MATRYCS

D2.3 | MATRYCS Reference Architecture for Buildings Data v1.0

WP2 – System Requirements and Specifications

July 2021

www.matrycs.eu

Modular Big Data Applications for Holistic Energy Services in Buildings

MATRYCS

Disclaimer

The sole responsibility for the content of this publication lies with the authors. It does not necessarily reflect the opinion of the European Union. Neither the EASME nor the European Commission is responsible for any use that may be made of the information contained therein.

Copyright Message

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Grant Agreement Number	101000158	Acronym	MATRYCS
Full Title	Modular Big Data Applications for Holistic Energy Services in Buildings		
Topic	LC-SC3-B4E-6-2020 | Big data for buildings		
Funding scheme	H2020- IA: Innovation Action		
Start Date	October 2020	Duration	36
Project URL	www.matrycs.eu		
Project Coordinator	ENG		
Deliverable	D2.3 – MATRYCS Reference Architecture for Building Data v1.0		
Work Package	WP2 – System Requirements and Specifications		
Delivery Month (DoA)	July 2021	Version	1.0
Actual Delivery Date	06/08/2021		
Nature	Report	Dissemination Level	Public
Lead Beneficiary	RWTH Aachen University		
Authors	Marco Pau [RWTH], Zhiyu Pan [RWTH], Stefan Lankes [RWTH], Dario Pellegrino [ENG], Pasquale Andriani [ENG], Leandro Lombardo [ENG], Francesco Saverio Nucci [ENG], Panagiotis Kapsalis [NTUA], Vaggelis Marinakis [NTUA], Zoi Mylona [HOLISTIC], Daniele Antonucci [EURAC], Miha Smolnikar [COMSENSUS]		
Quality Reviewer(s):	Marcelo Lampkowski [ICLEI], Panagiotis Kapsalis [NTUA]		
Keywords	Open Reference Architecture, Technical Specifications and Requirements, Big Data Management, Data Governance, Data Processing, Building Services, Big Data Analytics, Semantic Interoperability, Big Data Value Chain, Data Spaces		

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Preface

MATRYCS focuses on addressing emerging challenges in big data management for buildings with an **open holistic solution** for Business to Business platforms, able to give a competitive solution to stakeholders operating in building sector and to open new market opportunities. **MATRYCS Modular Toolbox,** will realise a holistic, state-of-the-art AI-empowered framework for decision-support models, data analytics and visualisations for Digital Building Twins and real-life applications aiming to have significant impact on the building sector and its lifecycle, as it will have the ability to be utilised in a wide range of use cases under different perspectives:

- Monitoring and improvement of the energy performance of buildings - **MATRYCS-PERFORMANCE**
- Design facilitation and development of building infrastructure - **MATRYCS-DESIGN**
- Policy making support and policy impact assessment - **MATRYCS-POLICY**
- De-risking of investments in energy efficiency - **MATRYCS-FUND**

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Who We Are

	Participant Name	Short Name	Country Code	Logo
1	ENGINEERING – INGEGNERIA INFORMATICA SPA	ENG	IT	ENGINEERING
2	NATIONAL TECHNICAL UNIVERSITY OF ATHENS	NTUA	GR	EPU N T U A
3	FUNDACION CARTIF	CARTIF	ES	CARTIF
4	RHEINISCH-WESTFAELISCHE TECHNISCHE HOCHSCHULE AACHEN	RWTH	DE	RWTH AACHEN UNIVERSITY
5	ACCADEMIA EUROPEA DI BOLZANO	EURAC	IT	eurac research
6	HOLISTIC IKE	HOLISTIC	GR	HOLISTIC
7	COMSENSUS, KOMUNIKACIJE IN SENZORIKA, DOO	COMSENSUS	SL	COM SENSUS
8	BLAGOVNO TRGOVINSKI CENTER DD	BTC	SL	BTC
9	PRZEDSIEBIORSTWO ROBOT ELEWACYJNYCHFASADA SP ZOO	FASADA	PL	FASADA
10	MIASTO GDYNIA	GDYNIA	PL	GDYNIA
11	COOPERNICO - COOPERATIVA DE DESENVOLVIMENTO SUSTENTAVEL CRL	COOPERNICO	PT	Coopérnico
12	ASM TERNI SPA	ASM	IT	ASM ASM Terni S.p.A.
13	VEOLIA SERVICIOS LECAM SOCIEDAD ANONIMA UNIPERSONAL	VEOLIA	ES	VEOLIA
14	ICLEI EUROPEAN SECRETARIAT GMBH (ICLEI EUROPASEKRETARIAT GMBH)	ICLEI	DE	ICLEI Local Governments for Sustainability
15	ENTE PUBLICO REGIONAL DE LA ENERGIA DE CASTILLA Y LEON	EREN	ES	EREN | Junta de Castilla y León
16	VIDES INVESTICIJU FONDS SIA	LEIF	LV	
17	COMITE EUROPEEN DE COORDINATION DE L'HABITAT SOCIAL AISBL	HOUSING EUROPE	BE	HOUSING EUROPE
18	SEVEN, THE ENERGY EFFICIENCY CENTER Z.U.	SEVEN	CZ	SEVEn

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Contents

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

The MATRYCS project has received funding from the European Union's Horizon 2020 research
and innovation programme under grant agreement no.101000158

Figures

The MATRYCS project has received funding from the European Union's Horizon 2020 research
and innovation programme under grant agreement no.101000158

Tables

The MATRYCS project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no.101000158

Abbreviation and Acronyms

Acronym	Description
AI	Artificial Intelligence
AIOTI	Alliance for Internet of Things Innovation
API	Application Programme Interface
BDC	Building Data Consumers
BDO	Building Data Owner
BDP	Building Data Provider
BDU	Building Data Users
BDVA	Big Data Value Association
BEMS	Building Energy Management System
BIM	Building Information Model
BSP	Building Service Provider
BVC	Building Value Chain
DaaS	Data as a Service
DAP	Data Analytics Provider
DL	Deep Learning
EPC	Energy Performance Certificate
GDPR	General Data Protection Regulation
GE	Generic Enabler
HLA	High Level Architecture
HPC	High Performance Computing
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology

Acronym	Description
IDS	International Data Space
IDSA	International Data Space Association
IIC	Industrial Internet Consortium
IIRA	Industrial Internet Reference Architecture
IoT	Internet of Things
IPR	Intellectual Property Right
IT	Information Technology
ML	Machine Learning
PaaS	Platform as a Service
PPP	Public-Private Partnership
QoS	Quality of Service
RAF	Reference Architecture Framework
RAM	Reference Architecture Model
RDF	Resource Description Framework
SaaS	Software as a Service
SME	Small-Medium Enterprise
SP	Service Provider
TP	Technology Provider
WG	Working Group

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Executive Summary

The process of digitalization and smartification of the European building sector is leading to the generation of a huge amount of data provided by sensors, smart meters, Internet of Things (IoT) devices and a multitude of other data sources. Exploiting these data and capturing their value through ad hoc analytics services is essential to increase the energy efficiency in the building domain and to move towards the decarbonisation and emission reductions' targets set by the European Commission. Unfortunately, available data are today mostly isolated and fragmented in non-interoperable data silos, which makes them hardly accessible to stakeholders interested in the design and provision of building services. Moreover, building data come from a broad range of heterogeneous data sources and are characterized by different data formats, size, resolution, levels of granularity, veracity, etc., which further hampers the efficient use of data. Overall, this makes it extremely difficult to extract value from these data and to create advanced services that could help to fasten the transition towards smart energy-aware buildings.

Similar to other sectors, the building domain would hence gain huge benefits from the definition of appropriate frameworks and architectures for the management and exchange of building-related (but not only) data. Having a shared and commonly accepted architecture for the deployment of data processing technologies, tools and services would in fact allow fully exploiting the value associated to building data, thus fostering the creation of new business cases and facilitating the introduction of innovation in the building ecosystem. This would clearly represent a key benefit for both the business and public sector, but also for individual citizens and for the society as a whole. To achieve this, therefore, it is of utmost importance to define the architectural solutions that should be used to ensure data interoperability, the deployment of the needed technologies and data analytics tools, and the seamless integration of new services to continuously create business value. Such an architecture has obviously to consider the existing needs and challenges, and should offer solutions to fulfil the specifications and requirements present in the building sector, taking into account the foreseeable use cases and the business interests of all the involved stakeholders.

Given this scenario, one of the main objectives of the MATRYCS project is to define and deploy a Reference Architecture for building data exchange, management and real-time processing. This architecture must provide an open, interoperable, scalable data-driven framework for the management of building-related data and it aims at providing a unifying approach for the deployment, use and access of the desired data analytics tools and services from the Building Value Chain (BVC) stakeholders. This Deliverable will present the first version of the MATRYCS reference architecture, which includes the main concepts, design principles and its functional description. The defined architecture takes into account the multiple efforts already done in other domains for the definition of big data architectural models and considers the specifications and requirements derived from the MATRYCS use cases to create a solution tailored to the specific needs of the building sector.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

1 Introduction

1.1 Purpose of the document

The goal of this Deliverable is to present and discuss the first version of the open Reference Architecture for Building Data conceived within the MATRYCS project. The second and final version of the Reference Architecture will be presented in the following Deliverable D2.4, which is due to M18 (March 2022).

In particular, this Deliverable firstly presents the concepts, principles and requirements taken into account for the definition of the proposed Reference Architecture. To this purpose, a review of already existing works aimed at the definition of reference architectural models for big data management is firstly presented, with the scope of underlining the main aspects and design principles typically considered for the definition of big data architectures. Such aspects and design principles are also considered as input for the definition of the MATRYCS Reference Architecture. In addition, the specifications, technical and non-technical requirements extracted from the MATRYCS use cases have been also considered as input for the MATRYCS Reference Architecture definition. These specifications and requirements were already presented in D2.2 and they are recalled here to highlight how they are fulfilled via the different architecture layers or components. Given the broad scope and the different focus of the Large Scale Pilots in the MATRYCS project, their use cases provide a heterogeneous set of requirements. Considering them has the goal to ensure that the proposed Reference Architecture fits the specific requirements possibly existing in the building domain.

In the second part of the Deliverable, the functional description of the different layers and components of the proposed Reference Architecture are presented. The adopted architecture is composed of three different layers:

- The **MATRYCS governance layer**, which aims mainly at ensuring the collection, curation and semantic interoperability of the available building data.
- The **MATRYCS processing layer**, which focuses on the deployment, training and provision of the Artificial Intelligence (AI)-based modules used to extract business value from the available data.
- The **MATRYCS analytics layer**, which contains the architectural components needed to expose data and building services to the final user.

The Deliverable will provide the overall view of these layers and of the related components, specifying how they interact each other and which technical and non-technical functionalities they must provide. Finally, the Deliverable presents the alignment of the MATRYCS Reference Architecture with the other main reference architectural models initially presented, in order to highlight existing analogies, possible differences or gaps, and to demonstrate how the proposed architecture can be mapped to different architectural views. A preliminary mapping of the MATRYCS Reference Architecture components into the building blocks of a future data space for buildings is finally presented.

1.2 Positioning within the project

This Deliverable is the outcome of the activities carried out until M10 in the Task T2.5 "Big Data, AI and IoT Reference Architecture and Alignment with Existing Frameworks and Architectures", which is the final task of the WP2 "System Requirements and Specifications".

As a matter of fact, Task 2.5 (and consequently this Deliverable) directly or indirectly uses as input the results of all the other tasks of WP2. In particular, the Reference Architecture here presented looks at the specifications and requirements identified in T2.3 ("Analytics Building Services Specifications") and T2.4 ("Regulatory Framework for Data Protection, IPR and Ethical Issues"), which have been presented in Deliverable D2.2. Specifications, technical and non-technical requirements have been extracted from the analysis of the MATRYCS use cases, which were defined as part of the activities in T2.2 ("User Stories and Requirements Analysis").

The activities and the Reference Architecture presented in this Deliverable represent an input for the different tasks of WP3, WP4 and WP5, which are all dedicated to the development of the specific modules, tools and services included in the three layers of the considered architecture (respectively, the MATRYCS governance layer, the MATRYCS processing layer and the MATRYCS analytics layer). It is worth noting that, since the proposed architecture provides a high-level abstraction of the required functionalities, it does not put any constraints on the technologies used for the practical implementation of the architecture components; the implementation work done in WP3, WP4 and WP5 (reported in the associated Deliverables) thus represents only a particular instantiation of the presented reference architecture.

1.3 Deliverable structure

In addition to the present Section, this Deliverable is composed of other four Sections:

- **Section 2** provides a review of some of the already existing works towards the definition of reference architectural models for big data management, which is used as input to underline the most important aspects and principles to be considered for the Reference Architecture definition.
- **Section 3** focuses on the aspects specifically related to the building sector and analyses the different stakeholders and their role with respect to the building data architecture, as well as the specifications and requirements coming from different use cases associated to operational performance, retrofitting, policies and investments in the building sector.
- **Section 4** presents and discusses the preliminary version of the proposed MATRYCS Reference Architecture. It will provide the details on the different architecture layers, the associated modules and components, together with their functional description.
- **Section 5** discusses the alignment of the proposed Reference Architecture to some of the existing frameworks for big data management already presented in Section 2, with the goal of highlighting similarities, possible differences and to show how the proposed architecture can be mapped into different architectural views.

2 Review of existing architectural models for big data management

The significant growth of generated data in almost all the industrial sectors, the business value these data have for the different stakeholders within the so-called big data value chain, as well as the continuous progress in the Information and Communication Technology (ICT) domain created an impellent demand for unified frameworks that set the principles, modalities, criteria and rules to provide, access, process, exchange and consume the available data. In this context, several projects, working groups and public or private consortia worked towards the definition of reference architectures tailored to specific domains or more generally oriented to the industry and the big data context.

This Section provides a review of some of the most important proposals of reference architecture models for big data management. The scope is to underline the major aspects that should be considered for the design of a reference architecture as well as to identify how the proposed MATRYCS architecture can be aligned and mapped to already existing architectures. In the following, the design principles and reference models promoted by the Big Data Value Association [1], the Industrial Internet Consortium [2], the Alliance for Internet of Things Innovation [4], the FIWARE Community [5], the International Data Space Association [10], [11] and the OpenDEI project [12] will be presented and discussed.

2.1 Big Data Value Reference Model

The Big Data Value Reference Model is an architectural model defined by the Big Data Value Association (BDVA) [1]. The BDVA is a not-for-profit organization composed of several industrial and research-oriented partners, which represents the private counterpart of the European Commission in the Big Data Value Public-Private Partnership (PPP) launched in 2014. One of the most important goals of the BDVA is to strengthen the competitiveness and leadership of the European industry by fostering the development of Big Data Value technologies and services. The design of a reference architectural model clearly goes in this direction, since it aims at facilitating and accelerating the implementation and uptake of such Big Data technologies.

The proposed Big Data Value Reference Model is shown in Figure 1 [1]. It is structured into horizontal and vertical concerns. **Horizontal concerns** mostly focus on specific aspects of the data processing chain, starting from the data collection and ingestion and all the way up to the data presentation and visualization. These horizontal concerns are not intended to strictly define a layered architecture; they mostly underline the different types of data operations that can be possibly present to process the data and to allow extracting or enriching their associated value. **Vertical concerns** refer instead to cross-cutting issues, sometimes not specifically related to technical aspects, which may affect all the horizontal concerns.

More in details, the horizontal concerns include:

The MATRYCS project has received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no.101000158

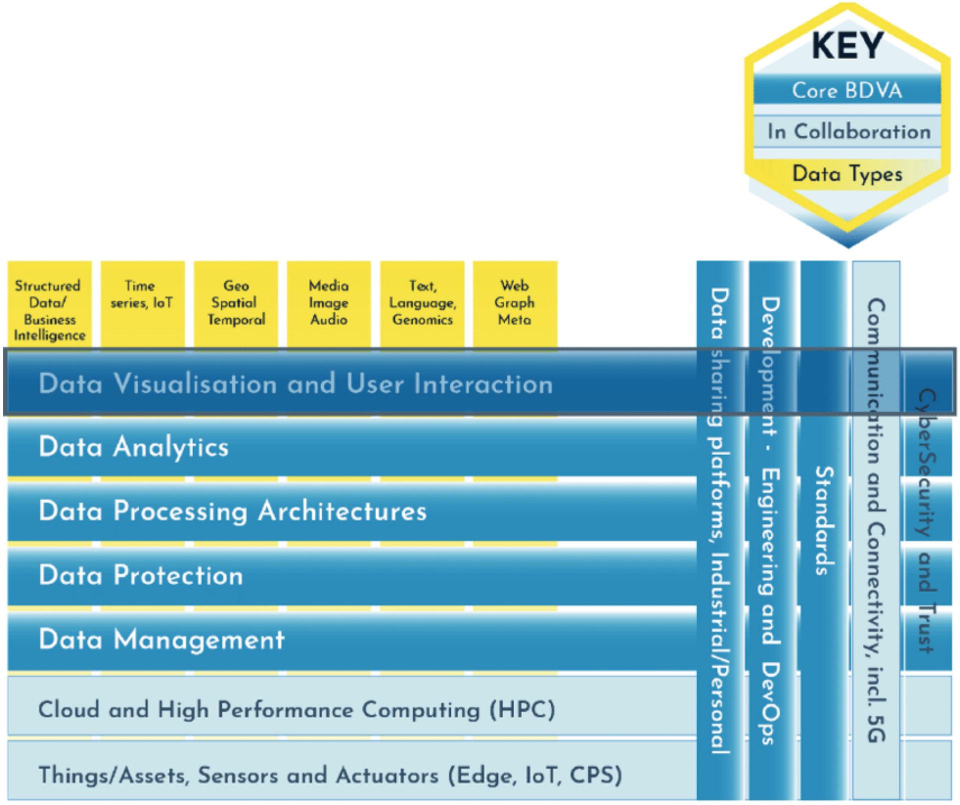

KEY
Core BDVA
In Collaboration
Data Types
Structured Data/ Business Intelligence
Time series, IoT
Geo Spatial Temporal
Media Image Audio
Text, Language, Genomics
Web Graph Meta
Data Visualisation and User Interaction
Data Analytics
Data Processing Architectures
Data Protection
Data Management
Cloud and 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 1: Big Data Reference Model defined by the Big Data Value Association

- **Things/Assets, Sensors and Actuators (Edge, IoT, CPS)**: this takes into account the sources of data as well as those entities that can be controlled by using data-driven intelligence. The architectural model has to guarantee the seamless integration of all these entities, ensuring the possibility to connect any type of smart device or object. To meet possible real-time requirements, edge cloud processing has to be possible in the architectural solution.
- **Cloud and High Performance Computing (HPC)**: it reflects the potential need of employing cloud and HPC infrastructures in order to deal with the large amount of data and to allow the deployment of sophisticated and, possibly, computationally demanding, data analytics services.
- **Data Management**: it refers to all the techniques and tools needed to handle different types of data (including also semi-structured and unstructured data), to manage multilingualism, to ensure data cleaning, curation and harmonization for semantic interoperability, to assess data provenance and assure data quality.
- **Data Protection**: this includes all the mechanisms employed to guarantee data protection in terms of privacy, anonymization and trust. This includes robust algorithms to guarantee anonymity, mechanisms to control data usage as well as risk assessment tools to analyse privacy vulnerabilities over large sets of data.

- **Data Processing Architectures**: it points to the frameworks and toolboxes necessary to deal with both batch and streaming data, to ensure real-time capabilities for some Big Data applications, and to handle large amounts of data in short times. This also includes the use of decentralized architectures and efficient methods for data processing and storage.
- **Data Analytics**: this includes all the analytics methods and techniques used to understand data, to recognize patterns, to classify information, to obtain additional learning and to provide meaningfulness to the data. As a matter of fact, this concern closely relates to the use of advanced Artificial Intelligence (AI), Machine Learning (ML) and Deep Learning (DL) tools to extract (or provide) business value to the available data.
- **Data Visualization and User Interaction**: it refers to the tools and visual interfaces used to present the data to the final users and to give them an enhanced user experience. It also takes into account the modalities provided to query the data and more in general the tools implemented to allow an interactive access to both data and services.

The vertical concerns include:

- **Data sharing platforms**: this concern refers to the need of ensuring the access to already existing personal or industrial data platforms as well as to marketplaces, research data platforms, urban/city data platforms, etc. This highlights the need for designed architectures to be able to exchange and trade data with external platforms, thus requiring platform-to-platform interoperability.
- **Development, Engineering and DevOps**: it points to the need of implementing tools and methodologies to allow testing, monitoring, verification and validation of the services in order to guarantee increased productivity as well as reliability, security and quality in the created Big Data Value.
- **Standards**: it relates to the need of unified and standardized approaches to foster integration, sharing and interoperability. These are indeed major challenges in the Big Data environment. Standardization applies not only to data (to foster sharing and interoperability), but also to the technologies used within the Big Data platforms.
- **Communication and connectivity**: it refers to the communication and connectivity mechanisms necessary to access, deliver, exchange and share data. This includes, among others, also the aspects associated to latency, reliability, availability and Quality of Service (QoS) of the communication infrastructure, which may be more or less relevant depending on the considered Big Data application.
- **Cybersecurity and trust**: it concerns all those aspects of cybersecurity that go beyond the privacy and anonymization issues. As clear, availability of Big Data and the manipulation of the associated information via AI and ML open to new security challenges. This concern thus incorporates all the countermeasures adopted to secure data repositories, to ensure data sovereignty, to guarantee protection of Intellectual Property Rights, to regulate the access and the rights to manipulate data, to detect possible cyber-attacks, threats or vulnerabilities, etc. In addition, this concern also includes the mechanisms adopted to guarantee data trust, which are often associated to the adoption of distributed ledger/blockchain technologies.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Data Types**: the Big Data Value Reference Model also underlines the concern, and the associated challenge, regarding the highly heterogeneous variety of data that architectural components must handle. Heterogeneity of the data exists under different standpoints, including the level of structure (structured, semi-structured or unstructured data), the time domain (time-series data or static data), their resolution (geographical or temporal), the format and representation (video, audio, images, text, etc.), the presence of metadata and semantic annotation, etc. Adequate algorithms and tools need to be available, at each of the data operation levels, in order to be able to deal with such a variety of data.

2.2 Industrial Internet Reference Architecture

The Industrial Internet Consortium (IIC) is a not-for-profit partnership among industrial, governmental and academic partners founded in 2014 to foster the acceleration of the digital transformation across industries. As part of its activities, the IIC has developed the Industrial Internet Reference Architecture (IIRA) [2], an architecture for IoT systems derived from the abstraction of common characteristics, features and patterns identified in a variety of industry IoT related use cases. The scope of this architecture is to drive interoperability, map applicable technologies and guide technology and standard development. The architectural concepts provided through the IIRA are general and at a high level of abstraction, since they are intended to have broad industry applicability and to cover without specific restrictions a multitude of heterogeneous industrial use cases.

According to the IIC vocabulary, which is derived from the standardized definitions in the ISO/IEC/IEEE Architecture Description standard [3], a Reference Architecture "*provides guidance for the development of system, solution and application architectures. It provides common and consistent definitions for the system of interest, its decompositions and design patterns, and a common vocabulary with which to discuss the specification of implementations and compare options. By staying at a* ***higher level of abstraction****, it enables the identification and comprehension of the most important issues and patterns across its applications in many different use cases*".

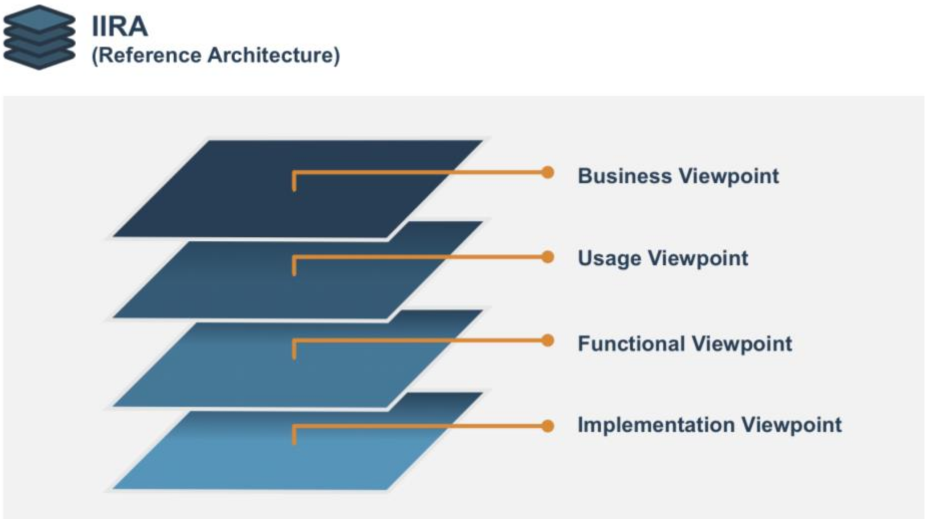
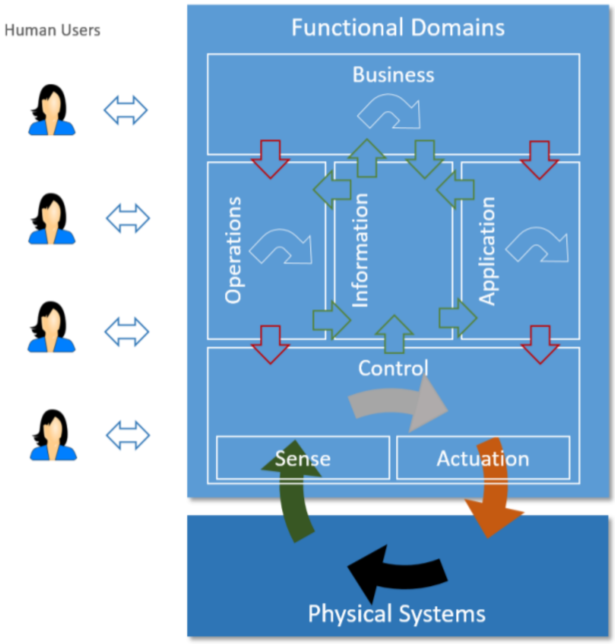


Figure 2: Industrial Internet Architecture Viewpoints

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

Similar to the BDVA Reference Model, the goal of an architecture framework is to discover, describe, organize and resolve concerns about the systems at hand. At the core of the IIRA are the **architecture viewpoints**, which are the collection of conventions used to frame the architectural solutions employed to resolve specific sets of system concerns. As shown in Figure 2 [2], the IIRA identifies four main architectural viewpoints, which have different architectural representation models:

- **The Business Viewpoint** refers to business-oriented concerns that come from stakeholders' needs and their business vision. Its associated architectural representation has thus the objective of identifying how the IoT system allows achieving the business objectives through the mapping to the system capabilities.
- **The Usage Viewpoint** deals with the concerns associated to the usage of the IoT system, thus detailing how the different activities, or sequence of activities, have to be executed to achieve the expected system functionalities. Its associated architectural representation has thus the goal of describing the different activities and of identifying the tasks and roles of the different entities interfaced to the system of interest.
- **The Functional Viewpoint** focuses on the functional components of the IoT system as well as on the definition of their interfaces and interactions in order to allow the system usage and to eventually provide the expected system capabilities. The functional view is further decomposed in domains (see Figure 3 [2]), which differentiate the functions used within the system according to the type of provided functionalities. For example, the business domain may include Enterprise Resource Planning, Payment and Billing functions, the Information Domain can contain Data Management and Data Analytics functions used to process and understand data, whereas the operations domain may entail functions for monitoring, optimization, diagnostic, etc.



Green Arrows: Data/Information Flows; Grey/White Arrows: Decision Flows; Red Arrows: Command/Request Flows

Figure 3: Domains associated to the IIRA Functional Viewpoint

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **The Implementation Viewpoint** specifically refers to the technical representation of the technological components needed to realize the fundamental system capabilities. Overall, this viewpoint thus includes the description of the general architecture together with its structure and the distribution of the different architectural components, a technical description of the components, of their interfaces, behaviour and properties, as well as an implementation map of the different key system characteristics.

Together with the different architectural viewpoints, the IIRA also introduces the concepts of **architectural patterns**, which are a simplified and abstracted view of IoT system implementations recurrent in the industrial sector. The most general architectural pattern is the three-tier architecture shown in Figure 4 [2], which is composed of:

- **Edge Tier**: it is the tier responsible for the data collection at the edge from the existing devices, sensors and IoT entities and for closing the applications' control loop via controllers and actuators.
- **Platform Tier**: it is the tier responsible for processing and analysing the data before making it available to the users; it also provides management functionalities for the devices and assets of the IoT system.
- **Enterprise Tier**: it is the tier where the applications, together with the associated visualization user interfaces, reside; this gives the opportunity to the user to perform its application and business functions while accessing the data in the platform.

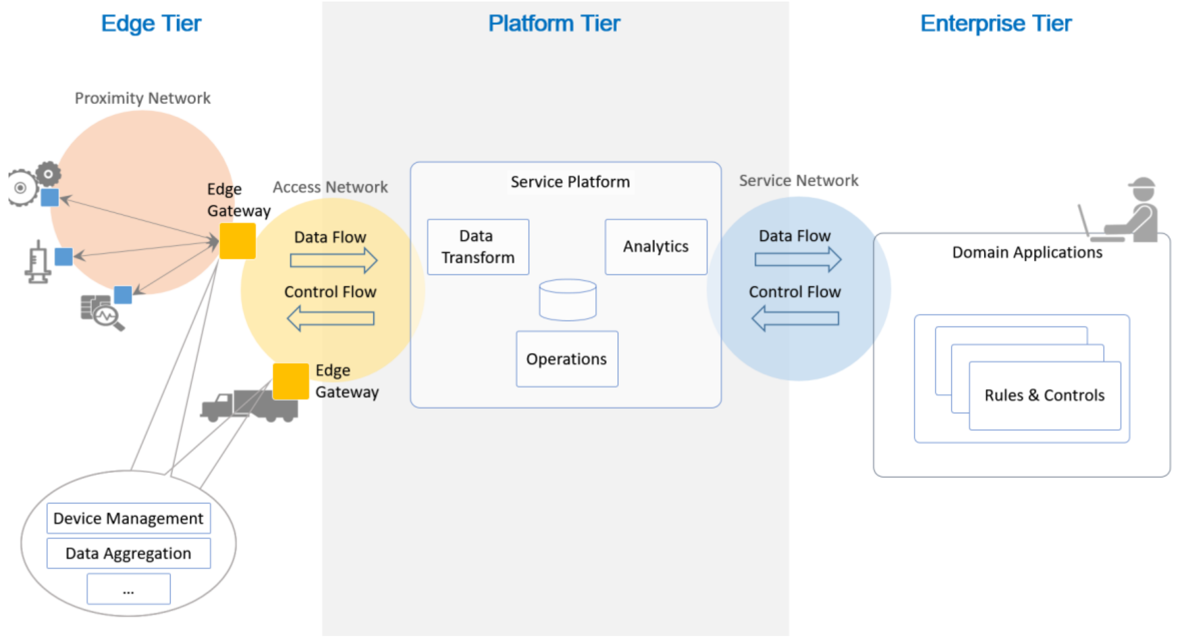


Figure 4: IIRA three-layer architectural pattern

This three-layer architecture is general enough and some specific instantiations may lead to particular variations, such as a layered databus architecture pattern. In this case, the industrial system is decomposed in multiple levels (smart machines, systems, system of systems, industrial internet) that may exchange data with the immediately lower and upper levels via specifically designed databus. Such an architecture pattern allows decoupling different hierarchical levels allowing to achieve large scalability.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

In addition to the architectural viewpoints and patterns previously described, the IIRA also considers the presence of cross-cutting concerns and system characteristics, which are transversally affecting all the viewpoints and architecture tiers. Safety and cybersecurity are probably the most obvious example of cross-cutting concerns extended to all the levels and viewpoints of the Reference Architecture.

2.3 AIOTI High Level Architecture

The Alliance for Internet of Things Innovation (AIOTI) is a partnership started in 2016 among a large number of stakeholders interested or involved, at different levels, in the development of IoT technologies. AIOTI members include several industrial actors, from large companies to Small and Medium Enterprises (SMEs), as well as universities, research institutes and user representatives. The mission of AIOTI is to drive its members towards the development and uptake of IoT and Edge Computing technologies to support the digitalization process of companies and increase the competitiveness of Europe.

In 2017, the Working Group (WG) on standardization developed a first version of the AIOTI High Level Architecture (HLA), which was recently (in 2020) reviewed and released in an updated version. The goal of such an architecture is to provide a coherent view of architectural concepts that can be used as a basis for the instantiation of Large Scale Pilot (LSP) deployments. The AIOTI HLA puts the "**Things**" of the IoT system at the center of the value creation. With reference to the possible architectural viewpoints, the AIOTI HLA focuses on the Domain and Functional Models:

- **The Domain Model** describes the entities in the IoT scenario and their relationships.
- **The Functional Model** describes the functions (and their interfaces) in the IoT domain.

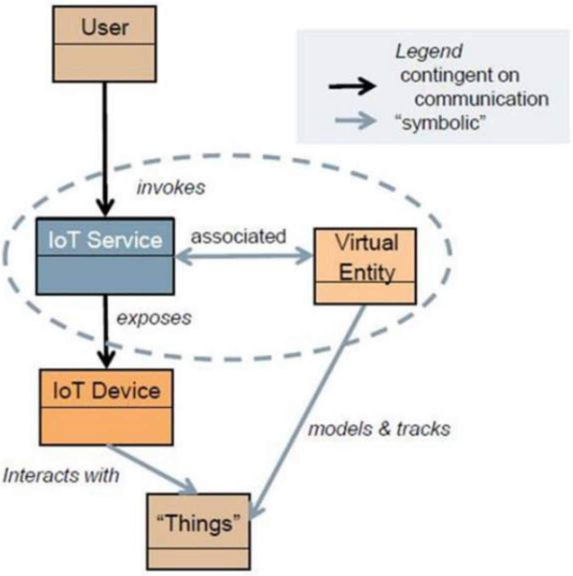


Figure 5: AIOTI Domain Model

The architectural view of the Domain Model is shown in Figure 5 [4]. This model representation relies on two main entities, the User and the Thing, which can interact by means of IoT devices and IoT services. The Thing is a physical entity or object that can be uniquely identified. In the IoT system, each Thing has

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

the ability to transfer its data and the possibility to expose its capabilities by means of an IoT device. The physical entity has a digital representation in the IoT system, called virtual entity, which is realized via an IoT Service. The IoT Service is the bridge allowing the interaction between User and Thing and the possible execution of specific operations in the IoT system.

The architectural view of the Functional Model is given in Figure 6 [4]. The Functional Model is composed of three layers (each layer is here intended in a software architecture sense, namely as a set of software components that provide a cohesive set of services):

- **The Network Layer**: it includes all those services put in place to guarantee the connectivity of all the entities in the IoT system, as well as other network services such as location, QoS, etc.
- **The IoT Layer**: it contains those groups of services responsible for the preliminary data processing and data management, including, for instance, data storage and data sharing. The IoT Layer exposes the collected data to the Application Layer through ad hoc Application Programming Interfaces (APIs).
- **The Application Layer**: it groups all those services that implement specific application logics used for data elaboration. They are the services directly used by the User to monitor, interact or control the Things within the IoT system.

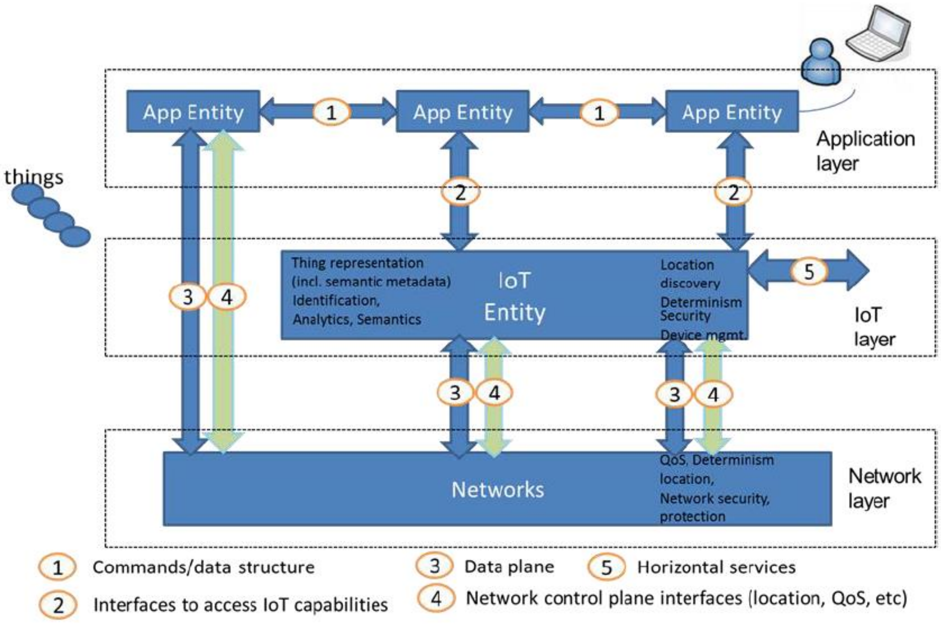

App Entity
1
App Entity
1
App Entity
Application layer
things
2
2
Thing representation (incl. semantic metadata) Identification, Analytics, Semantics
IoT Entity
Location discovery Determinism Security Device mgmt.
5
IoT layer
3
4
3
4
3
4
Networks
QoS, Determinism location, Network security, protection
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 6: AIOTI HLA Functional Model

In the Functional Model presented in Figure 6, the group of services implemented at the different layers of the HLA are associated to specific entities. In particular:

- **IoT Entities**: they are those entities exposing specific IoT functionalities to the upper-level applications, via dedicated APIs. Examples of this type of entities are data storage services, publish/subscribe services, notification services, access management services, etc.

- **App Entities**: they are the entities at the Application Layer that implement the application logics within the overall system. These entities can be located everywhere, like in remote servers, in smartphones or directly on-board in the devices.

In addition to the architectural view just described, the AIOTI HLA recommendations also include a set of deployment considerations, mostly associated to cross-cutting concerns that must be duly taken into account in each architecture representation and instantiation. As an example, security and management are recognized as essential aspect of the AIOTI HLA and, even if not explicitly represented in the HLA Functional Model, are considered to be intrinsically present in each interface design. The HLA interfaces should support authentication, authorization and encryption at each level. Similarly, digital rights, such as identity, access, rights of use, management and control at all the layers are recognized as important features to be guaranteed.

Other specific deployment considerations concern the use of Cloud and Edge Computing, the framing of the IoT system in the context of Big Data, the security and trust of both data and technologies, the privacy issues and the employment of virtualization technologies for the IoT system deployment. All these aspects (and the specific recommendations and considerations underlined by the AIOTI), even if not further detailed in this Section, are considered and treated in Section 3, when dealing with the architecture design principles.

2.4 FIWARE Open Reference Architecture

The FIWARE Foundation is a non-profit organization founded in 2016 that drives the definition, and adoption of open standards to develop smart solutions, based on open-source technologies, in a multitude of industrial domains. It has more than 400 members from all over the world, including large and small companies as well as research and academic institutions. The mission of FIWARE is "*to build an open sustainable ecosystem around public, royalty-free and implementation-driven software platform standards that will ease the development of new Smart Applications in multiple sectors*". This is done via the development of open-source software platform components that can be used within a fully FIWARE-powered platform or together with third party components to build hybrid platforms aimed at implementing and deploying smart applications and services in different domains.

The FIWARE software architecture builds upon the concepts of **Context** and **Context Information** [5]. The Context is the collection of the **Digital Twins** associated to the system under analysis. Each Digital Twin, in turns, can be defined as the digitalized, virtual representation of either a physical asset or a specific concept belonging to the considered system. The data associated to the different attributes of each Digital Twin represent the Context Information. Handling and sharing, via ad hoc methods, the Context Information is the key aspect of the FIWARE architecture. To this purpose, the FIWARE architecture includes a specific Generic Enabler (GE), the Context Broker, which takes care of this task (GEs are the building blocks of the FIWARE architecture; they can be defined as software components able to provide a well-defined functionality and they expose open APIs for the integration in the platform). Having a FIWARE Context Broker [6] is the only requirement to define a platform "powered by FIWARE" (other platform software components can come from the FIWARE Catalogue of GEs or can be taken from third parties).

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

The central role of the Context Broker is reflected also in the FIWARE Open Reference Architecture (Figure 7 [6]). This is a three-layered architecture composed of (starting from the bottom):

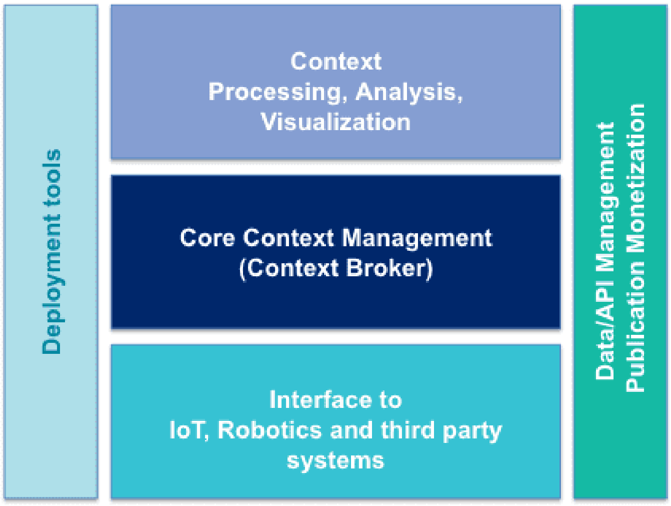

Deployment tools
Context Processing, Analysis, Visualization
Core Context Management (Context Broker)
Interface to IoT, Robotics and third party systems
Data/API Management Publication Monetization

Figure 7: FIWARE Open Reference architecture

- **Interface to IoT, Robotics and third-party systems**: it is the bottom layer of the architecture; it includes all the agents required for the interconnection to field devices or other systems, from which the data are collected and towards which the actuation commands are sent. The FIWARE catalogue contains a number of GEs conceived to facilitate the connection to several physical components (see for example [7] or [6] for a full list).
- **Core Context Management (Context Broker)**: as already anticipated, this is the heart of the FIWARE architecture. It keeps, handles and exposes all the information related to the digital twins of the physical (or non-physical) objects in the considered system.
- **Context Processing, Analysis and Visualization**: it is the upper layer of the architecture; it includes all those GEs or third-party software components employed for the further processing of the Context Information, for enriching the value of the available information (data analytics tools) or for presenting the information to the user by means of dedicated graphical interfaces.

In parallel to the three layers, the FIWARE Reference Architecture has additional GEs developed to address cross-cutting concerns, such as security (such as the Keyrock [8] and Wilma [9] GEs). In this regard, the FIWARE catalogue [6] offers a set of GEs aimed at dealing with the aspects related to Identity and Access Management, as well as additional GEs that foster the publication and monetization of data.

Starting from the Open Reference Architecture given in Figure 7, FIWARE Reference Architectures tailored to specific domains have been also defined. Figure 8 shows, as an example, the FIWARE Reference Architecture for the Smart Energy Management Services domain in the power system sector. Starting from the bottom of the architecture, it is possible to find the physical devices (or objects) that are relevant in the Energy Management domain together with other possibly available sources of information. These can be connected to the software platform either via specific agents available in the FIWARE Catalogue or using third parties IoT systems. This second layer thus guarantees the required connectivity. Data coming from the physical system via the above-mentioned interfaces must be then

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

transmitted to the Context Broker. As it is possible to see in Figure 8, the Context Broker contains digital twins for different power system entities and, accordingly, it handles the associated data and exposes them using open APIs. The upper layers then contain different processing and visualization tools that exploit the Context Information shared by the Context Broker to manage, control and optimize the energy system. Transversally, specific GEs (or third-party components) can be implemented to guarantee security and to allow the data monetization.

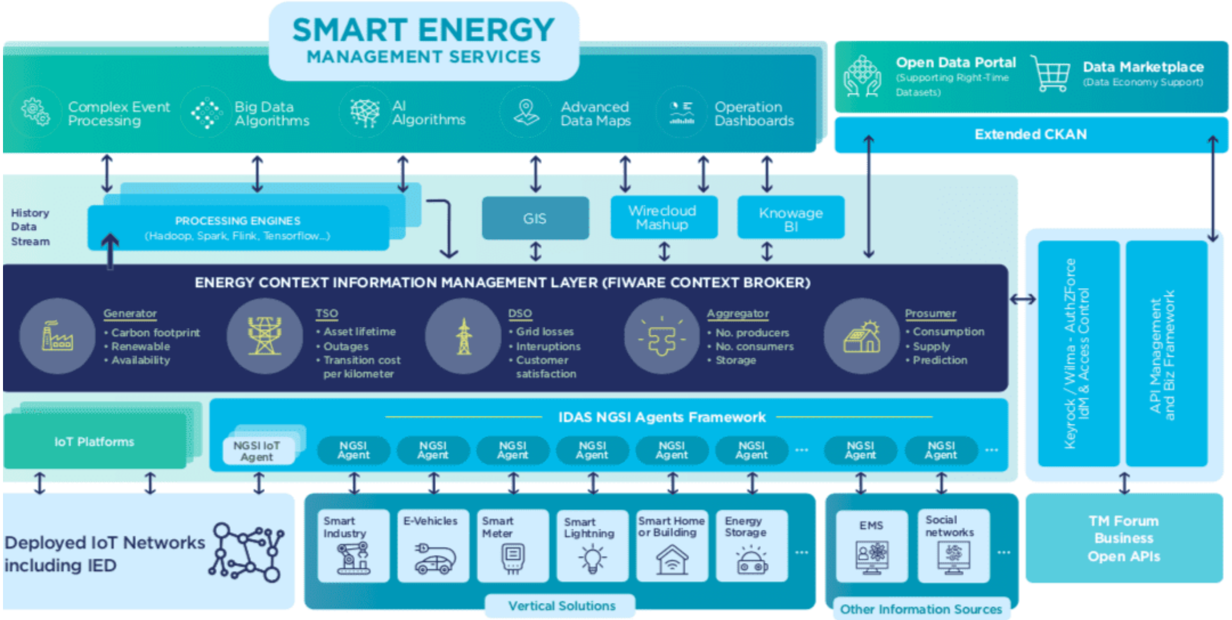


Figure 8: FIWARE Reference Architecture for the Smart Energy domain

2.5 IDS Reference Architecture Model

The International Data Space Association (IDSA) is a not-for-profit organization with more than 130 member companies across the European Union and around the world, whose mission is to create a secure and sovereign system of data exchange in which all the involved participants can develop their business around data, thus unlocking a trusted environment for the data economy of the future. The International Data Space (IDS) Reference Architecture Model (RAM) is the core of the proposed data exchange system [10]. It is the framework defining the concepts, aspects, processes and policies pertaining the data exchange, while enforcing data sovereignty, trustworthiness, security and privacy.

The IDS-RAM is defined around the concept of **Data Spaces**. Recalling the definition provided by the IDSA, Data Spaces are "*virtual data spaces leveraging existing standards and technologies, as well as governance models well-accepted in the data economy, to facilitate secure and standardized data exchange and data linkage in a trusted business ecosystem. Data Spaces thereby provide a basis for creating smart-service scenarios and facilitating innovative cross-company business processes, while at the same time guaranteeing data sovereignty for data owners*". From a different standpoint, Data Spaces can be also defined as those frameworks that allow linking different stakeholders, IT systems, platforms and data sources through policies and mechanisms for trusted and secure data exchange/sharing, hence creating an ecosystem where participants team up to achieve mutual benefits via the development of

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

data-driven services and innovation. Coherently with the previous definition, Figure 9 [10] illustrates an exemplary schematic view of an IDS as a decentralized framework interconnecting different platforms, IT systems and companies into a unique ecosystem.

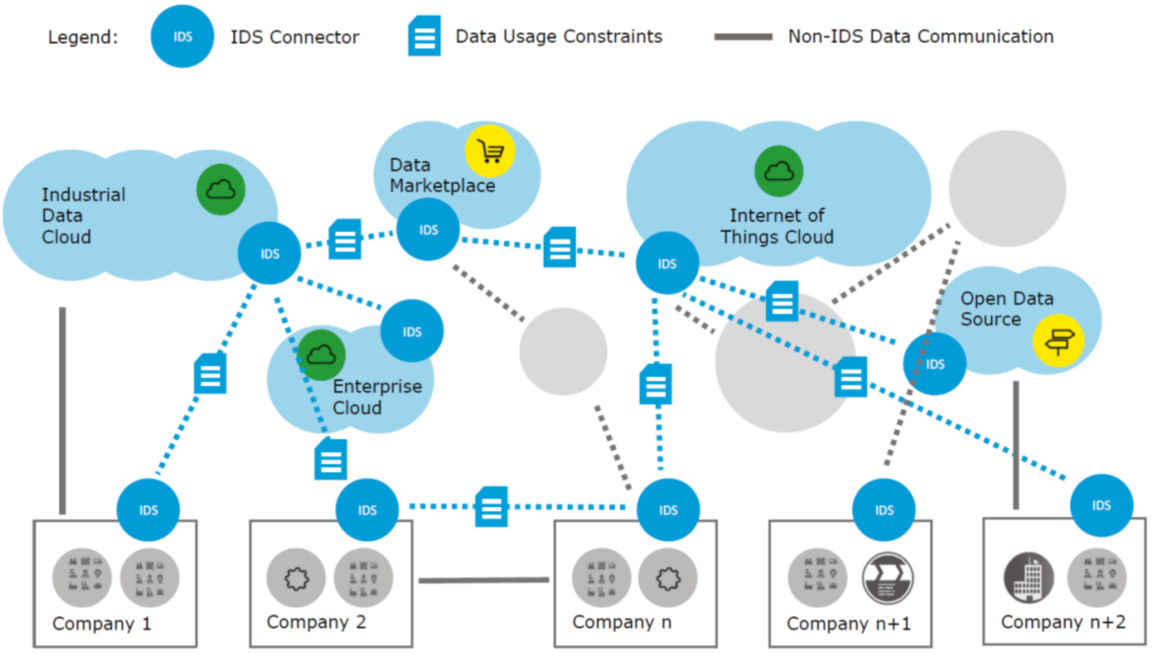


Figure 9: Visual representation of Data Spaces as a decentralized ecosystem

Given the above definition of Data Spaces, the IDS-RAM can be seen as a high-level architecture that defines and describes the frameworks, infrastructures, components, methods and policies for the design of such an ecosystem. The general high-level structure of the IDS-RAM is made up of five layers associated to different architectural viewpoints, which share three "perspectives" spanning across all the layers, as depicted in Figure 10.

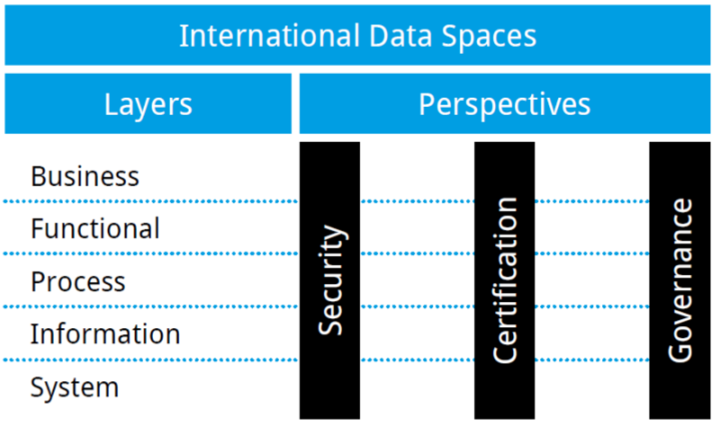


Figure 10: IDS Reference Architecture Model structure

The Business Layer defines and describes the different roles that participants may assume within the IDSs, and it specifies the basic interactions among them. The roles in the IDS-RAF are clustered in four categories, which include:

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Core participants**: are those directly involved in the data sharing/exchange as they own, provide, use or consume the data. In other words, these are the participants interested to exploit the business value associated to the data. Participants belonging to this category include data owners, data providers (owners and providers may coincide), data consumers, data users (consumers and users may coincide) and application providers.
- **Intermediaries**: are trusted actors of the IDS ecosystem responsible to create the trustworthiness in each data sharing/exchange process. These roles can be only taken by trusted organizations. Participants belonging to this category include broker service providers, identity providers, clearing house, app store providers and vocabulary providers.
- **Software/service providers**: are IT companies providing specific software or services to the core participants of the IDS for implementing the functionalities required in the IDS. Provided software and services may include, for example, also data cleansing, harmonization or semantic enrichment.
- **Governance body**: it consists of the certification bodies that are responsible of the certification process for both IDS participants and core technical components, which is one of the most important features in the IDS-RAM to guarantee security and trustworthiness. In addition, it also includes the IDSA, which is responsible for the continuous development of the IDS concepts.

Figure 11 shows a simplified example of the potential interactions among some of the main actors of the IDS [11].

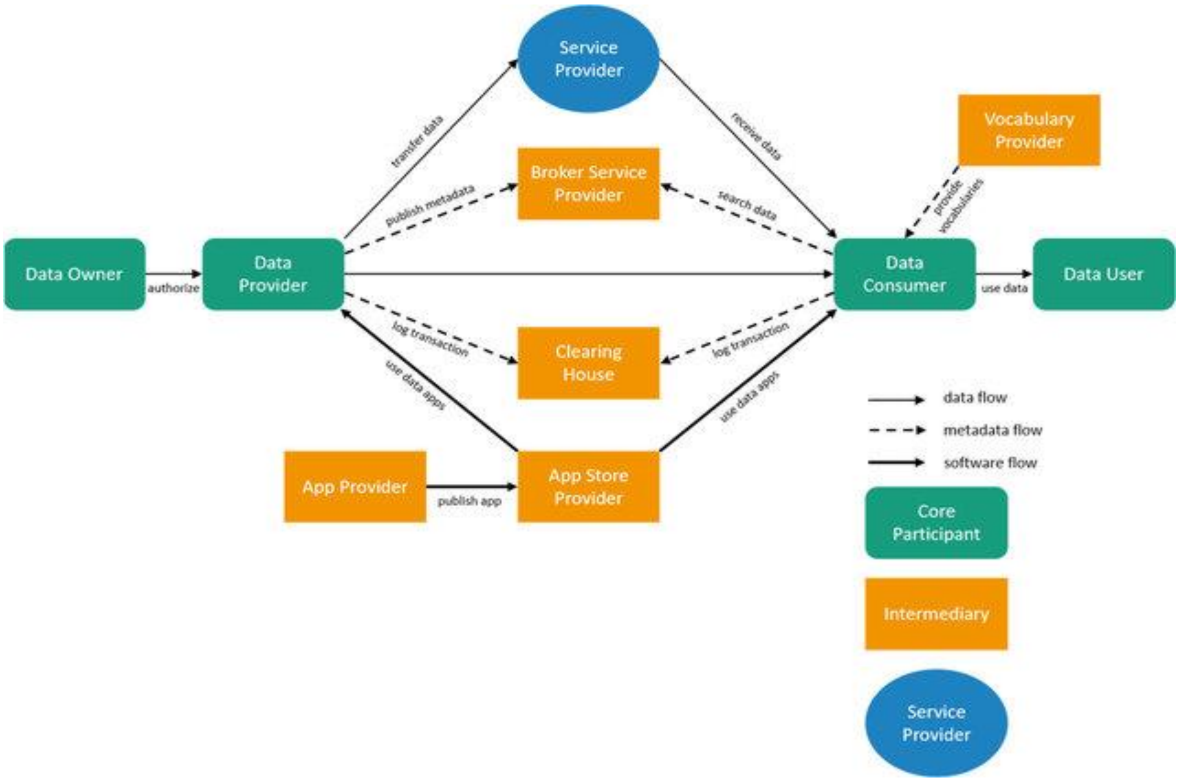
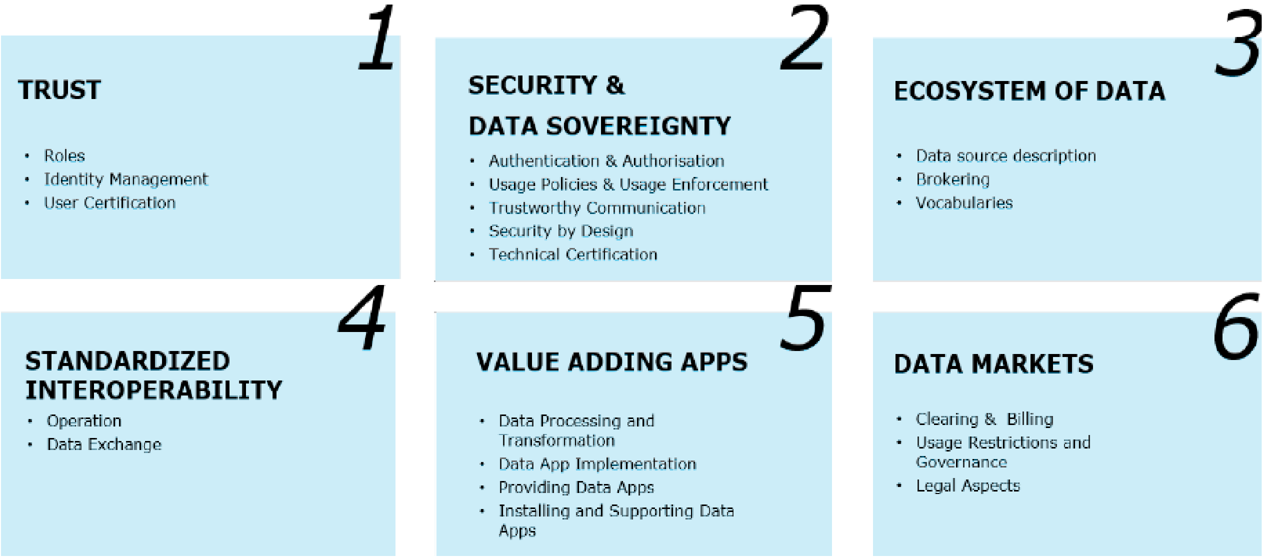


Figure 11: Schematic view of main roles and interactions in the IDS

The Functional Layer defines and describes the functional requirements of the IDS and, accordingly, the features to be implemented in the IDS ecosystem. Figure 12 shows the overall view of the considered functional requirements. Since these may be relevant also in view of the MATRYCS Reference Architecture, they are briefly illustrated in the following.



1 TRUST

- Roles
- Identity Management
- User Certification

2 SECURITY & DATA SOVEREIGNTY

- Authentication & Authorisation
- Usage Policies & Usage Enforcement
- Trustworthy Communication
- Security by Design
- Technical Certification

3 ECOSYSTEM OF DATA

- Data source description
- Brokering
- Vocabularies

4 STANDARDIZED INTEROPERABILITY

- Operation
- Data Exchange

5 VALUE ADDING APPS

- Data Processing and Transformation
- Data App Implementation
- Providing Data Apps
- Installing and Supporting Data Apps

6 DATA MARKETS

- Clearing & Billing
- Usage Restrictions and Governance
- Legal Aspects

Figure 12: Functional requirements of the IDS

- **Trust**: it is one of the fundamental features of the IDS. It is enforced by defining rights and duties of each role, assigning a unique identifier and a certificate to each IDS connector (used for the connection of each participant to the IDS), and establishing the certification processes that each IDS participant must undergo.
- **Security & Data Sovereignty**: they are fundamental features of the IDS. Sovereignty is achieved by means of usage policies attached to the shared data. Security includes the processes of authentication, authorization, check of trustworthiness, which are all established also by means of dedicated certification procedures.
- **Ecosystem of Data**: it is a requirement associated to the ability of finding and correctly interpreting the available data. It implies the use of the Information Model described at the Information Layer of the IDS-RAM.
- **Standardized Interoperability**: it is achieved by means of dedicated IDS Connectors, which allow adopting push/pull or subscribe mechanisms to exchange data.
- **Value Adding Apps**: it refers to the possibility to develop specific applications capable of transforming data according to the potential needs of IDS participants. The data apps need to be clearly described in terms of their functionalities and of their input/output characteristics. Data apps can be then made available in the app stores.
- **Data Markets**: it brings the requirements associated to the mechanisms needed to ensure the monetization of the data value. This includes the definition of pricing models from data owners and the logging of transactions as well as the implementation of processes for clearing and billing and for establishing legal contracts.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

The Process Layer defines and describes in a detailed way the steps and flowchart of different processes that may take place within the IDS. The current version of the IDS-RAM reports the details of the "onboarding", "exchanging data" and "publishing and using data apps" processes.

The Information Layer defines and describes the Information Model used within the IDS, namely the domain-agnostic, common language to be used into the IDS. This represents an essential feature to foster interoperability. The IDS Information Model relies upon the definition of digital resources, which are modelled and described using the so-called "concern hexagon" (Figure 13). The concern hexagon reports the different details to be provided for each digital resource, which include information about what the digital resource represents (content, concept and context), how the content is exchanged (communication and commodity) and the certification and other details to guarantee trust.

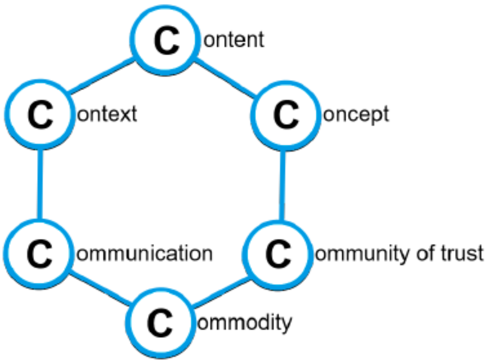


Figure 13: Basic view of the concern hexagon for the IDS Information Model

The System Layer concerns aspects related the integration, deployment, execution, configuration of the logical software components in virtualized environments such as virtual machines and application containers.

Regarding the three cross-cutting perspectives considered in the IDS-RAM, these include:

- **Security**: it includes the procedures employed, at all the architectural levels, to ensure the identification of participants, to protect communications and data exchange transactions, and to control the usage of the data after the exchange.
- **Certification**: it concerns all processes, rules and practices employed for the certification of both the IDS participants and the software components adopted (for instance the IDS Connectors) for the data sharing/exchange.
- **Governance**: it deals with the definition of all the roles, functions, processes that the IDS should have to meet the defined functional and non-functional requirements, as well as with the mechanisms used to check compliancy.

2.6 OPEN DEI Reference Architecture Framework

OPEN DEI (Aligning Reference Architectures, Open Platforms and Large Scale Pilots in Digitising European Industry) is a European H2020 project started in 2019 that aims at supporting the European Union efforts for the creation of common data platforms based on unified architectures and established standards. To this purpose, one of the main goals of the project is to compare existing reference architectures, possibly belonging also to different domains, in order to allow the creation of a unified data platform that can contribute to fasten the digital transformation of Europe and to rise the competitiveness of European industries and companies. In particular, OPEN DEI focuses on the architectures, platforms and pilots in four strategic industrial domains, namely manufacturing, agriculture, energy and healthcare.

As clear from the above description, the activities carried out in the context of the OPEN DEI project are of great importance also for the design and definition of the MATRYCS Reference Architecture. In October 2020, OPEN DEI published the first version of the Deliverable "Reference Architecture for Cross Domain Digital Transformation" [12]. This document first contains a short review of state-of-the-art reference architectures, and then it presents the design principles and the specifications for the proposed OPEN DEI Reference Architecture Framework (RAF), which tries to harmonize and coordinate different digital transformation architectural approaches under a unified framework.

The underlying architectural design principles are defined as those principles that are relevant for the design of data-driven services enabling the digital transformation as well as the digital platforms that support the deployment of such services. Six major principles (and associated recommendations) are identified, which are reported in the following (as given in [12]):

- **Interoperability through data sharing**: the RAF "*should allow technical interoperability at syntactic and semantic level via data sharing mechanisms grounded on well-established standards and design/implementation patterns*". This recommendation aims at contrasting the currently present fragmentation in terms of data formats, protocols, semantic description, which is present among different domains and, very often, also internally to the same industrial sectors.
- **Openness**: the RAF "*should ensure a level playing field based on open-source datasets/software/standards and demonstrate active and fair consideration of the coverage of functional needs, maturity and market support and innovation*". The openness recommendation thus refers to the possibility of having: i) open data (for free or under fair conditions), at least if this does not violate any protection of personal data, confidentiality or IPR; ii) open API specifications, namely royalty-free specifications, to foster the re-use of software implementing those specifications; iii) open-source software technologies, where possible, to allow saving development costs and to guarantee fast adaptation to specific business needs.
- **Reusability**: the RAF "*must support reusing and sharing of data and solutions, enabling cooperation in the collaborative development of data models and solutions when implementing Digital Transformation pathways*". This recommendation concerns software components, APIs, information and data, and it is a means to further support interoperability and quality. Applying this recommendation clearly requires the involved stakeholders to be open to share their solutions (for free or under fair conditions).
- **Avoidance of vendor lock-in**: the RAF "*should foster access and reuse of their digital services and data irrespective of specific technical implementations or products*". In other words, the RAF has to be conceived in a way that it is agnostic to any specific technological implementation or instantiation of the architecture.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Security and privacy**: the RAF "must define a common security and privacy framework and establish processes for digital services to ensure secure and trustworthy data exchange between the involved stakeholders and in interactions with organization and businesses". The identified architectural and technological solutions must thus guarantee confidentiality, authenticity and integrity of information as well as suitable methods and mechanisms to ensure identification, authentication, authorization, trust monitoring and certification of solutions.
- **Support to data economy**: the RAF "must define a data marketplace framework enabling parties to publish open and priced data, supporting the creation of multi-side markets and innovative business models which bring support to the materialization of a Data Economy". Accordingly, suitable tools should be in place in the architecture to allow the monetization of the data value, including data usage accounting, rating, payment settlement and billing.

In addition to the above principles, the OPEN DEI RAF has been developed also according to the following 6C architectural model (see Figure 14), which, in a way, it represents the different levels of processing operations that can be performed on data:

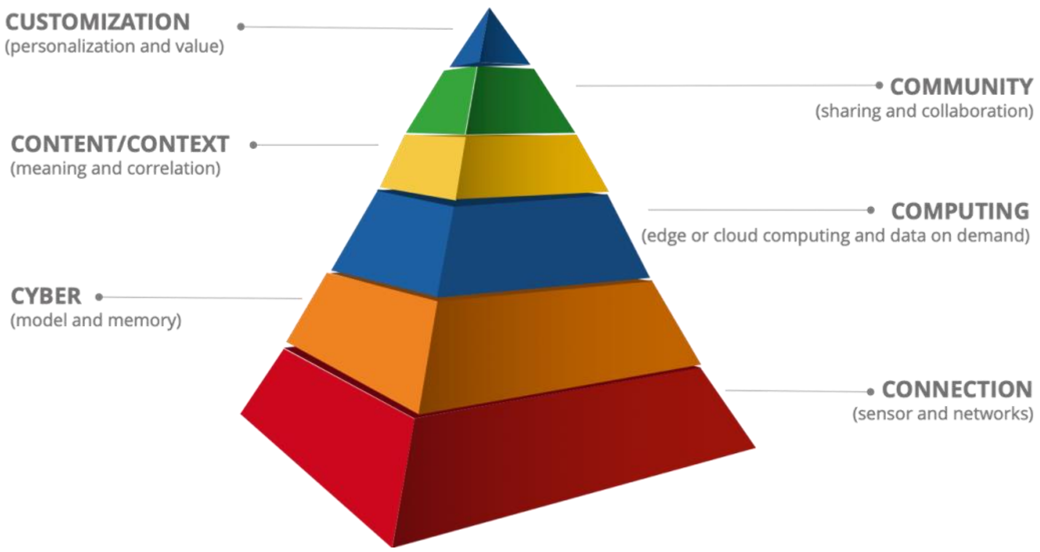

CUSTOMIZATION
(personalization and value)
COMMUNITY
(sharing and collaboration)
CONTENT/CONTEXT
(meaning and correlation)
COMPUTING
(edge or cloud computing and data on demand)
CYBER
(model and memory)
CONNECTION
(sensor and networks)

Figure 14: 6C Architectural Model

- **Connection**: it refers to the process of collection of the data from the sensors, IoT devices or already existing IT systems.
- **Cyber**: it refers to the conversion mechanisms adopted to transform the raw data into usable information.
- **Computing**: it refers to the storage and usage of the data at edge or cloud level.
- **Content/context**: it refers to the process of correlating different types of data in order to extract more detailed and sophisticated information with additional business value.
- **Community**: it refers to the presentation and sharing of the data (or the enriched information created at lower level) among different stakeholders.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Customization**: it refers to the further elaboration of the available data and information to create complex applications able to create business value.

Following these different levels of data enrichment, the OPEN DEI RAF has been eventually designed around the concept of Data Spaces using three layers differentiated on the basis of where and what data processing actions are performed (Figure 15).

- **Field Level Data Spaces**: they operate at field level and they involve closed-loop interactions within IoT systems, automation systems and human-supervised systems.
- **Edge Level Data Spaces**: they operate in the edge cloud using data generated at field level and they include data processing and data brokering, as well as a first level of data analytics.
- **Cloud Level Data Spaces**: they operate at cloud level and they involve a variety of more complex data operations aimed at enriching the information associated to the retrieved data, also via the integration with more complex systems, the correlation with other datasets, and the execution of advanced reasoning and data analytics.

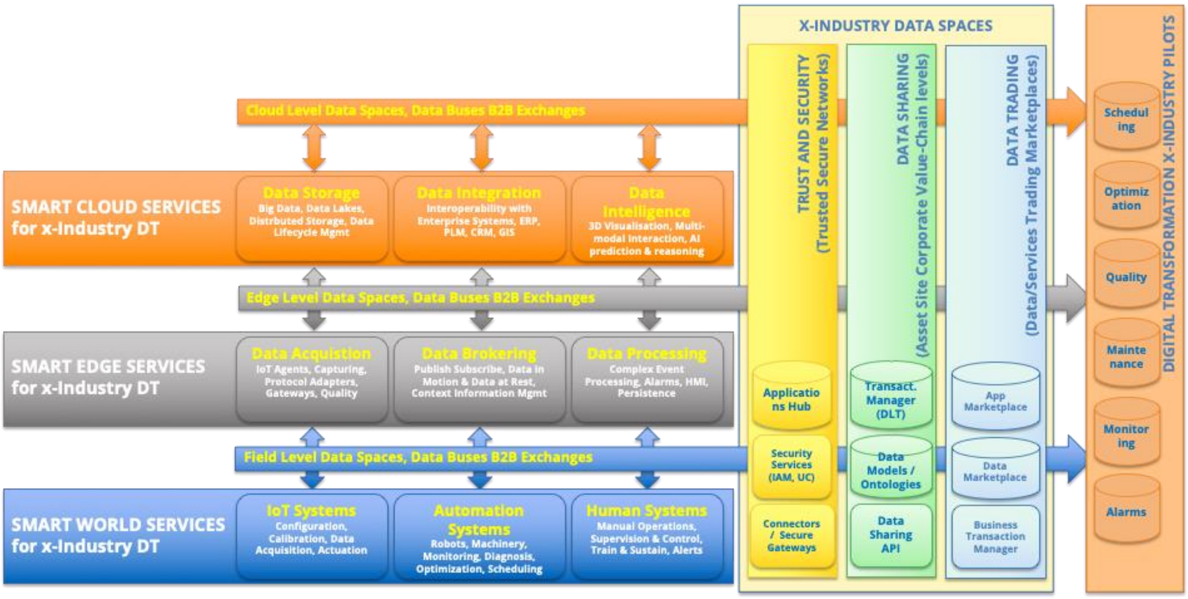

X-INDUSTRY DATA SPACES
Cloud Level Data Spaces, Data Buses B2B Exchanges
SMART CLOUD SERVICES for x-Industry DT
Data Storage
Big Data, Data Lakes, Distrbuted Storage, Data Lifecycle Mgmt
Data Integration
Interoperability with Enterprise Systems, ERP, PLM, CRM, GIS
Data Intelligence
3D Visualisation, Multi-modal Interaction, AI prediction & reasoning
Edge Level Data Spaces, Data Buses B2B Exchanges
SMART EDGE SERVICES for x-Industry DT
Data Acquistion
IoT Agents, Capturing, Protocol Adapters, Gateways, Quality
Data Brokering
Publish Subscribe, Data in Motion & Data at Rest, Context Information Mgmt
Data Processing
Complex Event Processing, Alarms, HMI, Persistence
Field Level Data Spaces, Data Buses B2B Exchanges
SMART WORLD SERVICES for x-Industry DT
IoT Systems
Configuration, Calibration, Data Acquisition, Actuation
Automation Systems
Robots, Machinery, Monitoring, Diagnosis, Optimization, Scheduling
Human Systems
Manual Operations, Supervision & Control, Train & Sustain, Alerts
TRUST AND SECURITY (Trusted Secure Networks)
Applicatio ns Hub
Security Services (IAM, UC)
Connectors / Secure Gateways
DATA SHARING (Asset Site Corporate Value-Chain levels)
Transact. Manager (DLT)
Data Models / Ontologies
Data Sharing API
DATA TRADING (Data/Services Trading Marketplaces)
App Marketplace
Data Marketplace
Business Transaction Manager
DIGITAL TRANSFORMATION X-INDUSTRY PILOTS
Schedul ing
Optimiz ation
Quality
Mainte nance
Monitor ing
Alarms

Figure 15: OPEN DEI Reference Architecture Framework

As shown in Figure 15, all the three data space layers have a set of vertical, cross-cutting concerns, which are summarized as follows:

- **Trust and security**: it refers to all the mechanisms to be adopted in order to enforce access/control usage policies, identity access management, trusted communication among stakeholders, etc.;
- **Data sharing**: it refers to all the components or infrastructures to be put in place in order to have effective and auditable data sharing, including logging of data sharing transactions, data harmonization according to standardized data models, exposure of data sharing APIs.
- **Data trading**: it refers to the frameworks and infrastructures needed to allow the monetization of the data-related transactions, here including both a data and an applications' marketplace.

3 MATRYCS Stakeholders and Requirements Analysis

This Section aims at determining the requirements and design principles to be followed for the definition of the MATRYCS Reference Architecture. These are identified taking into account: i) already existing works about the design of Reference Architectures (presented in Section 2), since they contain aspects that have general relevance, regardless of the specific field or domain of application; ii) the MATRYCS use cases (presented in the MATRYCS Deliverable D2.1) together with the associated specifications and requirements (presented in the MATRYCS Deliverable D2.2), since they may highlight needs or aspects specific to the building sector, thus being relevant for the discussion of building data spaces and for the definition of a Reference Architecture tailored to the building domain.

Keeping this twofold view in mind, in the following, this Section will first present the roles of the different stakeholders involved in the process of value creation for the building data, with the goal of identifying specific technical or business needs that should be considered in the Reference Architecture definition. Then, the existing functional and non-functional requirements will be analysed, also based on the considerations derived from the use cases proposed in the MATRYCS Large Scale Pilots. Finally, some more general design principles and recommendations will be discussed, which are at the basis of the definition of the MATRYCS Reference Architecture.

3.1 MATRYCS stakeholder roles

Data are at the heart of the MATRYCS Reference Architecture. The goal of the MATRYCS Architecture is to define the framework under which building data can be efficiently and effectively exploited to create the business value they intrinsically have. Similar to other domains, building data may undergo a series of steps, which allow incrementally generating new value and extracting additional insights about the associated system [13]. This series of steps is commonly referred to as the big data value chain. Multiple views of the big data value chain can be typically found. Figure 16 shows one of these possible views, which highlights how the data are progressively transformed first into information and then into knowledge. A set of different methods, tools, services are clearly needed to carry out such data transformation process.

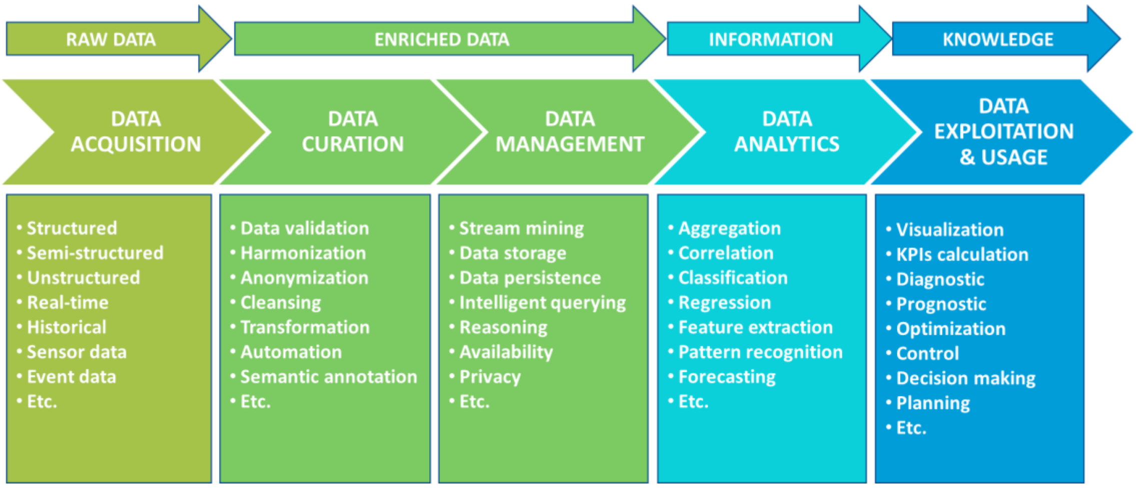


Figure 16: Big Data Value Chain

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

The value created around data can be typically associated to business value. The need of a multitude of steps and components to create this business value automatically implies also that different stakeholders may play a role in this scenario, at each step of the big data value chain. The MATRYCS Reference Architecture defines and classifies different stakeholders' roles according to the position they have in the data value creation process (at any of the steps of the big data value chain). Figure 17 shows a schematic view of the interactions among them during a generic process of data value creation.

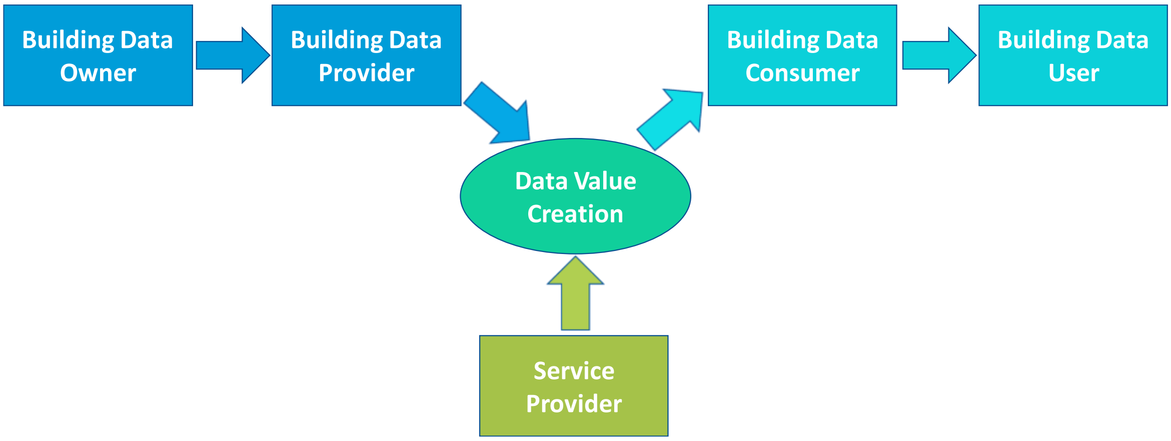


Figure 17: Stakeholders interactions in the MATRYCS Reference Architecture

Stakeholders directly participating in the big data value chain may take one or more of the following roles in the building data ecosystem.

- **Building Data Owners (BDOs)**: are the entities that have the ownership of the data. Specifying the legal terms under which the ownership of data is given is out of the scopes of this document. BDOs have the rights to execute full control over them, deciding the policies and conditions under which these data may be shared with other stakeholders. Example of BDOs can be the owners of the building itself (e.g., when the data describe the characteristics of the building), governmental institutions, but also companies creating complex data information via the processing of other shared data, etc.
- **Building Data Providers (BDPs)**: are the entities that share the data. Most of the times BDPs coincide with the BDOs. However, there are also cases where BDPs and BDOs are actually distinct entities. An example can be when a set of building data is shared via a dedicated platform. When BDPs and BDOs are different entities, BDPs must have the authorization to share the data.
- **Building Data Consumers (BDCs)**: are the entities that receive the shared data. The data is generally received through a service, that can be a message broker, a visualization tool, a database or an application that processes the input data to reshape it in a transformed data output, thus creating in this way an added value for the shared data.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Building Data Users (BDUs)**: are the final recipients of the output data received through the data transaction. Sometimes BDUs can coincide with the BDCs, but, in general, they are distinct entities. The BDU has the rights to use the received data and thus to exploit its value, for example from a business perspective. In the building domain, examples of final users of the enhanced building data are national and local governments, network operators, energy service companies, building managers, construction and renovation companies, etc.

In addition to the above categories, stakeholders essential for the creation of the value in the shared data are the **Service Providers (SPs)**. The category of SPs refers, in abstract terms, to any participant of the building data space that offers a service (it may be in the form of an algorithm, a software, a tool, an infrastructure or other) that can be used to perform an operation on the data (interpret, understand, process, manage, store, etc.). Depending on the specific type of service provided, it is possible to further differentiate the Service Providers into:

- **Technology Providers (TPs)**: are those entities providing services for the acquisition, curation and management of the data. This category of services generally allows transforming the raw data into useful (and usable) data within the Big Data ecosystem.
- **Data Analytics Providers (DAPs)**: are those entities providing AI, ML and DL models for the transformation of the data into richer and more complex information. These models or tools may be potentially used by other entities to create more sophisticated services.
- **Building Service Providers (BSPs)**: are those entities providing algorithms, tools, or other software packages for running building domain operations over sets of data, hence transforming the available information into knowledge that may be valuable for both operational and business purposes.

It is worth noting that in addition to the described stakeholders, which play a direct role in the creation of the data value, additional stakeholders playing an indirect role can also be present. Indirect stakeholders can be defined as those stakeholders that do not participate actively in the big data value chain and in the associated data space, but that may get benefits (or, vice versa, be negatively affected) by the created ecosystem. An example of indirect stakeholders is given by the individual citizens and the society as a whole. Even if these stakeholders do not belong to the business domain of the considered Big Data Architecture, they may still bring some specifications and requirements that should be considered in the architectural models' definition. In this version of the document, however, these stakeholders are not taken into account for the moment and only direct stakeholders are considered for the identification of the stakeholders' needs that should be taken into account in the MATRYCS Reference Architecture.

Considering the role of the different stakeholders, the following (non-exhaustive) list of potential stakeholders' needs can be identified:

- **Sovereignty**: the IDSA defines data sovereignty as a natural person's or corporate entity's capability of being entirely self-determined with regard to its data [10]. In other words, sovereignty concerns the rights of the data owner to decide the policies, conditions, restrictions under which its data can be used. Having mechanisms that guarantee keeping the sovereignty over data is a key aspect for BDOs to agree on sharing their data and unlock their availability for other stakeholders. The designed architecture framework should therefore include mechanisms, methods or tools to ensure data sovereignty for the BDOs.
- **Open API specifications**: APIs are essential for the integration of technologies, applications and services into platforms and systems. Having open API specifications on one hand facilitates the creation of the Big Data ecosystem, on the other hand assures that the services offered by SPs can be easily integrated in the system and used by BDCs. While this is not a constraint for the architecture definition, it is however a plus for the easy implementation and deployment of specific architecture representations.
- **Standardization**: closely related to the previous point, standardization is a key aspect to ensure interconnectivity and interoperability in the ecosystem. As indicated in [1], the standardization may concern both technologies and data. For SPs, the existence of official or de facto standards is key to offer services that can be easily integrated in the ecosystem.
- **No vendor lock-in**: having an open market with no vendor lock-in solutions is an important aspect for both SPs and BDCs. For SPs, this is important to create competitiveness and to open the possibility also to new SPs to enter the market and to bring innovative solutions. For BDCs, this is relevant due to the resulting chance to choose from a wide set of options that may differ in terms of quality, trustworthiness, pricing conditions, etc. The defined architecture has thus to foster the integration and interoperability of solutions coming from different vendors.
- **Trust**: it can be seen as a major need for BDCs, since the quality of the data they receive depends on the quality and integrity of the data provided by the BDP as well as on the quality and performance of the service employed during the data exchange. Consequently, trust is needed with respect to both data and services. The architecture solution should foresee the integration of technologies and methods able to guarantee trusted data exchange and should include tools and methods to assess the quality of data and applications (or, more in general, services).

3.2 MATRYCS requirements and specifications

This Section recalls the MATRYCS requirements and specifications extracted from the MATRYCS LSP use cases, as reported in the MATRYCS Deliverable D2.2 [14]. In [14], an in-depth analysis of the use cases was performed to understand the associated functional and non-functional requirements. The resulting requirements were then clustered according to their typology. The following Table presents the found categories of requirements and provides their underlying definition. These requirements are considered in this Deliverable in combination with other requirements derived from other Reference Architecture proposals that were deemed relevant also for the MATRYCS scenario. The merge of these requirements eventually provides the full list of macro-requirements that are considered in the design of the MATRYCS

Reference Architecture. Such general macro-requirements are further discussed in the following, indicating the associated implications they bring from an architectural perspective.

Table 1: List of MATRYCS functional and non-functional requirement categories

Requirement Category	Type	Description	Macro-requirement
Analytics	Functional	Need of advanced data analytics functionalities aimed at enhancing the information associated to the input data	Data analytics (Section 3.2.6)
Collection	Functional	Need of dedicated pre-processing routines aimed at cleaning, filtering or integrating the raw data provided by the physical system or by end-users	Streaming & batch data, Data management (Section 3.2.4 & 3.2.5)
Compatibility	Functional	Need of having interconnectivity and interoperability among specific technology components in the ecosystem	Interoperability (Section 3.2.1)
Integration	Functional	Need of having syntactic and semantic interoperability	Interoperability (Section 3.2.1)
Interaction	Functional	Need for end-users to freely query, visualize and interact with the ecosystem data they have access to	User-friendliness & interactivity (Section 3.2.7)
Notification	Functional	Need of specific functionalities aimed at notifying the end-used in case of occurrence of specific events	User-friendliness & interactivity (Section 3.2.7)
Reporting	Functional	Need of specific functionalities able to present in a clear and effective way the information extracted from sets of data	User-friendliness & interactivity (Section 3.2.7)
Saving	Functional	Need of specific functionalities aimed at saving the processed data and at allowing their possible query afterwards	Data management (Section 3.2.5)
Visualization	Functional	Need of specific functionalities aimed at presenting data or service results in an effective and attractive form	User-friendliness & interactivity (Section 3.2.7)
Modularity	Non-functional	Feature for which the components and modules of the overall ecosystem are able to	Scalability (Section 3.2.2)

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

		work independently	
Availability	Non-functional	Feature for which data, technologies and services are continuously available, with no (or minimal) downtime.	Security & Privacy (Section 3.2.8)
Efficiency	Non-functional	Feature for which the overall system or its components are able to perform complex operations in a relatively short time	Performance (Section 3.2.3)
Interoperability	Non-functional	Feature for which the system is able to successfully collect input data coming from the physical world or from other systems	Interoperability (Section 3.2.1)
Reliability	Non-functional	Feature for which the system is still able to provide its fundamental functionalities also in presence of unexpected errors or failures	Security & Privacy (Section 3.2.8)
Security	Non-functional	Feature for which the system is capable to prevent any unauthorized operation towards data and system components	Security & Privacy (Section 3.2.8)
Usability	Non-functional	Feature for which end-user can easily use the functionalities provided by the system	User-friendliness & interactivity (Section 3.2.7)

3.2.1 Interoperability

Guaranteeing interoperability is one of the major challenges in most of the industrial domains and the building domain does not represent an exception to it. The concept of interoperability concerns both the data and the software technologies employed in the building ecosystem, and it must be considered under multiple points of view.

A first view of the interoperability challenge is associated to the large **heterogeneity** of data that can be foreseen in input to the building ecosystem. This heterogeneity is also associated to the large variety of possible data sources. The building ecosystem will need to deal with diverse input data, such as Building Information Models (BIM), sensor data from Building Energy Management Systems (BEMS), cadastral data, Energy Performance Certificates (EPC), data from enterprise systems (e.g., working schedule, occupancy data, performance reporting), and so on. These data may differ due to a number of characteristics, such as their structure (structured, semi-structured or unstructured data), their time resolution, the used communication protocol, the data type (text, image, other), the adopted language (multilingualism), etc.

As a matter of fact, it is thus clear that to guarantee a seamless integration of the data in the system architecture, specific software components and technologies are necessary. These will be in charge to handle the data acquisition and to take care of the syntactic and **semantic description** of the received data. From this perspective, to enforce interoperability, it is important to make use, as far as possible, of standard data models and ontologies.

When looking at the interoperability challenge from a connectivity perspective, it is necessary to ensure interoperability towards the field devices as well as towards other IT systems, which may belong specifically to the building domain or not. Interoperability and interconnectivity with building-related IT systems is required because building data are generally decentralized and they can be offered by diverse IT systems. Interoperability and interconnectivity to the IT systems associated to other domains is instead necessary because services for the building domain in most of the cases still need additional data (such as electrical power consumption data, weather data, geospatial data, etc.) to complement the available information related to buildings. Platform-to-platform interoperability by means of suitable interfaces is thus required for offering, exchanging and consuming both domain and cross-domain data. The interoperability requirement from this point of view should be achieved by adopting dedicated APIs with **open specifications**.

Last but not least, interoperability is also required internally to same building IT systems in order to guarantee the interconnection of different technologies, components or services. Also in this case, interoperability should be enforced by using open APIs.

3.2.2 Scalability

The broad variety of data sources as well as the fine-grained resolution of time-series data associated to sensors clearly highlight the need of dealing with very large amounts of data, thus leading to a big data challenge. The architecture design has thus to duly consider the resulting requirements of scalability. Scalability is typically obtained either by scaling up, namely by adopting more performing hardware (vertical scaling), or by scaling out, i.e., by deploying additional resources operating in parallel (horizontal scaling).

To address the scalability challenge, a key aspect is to adopt technologies that can be deployed in a distributed fashion. Moreover, some architectural solutions may also help in pursuing scalability. As an example, adopting microservice-based architectures and last generation containerization principles can help in obtaining application scalability, since different services may be flexibly distributed. Cloud computing and virtualization approaches also support scalability, as they natively support horizontal and vertical scalability. Combining cloud and edge cloud computing could further help distributing the processing needs, hence unlocking higher scalability. In fact, the concept of edge cloud intrinsically implies that the processing tasks are carried out closer to the data sources. In some cases, beyond distributing the processing, this could also lead to avoid unnecessary data transmissions and archiving in the cloud, keeping therefore these operations only locally in the edge.

3.2.3 Performance

Complex services, and particularly those based on machine learning, may need to process very large amounts of data and may depend on extremely compute-intensive algorithms. At the same time, some of these services may also need very short response times in order to execute near-real-time control applications. In this scenario, it becomes important to look carefully at the performance of both the

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

underlying system and of the technologies adopted in the architecture. Efficient solutions to improve performance typically rely on the parallelization of the processing tasks. In scenarios where performance is a critical aspect, solutions based on High-Performance Computing (HPC) and/or on hybrid edge-HPC architectures should be considered to boost the performance.

3.2.4 Streaming & batch data processing

Building data can be either streaming data, i.e., time series data, sent periodically or not, that dynamically change over time (e.g., sensor data), or batch data, namely static data that do not change over time (e.g., BIM models). Different building services can clearly have the need to process either one or both of these types of data. The proposed architecture has thus to offer solutions to suitably handle and to foster the processing of both these types of data. It is also worth noting that often, services processing streaming data may also have near-real-time processing requirements. As a consequence, this aspect has to be additionally considered in the way streaming data are handled within the architecture.

To deal with these requirements, the Reference Architecture must support the use of both publish/subscribe and request/response communication mechanisms. The first one is often the preferred choice to handle streaming data and sporadic event notifications. The second one is instead more commonly adopted for pushing or pulling static data.

3.2.5 Data management

Data management is a functional requirement present for every big data architecture. In the case of buildings, particular attention has to be paid to the large heterogeneity of the associated data. The data management requirement entails the implementation and deployment of a number of methods, tools and technologies to carry our data curation, cleansing, harmonization, validation, annotation and storage. Through the data curation and management services it is possible to transform the raw data into enhanced data that can be then exposed for the upper-level applications and services.

3.2.6 Data analytics

In the building domain, applying advanced data analytics to extract information from the set of available data is a major requirement, as demonstrated by the large number of use cases and business cases that rely on AI and ML-based services. The Reference Architecture must thus be able to host these data analytics services and to facilitate their integration and use within the system. This requirement, therefore, does not only involve the implementation and deployment of a variety of AI and ML algorithms performing different tasks and providing different types of post-process information. It also includes the definition of the methods, technologies and software components that are necessary for training the analytics models, for their testing and validation, as well as for making them available in view of the creation of more complex building services.

3.2.7 User-friendliness & interactivity

Many of the services that can be conceived in the building domain are services having human operators actively involved in the loop and in the process of decision-making. From this standpoint, end-users would benefit from the availability of attractive graphical user interfaces, effective tools to visualize the data, intelligent query engines to look for business valuable information, etc. In this context, the presentation and visualization of the data assumes a relevant and strategic role, and it can be seen as one of the ways to add value to the available data and to create business around them. The Reference Architecture has thus to have a dedicated space to host the set of modules, tools and software components for the visualization, reporting and notification of data and information as well as for the interaction of the end-users with the available building services.

3.2.8 Security

The digitalization of systems and processes automatically poses challenges related to cybersecurity. Security is a cross-cutting concern that affects all the different layers of an architecture, as it involves data, technologies, and the system as a whole. In this context, it is possible to identify the following subgroups of security concerns:

- **Confidentiality**: it implies restrictions on the data sharing and that data can be accessed or correctly interpreted only by the data consumers authorized for it. Confidentiality is typically assured by using specific encryption methods.
- **Integrity**: the architecture must contain the security tools or methods needed by the data (or service) providers to ensure that the data (or service) they provide has not been changed, modified or manipulated by other parties that can have access to it. Security means to assure integrity may for example involve the use of cryptographic hash functions.
- **Availability**: data has to be accessible to authorized parties when needed and when legitimately demanded. This includes having dedicated functionalities in the system for Identity Management, Authentication and Access Control.
- **Reliability**: it is intended has the capability of a system to provide its fundamental functionalities also in presence of unexpected errors or failures. From an architectural perspective, this implies avoiding having single points of failure. Cloud-based systems natively support this feature.
- **Cyber-attacks processing**: as cybersecurity is a major challenge, adequate cyber countermeasures have to be put in place to discover and promptly handle possible cyber-attacks. A cyber countermeasure can be defined as any action, process or technology that allows preventing and mitigating the effects of a cyber-attack. The IT system has to be continuously updated to follow the best practices in this regard, which may include keeping track of attempted attacks, implementing suitable procedures to discover new cyber threats, as well as continuously monitoring the system to discover possible vulnerabilities.

3.2.9 Privacy

Close to the security aspects, privacy is another major requirement that has to be guaranteed by design within the IT systems. The recent General Data Protection Regulation (GDPR) introduced the principles of accountability and obligation of privacy by design. Accountability points to the responsibility for each party to demonstrate that they process personal information according to what expected following GDPR. Privacy obligations imply instead the adoption of adequate procedures to anonymize data and remove any information that could potentially lead to personal identification.

3.2.10 Marketplace

Since creating business value around data is the main goal of every big data architecture, having a marketplace that facilitates the buying and selling of data and services is a major requirement to bear in mind. The marketplace can potentially span over multiple cloud-based deployment models:

- **Data as a Service (DaaS)**: it refers to the possibility for stakeholders to get access and visualize data on demand;
- **Software as a Service (SaaS)**: it refers to possibility for stakeholders to use specific software (business applications) on-demand to process data;
- **Platform as a Service (PaaS)**: it refers to the possibility for stakeholders to use a remote computing platform to provision, instantiate, run and manage services and applications;
- **Infrastructure as a Service (IaaS)**: it refers to the possibility for stakeholders to use a remote virtualized infrastructure (managed with cloud orchestration) to provision processing, storage and other computing resources.

The functional domain of the architecture must foresee the presence of such a marketplace and, accordingly, of all the software components, tools and services required for handling the marketplace and for logging transactions, offering assets, exploring available resources, providing clearing mechanisms and billing functions, etc.

3.3 General architecture design principles

In addition to the above-mentioned requirements and stakeholders' needs, the definition of the MATRYCS Reference Architecture also considers some general design principles, some of which are derived from the recommendations given in the OPEN DEI project [12]. In particular, the following are taken into account as general principles for the definition of the Reference Architecture or as recommendations for its instantiation.

3.3.1 Modularity via microservices

Modularity is one of the main features the architecture must support, as it allows for easy integration, deployment, upgrade, replacement of a large number of services in order to flexibly accommodate the technical and business needs of different stakeholders. From a functional viewpoint, modularity is obtained by designing the architecture following a microservice-based approach. Microservices are defined as small and loosely coupled processes executed in parallel, each performing a well-defined task for the achievement of a specific business or technical goal [15].

From a technical perspective, using microservices brings benefits, among others, in terms of:

- Fully independent development of each microservice;
- Easier understanding and maintenance within the overall IT system;
- Easier load balancing supporting horizontal scalability;
- Possibility of polyglot software development.

Following these technical advantages, from a business perspective, a microservice-based architecture allows unlocking the following benefits:

- Competitiveness, due to the possibility of having solutions developed by different vendors addressing the same stakeholder functional/technical needs;
- Cost-efficiency, due to the unlocking of a multi-vendor ecosystem;
- Fast pace innovation, since the quality of the offered service would be one of the main criteria to choose a technological solution over a similar one;
- Openness to new service providers, as technological solutions are interchangeable and new service providers have the possibility to easily plug and test their solution in the real environment.

3.3.2 Cloud computing

Cloud Computing is the most common approach for developing, sharing and using applications between users. Resources are dynamically scaled according to the demand for the service, as it is the case for the electric network, where more or less power is provided according to the customer demand.

In general, some of the advantages of adopting cloud computing for a company are [16]:

- The offered web services enable companies to connect and make collaborations possible globally without setting up additional infrastructures like servers, datacentres and more;
- All applications, data, middleware, operating system, visualization, server, storage and network are managed by a third party, namely the provider of the cloud computing;
- Costs of hardware can be reduced. Indeed, organizations purchase IaaS resources on an as-needed basis, for as long as they need them, without long procurement and deployment cycles. In addition, hosting the required hardware and services in the cloud reduces the work required to host and maintain these services on-premises.

The services offered via cloud computing can be classified in:

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)

The IaaS is a model in which virtualized hardware resources are made available so that users can create and manage their own infrastructure on the cloud. It is the base layer for cloud computing and it basically involves the provision of virtual machines, servers, networks, load balancers. The IaaS cloud providers supply these resources on-demand.

PaaS provides computing platform which primarily includes resources like operating systems, programming language, database, web server that automatically scale to meet the application demands. In this case, the user can develop and run his own applications through the tools provided by the provider, who guarantees the proper functioning of the underlying platform. PaaS reduces the cost and complexity of application deployment by casting off the need to buy and manage the underlying hardware and software and to provision hosting capabilities.

SaaS is the top layer and most basic form of cloud computing. It is a model that encompasses applications and software systems, accessible from any type of device (computer, smartphone, tablet, etc.), through the simple use of a client interface. In this way, the user does not have to worry about managing the resources and infrastructure, as the provider controls them. In this layer, many customers can use a single application with customizable configuration. The advantage is that it requires no installation of the software and it can be accessed from anywhere with an internet connection

In the building domain, different stakeholders can have different needs or may be in the position to offer different services complementary to the Big Data Value Creation. Consequently, the MATRYCS Reference Architecture has to consider and must support all these options to offer diverse levels and types of services functional to the creation of innovation and business around data.

3.3.3 Data sharing towards other IT systems

The proposed architecture must allow sharing the data with other IT systems and data spaces. To this purpose, it needs to expose open APIs so that the other systems can effectively access the available data. The underlying principle is that bringing together data and information belonging to different domains is the key to develop more sophisticated and powerful services. In the same way building services may need data related to other domains (e.g., weather data, socio-economic data, etc.), other platforms (or data spaces) should be also put in the conditions to access and retrieve the available building data (under the conditions established by the relevant BDOs).

3.3.4 Openness

As indicated in [12], openness may lead to a set of competitive benefits in the development of the data economy. Openness must be intended at all levels, thus including open data, open APIs and open-source services. Open data are clearly strategic to create the business data value. Data should be made available to other stakeholders, either for free or under fair conditions. Openness in this context thus means offering the data, with as few restrictions as possible, and defining the policies to access and use them.

Open APIs are also key as they allow for easy integration of different services and technological solutions, also coming from different vendors. In the context of APIs, openness means that the related specifications are given as public and royalty-free. Having open APIs is a key aspect to remove technological barriers, to unlock fair participation to the digital market of data for different (including new) stakeholders, to enhance quality via shared development and to bring the experience associated to the integration with different software. Thanks to these set of advantages, open APIs also have better chance to be stable and to evolve in de facto standards.

Open-source software/technologies represent the last level of openness in the system. Open source in this context is intended as full visibility of the source code. Similar to the open APIs, development of open-source services bring benefits, among others, in terms of stability and reliability (since developers of multiple stakeholders may use such code as basis for their implementations), faster time to market (thanks to the shorter time for development and testing), better security (since the community using this software may discover possible vulnerabilities), faster innovation pace and capability of promptly address new business needs. The availability of open-source technologies can be a big advantage not only for other stakeholders gaining the possibility to use free software, but also for the enterprises that launch it, since this can represent a way to progress quickly and a competitive advantage in the process of creating innovation.

3.3.5 No vendor lock-in

The underlying principle behind the MATRYCS Reference Architecture is that it enables open and fair participation of any stakeholder to the process of data value creation. Consequently, no vendor lock-in solutions have to be present, for any of the software stacks and for any of the functionalities required to fulfil the presented functional and non-functional requirements. As already described in this Section, some of the architectural solutions may support this principle and foster competitiveness through the development of diverse vendor solutions for addressing a same technical/business need.

3.3.6 Support data economy

The main goal of defining a big data architecture is to allow unlocking the value (technical and business) around data, and thus to support the creation of digitalised market of data. Supporting the European data economy has a goal wider than supporting the European industry to create innovation and thus to prosper. It has also the objective to indirectly address the raising needs of individual citizens and society, which are indirect stakeholders of the big data value chain. Thanks to a well-established market of data, grounded on shared and accepted instantiations of higher-level Reference Architectures, it is possible to foster productivity, address technological and societal challenges, create wealthness and comfort, move towards energy efficiency and sustainability, etc., thus creating value for the entire society.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

4 MATRYCS Reference Architecture

The MATRYCS Reference Architecture aims at defining a unified framework for Big Data management in the building domain that allows the stakeholders operating in this sector (but not only) to create new business models and business opportunities relying on the value extracted from shared data. In particular, the MATRYCS Reference Architecture mostly focuses on the functional/implementation viewpoint of the architectural model and it describes the set of basic components needed for the creation of the overall building big data ecosystem and of a building sector-related data economy.

The MATRYCS Reference Architecture presented in the following represents the definition in use until M10 of the project. Modifications and integrations to such architecture definition could thus still occur during the continuation of the project. The final version of the MATRYCS Reference Architecture will be presented within Deliverable D2.4 at M18 (March 2022).

The MATRYCS Reference Architecture currently in use is composed of three software layers on top of the physical layer, which can be directly mapped to the different stages of the Big Data Value Chain (Figure 18). This architectural view reflects the goal of the Reference Architecture of providing a framework to support the creation of data value. In this context, the layers indicate a logical partition of the architecture and are not strictly representative of the communication flows in the architecture implementation. The layers are here intended in a software architecture sense; they represent a cohesive set of software components whose combined operation allows achieving a high-level functionality.

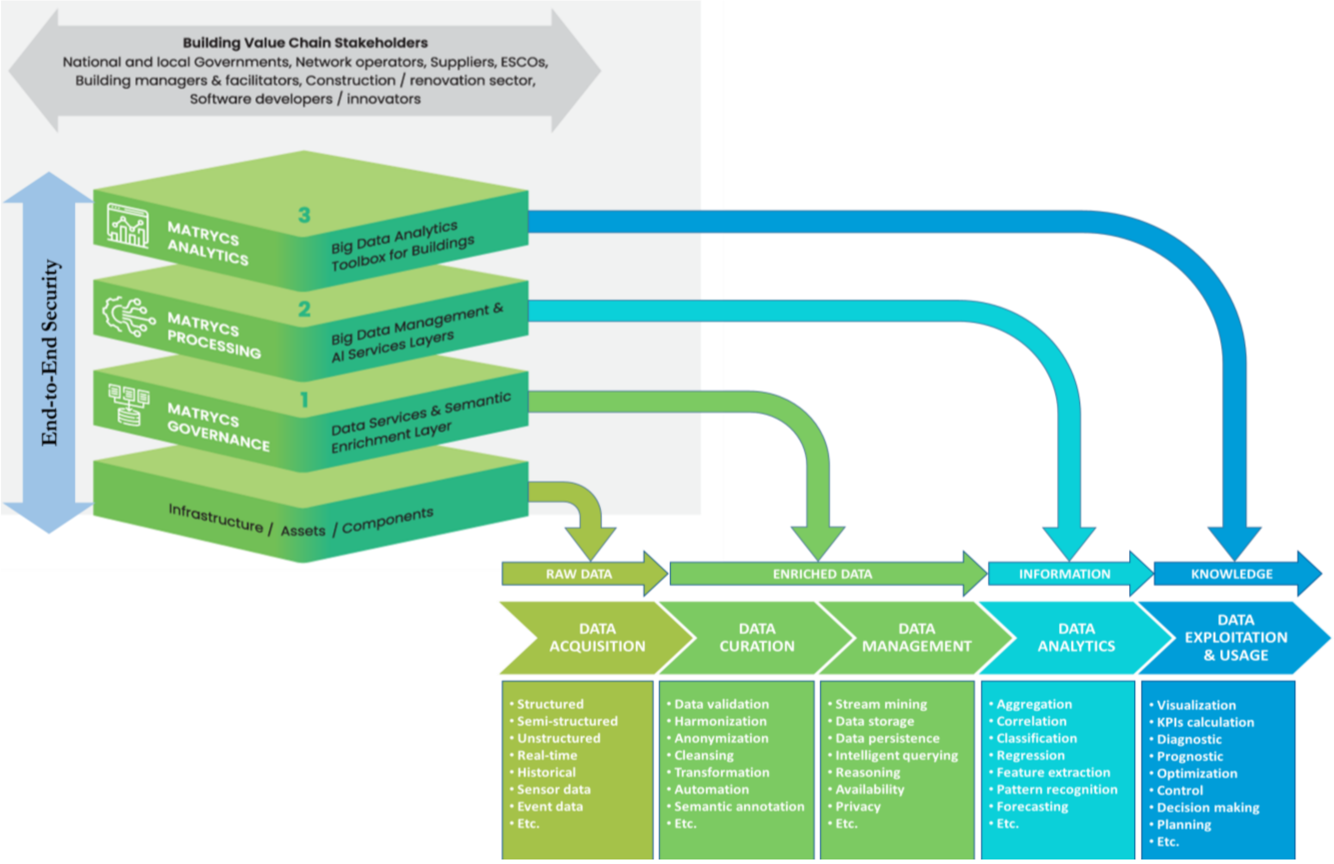


Figure 18: High-Level MATRYCS Architecture and mapping to Big Data Value Chain.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

The three software layers of the MATRYCS Reference Architecture are:

- **MATRYCS Governance**: it is composed of those services that realize the middleware needed for acquiring, managing and exposing the data. It includes the services required to guarantee data interoperability, cleaning, validation and storage, thus ensuring that collected raw data are converted into a usable form for the upper-level applications.
- **MATRYCS Processing**: it includes the components needed for the modelling, training, testing and validation of AI and ML based algorithms. It thus allows developing AI and ML models that are used to add value to the data in form of feature extraction, pattern recognition, event detection, forecasting, etc. These models are then exposed to the upper layer, which can employ one or more of them to design applications with complex logics.
- **MATRYCS Analytics**: it includes the set of services and tools offered to end-users for implementing complex building management applications. As the architecture aims at supporting end-users in the creation of innovation and business, the available services/tools are exposed through the so-called MATRYCS toolbox via different deployment options, which include DaaS, SaaS, PaaS and IaaS (see Section 3.2.10).

Transversally to all the layers, there is the **end-to-end security** framework, which is intended to implement, at all the levels of the architecture, the needed security functionalities for confidentiality, integrity, availability, privacy and, more in general, for cyber-attack prevention and mitigation.

A more detailed view of the MATRYCS Reference Architecture is shown in Figure 19. The following of this Section gives more details about the software components included at each layer of the architecture.

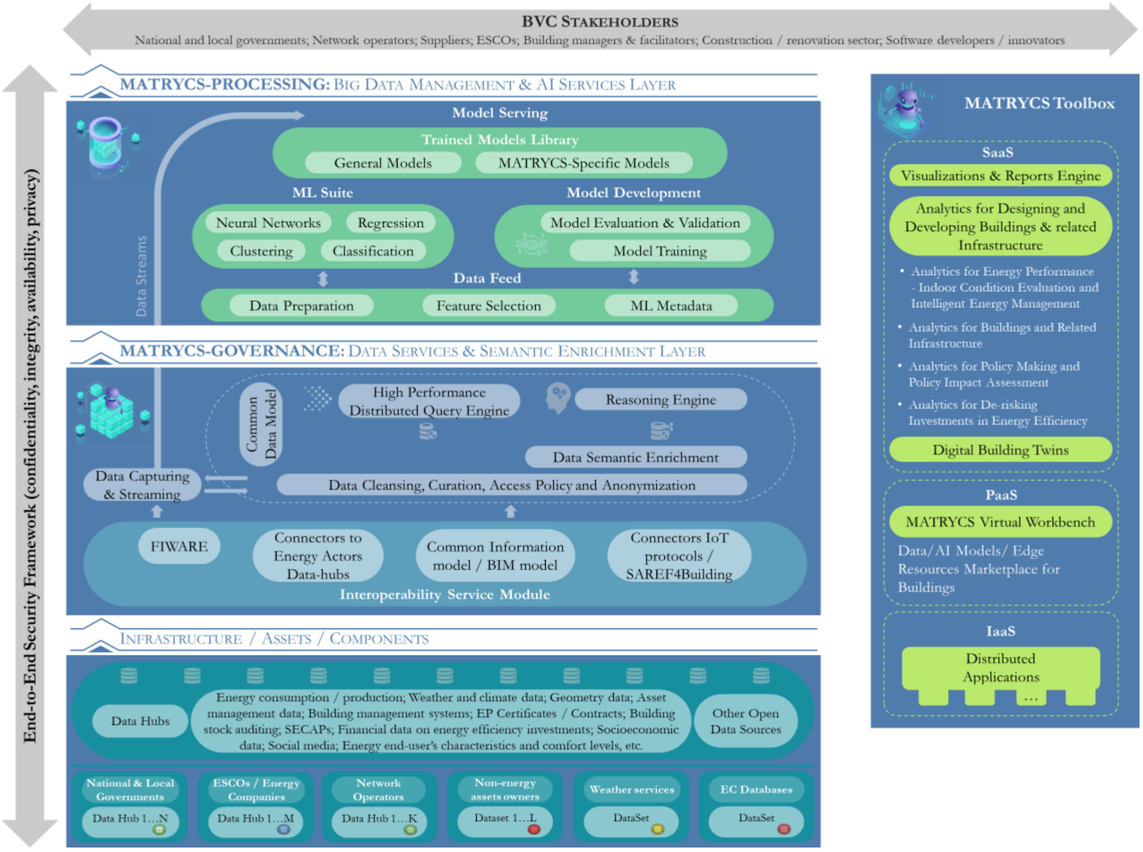

BVC STAKEHOLDERS
National and local governments; Network operators; Suppliers; ESCOs; Building managers & facilitators; Construction / renovation sector; Software developers / innovators
End-to-End Security Framework (confidentiality, integrity, availability, privacy)
MATRYCS-PROCESSING: BIG DATA MANAGEMENT & AI SERVICES LAYER
Model Serving
Trained Models Library
General Models
MATRYCS-Specific Models
ML Suite
Neural Networks
Regression
Clustering
Classification
Model Development
Model Evaluation & Validation
Model Training
Data Streams
Data Feed
Data Preparation
Feature Selection
ML Metadata
MATRYCS-GOVERNANCE: DATA SERVICES & SEMANTIC ENRICHMENT LAYER
Common Data Model
High Performance Distributed Query Engine
Reasoning Engine
Data Semantic Enrichment
Data Capturing & Streaming
Data Cleansing, Curation, Access Policy and Anonymization
FIWARE
Connectors to Energy Actors Data-hubs
Common Information model / BIM model
Connectors IoT protocols / SAREF4Building
Interoperability Service Module
INFRASTRUCTURE / ASSETS / COMPONENTS
Data Hubs
Energy consumption / production; Weather and climate data; Geometry data; Asset management data; Building management systems; EP Certificates / Contracts; Building stock auditing; SECAPs; Financial data on energy efficiency investments; Socioeconomic data; Social media; Energy end-user's characteristics and comfort levels, etc.
Other Open Data Sources
National & Local Governments
Data Hub 1…N
ESCOs / Energy Companies
Data Hub 1…M
Network Operators
Data Hub 1…K
Non-energy assets owners
Dataset 1…L
Weather services
DataSet
EC Databases
DataSet
MATRYCS Toolbox
SaaS
Visualizations & Reports Engine
Analytics for Designing and Developing Buildings & related Infrastructure
• Analytics for Energy Performance - Indoor Condition Evaluation and Intelligent Energy Management
• Analytics for Buildings and Related Infrastructure
• Analytics for Policy Making and Policy Impact Assessment
• Analytics for De-risking Investments in Energy Efficiency
Digital Building Twins
PaaS
MATRYCS Virtual Workbench
Data/AI Models/ Edge Resources Marketplace for Buildings
IaaS
Distributed Applications
…

Figure 19: Functional View of the MATRYCS Reference Architecture

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

4.1 MATRYCS Governance Layer

The main objective of the MATRYCS Governance Layer is to provide the necessary middleware (see Figure 20) to act as a mediator between MATRYCS data providers and MATRYCS data users (analysis tools and services). Adopting a holistic approach to data management, the MATRYCS Governance layer has to take into account different domain and non-domain data such as building data, energy data, sensors data, energy usage data (heating, lighting, cooling, air conditioning, ventilation), weather data, etc., and to provide interoperability across this heterogeneous set of data via harmonization services and semantic enrichment.

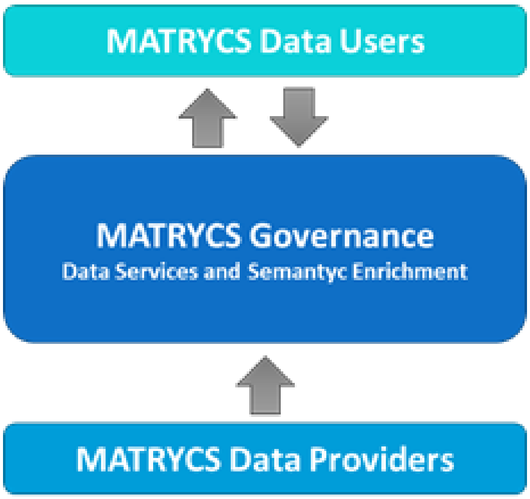


Figure 20: MATRYCS Governance overview

The MATRYCS Governance Layer provides data services that allow the integration, pre-processing, semantic annotation and querying of heterogeneous data, guarantying traceability, provenance tracking and accountability of the MATRYCS data. In this regard, the MATRYCS Governance architecture has been designed identifying the key high-level big data management phases needed to cover the whole MATRYCS Governance data pipeline processes (see Figure 21).

These processes include:

- **Data acquisition**: it is guaranteed via the *Interoperability Service Module*, which is designed to facilitate the data integration of heterogeneous data sources and/or platforms belonging to different MATRYCS actors. A *Streaming module* is in place to manage the data streaming to the MATRYCS Processing layer and for the in-memory processing of the low latency near real-time data.
- **Data Curation**: it is guaranteed via the *Data Pre-processing Module*, which is in charge of the data pre-processing activities for data cleansing, data curation, data anonymization and semantic annotation. This module also covers data modelling activities leveraging on pre-existing vocabularies and ontologies with the aim to define the MATRYCS Common Data Model.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- **Data storage**: it is guaranteed through two different modules: the *High Performance Query Engine* and the *Reasoning Engine*. The *High Performance Query Engine* builds on top of a NoSQL data storage and has the aim to perform complex queries in very efficient and highly scalable way. The *Reasoning Engine* is able to persist semantic datasets and any Resource Description Framework (RDF) information produced by the *Pre-processing & Semantic Enrichment Module*. Moreover, it can also consume JSON data and be used to construct the graph entities and relationships. Both modules expose intelligent querying systems and APIs to be used both in the MATRYCS Processing and in the MATRYCS Analytics layers.

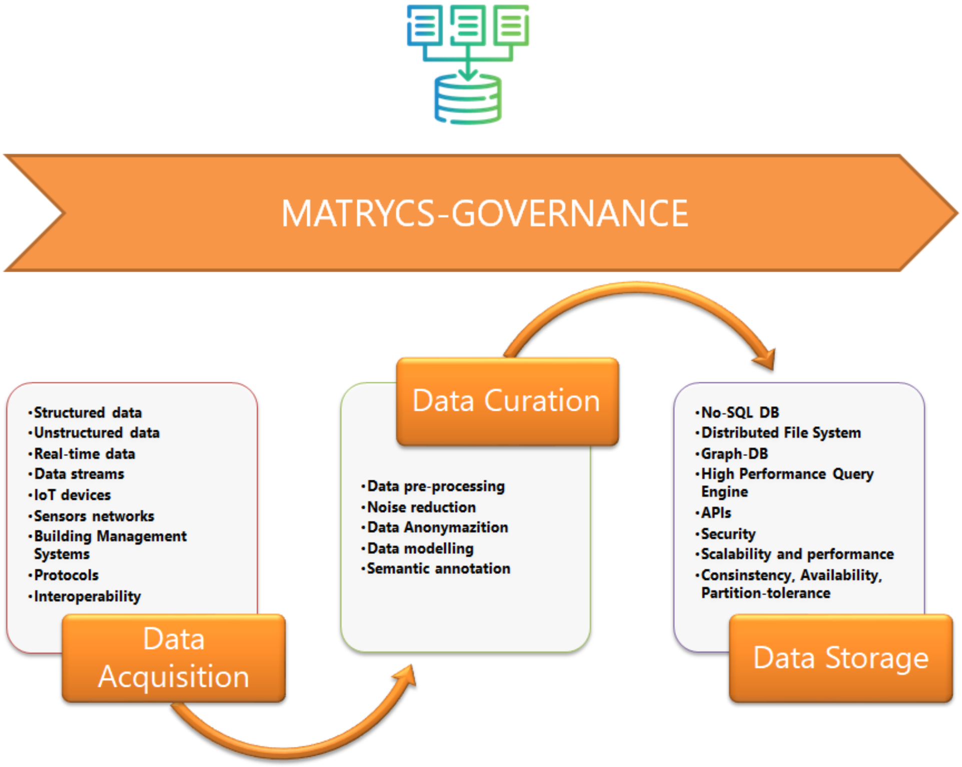


Figure 21: Big Data Management Phases in the MATRYCS Governance Layer

Figure 22 shows the overall conceptual architecture of the MATRYCS Governance Layer, which includes the aforementioned Governance components. More details about the MATRYCS Governance Layer with examples of the technologies that can be adopted for its instantiation can be found in the MATRYCS Deliverable D3.1 (released in M11, i.e., August 2021).

4.1.1 Interoperability Module

The *Interoperability Service Module* (see Figure 23) is in charge of integrating and sharing of the heterogeneous data belonging to the different MATRYCS Data Providers, as well as external services, in an interoperable manner. This accounts for the fact that data ingested via the Interoperability Module can potentially come from a variety of data sources, such as Building Management Systems, Building Information Models, third-party platforms, sensors, IoT devices, open data repositories, etc.

This component shall include microservices (data connectors) to handle the different data sources provided with different data format and different communication protocols (e.g., Rest APIs, SFTP, IoT protocols, Sensor Network) providing also interfaces to other third-party systems and external datasets/platforms. This module can cover also data exchange mechanisms at edge level, with the aim to offer both local computational power and storage capacity to services and functions that may need ultra-low latency, as well as local processing without leveraging on core cloud remote services.

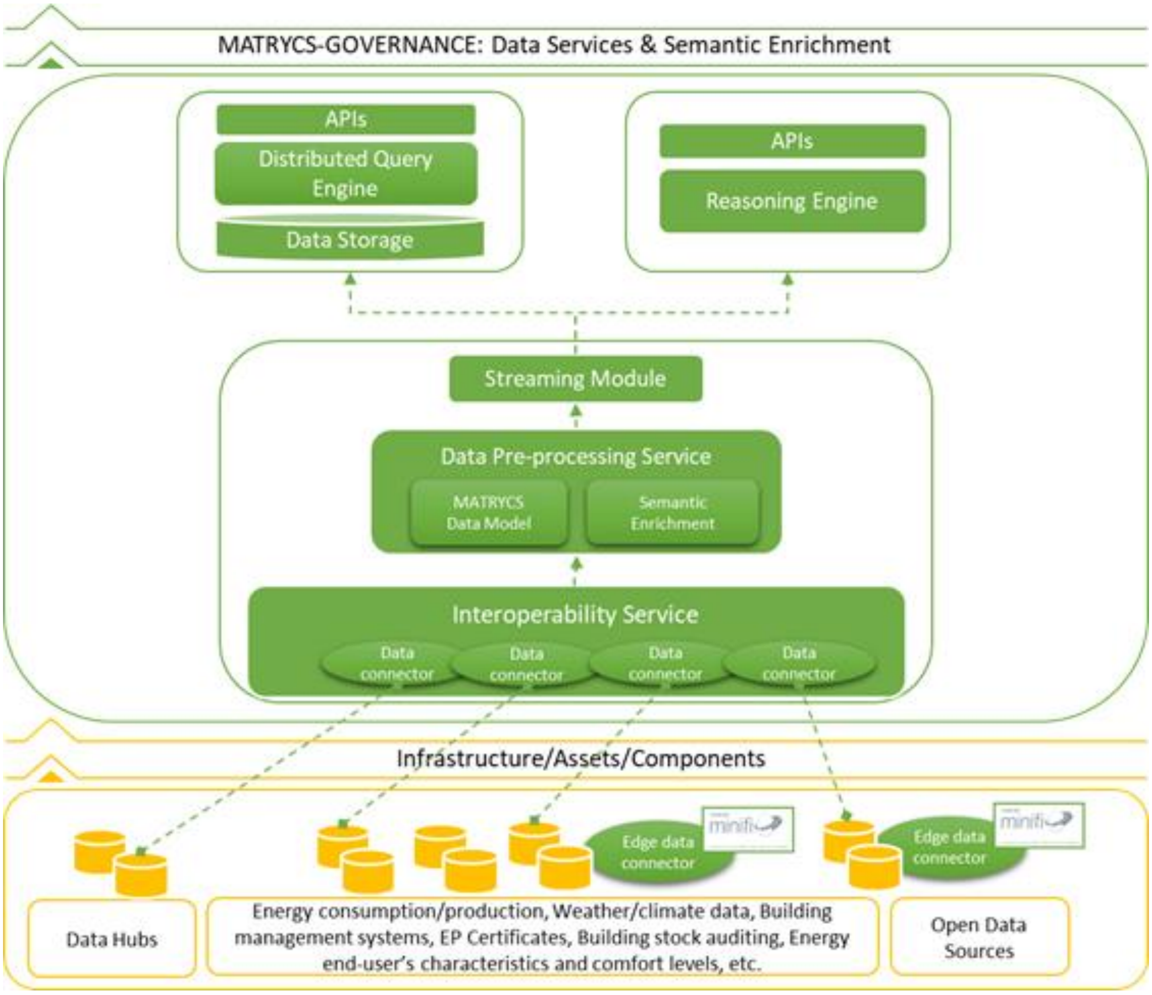

MATRYCS-GOVERNANCE: Data Services & Semantic Enrichment
APIs
Distributed Query Engine
Data Storage
APIs
Reasoning Engine
Streaming Module
Data Pre-processing Service
MATRYCS Data Model
Semantic Enrichment
Interoperability Service
Data connector
Data connector
Data connector
Data connector
Infrastructure/Assets/Components
Edge data connector
minifi
Edge data connector
minifi
Data Hubs
Energy consumption/production, Weather/climate data, Building management systems, EP Certificates, Building stock auditing, Energy end-user's characteristics and comfort levels, etc.
Open Data Sources

Figure 22: MATRYCS Governance conceptual architecture

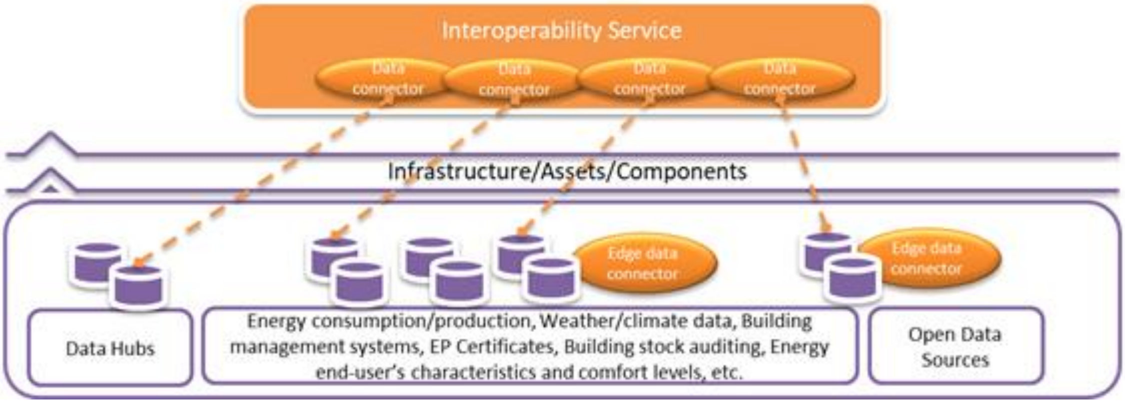

Interoperability Service
Data connector
Data connector
Data connector
Data connector
Infrastructure/Assets/Components
Edge data connector
Edge data connector
Data Hubs
Energy consumption/production, Weather/climate data, Building management systems, EP Certificates, Building stock auditing, Energy end-user's characteristics and comfort levels, etc.
Open Data Sources

Figure 23: Interoperability Service Module conceptual architecture

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

4.1.2 Data Pre-processing Module

The *Data Pre-processing Module* is composed of two main sub-components, namely the *Data Cleansing & Anonymization Module* and the *Semantic Enrichment Module*.

The goal of the *Data Cleansing & Anonymization Module* is to enhance the quality of the data and to guarantee privacy protection**.** To improve the data quality, dedicated functions perform data cleansing, which include operations such as data restructuring, predefined value substitution, noise reduction, outlier detection, and data inconsistencies handling. All data are then transformed into the MATRYCS Data Model, which ensures all the data processed by the system adhere to the predefined standardized vocabularies and data models. In addition, to protect privacy, sensitive information is detected and anonymized in this module. The *Data Cleansing & Anonymization Module* is connected to the *Data Steaming Module*, which manages the data steaming towards different consumers.

The goal of the *Semantic Enrichment Module* is to overcome the heterogeneity of data and to guarantee semantic interoperability. The semantic enrichment process relies upon predefined ontologies, which cover endogenous buildings technical measurements as well as other context data, such as weather, geographical information, or energy networks data. Ontologies contain the vocabulary and the set of possible relationships, which follow well-established standards. Based on those vocabulary and relationships, it is possible to guarantee the semantic interoperability needed at application level. The *Semantic Enrichment Module* publishes the semantically annotated data through the *Data Steaming Module*.

4.1.3 Data Streaming Module

The Data Streaming Module (see Figure 24) is a key component within the MATRYCS Governance Layer, as it allows re-routing the data to the different software components. It is in charge of managing the stream of the MATRYCS data, handling dynamically the frequency rate of the data streaming for the subsequent in-memory processing of the low latency near real time data.

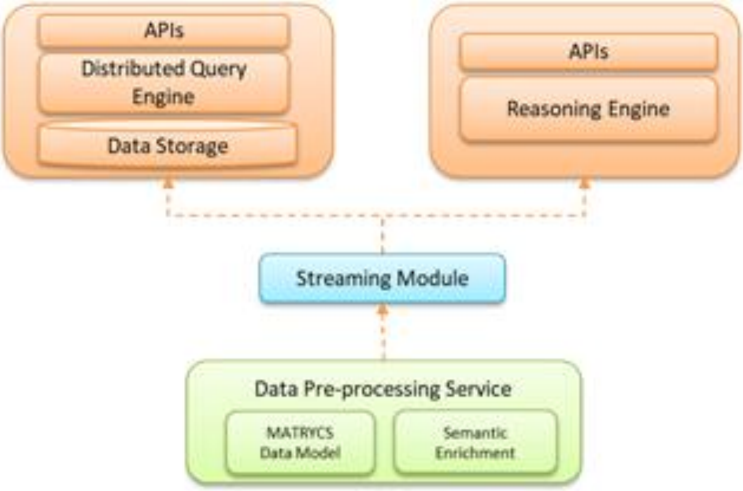


Figure 24: Data Streaming Module view

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

4.1.4 Data Storage & Query Engine

In the MATRYCS governance layer, the persistent/temporary local storage of big datasets is enabled by the *Data Storage Module*. It consists of a distributed storage layer, encompassing both edge and cloud infrastructure, which provides efficient and indexed NoSQL storage capacities. Based on persistence longevity, data size, data structure and availability constraints, the data storage module is split into three components: temporary storage (also staging area), structured/semi-structured data storage and networked/cloud file storage. For integration purposes, the data storage module exposes a push/pull interface via a dedicated data storage service. Additionally, a predefined access to the temporary storage is available.

Built on top of the data storage module, the *High Performance Distributed Query Engine* enables running complex analytic queries in real-time. The query engine minimizes network usage and enforces data privacy while providing a high-performance data retrieval and data processing to other MATRYCS Governance components. Queries may be executed using a REST API or via a graphical user interface. Direct interaction and usage via temporary data storage is also possible.

4.1.5 Reasoning Engine

The *Reasoning Engine* is a component that provides intelligent querying, insights and search capabilities by leveraging the available knowledge for digital twins and other building analytics services. Furthermore, it provides functionalities for adding more information to the existing knowledge base exploiting metadata and relationships among data. The Reasoning Engine must be capable of consuming data that are received, via the *Streaming Module*, from the *Semantic Enrichment Module*. These data could be expressed with different data models, including metadata data models (such as RDF) and when received they are persisted to a *Reasoning Engine*'s graph database. The graph database is a powerful inference engine that enables graph functionalities over entities and allows connections for extracting new insights and patterns from available datasets.

4.1.6 Blockchain/Distributed Ledger system

The goal of this module is the securitization of data storage, specifically the primary purpose of blockchain technologies is to remove the need for intermediaries and replace them with a distributed network of digital users who work in partnership to verify transactions and safeguard the integrity of the ledger. Use of blockchain in the data-sharing model has these main key features: it ensures the traceability and data storage throughout a decentralised and distributed system, with which it will be possible to implement a secure way to track changes in information over time. In this way, the module can warrant the creation of trust among untrusted participants; in addition, the absence of intermediaries can promote a more transparent data sharing. In order to implement these functionalities this module will use the blockchain. The Blockchain is a distributed ledger, based on a shared and distributed database, containing a log of transactions in chronological order. Transactions are grouped into blocks and chained through cryptographic hashes into an ongoing chain of hash-based proof-of-work, forming a record that cannot be changed without redoing the proof-of-work.

Within the MATRYCS Governance Layer, this module interacts with the Data Storage for the extraction of harmonized data and with the Blockchain network (Ethereum) for the physical fingerprints (hash codes) storage in the chain blocks to assure data immutability. This will be the main mechanism to assure immutability of data stored into the Governance Data Layer; optionally, the module could also interact directly with the MATRYCS assets, and external data providers; in this case, the interaction will require specific data connectors to integrate the datasets with different standard and access interfaces.

4.2 MATRYCS Processing Layer

The MATRYCS Processing Layer is responsible for advanced data processing via AI-based services. It aims at encapsulating the intelligent building blocks of the MATRYCS solution by providing a library of reusable ML/DL models that are made available with a view to promote quick adaptation and reuse of ML models along different contexts. Figure 25 presents a graphical representation of the MATRYCS Processing Layer with its major components and their interdependences and connections.

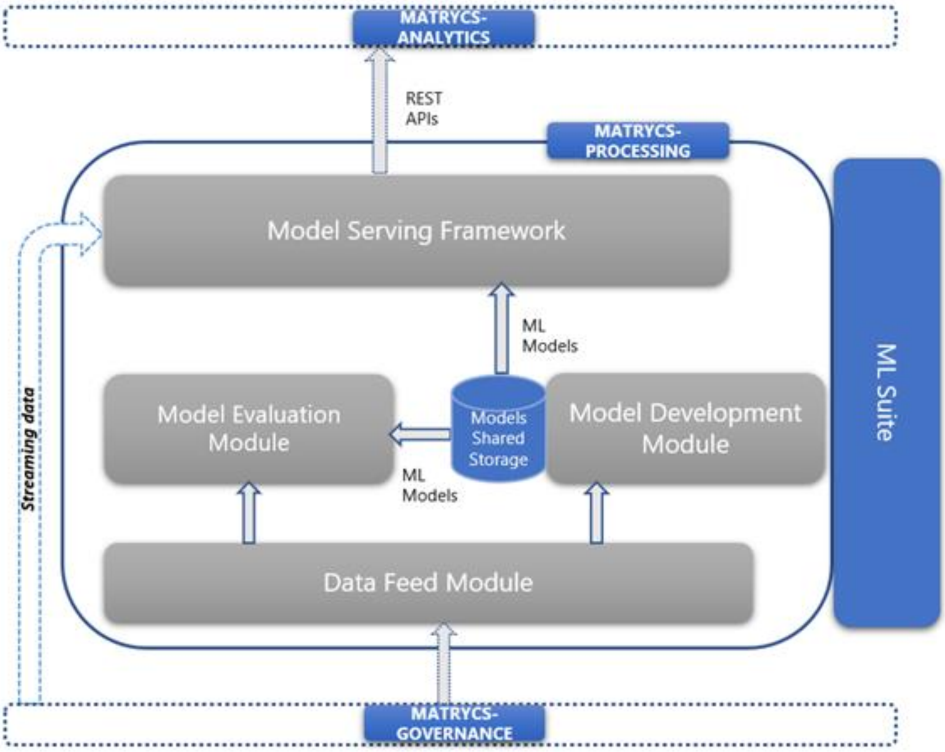


Figure 25: MATRYCS Processing conceptual architecture

The MATRYCS Processing Layer is connected to the MATRYCS Governance Layer through the *Data Feed Module* and the *Model Serving Framework*. The *Data Feed Module* is the mediator that takes the data from the MATRYCS Governance Layer and makes them available into the MATRYCS Processing Layer for performing the data preparation, the model training and the final evaluation of the AI-based services. Models successfully trained and validated are finally made available through the *Model Serving Framework*. The *Model Serving Framework* is the second connection point to the MATRYCS Governance Layer and it serves to receive the streaming data from the *Streaming Module* for providing advanced analytics information based on the results of the AI-based processing (such as forecasts, regressions, feature extraction, etc.). The different components of the MATRYCS Processing Layer modules are further described below. More details about the components in the MATRYCS Processing Layer with examples

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

of the technologies that can be used for its instantiation can be found in the MATRYCS Deliverable D4.1 (released in M11, i.e., August 2021).

4.2.1 Data Feed Module

The *Data Feed Module* works as the main connection between the MATRYCS Governance Layer and the MATRYCS Processing Layer. It retrieves the data stored in the MATRYCS Governance *Data Storage* and makes them available in the MATRYCS Processing Layer for the subsequent training of the AI models. To this purpose, at first, a series of preparation steps over the data is conducted, such as drop-missing values, data normalization and encoding of categorical variables. After the data preparation, the data are stored into the MATRYCS *Data Storage*. The other modules of the MATRYCS Process Layer can gain access to *Data Feed Module*'s output by using either the *Data Storage* or the *Data Feed Module*'s API.

4.2.2 Machine Learning Suite

The *Machine Learning Suite* component contains a catalogue of state-of-the-art technologies and software libraries (i.e., Scikit-Learn, Keras, Tensorflow) that can be utilised to train the ML/DL models. More specifically, this component is designed to be an enriched software library that can be used for the purposes of the AI-model training. The ML suite may include, for example, neural networks, and other ML/DL-based services for classification, reasoning, forecasting and regression analysis.

4.2.3 Model Development Module

The *Model Development Module* exploits the *ML Suite* for training the data driven algorithms using the output created by the *Data Feed Module*. At the end of the training process, the output of this module is a trained model that should be stored in a file format into a dedicated *Models' Shared Storage*. This storage will host all the trained models created in the MATRYCS Processing Layer and will serve as database for them.

4.2.4 Model Evaluation Framework

The *Model Evaluation Framework* is used for evaluating the trained models created by the *Model Development Module*. Therefore, it is interconnected with the *Models' Shared Storage* for getting the trained models and with the *Data Feed Module* for getting the dataset to be used for the evaluation purposes. These two inputs are combined for testing and evaluating specific metrics representative of the quality of the trained models. The assessment (and possible validation) of the trained models canbe done exploiting metrics commonly adopted in the AI and ML/DL domain, such as the R^2 score, Mean Squared Error and Mean Absolute Error for forecasting and regression analysis, or the F1 Score, log loss/binary cross-entropy and categorical cross-entropy for classification problems. Using these metrics, the quality of the produced AI models can be ensured, therefore increasing the trustworthiness about the information created via such models.

4.2.5 Model Serving Framework

The *Model Serving Framework* is the bridge between the MATRYCS Processing and the MATRYCS Analytics layers. It serves the ML/DL models saved under the trained models library (hence those models

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

already trained by the ML developers and subsequently evaluated and validated via the *Model Evaluation Framework*), making them available for potential use within more complex applications implemented in the MATRYCS Analytics Layer. Through the gathering of streaming data from the *Streaming Module* (in the MATRYCS Governance Layer), the *Model Serving Framework* allows running the trained models in near real-time and it can thus provide the results of the ML model to the applications and services running in the MATRYCS Analytics Layer.

4.3 MATRYCS Analytics Layer

The MATRYCS Analytics Layer contains all the services that will be exposed to the end users. These services can be clustered into the Visualisations and Reports Engine, the Analytics Building Services, the Building Digital Twin and the Virtual Workbench. They are offered to the end-users via the MATRYCS Toolbox, leveraging on the different options of delivery and deployment of the cloud-based services.

At the SaaS level, the MATRYCS Toolbox incorporates:

- The *Visualizations and Reports Engine*, responsible for the visual representation of the stored data and of the results produced by the analytical components. It can offer a variety of visual representations including charts, map visualizations and other;
- A range of innovative *Analytics Building Services* covering different scales of the building domain
- The *Building Digital Twin*.

At the PaaS level, the MATRYCS Toolbox incorporates the Virtual Workbench. This provides to the user an intuitive and easy GUI to create new services in the public sector and it integrates two modules:

- The *Building Data & Services Innovation Hub*: it provides SMEs, developers and potential innovators a graphical interface that facilitates the development and deployment of new applications on the base of the existing data sets and the available catalogue of developed and trained AI models;
- The *Data Analytics Environment*: it allows users (e.g., analysts) to perform freeform queries and data analytics on the assets which are accessible to the platform (models and data).

The IaaS is a layer in which virtualized hardware resources are made available so that the user can create and manage his own infrastructure on the cloud according to his needs, without worrying about where the resources are allocated.

In the following, the different functional components included in the MATRYCS Analytics Layer are further discussed. More details about these components can be found in the MATRYCS Deliverable D4.4 (released in M11, i.e., August 2021).

4.3.1 Visualization & Reporting Engine

The *Visualisations and Reports Engine* is the functional module responsible for creating advanced visualisations and reports on flexible dashboards over the stored data that are available to the end users. It supports a variety of visual representations including charts and map visualizations able to be customised upon the different needs. More specifically, by leveraging the functionalities of frontend

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

framework, the Visualisation engine is a user-friendly environment that offers flexible, fully extendable and reusable dashboards to the end users . In general, this solution is created and provided by deploying different microservices. Figure 26 shows a visual representation of the conceptual schema behind the *Visualisation & Reporting Engine*.

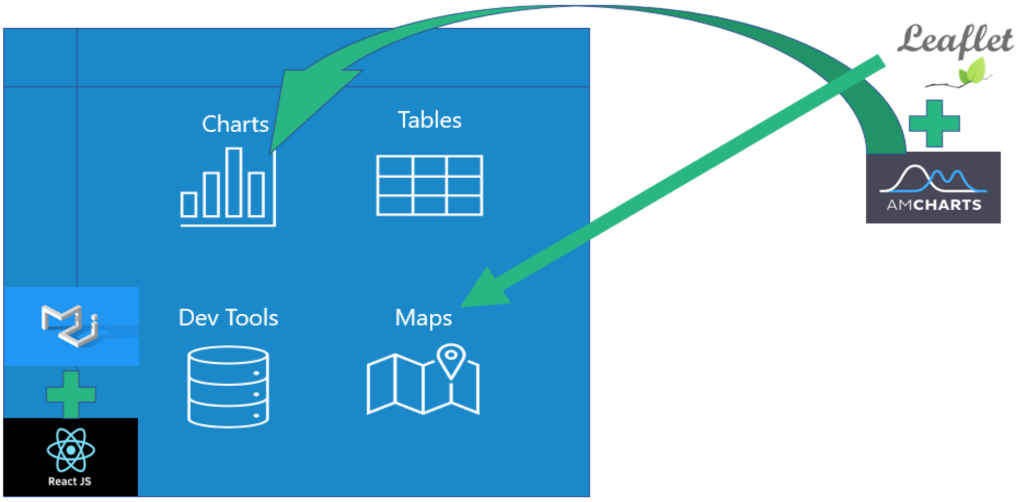


Figure 26: Visualization & Reports Engine conceptual schema

4.3.2 Analytics & Building services

In the MATRYCS Reference Architecture, the *Analytics & Building services* is the functional block under which all the different analytics services for buildings are deployed and offered to the end-users. The services included in this module can vary and are intended to answer the specific needs of diverse building sector stakeholders. These services can be thus heterogeneous and can serve to cover the needs at different levels and scales, including for example:

- Analytics for optimal management at building level, namely analytics services aimed at managing, monitoring, planning, controlling, optimizing the operation of building assets. Services in this category can include smart algorithms in support of Building Energy Management Systems, smart scheduling of building assets' operation to reduce operational costs, air quality monitoring, etc.
- Analytics for building systems and infrastructures, namely services aimed at identifying best design options or possible refurbishment measures for building systems. Services in this category may include for example analytics to identify the best retrofitting options for specific typologies of buildings or building stock, etc.
- Analytics for policy making and policy impact assessment in the building sector, namely services aimed at providing tools, visualization, statistics, indicators to support policy makers in their decision-making. Services belonging to this cluster may include services providing insights at local, regional or national level for the definition of Sustainable Energy and Climate Action Plans, services to estimate the energy consumption levels in different areas and with different geographical scales, etc.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

- Analytics for building efficiency investments, namely services aimed at supporting in identifying and defining meaningful investments in the building sector. Services related to this category may include analytics for de-risking investments in energy efficiency, analytics for measurement and verification of energy savings, services providing economic indicators to financers and investors to support them in their decision-making, etc.

The services, application and tools included in the *Analytics & Building services* architecture block work by using the data (building and non-building related) collected from the other layers of the architecture. In addition, they may access and exploit the functionalities offered by the ML/DL models exposed by the MATRYCS Processing Layer. Each application should provide a specific front-end to the user that can be used to configure possible settings and perform the desired analysis.

4.3.3 Digital Twin

The *Building Digital Twin* is a service that allows creating a digital representation of physical buildings, with every detail, mapping available data from different sources into intelligent 2D and 3D models and other enriched visualization options, to help stakeholders exploring possible design options and enabling them to take informed and effective decisions. The *Building Digital Twin* can be used also to generate the design documentation for construction and, once available in the cloud, it can serve as a repository of all the building data throughout its life cycle. Overall, stakeholders can exploit the *Building Digital Twin* as a stand-alone service or exploits its features (available information and visualization options) in conjunction with other analytics services to design complex applications.

4.3.4 Virtual Workbench

The Virtual Workbench provides a user-friendly User Interface (UI) to help in building new services that rely on Big Data and ML models. It can combine various data sets, ML models, into an environment suited for new services creation. This environment enables a wide range of potential stakeholders, such us SMEs, developers and innovators to use the provided set of tools for the design and development of new services for the building sector. The possibility to have both data and ML models easily accessible under a single environment facilitates the creation and testing of potential new services at scale and reduces the time required for developers and SMEs to create them.

Two main sub-modules are included in the MATRYCS Virtual Workbench:

- The *Building Data and Services Innovation Hub*;
- The *Data Analytics Environment*.

The *Building Data and Services Innovation Hub* is the module aimed at providing the functionalities to facilitate the creation of services quickly and easily through an intuitive user interface. The available datasets and the pre-trained models exposed by the MATRYCS Processing Layer through the *Model Serving Framework* are made available in the Virtual Workbench for analysis and reuse.

The *Data Analytics Environment* is an additional component that allows users (e.g., analysts, data scientist) to perform freeform queries and data analytics on the assets that are accessible to the platform (models and data).

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

4.4 End-to-end security

The end-to-end security in the MATRYCS Reference Architecture is implemented as an end-to-end security framework based on four pillars. The framework provides a set of design decisions and implementation aspects related to high-level security, fine-grained access control and privacy, encompassing various MATRYCS entities – also associated with big data for buildings: infrastructure, assets, services, end-users, and data. The end-to-end security framework supports inter-entity secure encrypted communication and enables mechanisms for maintaining and enforcing appropriate legal, security and privacy policies, thus increasing the system's trustfulness. Additionally, it ensures anonymization and stored data encryption where required. For auditing purposes, a logging system is employed. Moreover, the end-to-end security framework provides means for a system-wide authentication and authorization as well as software vulnerabilities/flaws detection and mitigation tools. More details about the end-to-end security framework can be found in the MATRYCS Deliverable D3.2 (released in M11, i.e., August 2021).

5 Mapping of MATRYCS Reference Architecture to existing architecture models

This Section aims at showing how the MATRYCS Reference Architecture relates to the other main architecture solutions presented in the context of big data (see Section 2). The goal is to determine if the different architectural views are consistent and, in case, to identify possible existing gaps in the architecture definitions. The following of this Section presents the alignment of the MATRYCS Reference Architecture with respect to the BDVA Reference Model, to the AIOTI HLA, and to the FIWARE Open Reference Architecture. The mappings and notes reported in the following are the outcome of the preliminary considerations made in this regard in the first part of the project. A more in-depth analysis, including also the mapping to other reference architecture models will be released together with the final version of the Deliverable on the MATRYCS Reference Architecture (D2.4, in March 2022).

5.1 Alignment with BDVA Reference Model

The BDVA Reference Model does not provide a strict definition of architectural layers, but rather it presents the horizontal and vertical concerns that are relevant within a big data reference architecture. The MATRYCS Reference Architecture maps very well to the BDVA Reference Model, namely it covers all the existing horizontal and vertical concerns. Figure 27 shows the mapping between the two architecture views.

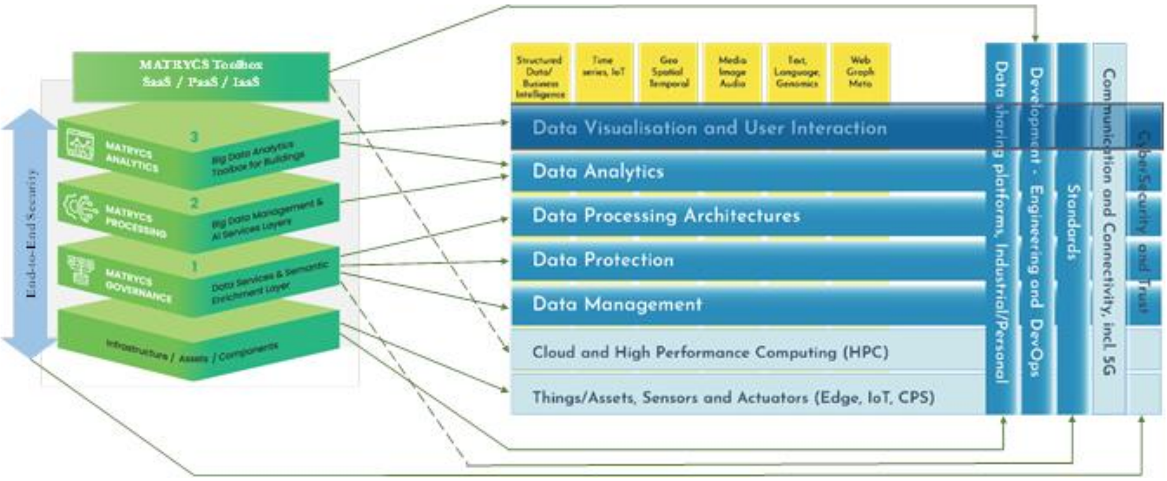

MATRYCS Toolbox
SaaS / PaaS / IaaS
End-to-End Security
3
MATRYCS ANALYTICS
Big Data Analytics Toolbox for Buildings
2
MATRYCS PROCESSING
Big Data Management & AI Services Layers
1
MATRYCS GOVERNANCE
Data Services & Semantic Enrichment Layer
Infrastructure / Assets / Components
Structured Data/ Business 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
Cyber Security and Trust

Figure 27: Mapping of MATRYCS Reference Architecture to BDVA Reference Model

Looking at the horizontal concerns, the bottom layer of the MATRYCS Reference Architecture includes all the possible data sources, thus including sensors, actuators, IoT devices, but also other platforms and open datasets. The MATRYCS Governance Layer covers three of the BDVA horizontal concerns. The MATRYCS *Interoperability Module* and *Data Pre-processing Module* cover in fact the concerns within the BDVA Data Management and specific functions are foreseen in the *Data Pre-processing Module* to guarantee data protection. The management of both data at-rest and data-in-motion required in the BDVA Data Processing block is instead guaranteed in the MATRYCS architecture via the *Data Streaming Module* and the *Query Engines* associated to the *Data Storages*. The suite of ML/DL trained models in

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

the MATRYCS Processing Layer and the *Analytics & Building Services* in the MATRYCS Analytics Layer provide the analytics capabilities highlighted in the BDVA Data Analytics, whereas the *Visualization and Reporting Engine* of the MATRYCS Analytics Layer allows visualization and user interaction.

For the vertical concerns, cybersecurity is of course one the major challenges to be addressed in both architectures. Regarding the concern of standards, the MATRYCS Reference Architecture supports developments in this direction above all in the MATRYCS Governance Layer, where there are the functions for harmonization and conversion towards a common data model. As highlighted also in the BDVA architecture description, syntactic and semantic interoperability are indeed among the most important challenges to be taken into account. Finally, regarding the concern of Development, Engineering and DevOps, it is worth noting that the MATRYCS architecture offers the MATRYCS Toolbox, where users, engineers, developers can all have access at different functionalities provided as SaaS, PaaS and IaaS. In this way, they are put in the condition to easily and flexibly design, test, and validate ideas, algorithms and services. At the moment, the Communication and Connectivity concern does not have a direct mapping into the MATRYCS Reference Architecture. The communication dimension is not explicitly mentioned in the MATRYCS Architecture because the architecture is agnostic to the type of communication being used.

5.2 Alignment with AIOTI High Level Architecture

The AIOTI HLA provides a high-level view of a big data architecture from an IoT perspective. Figure 28 shows the way in which the MATRYCS Reference Architecture can be mapped to the AIOTI HLA model.

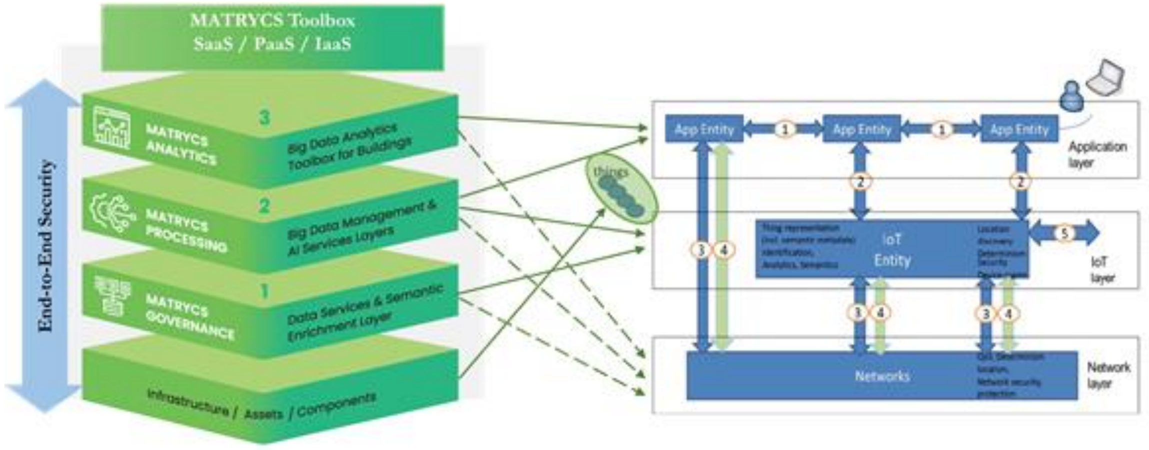


Figure 28: Mapping of MATRYCS Reference Architecture to AIOTI High-Level Architecture

Starting from the bottom of the MATRYCS Reference Architecture, the data sources in the infrastructure/ assets/ components layer have a clear and direct mapping to the things of the AIOTI architecture. In a similar way, the microservices included in the MATRYCS Governance Layer can be directly mapped to IoT entities of the AIOTI model, as these are the components guaranteeing interoperability towards the data sources and performing the pre-processing of the raw data to make them usable for the upper-level applications. The ML/DL models included in the MATRYCS Processing Layer, instead, can be both IoT Entities and App Entities in the AIOTI model. As described in [4], in fact, IoT Entities can also have analytics capabilities, above all when installed at the edge, and the App Entities are those that have the

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

intelligence associated to the specific IoT applications. As a consequence, it is straightforward to identify the link between MATRYCS Analytics Layer and App Entities. Finally, the MATRYCS Reference Architecture does not have the representation of an explicit network layer, but all the microservices have to rely on some type of communication to interact with the other architecture components. The dashed lines thus indicate here the fact that all the microservices, at each layer of the MATRYCS architecture, build on top of a network layer. Vice versa, the AIOTI architecture does not have an explicit representation of a security block. However, all the Entities of the AIOTI architecture clearly need to have specific security functionalities.

5.3 Alignment with FIWARE Reference Architecture

FIWARE proposes an Open Reference Architecture that has many aspects in common with the MATRYCS Reference Architecture, also because both focus mostly on a software/implementation perspective. As a consequence, not only the mapping of the two architecture sounds straightforward, but also the MATRYCS Reference Architecture could potentially adopt FIWARE components, leading in this way to the implementation of instances of the MATRYCS architecture powered by FIWARE. Figure 29 shows the mapping between the two architectures.

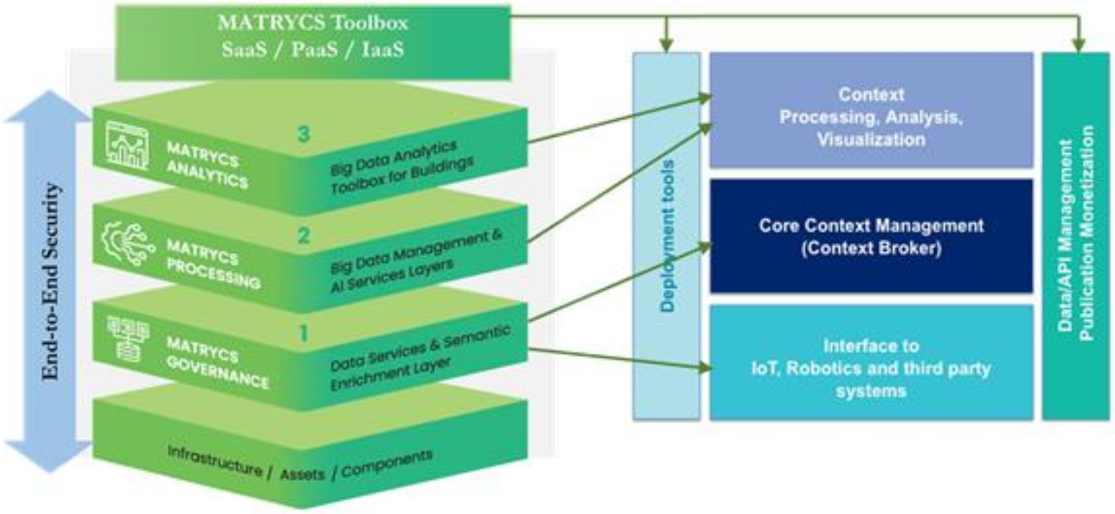

MATRYCS Toolbox
SaaS / PaaS / IaaS
End-to-End Security
3
MATRYCS ANALYTICS
Big Data Analytics Toolbox for Buildings
2
MATRYCS PROCESSING
Big Data Management & AI Services Layers
1
MATRYCS GOVERNANCE
Data Services & Semantic Enrichment Layer
Infrastructure / Assets / Components
Deployment tools
Context Processing, Analysis, Visualization
Core Context Management (Context Broker)
Interface to IoT, Robotics and third party systems
Data/API Management Publication Monetization

Figure 29: Mapping of MATRYCS Reference Architecture to FIWARE Reference Architecture

The FIWARE architecture has a software structure similar to the MATRYCS architecture, but it puts the data management done by the context broker at the center of the architecture, separating this part from the process of interconnection and gathering of the data from the data sources. In the MATRYCS Reference Architecture, both these functionalities are present but included in the MATRYCS Governance Layer. The MATRYCS Reference Architecture, instead, divides the ML/DL models' preparation from the services and applications that may make use of them (MATRYCS Processing and Analytics Layer, respectively). Both these sets of functionalities are represented in the upper layer of the FIWARE Reference Architecture. Similar to MATRYCS, the FIWARE architecture also comes with a marketplace (and the associated functionalities for buying and selling data) as well as with the possibility of adding and developing tools. This is somewhat similar to the Toolbox present in MATRYCS.

Even if not captured in the architecture representation, it is worth noting that FIWARE also tries to address concerns very similar to those highlighted for MATRYCS. For example, particular attention is given to the problem of interoperability, for which a dedicated, cross-domain FIWARE data model [17] is being created. From this perspective, not only the architectures of MATRYCS and FIWARE are well aligned, but also there is a clear possibility to build synergies to cope with some of the pending challenges behind the implementation of big data architectures. A more detailed analysis of the similarities and synergies that there might be between the MATRYCS and FIWARE will be presented in the final version of this Deliverable D2.4.

The MATRYCS project has received funding from the European Union's Horizon 2020 research
and innovation programme under grant agreement no.101000158

6 Conclusion and future steps

In this Deliverable, a first version of the MATRYCS Reference Architecture for the building domain has been presented. The proposed architecture takes into account already existing concepts and developments in the area of Big Data Architectures as well as the specifications and requirements derived from the analysis of the MATRYCS Large Scale Pilots use cases to provide a high-level, but at the same time comprehensive, description of the proposed Reference Architecture for building data spaces.

The description of the MATRYCS Reference Architecture mainly focuses on the implementation/function viewpoint of the architecture model, since this represents the central activity of the project. To this purpose, a short overview of the core components within each of the architecture layers is provided, together with relevant details about the needed interfaces and the functional interactions with other modules of the architecture. In addition, the business viewpoint is briefly touched, via the definition and description of the stakeholders involved in the building Big Data Value Chain.

The Deliverable is concluded with a preliminary analysis of the mapping of the conceived MATRYCS Reference Architecture to other well-known Reference Architecture Frameworks. This allows identifying the connections among different architecture models as well as the possible gaps to be considered for refining the design of the MATRYCS architecture.

It is worth noting that, as this Deliverable concerns the first draft of the MATRYCS Reference Architecture, the proposed concepts, definitions and models are still under discussion and can be rethought and partially modified over the course of the project activities. In this regard, future steps include the extension of the survey on existing Reference Architecture specifications as well as the review of the specifications and requirements derived from the MATRYCS use cases, following the further developments in each one of the Large Scale Pilots.

A refined description of the specifications of each architecture component will be also provided according to the progress made in the project. Finally, the alignment of the MATRYCS Reference Architecture with other architecture descriptions (and in particular with the IDS Reference Architecture Framework) as well as with other architecture proposals conceived in similar projects in the domain of building platforms and big data will be conducted and detailed.

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158

References

[1] Big Data Value Association, "European Big Data Value Strategic Research and Innovation Agenda (Version 4.0)", Oct. 2017, [online] http://bdva.eu/sites/default/files/BDVA_SRIA_v4_Ed1.1.pdf

[2] Industrial Internet Consortium, "The Industrial Internet of Things Volume G1: Reference Architecture", 2017, [online] http://www.iiconsortium.org/IIRA.htm

[3] ISO/IEC/IEEE: "ISO/IEC/IEEE 42010:2011 Systems and software engineering - Architecture description", 2011, [online] http://www.iso.org/iso/catalogue_detail.htm?csnumber=50508

[4] Alliance for Internet of Things Innovation, "High Level Architecture (HLA) – Release 5.0", Dec. 2020, [online] https://aioti.eu/wp-content/uploads/2020/12/AIOTI_HLA_R5_201221_Published.pdf

[5] FIWARE Foundation, "FIWARE for Data Spaces – version 1.0", https://www.hannovermesse.de/apollo/hannover_messe_2021/obs/Binary/A1085838/FIWARE%20for%20Data%20Spaces%20%281%29.pdf

[6] FIWARE Foundation, "FIWARE Developers Catalogue - Components", [online] https://www.fiware.org/developers/catalogue/

[7] FIWARE Foundation, "FIWARE IoT Agents for JSON", [online] https://fiware-iotagent-json.readthedocs.io/en/latest/

[8] FIWARE Foundation, "FIWARE Identity Manager Keyrock", [online] https://fiware-idm.readthedocs.io/en/latest/

[9] FIWARE Foundation, "FIWARE PEP Proxy Wilma", [online] https://fiware-pep-proxy.readthedocs.io/en/latest/

[10] International Data Space Association, "Reference Architecture Model – Version 3.0", Apr. 2019, [online] https://www.fraunhofer.de/content/dam/zv/en/fields-of-research/industrial-data-space/IDS-Reference-Architecture-Model.pdf

[11] International Data Space Association, "GAIA-X and IDS – version 1.0", Jan. 2021, [online] https://internationaldataspaces.org/wp-content/uploads/IDSA-Position-Paper-GAIA-X-and-IDS.pdf

[12] OPEN DEI Project, "D2.1 – Reference Architecture for Cross-Domain Digital Transformation V1", Nov. 2020, [online] https://www.opendei.eu/case-studies/d2-1-reference-architecture-for-cross-domain-digital-transformation/

[13] E. Curry, "The Big Data Value Chain: Definitions, Concepts, and Theoretical Approaches", in New Horizons for a Data-Driven Economy. Springer, 2016.

[14] MATRYCS Project, "D2.2 – MATRYCS Technical and Security Specifications", Apr. 2021, [online] https://matrycs.eu/resources/results

[15] S. Newman, "Building Microservices". O'Reilly Media, Inc., 2015.

[16] S. K. Sowmya, P. Deepika, J. Naren, "Layers of cloud – IaaS, PaaS and SaaS: a Survey", in International Journal of Computer Science and Information Technologies, Vol. 5 (3), 2014.

[17] FIWARE Foundation, "FIWARE Data Models", [online] https://www.fiware.org/developers/data-models/

The MATRYCS project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement no.101000158