embedded into C09 and C10 components, thus linking directlty into the context broker. In brief, the IDS Connector type P associated with GESTAMP shopfloor data will include at least two System Adapters, linking to same Data Broker.

The full descriptions of these Components (C02, C09 and C10) that collect and prepare data from the Shop Floor level in GESTAMP use case are to be provided elsewhere. In fact, both the sensors system at the welding machine and the M3 Gage system are expected to evolve during the Z-BRE4K project, namely due to addition of new sensors, for which no closed definition of data exists. For the matter, we only need to focus on the type, quantity and quality of data that these bring into the IDS connector.

The data coming from the welding cell, in real time, includes:

- video streams,
- other real time data streams,

- TXT files,
- XML files,
- FAGOR specific data.

These will be collected by Component C02, and the System Adapter will be embedded into it. The System Adapter will convert these data into NGSI files, grouped in accordance with higher-level specifications from the main Z-BRE4K data processing and knowledge extraction components.

Considering the metrological data, coming from CMM, TRIMEK has built a custom agent (system adapter) for the M3 software that can easily connect to NGSI Orion Context Broker to be used in Z-BRE4K project.

The idea is to deploy and configure the CMM FIWARE System Adapter in the IDS Connector. Thus, CMM produced part measurements are sent to the product quality control management platform in order to be able to analyse the performance of the CMM, ensuring zero break downs. Likewise, the IDS-ready CMM system could also send product quality measurements to the Gestamp predictive maintenance service.

The use of the IDS connector at the factory will bring the necessary warranty to TRIMEK and Innovalia that only the measurements which have been approved can be delivered to the Gestamp welding robotic cell for predictive maintenance.

A complementary dashboard application will alert the shop floor operator about quality detected issues and suggests recommendations for the production adjustment and maintenance of the machine. This may be valid for Edge-to-ShopFloor Communication as well as for Cloud-to-Cloud and Cloud-to-ShopFloor communication. Hence, several routes may be associated to the first level System Adapter for this use case.

As one may see in Figure 4, the data from the CMM measurements (C09 + C10) are expected to incorporate the same Connector (but through a separated System Adapter) as the data from the welding machine. This will require additional effort to embed these files into the basic functional definitions of it. Synchronization and/or correlation will not be attained at this level, since other components will deal with such relationships. However, considerable differences in terms of data standards or formats are expected, since there are specific QIF-based definitions for the CMM outputs.

The outputs from the CMM unit (outputted from Components C09 and C10) are set for each measurement (each inspected part) and are the following:

- Point cloud TXT file;
- Measurement results XML file (QIF-based);
- Deviation Map STL + Annotated file;
- Customized Reports PDF files.

As becomes evident, several different formats will flow into the System Adapter, which shall be converted into NGSI file formats, in order to be forwarded to the true Z-BRE4K environment. Strict collaboration between AIMEN and TRIMEK will allow to deliver the expected result on a comprehensive System Adapter for this specific use case. Proper definition of data requisites from all other Z-BRE4K components will be determinant for the success and efficiency of this solution.

The system adapter currently deployed by TRIMEK allows that the GD&T results (data at rest) can be accessed and offered to data apps. It has been proposed the use of QIF standards to leverage metrology information interoperability using industry consensus standards to communicate data between component producers and consumers of manufacturing quality systems. For this reason, TRIMEK is integrating and developing QIF schema as the data model of the GD&T results. An XML data file conforming to an XML schema is called an instance file. Every instance file conforming to the QIF model will have QIFDocument as the root of a hierarchy of information.

This strategy fits the purpose of a complementary route for data processing and flow at the shopfloor and edge levels. However, it is not fully compliant with the remaining Z-BRE4K components which require NGSI files. Effort will be made to add this conversion at the System Adapter level, so that the Broker may decide how and where to split those.

5.2 PHILIPS

The PHILIPS use case has a unique configuration, since the Data Owner provides the primary collection of data at the shop floor level, placing it in a proprietary database (Philips DataBase). Eight different sources of data exist in the production machine (press), and these will be stored in the mentioned database:

- Data from sensors (TXT + JSON files);
- Machine status data (CSV or JSON files);
- Maintenance logs (CSV or JSON);

The machine status data has time related information, thus showing any shutdown circumstance for the record.

Currently all these data are collected and stored in files or oracle databases, from where the Z-BRE4K dedicated System Adapter will collect those. Files are made available in batches (non-real time). Then (still during the Z-BRE4K project) it is expected that the system evolves up to providing all data (except for 1 of the 8 sources) directly as a URL with 5 minutes update. However, all sources may return values in a non-synchronized manner.

Therefore, different files need to be generated and converted into NGSI files. Also, since the Knowledge Extraction and Machine Learning algorithms require huge amount of data, one is expected to transfer all data sets with no truncation into the Z-BRE4K environment.

The suggested design is to convert each file independently of the number of files and each one's size. Then, data combination at the database level or at the System Adapter level, may be implemented, but the flexibility of this architecture is friendly to such evolution.

The IDS Connector Provider (IDScon P) will create on ORION context broker entities corresponding to the machines according to the NGSI standard. Sensors and maintenance related information will be created as "attributes" as can be seen in the following example:


```
{
  "contextElements": [
    {
      "type": "MFL1_MP26",
      "isPattern": "false",
      "id": "Machine3",
      "attributes": [
        {
          "name": "Angle",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75","96.00","96.25"]"
        },
{
          "name": "RAM_AE",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75","96.00","96.25"]"
        },
{
          "name": "G0_AE",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75","96.00","96.25"]"
        },
{
          "name": "G2_AE",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75","96.00","96.25"]"
        },
{
          "name": "G3_AE",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75","96.00", null]"
        },
{
          "name": "G4_AE",
          "type": "ArrayList",
          "value": "["95.00",null,"95.50","95.75","96.00", null]"
        },
{
          "name": "G7_AE",
          "type": "ArrayList",
          "value": "["95.00","95.25","95.50","95.75", null, null]"
```

```
            }
          ]
        }
    ],
    "updateAction": "APPEND"
}
```

"APPEND" allows to create new context elements or to update the existing ones.

5.3 SACMI

In the SACMI use case, a press will be monitored by means of installed sensors (several already installed and a few yet to be possibly installed during the Z-BRE4K project), whose signals will be gathered by an integrated PLC.

These data will be exchanged into the Data Collection component (C01a) developed by HOLONIX in JSON format.

In addition, data from operators with maintenance related information (when, what) will be collected as well, through a dedicated user-friendly interface to be developed, in JSON and TXT formats.

Therefore, a similar approach to the PHILIPS use case will be conducted, in order to convert these files into compliant NGSI files.

The next figure shows the current scheme for the data collection and pre-processing for the SACMI use case, in which the System Adapter is embedded. Its representation will, nevertheless, evolve to homogenize the conceptual representation of the IDS Connectors coupled into specific components (as is the case in all three use cases).

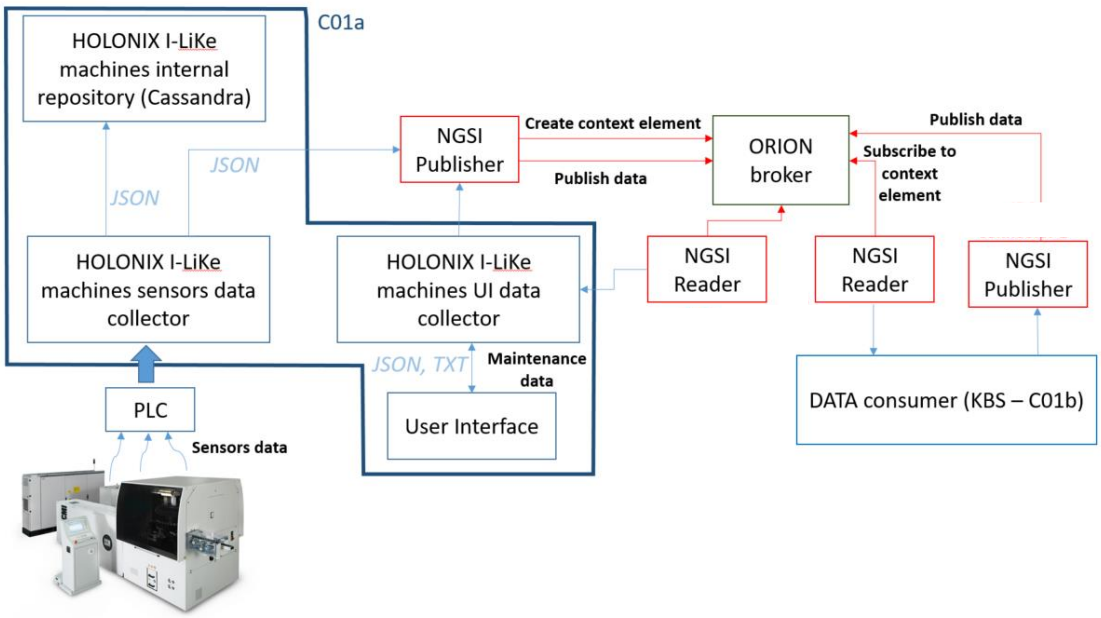

Figure 11. SACMI use case data collection by C01a, developed by HOLONIX. Integration with IDS Connector is provided by the System Adapters (herein called NGSI reader/publisher)

The System Adapter (NGSI Publisher) will create on ORION Context Broker entities corresponding to the machines according to the NGSI standard. Sensors and Maintenance related information will be created as "attributes" as can be seen in the following example:

```
{
  "contextElements": [
    {
      "type": "SACMI_Machine",
      "isPattern": "false",
      "id": "Machine1",
      "attributes": [
        {
          "name": "polymer_temperature_EX",
          "type": "float",
          "value": "125"
        },
        {
          "name": "motor_speed_EX",
          "type": "integer",
          "value": "720"
        }
      ]
    }
  ],
  "updateAction": "APPEND"
}
```

"APPEND" allows to create new context elements or to update the existing ones.

If/whenever an IDS Connector Provider + Consumer (P+C) is required for this interface (when feedback information will be able to be visualized at the machine operation level), the following Data App will be put in place. This implies that the dedicated connector to SACMI use case may be upgraded to P+C architecture.

The Data App (NGSI Reader) will subscribe to context elements to acquire data and make them available for data consumers (KBS or user interface) with an NGSI10 query Context request having the following structure:

```
{
  "entities": [
    {
      "type": "SACMI_Machine",
      "isPattern": "false",
      "id": "Machine1"
```

```
      }
   ]
}
```

The response will include all the attributes belonging to the Machine1 (SACMI) if not differently specified (as shown in the following example):

```
{
   "entities": [
      {
         "type": "SACMI_Machine",
         "isPattern": "false",
         "id": "Machine1"
      }
   ],
   "attributes": [
      "polymer_temperature_EX"
   ]
}
```

6 COLLECTIVE INTELLIGENCE PREDICTIVE MAINTENANCE

The Z-BRE4K goal is to provide improved knowledge and advice to manufacturing industries that face maintenance related challenges to keep their production assets at optimum conditions and performance. For this, large quantities of data, with heterogenous sources, formats, sizes, timings and relevances are collected from a number of shop floor production cells. Then, these data are interpreted and processed across several Z-BRE4K components allowing for retrieving effective knowledge on the behaviour, trends and expectations on those production cells. New knowledge is generated and decision support advices are returned back to the production managers/engineers/operators.

Such a complex system requires trust, intensity, accuracy and relevance of data. Moreover, with such heterogenous types of sources, locations, formats and de-synchronization there is need for reliable, trustful and efficient gateways.

The Z-BRE4K dedicated IDS Connectors, incorporating the context brokers within, grant this role to the platform. Despite these don't treat the data nor take technical decisions over them, IDS Connectors act as the reliable file conversion and reading mechanisms in which the overall system relies on.

The end-to-end communication exclusivity ensure expandability, upgrades and modular architecture for the overall network, thus attributing resilience and confidence to it. Although these connectors could be replaced by many other alternative approaches (e.g., independent bilateral connections between pairs of components), they represent the best available approach to get different players, participants and devices to effectively work together towards a common objective (in this case a collective intelligence for improved predictive maintenance knowledge).

7 CONCLUSION

A set of definitions and concepts have been studied and tailored to the specific needs of Z-Bre4k. Dedicated IDS connectors, with simplified architectures, compliant with Fiware environment and providing NGSI-type files for all relevant communications within the platform were designed and described in detail.

Further developments shall be provided at a later stage of the project in what concerns the "upper" levels of the platform (the modelling and knowledge extraction components), since these are yet to be fully defined and thus their data and communication requirements as well.

At the shopfloor level, the types of data to be made available are set and understood, and therefore a well-defined structure for the data flow as well as for the required System Adapters at the first entry-level into the Z-Bre4k platform have been designed and specified.

In summary, the IDS Connectors architecture designed for the Z-BRE4K platform set a simple way of connecting the multiple nodes of the complex network, homogenizing and agilizing all communications within the system. Heterogeneous data gets uniformized in terms of ontology, standards, formats and identifiers, then gets redirected to the correct endpoints through push/pull mechanisms prescribed in its fundamental definition and setup.