energies

MDPI

Article

MATRYCS—A Big Data Architecture for Advanced Services in the Building Domain

Marco Pau [1,*], Panagiotis Kapsalis [2], Zhiyu Pan [1], George Korbakis [2], Dario Pellegrino [3] and Antonello Monti [1]

[1] Institute for Automation of Complex Power Systems, RWTH Aachen University, 52074 Aachen, Germany; zhiyu.pan@eonerc.rwth-aachen.de (Z.P.); amonti@eonerc.rwth-aachen.de (A.M.)
[2] Decision Support Systems Laboratory, School of Electrical and Computer Engineering, National Technical University of Athens, 15773 Athens, Greece; pkapsalis@epu.ntua.gr (P.K.); gkorbakis@epu.ntua.gr (G.K.)
[3] Engineering Ingegneria Informatica S.p.A., 90146 Palermo, Italy; dario.pellegrino@eng.it
* Correspondence: mpau@eonerc.rwth-aachen.de

Abstract: The building sector is undergoing a deep transformation to contribute to meeting the climate neutrality goals set by policymakers worldwide. This process entails the transition towards smart energy-aware buildings that have lower consumptions and better efficiency performance. Digitalization is a key part of this process. A huge amount of data is currently generated by sensors, smart meters and a multitude of other devices and data sources, and this trend is expected to exponentially increase in the near future. Exploiting these data for different use cases spanning multiple application scenarios is of utmost importance to capture their full value and build smart and innovative building services. In this context, this paper presents a high-level architecture for big data management in the building domain which aims to foster data sharing, interoperability and the seamless integration of advanced services based on data-driven techniques. This work focuses on the functional description of the architecture, underlining the requirements and specifications to be addressed as well as the design principles to be followed. Moreover, a concrete example of the instantiation of such an architecture, based on open source software technologies, is presented and discussed.

Keywords: high-level architecture; building services; building value chain; big data; Internet of Things; data analytics

Citation: Pau, M.; Kapsalis, P.; Pan, Z.; Korbakis, G.; Pellegrino, D.; Monti, A. MATRYCS—A Big Data Architecture for Advanced Services in the Building Domain. *Energies* **2022**, *15*, 2568. https://doi.org/10.3390/en15072568

Academic Editor: Benedetto Nastasi

Received: 28 February 2022
Accepted: 30 March 2022
Published: 1 April 2022

1. Introduction

The built environment is one of the most energy-demanding sectors which is responsible for a significant share of the total amount of final energy consumption. In the European Union (EU), the energy used over the entire building life cycle, thus including the construction, usage, refurbishment and demolition, accounts for approximately 40% of the overall EU energy consumption and 36% of the greenhouse gas emissions [1]. As a consequence, buildings are among the main targets of the current policies aimed at decarbonization and greenhouse gas reduction. This is also further motivated by the fact that the existing building stock is in many cases quite old and energy-inefficient [1] which leaves room for undertaking actions to curtail the energy consumption and enhance the energy performance. To this purpose, the EU issued an Energy Performance of Building Directive [2] and an Energy Efficiency Directive [3], which pushed member countries to develop a legislative framework to pursue better energy efficiency and climate neutrality for the building stock by 2050. These directives have been revised [4] as part of the Clean Energy Packet [5], more strongly pushing the adoption of new technologies to modernize the building sector and asking for the definition of clear building renovation strategies. Recently, a proposal for a further upgrade of these directives has been released [6] which sets more ambitious goals and calls for additional efforts in digitalization.

The digital transformation is commonly recognized as one of the main ingredients for the smartification of the building sector [7]. Digitalizing the built environment and facilitating data sharing is key to achieve the aforementioned energy efficiency targets, as it unlocks the possibility to process cross-domain data and to develop innovative data driven services. Heterogeneous data are used for example in [8] for balancing the building energy demand with the local renewable generation and in [9] to enable the measurement and verification of buildings' energy performance in real-time. Data-driven approaches based on different machine learning techniques are proposed for a large variety of building applications. Some of them directly address the problem of improving the building energy performance by means of smart energy management or control [10–13]. Other applications are instead complementary to the task of the optimal energy management of the building and provide, for example, load prediction [14–16], demand estimation [17,18], anomaly detection and diagnosis [19], or models calibration [20]. Finally, some other applications do not directly focus on the building energy management but they put the basis for the identification of refurbishment needs or other energy saving measures. Applications belonging to this category are for example those aimed at the enrichment (or error detection) of energy performance certificates [21] at the measurement and verification of energy conservation measures [9,22], etc. Directly or indirectly, therefore, all the applications above eventually either help to improve the energy performance or to conceive strategies for energy saving, hence contributing to pursuing the defined energy efficiency targets.

The concept of digitalization is multifaceted and encompasses several aspects. First of all, it involves the conversion of the available building-related data into digital models that can be easily accessed and processed via dedicated software. The Building Information Model (BIM) is one of the most important and well-known examples of models providing a digital representation of building information [23–25]. Other efforts in this direction include the definition of open ontologies, such as SAREF [26] and Brick [27], aimed at providing unified representations of specific classes of building data. Beyond this, the concept of digitalization also concerns the integration of digital tools to deploy building automation, optimize operations and streamline processes. From this standpoint, the adoption of building management systems (BMSs) [28] is becoming a consolidated practice, but other innovative digital tools such as last generation digital twins [29,30] are also emerging and increasingly gaining importance. In general terms, digitalization also includes the employment of any digital technology to access, store, interpret, analyze and process existing data in order to support business planning and decision making. In this regard, a large set of opportunities exist nowadays thanks to the large spread of technologies coming from the Internet of Things (IoT) domain [31] and to the possibilities made available via cloud computing [32].

Overall, the digitalization process has the potential to bring a number of benefits for all building value chain (BVC) stakeholders, including higher productivity, energy efficiency improvement, reduction in the costs of building constructions, better situational awareness in support of decision making, etc. Moreover, it can unlock new business opportunities through the smart integration of buildings within the electrical grid [33,34] or with other inter-dependent domains [35]. Nevertheless, digitalization also comes with a set of challenges related to data management. Today, huge amounts of building data are generated by sensors, smart meters, IoT devices and a multitude of other data sources. These data are largely heterogeneous and they come with very diverse formats, sizes, varying granularity, quality, etc., thus posing serious obstacles to the effective handling and exploitation of data. Furthermore, data are typically dispersed in different locations and non-interoperable platforms [36] , which prevents extracting their full value, using them for multiple use cases and creating advanced applications based on cross-domain information. As data may contain sensitive information, ensuring data privacy is also a main concern, similar to providing data sovereignty and cybersecurity to protect the business value of the data [37].

All the above challenges call for the adoption of ad hoc data science solutions to exploit the full potential of the available data [38]. To this aim, technologies specifically conceived for big data management are increasingly required to deal with the volumes, variety and veracity of incoming data. IoT-based platforms are also necessary to put in place the software functionalities needed to support data handling and to deploy software infrastructures that can be easily scaled, extended and upgraded. Blockchain and other distributed ledger technologies (DLTs) may be used to enforce trustworthiness and data sovereignty, while cloud computing options can be implemented not only to guarantee elasticity in the allocation of computational resources, but also to facilitate the creation of a market of turnkey services in the building sector. Finally, the most recent machine learning (ML) and artificial intelligence (AI) techniques can be employed to process, understand, classify and correlate data, thus allowing to generate more complex and meaningful information out of the raw samples, which would eventually lead to the implementation of more advanced services and applications [39–41].

Given this context, the goal of this paper is to present a high-level architecture for building data management that unlocks the sharing and interoperability of heterogeneous data, the interconnection of advanced data analytics tools and the seamless integration of new services to continuously create business value. The presented architecture stems from the big data architecture presented in [42] and it is designed to fulfill the requirements of different smart building use cases defined during the European project MATRYCS [43]. Overall, it aims to serve as a reference architecture for the deployment of the software components needed for a proper data governance and for an easy integration of third-party services. In this paper, a detailed view of the architectural requirements and functionalities is provided together with concrete examples of technologies that can be used for the instantiation of the architecture. More specifically, this paper provides the following contributions:

- It analyses the requirements and specifications coming from different use cases about smart buildings in order to define the main concepts and design principles for a big data architecture for the building domain;
- It shows the functional view of a high-level architecture for building data management which unlocks data sharing, interoperability and the easy connection of advanced turnkey services for the built environment;
- It provides a view of exemplary technologies that can be used for the instantiation of such an architecture, together with the discussion of a real use case that highlights the key benefits that this architecture may bring with respect to in silo systems.

The rest of this paper is organized as follows. Section 2 provides a review of related works concerning reference architectures for big data, which can also be relevant for the building domain. Section 3 discusses the steps for the creation of value in a big data value chain and defines the roles of the actors participating in it. Section 4 analyzes the main needs of the stakeholders and derives the most important requirements, specifications and design principles to be considered for the definition of the building data architecture. Section 5 describes the proposed architecture, focusing on the functional role of each software component and their inter- and intra-layer interactions. Section 6 presents a specific use case that shows how the proposed architecture allows integrating different types of data useful for developing advanced building applications and provides a view of the technologies used for the concrete instantiation of such an architecture. Finally, Section 7 provides the final remarks and concludes this work.

2. Review of Other Big Data Architectures

With the explosion of the amount of data being generated by sensors, smart meters and other IoT devices as part of the digitalization process of many industrial sectors, using and exploiting these data in an optimal way via ad hoc data science and data analytics tools is becoming increasingly strategic. In this scenario, a number of architectures have been proposed in the recent past to allow for proper data processing and usage. Some important

architecture proposals are not domain-specific, but more generally oriented towards the industry and big data context.

The big data value reference model, defined by the big data value association (BDVA), is a first example of such architectures [44]. It is structured along horizontal and vertical concerns. The horizontal concerns focus on different steps of data handling. At the bottom, there are the physical devices (sensors, actuators, edge things, etc.) that generate the data and the physical cloud infrastructure where the data will be received and processed. On top of them, there are four software layers which take care of: (i) data management, namely all the preprocessing needed to clean, translate and harmonize the incoming data; (ii) data protection, which addresses the issues of anonymization, trust and privacy for the received data; (iii) data processing, which handles the queries, streaming or storage of different types of data; and (iv) data analytics, where machine learning and other analytics tools are employed to extract further value from the received raw data. Finally, the upper horizontal concern refers to the visualization for the end user and to the set of tools needed to allow their interaction with the available data and tools. The vertical concerns instead refer to the cross-cutting aspects that have to be addressed along the entire chain of data handling. These include aspects such as cybersecurity, communication and connectivity, standardization, data sharing and access to marketplace.

The Industrial Internet Reference Architecture (IIRA), created by the Industrial Internet Consortium, is an abstract architecture that aims at addressing different industrial sectors, providing unified definitions and patterns that can be applied across different use cases [45]. It is structured in four architectural viewpoints, each one providing different architectural details that are relevant for the definition and description of use cases. At the top is the business viewpoint, which provides a view of how the business objective has to be reached and how the involved stakeholders interact. Moving downwards, there is the usage viewpoint which describes the tasks to be performed by each actor of the use case and how these are interfaced with the IoT system. The third layer is the functional viewpoint and it focuses on the methods, tools and interfaces that are needed in the IoT system to support the defined use case from a functional perspective. Finally, the implementation viewpoint more concretely describes the technological components needed to perform the functionalities described in the above layer. Beyond this partition in viewpoints, the IIRA also provides a discussion of different architectural patterns underlying the different options of implementing the IoT system, or parts of it, at the edge, in the cloud, or directly in the servers of the involved enterprise.

The Alliance for Internet of Things Innovation (AIOTI) is concerned with the development of innovative IoT solutions and it proposed a high-level architecture specific for the IoT domain [46]. The scope was to create a coherent architectural view that can be adopted for the development of large-scale pilots where "things" are at the center of the use case. The AIOTI architecture provides two views, a domain model and a functional model. The domain model aims to describe how things, users and services are connected and interact within a specific use case. The functional model gives a view of the software components over different layers according to the type of service they provide. This includes a network layer for the provision of all the communication and connectivity services, an IoT layer for the middleware services and an application layer for the implementation of the specific industrial use case.

The Fiware Open Reference Architecture [47] was developed by the Fiware consortium, which develops and provides open source software for the implementation of smart IoT platforms for different use case scenarios. The Fiware architecture looks more closely at the functional and software implementation perspectives. It builds upon a Fiware Context Broker, a key element of each Fiware platform, which is the software component responsible for handling and re-routing the data to the different applications in the platform. The rest of the platform is built in a modular way using generic and/or specific enablers which are software packages responsible for some generic or specific data processing tasks. The Fiware architecture has three layers with the context broker at the center, a set of enablers

responsible for the connectivity towards field devices at the bottom, and a set of enablers for data processing, analysis and visualization at the top. Each enabler exhibits open APIs as interfaces for the flexible and modular interconnection within a Fiware-powered platform.

More recently, large efforts have been devoted to the definition of data spaces. Data spaces are intended to act as a decentralized ecosystem of shared data that aim to support the trusted exchange of data for the creation of a data economy and for leveraging their business value. In this context, the International Data Space (IDS) Association released the IDS Reference Architecture Model [48]. Similarly to IIRA, the IDS architecture consists of different viewpoints which are associated with businesses, functions, processes, information and systems. The business layer defines and describes the different roles of the participants to the IDSs in relation to their task with respect to the data exchange. The functional layer focuses on the functional requirements to be addressed in the IDS ecosystem. Particular attention was paid to the topics of trust, security and data sovereignty, ecosystems, standardized interoperability, value-adding apps and the data marketplace. The process layer describes the process steps to onboard exchange data and uses value-adding apps in the data spaces. The information layer deals with the methods to define the domain-agnostic common languages to be used within IDSs. Finally, the system layer concerns aspects related to the integration, deployment, execution, and configuration of the logical software components in virtualized environments such as virtual machines and application containers. In addition to these horizontal layers, cross-cutting layers associated with security, governance and certification are also considered.

The architecture proposed in this paper is not in contrast with the proposals above, but rather focuses on a different architectural view. The definition of the stakeholders' roles presented in Section 3 is in line with the roles' definition provided at the business layer of the IDS architecture. The software architecture instead mostly focuses on the functional and implementation viewpoints. Similarly to the BDVA architecture, it describes the set of functionalities to be provided within the architecture to guarantee the handling of data at the different levels of the data value chain and it offers a view of the interconnections of the different software components together with practical examples of the technologies that may be used for architecture instantiation.

3. Building Data Value Chain

The smart building sector involves many stakeholders (e.g., building owners, facility managers, constructors, utilities, energy service companies, governmental institutions), each one with a different role, also depending on the considered use cases. In this section, stakeholders are considered in more general terms, abstracting from their particular role within a specific building application scenario, and rather looking at their role from a data value creation perspective.

As described in [49], the data value chain may involve multiple steps, each one bringing additional insights and business value to the available data. Figure 1 shows the flow of subsequent steps for the creation of data value within an IoT data-driven architecture, where raw data are first incrementally transformed into information and then into advanced knowledge. As visible in Figure 1, each one of these steps may involve different types of data processing which require dedicated methods, tools, services or applications to be carried out. Within this data value chain, stakeholders can assume different roles depending on their position with respect to the considered data transformation process. In particular, each operation of data transformation can be generically seen as a process that involves stakeholders who make available the data in input, stakeholders that provide the software applications to process the data and stakeholders that eventually receive and make use of the transformed data. Figure 2 gives a schematic view of the interactions among these stakeholders within a generic flow of data transformation. Such a classification of the stakeholders, described in greater detail in the following, is in line with the roles defined for the data space ecosystems [48].

- *Building data owners*: these are the entities that have legal ownership of the data to be processed. They have full control of the data, namely the rights to decide the terms and conditions with which the data can be made available and accessed by other stakeholders. Examples of this category may be building owners or facility managers that own meters and sensors generating raw data, or service companies that create enriched data or information exploiting specific software (in this case, they could own the enriched data or information they create).
- *Building data providers*: these are the entities responsible for making the data available to third parties. In many cases, this entity coincides with the building data owner, but there could also be cases in which an external provider (e.g., an IT provider) is in charge of setting up the tools required to give access to the considered data.
- *Service providers*: these are the entities that provide the services, namely the tools, software or applications necessary to process the input data for transforming them into higher-complexity and higher-value data. Depending on the type of data transformation they perform and where this transformation step is positioned in the data value chain, services can be further distinguished in different sub-categories (e.g., with reference to Figure 1, data analytics service providers offer software services to convert raw or enriched data into information).
- *Building data consumers*: these are the entities that receive the added-value data created via a service and make them available to the final recipient. This role in many cases can coincide with the building data user, however, similarly to the distinction between data owners and data providers, these entities are generally different.
- *Building data users*: these are the entities that make use of the processed data and that exploit their business value. Depending on the conditions and terms with which they are granted access to the starting (input) data, they may become owners of the added-value data. Examples of this category may be building managers, energy service companies, governmental institutions or other stakeholders that make use of elaborated data to run their business.

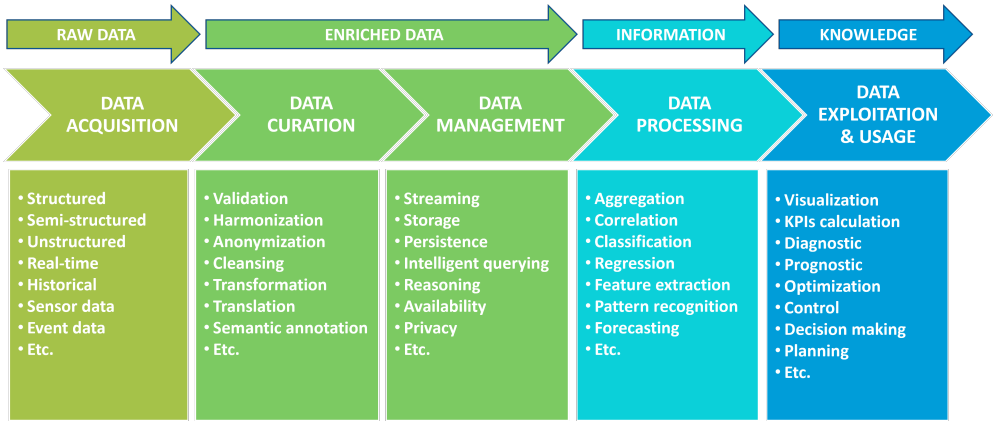

Figure 1. Data value creation within an IoT data-driven platform (adapted from [49]).

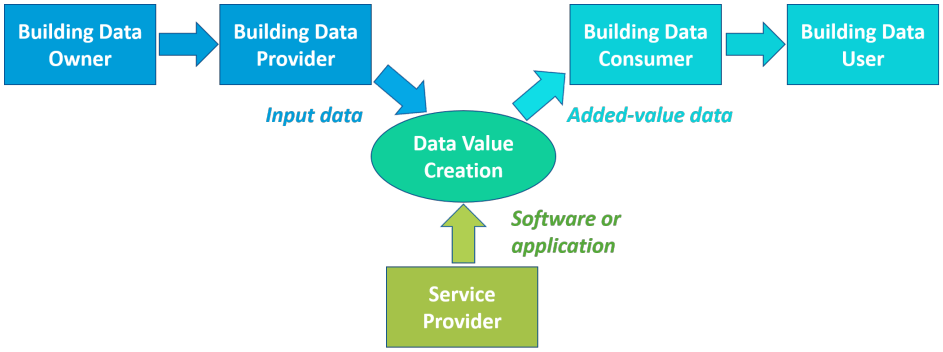

Figure 2. Stakeholders' roles within the building data value chain.

4. Identification of the Main Architectural Requirements

From a methodological point of view, the requirements for the designed building data architecture are collected by analyzing: (i) the specific needs of different data value chain stakeholders; (ii) the functional requirements derived from a heterogeneous set of smart building use cases; (iii) the non-functional requirements derived from use cases and from general good practice considerations associated with the development of IoT frameworks.

4.1. Data Value Chain Stakeholder Requirements

Table 1 shows the main needs associated with each one of the stakeholders in the data value chain. For building data owners, the main concern is to keep the sovereignty of the data [50] and to ensure that the terms and conditions imposed for their use are fulfilled. As such, they can retain ownership of the shared data and exploit their business value. On the other end of the data value chain, building data users may be concerned about the trustworthiness of the data they obtain since their decisions and business may be highly dependent on such data. Blockchain and DLT can help both to empower data sovereignty and to guarantee the trustworthiness of the data transactions [51,52]. The use of these technologies should thus be envisioned in the overall architectural solution.

Table 1. Main stakeholder requirements in a big data ecosystem.

Stakeholder	Requirement
Data owners	Data sovereignty
Data publishers and consumers	Open APIs
Service providers	Standardized data models
Service providers	No vendor lock-in
Data users	Trustworthiness

For data providers and consumers, the main role is to make available and access the data, respectively. For this, open application programming interfaces (APIs) should be possibly adopted, so that no technological barriers exist for access to the data [53]. On the service providers' side, an important feature is the use of standardized data models. To this purpose, de facto standard ontologies and data schemes may be used, such as SAREF [26] and Brick [27]. This in fact avoids the need for developing customized interfaces to interpret and convert the input data. In addition, the architectural solution should allow the easy integration of third-party services and the co-existence of multi-vendor technologies. From this standpoint, it is thus essential that the architectural model is highly modular, with a clear decoupling of the functionalities performed by each module.

4.2. Functional Requirements

Table 2 gives an overview of the main functional requirements identified from the pilot use cases of the MATRYCS project which span over the different business objectives and scales of the built environment [43]. First requirements are related to the need of processing two different types of data: streaming and batch data. Stream data are usually measurements generated by field devices which are transmitted with a certain reporting rate and need to be collected and processed in near real-time. An example is given by the data generated by building sensors as input to a BMS [8,54,55]. Batch data instead generally consist of large datasets that need to be processed a posteriori all at once. Examples of these data are large sets of cadastral data or of energy performance certificates (see [36] for an exhaustive list of building-related data repositories). Due to the diverse nature of the data, different solutions are typically employed for their communication. Publish/subscribe patterns [56] are typically preferred for handling streaming dataflows since they allow the easy re-routing of the data to multiple consumers as well as generating event-based

triggers. On the other hand, request/response mechanisms are generally preferred for querying and collecting batch data in one-to-one communications.

Other requirements are related to the management or preprocessing of the incoming data. As the quality of the used data is of utmost importance, dedicated services are necessary to check the possible presence of outliers, incorrect entries, inconsistencies, duplicates or incomplete data over existing or incoming datasets. Specific data curation routines should be also in place to ensure the proper organization and structuring of data coming from a variety of heterogeneous data sources. Finally, data management requirements also include the use of ad hoc databases and data persistence logics to ensure the availability of both raw and processed data over time.

Table 2. Main functional requirements for smart building use cases.

Functional Need	Requirement
Need to handle streaming of near real-time data.	Stream processing
Need for efficient querying and collection of large datasets.	Batch processing
Need for preprocessing services to handle outliers, duplicates, inconsistent or incomplete data.	Data cleaning
Need for services that automatically organize and structure heterogeneous data.	Data curation
Need for efficient data storage and persistence solutions.	Data management
Need to unlock interconnectivity among different technological components.	Interoperability/modularity
Need for dedicated interfaces to import data from other platforms and repositories.	Interoperability
Need for intelligence to extract meaningful information over large sets of heterogeneous data.	Data analytics
Need for user-friendly interfaces and querying systems to access and interact with available data.	User friendliness and interactivity
Need for immediate notification of possible events, warnings and alarms.	User friendliness and interactivity
Need for user-friendly graphical interfaces for the clear and effective presentation of reports and/or results.	User friendliness and interactivity
Need for attractive dashboards with interactive graphs, charts, and maps and other suitable visualization options.	User friendliness and interactivity

Interoperability is another main requirement. First of all, the architectural solution should be designed to facilitate the co-existence of the different technologies needed to simultaneously run all the middleware and application-oriented services foreseen to cope with complex use cases. To this aim, specific software components may be needed to guarantee syntactic and semantic interoperability [57]. Moreover, many use cases require the data provided by third-parties or external repositories. For this, dedicated interfaces are needed to ensure the interconnection of multiple platforms in the overall building ecosystem. The complexity of many use cases and the availability of large amounts of data also leads to the necessity to adopt appropriate data analytics and AI tools to synthesize information, extract patterns, discover correlations and create additional insights that are not directly deducible from the raw data. Since this plays a key role for the creation of business value from the data, a dedicated framework should be envisioned in the architectural model to easily plug different AI modules tailored to the specific needs.

A last important requirement from a functional point of view is associated with the final presentation and visualization of the information for the final users. To this aim, specific dashboards and graphical user interfaces should be provided to ensure interactive

access to information, the immediate notification of events, visualization of graphs, maps, charts, etc.

4.3. Non-Functional Requirements

In addition to the above specifications, an additional set of non-functional requirements can be identified in relation to common practices of big data management within IoT frameworks (see Table 3). When dealing with big data, a first major requirement is scalability. The IoT framework must be able to easily scale up to cope with increasing amounts of incoming data and integrated services. Microservice-based architectures are typically recommended to this purpose for the deployment of the software [58]. Cloud computing can offer elasticity in the allocation of computational resources typical of cloud systems [32]. Virtualization approaches and the distributed deployment of software components at the edge [59] (where applicable) can also help in reaching scalability. The choice of ad hoc middleware technologies (databases, brokers, etc.) that can be easily distributed over clusters of servers is also of paramount important from this point of view.

Table 3. Non-functional requirements for IoT-based architectures.

IoT Need	Requirement
Need to flexibly scale horizontally to integrate increasing amounts of data and services.	Scalability
Need to process very large amounts of heterogeneous data in a computationally efficient way.	Performance
Need to easily replace outdated software or integrate new components without affecting system operation.	Upgradeability and extensibility
Need for solutions that ensure proper operation including in presence of errors or failures.	Reliability
Need for solutions to prevent unintended or unauthorized operations on data and system components.	Security
Need to avoid any technological barrier towards the integration of data and services from any stakeholder.	Cost effectiveness

Computational efficiency and performance are other important requirements when having to process huge amounts of data. Cloud computing and high performance computing (HPC) can be essential to run some of the services and this again brings the need for virtualization and distributed deployment in the proposed solution. Easy upgradeability and extensibility are other key aspects. These take into account the need for dynamically updating the available software to address new use cases or to exploit new technologies and technical solutions that may become available over time. The modularity provided by a microservice-based philosophy is important in this perspective, since it can enable the integration of new software modules in a sandbox environment and the smooth replacement of obsolete software or technologies with novel ones.

Reliability and security are two other critical requirements. Reliability is generally intended as the capability to provide fundamental functionalities including in the presence of errors or failures (see [60] for more details on the concept of reliability in the IoT domain). In loose terms, it also incorporates the concept of the availability and survivability of the system. To ensure high reliability, the architectural solution should not have single points of failure, should employ suitable mechanisms for managing the redundancy/replication of software components as well as the mirroring of data, and it should have high modularity to ensure that possible issues are isolated within the affected subsystem and do not compromise the operation of other components. The use of recent containerization approaches may represent a technological solution that allows addressing some of the above points, thus leading to higher reliability [61]. Security is also quite a broad concept as it encompasses aspects related to data confidentiality, privacy, integrity and cybersecurity [62,63]. The overall

IT system should have tailored solutions to allow, among others, the anonymization of the data, restricted access to confidential information, the application of authorization policies, the detection of undesired data manipulation and the encryption of data transmission. From a cybersecurity perspective, specific security policies should be in place to constantly monitor for malicious activities or other violations, to track and respond to incidents, and to search for known and unknown threats. Such policies should be constantly updated to reflect the best practices in the cybersecurity domain in order to ensure the security of data, software components, technologies and thus that of associated stakeholders.

Finally, the building data system should allow to offer, buy and sell both data and services in a cost-effective way. The service provision here generally refers to any deployment model including the options of data as a service (DaaS), software as a service (SaaS), platform as a service (PaaS) and infrastructure as a service (IaaS) [64]. From an architectural point of view, this means that the architecture design should permit the easy integration of new data and services and allow offering them into a marketplace without placing any technological barrier on any of the stakeholders. Once more, here modularity and virtualization represent two key specifications in this regard.

4.4. Architecture Design Principles

Based on the derived set of requirements and in alignment with the best practice recommendations given in [65], the following design principles were followed for the definition of the building data architecture presented as follows in Section 5.

- *Modularity via microservices:* the proposed architecture is a microservice-based architecture; namely, each of the indicated software components should be intended as an independent and loosely coupled process that performs a small and well-defined task that interacts with the rest of the system only via its I/O interfaces. As already discussed, a microservice philosophy helps achieve high modularity, which is essential to foster scalability, upgradeability, extensibility, reliability and to open to a fair, cost-effective and multi-vendor provision of services.
- *Cloud virtualization:* the designed architecture does not put any constraint on the deployment of the related software. This opens to the possibility to virtualize the microservices and implement any kind of cloud deployment model [64] where the provisioning and maintenance of hardware, IT platform and/or software could be handled by third-party providers. Thanks to the modular design of the architecture, if desired, some of the functional blocks may be flexibly moved at the edge, thus customizing the software deployment according to specific needs.
- *Openness and data sharing:* the proposed architecture aims to open the building data and services to boost the business opportunities in the building domain. In this context, openness refers to the possibility of having open data (available for free or under fair conditions), open API specifications to facilitate data sharing (also among different domains), and open source services to foster the creation of cost-effective building ecosystems.
- *Security:* the architecture must integrate a security framework that allows having trusted and secure data transactions, simultaneously fulfilling all the requirements of privacy, confidentiality and sovereignty that may exist in each application scenario.
- *No vendor lock-in:* the architecture design is conceived to allow for the easy integration of new services and applications coming from different vendors free from technological barriers, for example, being dependent on some proprietary solutions. The interfaces with the rest of the ecosystem must be clearly defined, transparent and possibly based on standardized solutions. This can lead to the development of an open market of turnkey services in the building sector with fair conditions for all participating stakeholders.
- *Distributed data ecosystem:* the proposed architecture takes into account the fact that building data will still be dispersed over several independent platforms. The aim here is to conceptually define the functional blocks that should exist, which can

then reside in different locations. In other words, the proposed architecture defines the functional layers to transform raw data into information and knowledge, but the software components can be flexibly distributed, giving place to a distributed ecosystem in support of data and building services economy.

5. MATRYCS Big Data Architecture for Building Services

The proposed architecture was designed to cover the main steps indicated in Section 3 for the creation of value from the available data. To this purpose, each enrichment step of the data value chain is mapped to a specific layer, which contains all the methods, tools and components necessary to this aim [66]. Figure 3 shows the conceptual partition of the proposed architecture in layers, together with their association with each step of the data value chain. Overall, the MATRYCS architecture consists of:

- *Infrastructure layer:* this encompasses all the sensors, meters, IoT devices as well as other data hubs or data sources that generate the (raw) data as input to the MATRYCS ecosystem.
- *Governance layer:* this contains all the software components necessary for the collection of the raw data, their preprocessing, cleaning, curation and management. At this level, raw and possibly unstructured data are thus transformed into enriched data structured according to the chosen syntactic and semantic models.
- *Processing layer:* this includes all the components for the training, validation and running of the ML and AI tools used to carry out advanced data processing and the transformation of data into more elaborate information.
- *Analytics layer:* this provides the toolboxes with the building applications offered to address specific use cases, together with the associated visualization tools and user interfaces. Here, the applications can use and assemble different pieces of information offered by the processing layer for creating complex knowledge.
- *Security layer:* this is a cross-cutting layer that spans over all the other layers with the scope of providing the software technologies and the framework necessary to guarantee the security of the building ecosystem at all of its levels.

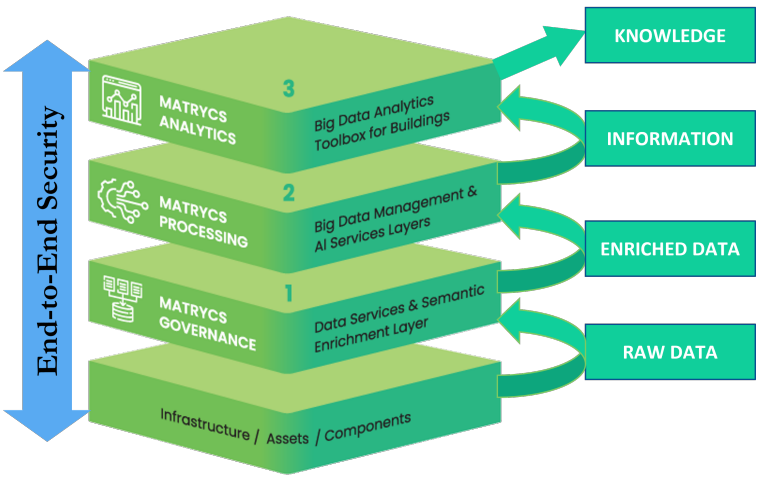

Figure 3. High-level view of the MATRYCS architecture layers and of their association to each transformation step of the data value chain.

In the following, a more in-depth view of the software components defined within each architectural layer is given together with the details of their intra- and inter-layer interactions.

5.1. MATRYCS Governance Layer

The MATRYCS governance layer offers a collection of data services in the MATRYCS ecosystem and acts as an orchestrator layer by providing event-driven pipelines that manage data from sources (MATRYCS Data Providers) up to the MATRYCS analytics tools and services. According to the Big Data Value Chain approach followed in the conceptualization of the MATRYCS solution, the governance layer was designed to identify and handle the related activities of integration, preprocessing, semantic annotation, harmonization, storage and querying of the largely heterogeneous data (building data, energy data, sensors data, weather data, etc.) that can be encountered in building scenarios.

Figure 4 provides a detailed view of the MATRYCS governance layer and its building blocks. As shown, the main components are the Interoperability Service, the Data Preprocessing Service, the Streaming Module, the Reasoning Engine, the Trusted Data Sharing component, and the Data Storage and Querying module. These modules ensure the acquisition of the data from their source as well as their curation, anonymization, preprocessing and (distributed) storage while harmonizing the format of the various datasets according to the chosen syntactic and semantic models. Moreover, specific modules ensure that the secure and trusted data sharing with the other layers of the MATRYCS ecosystem.

- *Interoperability Service:* it is the service responsible for connecting the data sources with the MATRYCS technical ecosystem. This service should allow accepting different protocols (such as SFTP, HTTP, AMQP datasets and events) and, leveraging on its mechanisms, it distributes the collected information to the other components and layers of the MATRYCS architecture. Data from external data hubs and other open data sources (e.g., weather data repositories) are also retrieved via this service for being included in the MATRYCS data collections and pipelines [67].
- *Data Preprocessing Service:* the Data Preprocessing Service is a mechanism responsible for the curation, anonymization, homogenization and semantic annotation of the data inserted into MATRYCS governance layer through the Interoperability Service. Specific ontologies and data models must be used here to ensure the harmonization of all the different incoming data so that the upper level analytical tools and services can have a more straightforward and efficient access to interoperable data with homogenized variables.
- *Streaming Module:* the Streaming Module is the mechanism responsible for the distribution of the streaming messages/events between MATRYCS components, modules and services. This must ensure a one-to-many communication, thus allowing the simultaneous distribution of the data to multiple microservices present in the MATRYCS ecosystem.
- *Reasoning Engine:* the Reasoning Engine condenses the MATRYCS metadata and semantic data in order to provide intelligent querying over data and pattern extraction, thus enhancing the analytical services' capabilities. It then exposes the extracted information via REST APIs.
- *Data Storage and Querying:* the streaming data ingested into the MATRYCS ecosystem are saved in the data storage module which consists of an object storage that ensures the retention of incoming events from the streaming module. A querying engine must also be integrated in order to allow for the fast and multiple queries of the different entities stored in the object storage.
- *Trusted Data Sharing:* the Trusted Data Sharing is a module which uses blockchain technologies to ensure integrity and trustworthiness in the MATRYCS datasets. The primary purpose of this component is to remove the need for intermediaries and replace them with a distributed network of digital users that work in partnership to verify and safeguard the data transactions between stakeholders.

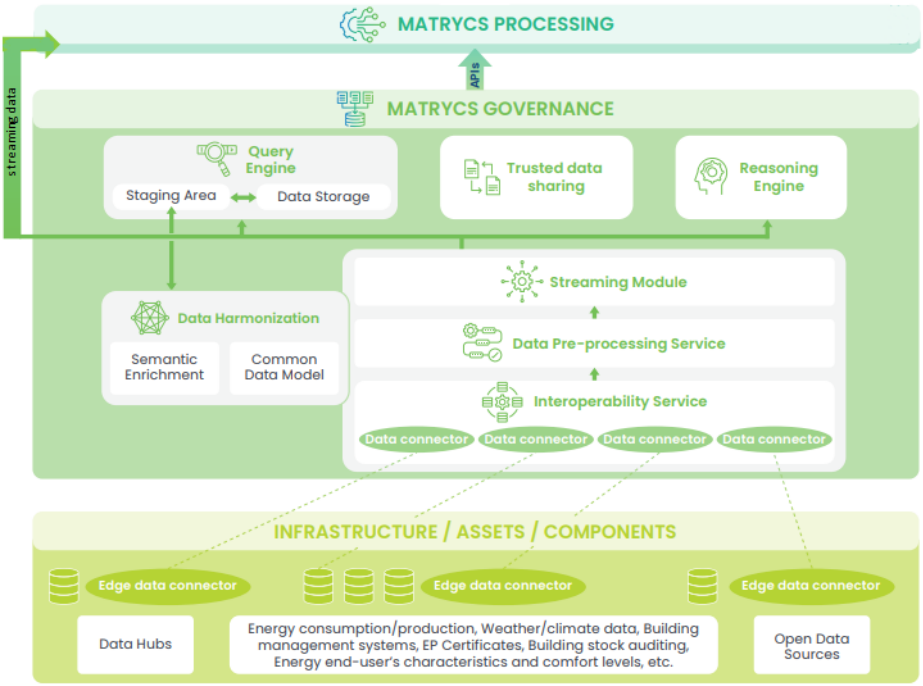

Figure 4. MATRYCS governance layer architecture.

5.2. MATRYCS Processing Layer

The MATRYCS processing layer encapsulates the machine learning and artificial intelligence components of the MATRYCS ecosystem and organizes them into a standalone sandbox to promote and facilitate quick adaptation along different contexts. This layer is responsible for retrieving the needed data from the storage, for their proper transformation, for the training, validation and final deployment of the machine learning models and for feeding these models with the necessary batch and/or streaming data. Figure 5 depicts in greater detail the software blocks included within the MATRYCS processing layer and their interconnections. These components are as follows.

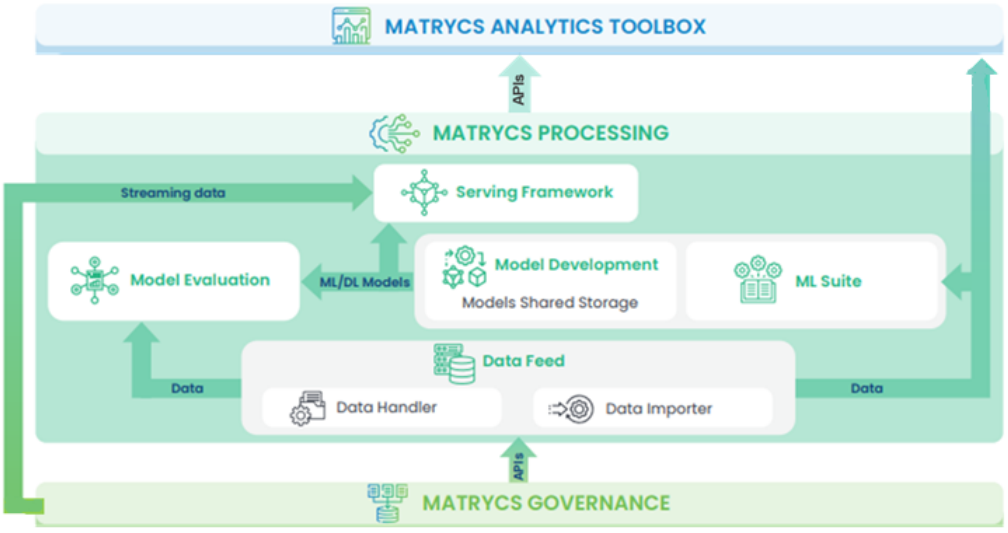

Figure 5. MATRYCS processing layer architecture and interconnections.

- *Data Feed Module:* the role of this module is to retrieve the underlying data from the storage, perform the needed transformations and finally pass the properly transformed data to the AI models. This stage is needed because the AI and ML models usually

cannot operate directly with the raw data in the format they are stored. In fact, each AI model requires that the data have a specific format in order to be able to handle them. Typical examples of required transformations are the handling of missing values, the normalization of input data, or the selection of the right features. Once this step is completed, the properly transformed (final) data can then be passed to the AI models through the ML suite.

- *ML Suite:* the ML Suite is a library of state-of-the-art AI data-driven tools and methods that is used for the development of MATRYCS AI models. Multiple technologies and software can be exploited for ML (scipy, scikit-learn, Spark MLib), DL (Keras, Pytorch, TensorFlow, Horovod) and Image Processing (OpenCV, scikit-image). The result is to expose a rich and flexible software library in order to define, train and deploy ML models, including ANN classifiers, knowledge representation and reasoning aiming to attach new knowledge and predictions on the existing extreme-scale streams of data.
- *Model Development Module:* this module concerns the exploitation of the ML Suite and the use of the available tools in order to create and train the models based on the existing data. By using well-established and stable methods such as regression analysis, clustering and neural networks, the properly transformed data are fed to the training models.
- *Model Evaluation Module:* during the Model Development phase, a certain number of ML models that are developed will be able to satisfy the needs expressed by the end-users (as defined in the developed use cases) and they will constitute the building blocks of the upper MATRYCS Analytics Layer. These ML models, after development and training, need a process of evaluation and refinement through appropriate techniques that determine for example the accuracy, performance and error level, which is necessary before the models can be eventually served. The Model Evaluation Module aims at covering this specific task.
- *Model Serving Framework:* The Model Serving Framework represents the bridge between the underlying MATRYCS Governance Layer and the upper-level MATRYCS Analytics Layer. This serves the ML models available under the Trained Models library and those already trained and evaluated by ML developers via the Model Evaluation Module to the upper level MATRYCS Analytics Toolbox in order to allow their use for the design of complex services and applications for the built environment.

5.3. MATRYCS Analytics Layer

The MATRYCS Analytics Layer hosts the collection of building services and applications that are eventually developed to address specific use cases in the building scenario. These analytical services aim to fulfill the specific needs of the building use cases as captured from different stakeholders, but also for offering innovative functionalities that can be important to improve the building management from different perspectives and scales. A non-exhaustive list of building services that can be provided at this level includes:

- Analytics for *building energy performance* evaluation and optimization, which may include services for indoor condition evaluation, intelligent building energy management and building automation control.
- Analytics to facilitate *building design* such as for the identification of refurbishment needs, the evaluation of energy conservation measures and the assessment of retrofitting actions.
- Analytics in support of *policy making and policy impact assessment* on different scales, such as sustainable energy and climate action plans as well as the evaluation and harmonization of energy performance certificates.
- Analytics addressing *business and financial* aspects, such as de-risking energy efficiency investments as well as the measurement and verification of energy services.
- Applications for general purposes, such as geoclustering and digital twin.

The Analytics Services constitute the overall MATRYCS Toolbox and they can be exploited to implement holistic energy services to create representations of physical systems

such as buildings and energy systems, and to perform simulations, examine scenarios and make predictions. Figure 6 depicts the overall MATRYCS Analytics Service Layer. As shown, in addition to the different applications, this layer also includes the tools for data visualization, for the interaction of the end users with the MATRYCS ecosystem and a virtual workbench to unlock the development of analytics services from external developers.

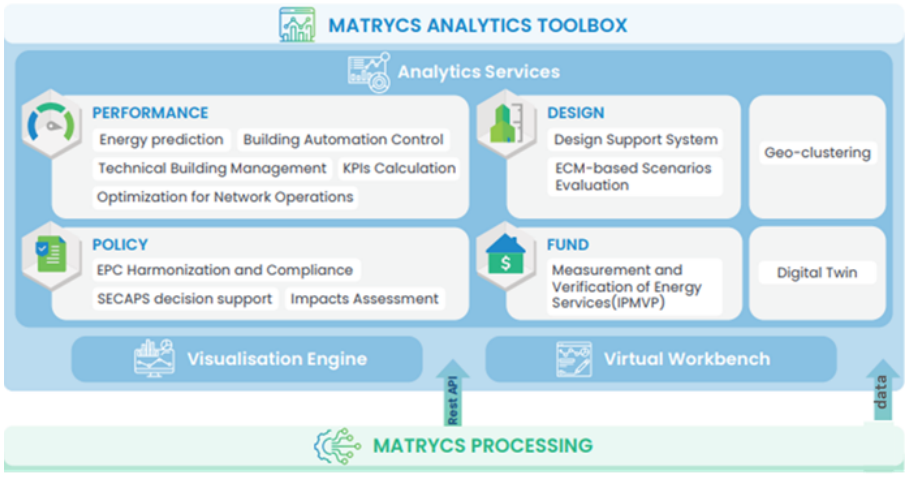

Figure 6. MATRYCS analytics layer architecture and development.

5.4. MATRYCS Security Layer

The MATRYCS architecture needs a vertical security layer spanning and interacting with the different blocks of the MATRYCS architecture for enabling the authentication, authorization and logging of various events in the system and the enforcement of security as well as privacy aspects. The MATRYCS end-to-end security framework is thus considered a cross-cutting security layer that covers the MATRYCS governance, MATRYCS processing and MATRYCS analytics layers. Specifically, the framework encompasses and relates to several entities in the MATRYCS architecture: infrastructure/assets, AI/ML services with a focus on big data, MATRYCS end-users, and data. The end-to-end security framework aims to secure the MATRYCS platform and its constituent information, thus enhancing the trustworthiness of the system by applying high-level security and fine-grained access control as well as appropriate mechanisms for maintaining and reinforcing legal and security policies over the MATRYCS resources. Role-based policies are created in order to grant permissions to resources such as the analytics services to be accessed users. Figure 7 shows the connection of the MATRYCS Toolbox services through clients to secure their interfaces.

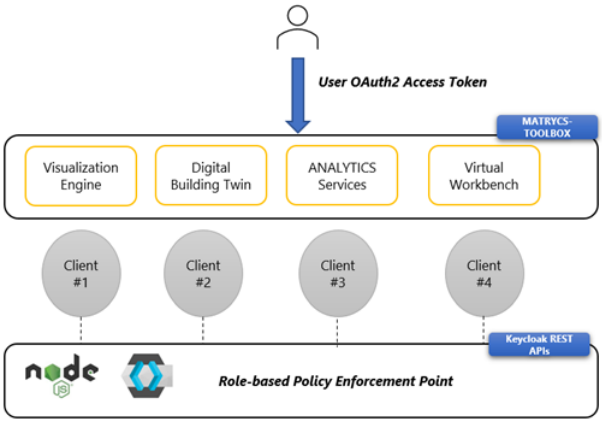

Figure 7. MATRYCS Toolbox integration with security layer.

6. Exemplary Architecture Instantiation and Use Case Study

The MATRYCS architecture presented in Section 5 is intended to be a high-level software architecture, namely an abstract and high-level view of the software functionalities that should be embedded when implementing a MATRYCS platform or ecosystem, together with their interfaces and interconnections. Accordingly, the proposed architecture only provides the software specifications without placing any constraint on the specific technologies that should be used for the practical implementation. Figure 8 shows, however, an example of open source technologies that may be adopted for a concrete instantiation of the different software components of this architecture.

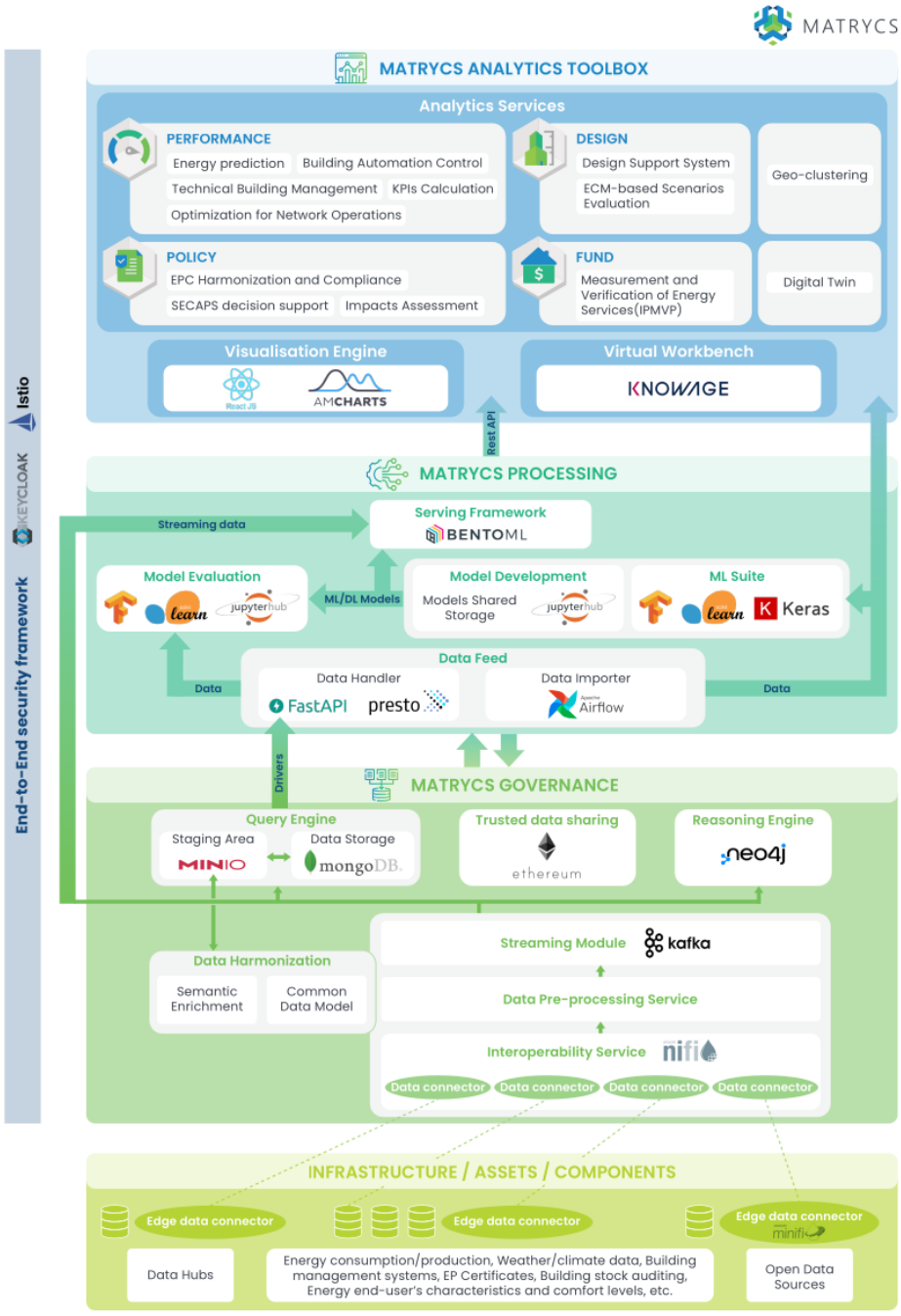

Figure 8. Practical instantiation of the MATRYCS architecture with open source technologies.

As mentioned in the previous sections, the main goals of the proposed architecture are as follows: (i) to ensure the integration and harmonization of large amounts of heteroge-

neous data provided by different data sources; (ii) to provide a framework for the provision of advanced AI methods able to extract additional value from the available data; (iii) to enable the design of complex building services and applications based on cross-domain data, which would not be otherwise possible when having closed data silos. To discuss how the MATRYCS architecture unlocks these features, the example of a large facility composed of business, commercial, entertainment and logistic centers is taken as a sample use case study. The considered facility is equipped with:

- Air handling units that are used to regulate and circulate air as part of a heating, ventilating and air-conditioning (HVAC) system as well as calorimeters at the heating substation and other temperature sensors. A dedicated building management system is responsible for controlling the operation of the HVAC system.
- An energy management system that collects data from electricity meters associated with the different centers of the facility and from the renewable energy sources installed on the facility premises.
- A management system that allows tracking the occupancy of different areas (via ad hoc sensors) and that is responsible for the execution of the operating schedules of the lighting system.
- An additional management system that is used for controlling the operation of refrigeration units and for monitoring the temperature in the refrigeration chambers of the cold storage of some existing warehouses.

At the current stage, these different processes are supervised by independent management systems without any specific coordination among them. Moreover, data related to each domain are handled in completely different ways, without a common approach and employing diverse technologies (e.g., different data models, databases). In this context, a MATRYCS-compliant platform can be implemented as follows, to obtain the following important benefits.

- *Governance layer:* via its interoperability connectors, the governance layer supports different communication protocols such as SFTP, HTTP, MQTT, AMQP and AVRO and it would allow the connection and retrieval of all sensors and other metering data. The Data Preprocessing Service would be responsible for the harmonization of the data of different facility centers, also belonging to different domains, by using a common data scheme based, for example, on the BRICK ontology or on a hybrid Fiware data model. This facilitates the re-use of these data and fosters interoperability. The Streaming Module can be used to distribute the real-time operational data to the Data Storage components in an asynchronous manner, which consist of a Data Storage and Distributed Query Engine (for storing the time series data) and a Reasoning Engine [68] (for the storage of metadata and other enriched data coming from the Semantic Enrichment module).
- *Processing layer:* the Data Feed Module acts as connection point to the governance layer and it collects and then transforms the stored data to prepare them for the Model Development component, wherein the training of the AI models is carried out. Multiple pre-selected machine learning algorithms (Regressors (Random Forest, Lasso, Linear, Decision Trees), Long Short Term Memory neural networks, etc.) may be executed to conduct predictions or to extract other features over the aggregated (and possibly cross-domain) data. When the training phase is complete, the Evaluation Module is employed to evaluate the metrics of the trained models. If the validation is successful, the Serving Framework would then expose the predictions (or other AI results) to the upper level MATRYCS Analytics services.
- *Analytics Layer:* the analytics layer contains the MATRYCS Toolbox where the building Digital Twin and other advanced building services may be offered. Front-end services can rely upon the served trained models from the MATRYCS processing layer to enable energy prediction and other energy efficiency services, and to provide advanced visualizations and reports to the end users.

- *Security:* security policies must be put in place to guarantee the security of the MATRYCS ecosystem. Role-based access control using, for example, the OAuth2 and UMA 2.0 security standards, can be adopted to permit or deny the access of different user groups to the MATRYCS ecosystem.

Figure 9 summarizes the described flow of operations over the different MATRYCS architecture layers for the considered use case. The data collection process is initiated from the interoperability service of the MATRYCS ecosystem. After ingestion, the building data are harmonized according to MATRYCS data model and organized in a directory format into the staging area. The Data Feed Module is responsible for the data cube integration as it receives harmonized building data from the MATRYCS staging area and performs pre-analytics steps such as the removal of duplicates, null values handling and outlier detection. Subsequently, the processed data are stored into MongoDB collections, where a REST API is utilized to enable aggregations and to feed a visualization engine that facilitates the exploration of building information. Furthermore, a collection of REST services called the MATRYCS Data Handler is used for preparing the data for training the machine learning models (pre-mining phase [69]) over stored processed data. This pre-mining phase consists of operations responsible for selections, group-bys, min–max scaling, categorical encoding, timeseries preparation and average smoothing. During the models training, techniques such as clustering timeseries regression are applied for supporting the building automation control and energy consumption prediction. Algorithms such as Random Forest Regressors, XGBoost and Arima are leveraged. Moreover, neural networks such as LSTMs and RNNs, which are suitable for regression and timeseries predictions when having significant amounts of data, are employed. After training the model, weights are stored into the MATRYCS Models storage where, on top of it, various applications for building automation control, energy prediction and building management have been implemented to provide the data-driven decision support for the final end users.

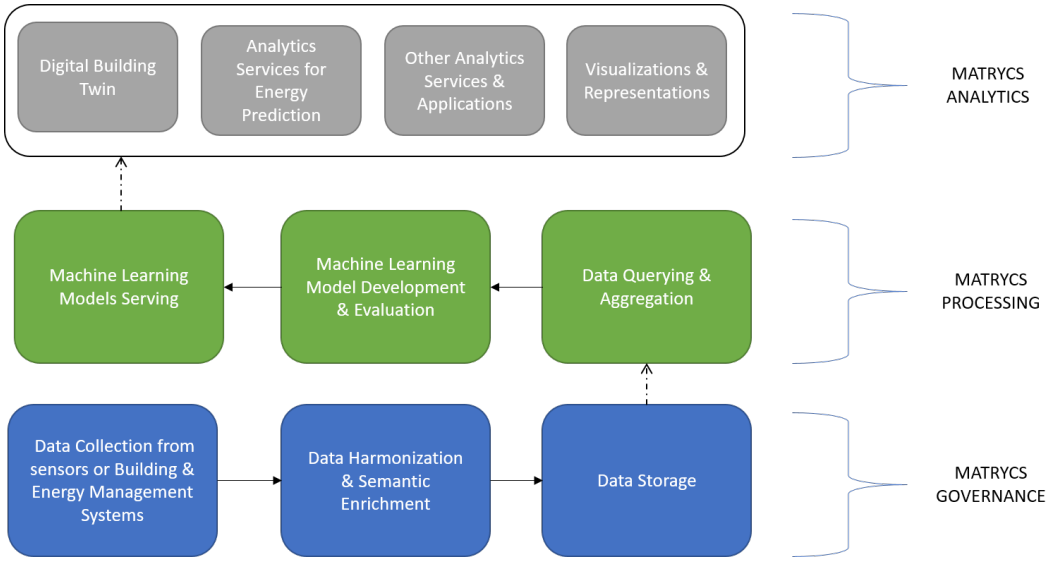

Figure 9. Data flow over the MATRYCS layers.

Concerning the benefits unlocked by the MATRYCS architecture, one of them is the interoperable and harmonized representation of heterogeneous data by means of a common data model. To this purpose, it is worth underlining that the common data model should be based on open ontologies and data schemas in order to guarantee the possibility of sharing the data with multiple applications potentially developed by different vendors (in line with the principles of openness, data sharing and no vendor lock-in mentioned in Section 4). This is particularly relevant because, in turn, it permits the use of the same data for multiple services and to design complex applications that make simultaneous use of data belonging to different domains. With reference to the presented use case, in

comparison to the siloed management and control currently in use, the deployment of open data eventually brings the possibility of implementing several additional services of potential interest for the facility manager, such as:

- A building digital twin based on the combination of the static building information (building registration number, building boundaries and geometry, building condition, building address, number of units, number of dwellings, number of floors, building U-values, etc.) and real-time operational data (e.g., sensors and meters data). The digital twin can then be simply used for energy performance monitoring or to verify energy savings, for anomaly detection and predictive maintenance.
- The coordinated energy management of subsystems associated with different energy vectors within a holistic BMS. This also involves the exploitation of the renewable energy sources to, for example, maximize the self-consumption and minimize the net power exchange with the electric grid.
- The evaluation of the overall flexibility coming from the different subsystems (heating, lighting, etc.) for offering it into future markets of power system flexibility.
- The accurate prediction of different quantities (e.g., heating and electricity demand) also using cross-domain information whenever beneficial.
- The computation of key performance indicators for the identification of possible needs for energy efficiency improvements or other energy saving measures in any of the buildings of the facility.

Overall, the above services would allow identifying and pursuing energy performance improvements that are not possible today due to the isolated deployment of different management systems. The proposed architecture would enable the design and provision of such services, thus opening opportunities for energy efficiency improvements and for further business cases for both the facility manager and other related stakeholders.

7. Conclusions

With the ongoing digitalization of the building sector, the role of data is becoming increasingly important. In this paper, an IoT-based architecture has been presented, which aims at providing a general framework for the development of a distributed ecosystem of platforms that interact to exchange data and to create business value via the development and connection of advanced data-driven services. The design of the architecture relies upon a detailed analysis of the stakeholders, functional and non-functional requirements associated to different applications and use cases related to the built environment, which have been here discussed to derive the architecture specifications. The proposed architecture is composed of several layers, each one addressing a specific step of the data value chain, thus allowing one to handle raw data, to elaborate them via machine learning and artificial intelligence modules and finally, to develop complex applications based on the enriched data and information created via the machine learning tools. The perspective presented in this paper mostly focuses on the identification of the needed software blocks in each layer and on their functional interconnection, thus serving as a reference for the possible implementation of platforms and building data ecosystems aiming to host turnkey services for the building scenario. While no constraints exist for the software technologies to be adopted, this paper additionally presents an exemplary instantiation of such an architecture, indicating specific technologies that may be used for the concrete implementation of a platform compliant with the proposed architectural view. This can thus serve as a reference for better visualizing how to translate the abstract architectural model into a more concrete and practical implementation of the software architecture. Moreover, a real use case study was presented to give a tangible view of how the proposed architecture may help in deploying innovative services and applications in support of energy efficiency and energy performance improvements. Future activities will mainly focus on the deployment of the described architecture in several pilots for testing its capabilities under different points of view and to analyze in greater depth the impact of the digitalization process on the energy performance of selected scenarios and use cases.

Author Contributions: Conceptualization, M.P., P.K. and D.P.; methodology, M.P. and P.K.; software, Z.P., P.K., G.K. and D.P.; validation, M.P., P.K. and D.P.; formal analysis, M.P., Z.P., P.K. and G.K.; investigation, M.P. and P.K.; writing—original draft preparation, M.P., Z.P., P.K. and G.K.; writing—review and editing, M.P., Z.P., P.K., G.K. and D.P.; supervision, M.P., P.K. and A.M.; funding acquisition, D.P. and A.M. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by MATRYCS, which is a European project funded by the European Union's Horizon 2020 research and innovation program under Grant Agreement No. 1010000158.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank all the MATRYCS consortium partners and especially Gema Hernandez Moral (CARTIF), Sofia Mulero Palencia (CARTIF), Zoi Mylona (HOLISTIC), Daniele Antonucci (EURAC) and Tomaz Damjan (BTC) for the fruitful discussions, remarks and observations during the project meetings.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. European Commission. Energy Efficiency in Buildings. 2020. Available online: https://ec.europa.eu/info/news/focus-energy-efficiency-buildings-2020-feb-17_en (accessed on 19 February 2022).
2. European Commission. Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the Energy Performance of Buildings. 2010. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32010L0031&from=EN (accessed on 19 February 2022).
3. European Commission. Directive 2012/27/EU of the European Parliament and of the Council of 25 October 2012 on Energy Efficiency, Amending Directives 2009/125/EC and 2010/30/EU and Repealing Directives 2004/8/EC and 2006/32/EC. 2012. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32012L0027&from=EN (accessed on 19 February 2022).
4. European Commission. Directive 2018/844/EU of the European Parliament and of the Council of 30 May 2018 Amending Directive 2010/31/EU on the Energy Performance of Buildings and Directive 2012/27/EU on Energy Efficiency. 2018. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32018L0844&from=EN (accessed on 19 February 2022).
5. European Commission. Clean Energy for all Europeans Package. 2019. Available online: https://energy.ec.europa.eu/topics/energy-strategy/clean-energy-all-europeans-package_en (accessed on 19 February 2022).
6. European Commission. Proposal for a Directive of the European Parliament and of the Council on the Energy Performance of Buildings. 2021. Available online: https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:52021PC0802&from=EN (accessed on 19 February 2022).
7. Lasarte, N.; Elguezabal, P.; Sagarna, M.; Leon, I.; Otaduy, J.P. Challenges for Digitalisation in Building Renovation to Enhance the Efficiency of the Process: A Spanish Case Study. *Sustainability* **2021**, *13*, 12139. [CrossRef]
8. Kyritsis, A.; Mathas, E.; Antonucci, D.; Grottke, M.; Tselepis, S. Energy improvement of office buildings in Southern Europe. *Energy Build.* **2016**, *123*, 17–33. [CrossRef]
9. Ke, M.T.; Yeh, C.H.; Su, C.J. Cloud computing platform for real-time measurement and verification of energy performance. *Appl. Energy* **2017**, *188*, 497–507. [CrossRef]
10. Nagy, Z.; Yong, F.Y.; Frei, M.; Schlueter, A. Occupant centered lighting control for comfort and energy efficient building operation. *Energy Build.* **2015**, *94*, 100–108. [CrossRef]
11. Yang, L.; Nagy, Z.; Goffin, P.; Schlueter, A. Reinforcement learning for optimal control of low exergy buildings. *Appl. Energy* **2015**, *156*, 577–586. [CrossRef]
12. Peng, Y.; Rysanek, A.; Nagy, Z.; Schlüter, A. Using machine learning techniques for occupancy-prediction-based cooling control in office buildings. *Appl. Energy* **2018**, *211*, 1343–1358. [CrossRef]
13. Agostinelli, S.; Cumo, F.; Guidi, G.; Tomazzoli, C. Cyber-Physical Systems Improving Building Energy Management: Digital Twin and Artificial Intelligence. *Energies* **2021**, *14*, 2338. [CrossRef]
14. Xu, X.; Wang, W.; Hong, T.; Chen, J. Incorporating machine learning with building network analysis to predict multi-building energy use. *Energy Build.* **2019**, *186*, 80–97. [CrossRef]
15. Seyedzadeh, S.; Pour Rahimian, F.; Rastogi, P.; Glesk, I. Tuning machine learning models for prediction of building energy loads. *Sustain. Cities Soc.* **2019**, *47*, 101484. [CrossRef]
16. Mohammadiziazi, R.; Bilec, M.M. Application of Machine Learning for Predicting Building Energy Use at Different Temporal and Spatial Resolution under Climate Change in USA. *Buildings* **2020**, *10*, 139. [CrossRef]
17. Robinson, C.; Dilkina, B.; Hubbs, J.; Zhang, W.; Guhathakurta, S.; Brown, M.A.; Pendyala, R.M. Machine learning approaches for estimating commercial building energy consumption. *Appl. Energy* **2017**, *208*, 889–904. [CrossRef]

18. Attanasio, A.; Savino Piscitelli, M.; Chiusano, S.; Capozzoli, A.; Cerquitelli, T. Towards an Automated, Fast and Interpretable Estimation Model of Heating Energy Demand: A Data-Driven Approach Exploiting Building Energy Certificates. *Energies* **2019**, *12*, 1273. [CrossRef]
19. Chiosa, R.; Piscitelli, M.S.; Capozzoli, A. A Data Analytics-Based Energy Information System (EIS) Tool to Perform Meter-Level Anomaly Detection and Diagnosis in Buildings. *Energies* **2021**, *14*, 237. [CrossRef]
20. Manfren, M.; Nastasi, B. Parametric Performance Analysis and Energy Model Calibration Workflow Integration—A Scalable Approach for Buildings. *Energies* **2020**, *13*, 621. [CrossRef]
21. von Platten, J.; Sandels, C.; Jörgensson, K.; Karlsson, V.; Mangold, M.; Mjörnell, K. Using Machine Learning to Enrich Building Databases—Methods for Tailored Energy Retrofits. *Energies* **2020**, *13*, 2574. [CrossRef]
22. Gallagher, C.V.; Leahy, K.; O'Donovan, P.; Bruton, K.; O'Sullivan, D.T. Development and application of a machine learning supported methodology for measurement and verification (M&V) 2.0. *Energy Build.* **2018**, *167*, 8–22.
23. Mannino, A.; Dejaco, M.C.; Re Cecconi, F. Building Information Modelling and Internet of Things Integration for Facility Management—Literature Review and Future Needs. *Appl. Sci.* **2021**, *11*, 3062. [CrossRef]
24. Charef, R.; Emmitt, S.; Alaka, H.; Fouchal, F. Building Information Modelling adoption in the European Union: An overview. *J. Build. Eng.* **2019**, *25*, 100777. [CrossRef]
25. Borrmann, A.; König, M.; Koch, C.; Beetz, J. *Building Information Modeling—Technology Foundations and Industry Practice*; Springer: Cham, Switzerland, 2018.
26. European Telecommunications Standards Institute. SAREF Extension for Buildings. 2020. Available online: https://saref.etsi.org/saref4bldg/v1.1.2/ (accessed on 21 March 2022).
27. Brick Consortium. Brick—A Uniform Metadata Schema for Buildings. 2021. Available online: https://brickschema.org/ (accessed on 21 March 2022).
28. Mariano-Hernandez, D.; Hernandez-Callejo, L.; Zorita-Lamadrid, A.; Duque-Perez, O.; Santos Garcia, F. A review of strategies for building energy management system: Model predictive control, demand side management, optimization, and fault detect and diagnosis. *J. Build. Eng.* **2021**, *33*, 101692. [CrossRef]
29. Khajavi, S.H.; Motlagh, N.H.; Jaribion, A.; Werner, L.C.; Holmström, J. Digital Twin: Vision, Benefits, Boundaries, and Creation for Buildings. *IEEE Access* **2019**, *7*, 147406–147419. [CrossRef]
30. Shahzad, M.; Shafiq, M.T.; Douglas, D.; Kassem, M. Digital Twins in Built Environments: An Investigation of the Characteristics, Applications, and Challenges. *Buildings* **2022**, *12*, 120. [CrossRef]
31. Minoli, D.; Sohraby, K.; Occhiogrosso, B. IoT Considerations, Requirements, and Architectures for Smart Buildings—Energy Optimization and Next-Generation Building Management Systems. *IEEE Internet Things J.* **2017**, *4*, 269–283. [CrossRef]
32. Buyya, R.; Broberg, J.; Goscinski, A. *Cloud Computing: Principles and Paradigms*; Wiley: Hoboken, NJ, USA, 2011.
33. Vivian, J.; Prataviera, E.; Cunsolo, F.; Pau, M. Demand Side Management of a pool of air source heat pumps for space heating and domestic hot water production in a residential district. *Energy Convers. Manag.* **2020**, *225*, 113457. [CrossRef]
34. Ma, Z.; Clausen, A.; Lin, Y.; Jorgensen, B.N. An overview of digitalization for the building-to-grid ecosystem. *Energy Inform.* **2021**, *4*, 36. [CrossRef]
35. Apanaviciene, R.; Vanagas, A.; Fokaides, P.A. Smart Building Integration into a Smart City (SBISC): Development of a New Evaluation Framework. *Energies* **2020**, *13*, 2190. [CrossRef]
36. Hernandez-Moral, G.; Mulero-Palencia, S.; Serna-Gonzalez, V.I.; Rodriguez-Alonso, C.; Sanz-Jimeno, R.; Marinakis, V.; Dimitropoulos, N.; Mylona, Z.; Antonucci, D.; Doukas, H. Big Data Value Chain: Multiple Perspectives for the Built Environment. *Energies* **2021**, *14*, 4624. [CrossRef]
37. Boyes, H. Security, Privacy, and the Built Environment. *IT Prof.* **2015**, *17*, 25–31. [CrossRef]
38. Molina-Solana, M.; Ros, M.; Ruiz, M.D.; Gomez-Romero, J.; Martin-Bautista, M. Data science for building energy management: A review. *Renew. Sustain. Energy Rev.* **2017**, *70*, 598–609. [CrossRef]
39. Qolomany, B.; Al-Fuqaha, A.; Gupta, A.; Benhaddou, D.; Alwajidi, S.; Qadir, J.; Fong, A.C. Leveraging Machine Learning and Big Data for Smart Buildings: A Comprehensive Survey. *IEEE Access* **2019**, *7*, 90316–90356. [CrossRef]
40. Hong, T.; Wang, Z.; Luo, X.; Zhang, W. State-of-the-art on research and applications of machine learning in the building life cycle. *Energy Build.* **2020**, *212*, 109831. [CrossRef]
41. Alanne, K.; Sierla, S. An overview of machine learning applications for smart buildings. *Sustain. Cities Soc.* **2022**, *76*, 103445. [CrossRef]
42. Marinakis, V. Big Data for Energy Management and Energy-Efficient Buildings. *Energies* **2020**, *13*, 1555. [CrossRef]
43. MATRYCS—Modular Big Data Applications for Holistic Energy Services in Buildings. Available online: https://matrycs.eu/ (accessed on 21 February 2022).
44. Big Data Value Association. European Big Data Value Strategic Research and Innovation Agenda (Version 4.0). 2017. Available online: http://bdva.eu/sites/default/files/BDVA_SRIA_v4_Ed1.1.pdf (accessed on 25 February 2022).
45. Industrial Internet Consortium. The Industrial Internet of Things Volume G1: Reference Architecture. 2017. Available online: http://www.iiconsortium.org/IIRA.htm (accessed on 25 February 2022).
46. Alliance for Internet of Things Innovation. High Level Architecture (HLA)—Release 5.0. 2020. Available online: https://aioti.eu/wp-content/uploads/2020/12/AIOTI_HLA_R5_201221_Published.pdf (accessed on 25 February 2022).

47. Fiware Foundation. FIWARE for Data Spaces—Version 1.0. 2021. Available online: https://www.hannovermesse.de/apollo/hannover_messe_2021/obs/Binary/A1085838/FIWARE%20for%20Data%20Spaces%20%281%29.pdf (accessed on 25 February 2022).
48. International Data Space Association. The Industrial Internet of Things Volume G1: Reference Architecture. 2019. Available online: https://www.fraunhofer.de/content/dam/zv/en/fields-of-research/industrial-data-space/IDS-Reference-Architecture-Model.pdf (accessed on 25 February 2022).
49. Curry, E. The Big Data Value Chain: Definitions, Concepts, and Theoretical Approaches. In *New Horizons for a Data-Driven Economy*; Springer: Cham, Switzerland, 2011; pp. 29–37.
50. Calzada, I. Data Co-Operatives through Data Sovereignty. *Smart Cities* **2021**, *4*, 1158–1172. [CrossRef]
51. Rouhani, S.; Deters, R. Data Trust Framework Using Blockchain Technology and Adaptive Transaction Validation. *IEEE Access* **2021**, *9*, 90379–90391. [CrossRef]
52. Tosh, D.; Shetty, S.; Liang, X.; Kamhoua, C.; Njilla, L.L. Data Provenance in the Cloud: A Blockchain-Based Approach. *IEEE Consum. Electron. Mag.* **2019**, *8*, 38–44. [CrossRef]
53. Borgogno, O.; Colangelo, G. Data sharing and interoperability: Fostering innovation and competition through APIs. *Comput. Law Secur. Rev.* **2019**, *35*, 105314. doi: 10.1016/j.clsr.2019.03.008. [CrossRef]
54. Jung, S.; Jeoung, J.; Hong, T. Occupant-centered real-time control of indoor temperature using deep learning algorithms. *Build. Environ.* **2022**, *208*, 108633. [CrossRef]
55. Sembroiz, D.; Careglio, D.; Ricciardi, S.; Fiore, U. Planning and operational energy optimization solutions for smart buildings. *Inf. Sci.* **2019**, *476*, 439–452. [CrossRef]
56. Eugster, P.T.; Felber, P.A.; Guerraoui, R.; Kermarrec, A.M. The Many Faces of Publish/Subscribe. *ACM Comput. Surv.* **2003**, *35*, 114–131. doi: 10.1145/857076.857078. [CrossRef]
57. Rahman, H.; Hussain, M.I. A comprehensive survey on semantic interoperability for Internet of Things: State-of-the-art and research challenges. *Trans. Emerg. Telecommun. Technol.* **2020**, *31*, e3902. [CrossRef]
58. Newman, S. *Building Microservices*; O'Reilly Media, Inc.: Newton, MA, USA, 2015.
59. Pan, J.; McElhannon, J. Future Edge Cloud and Edge Computing for Internet of Things Applications. *IEEE Internet Things J.* **2018**, *5*, 439–449. [CrossRef]
60. Xing, L. Reliability in Internet of Things: Current Status and Future Perspectives. *IEEE Internet Things J.* **2020**, *7*, 6704–6721. [CrossRef]
61. Pahl, C.; Brogi, A.; Soldani, J.; Jamshidi, P. Cloud Container Technologies: A State-of-the-Art Review. *IEEE Trans. Cloud Comput.* **2019**, *7*, 677–692. [CrossRef]
62. Hassija, V.; Chamola, V.; Saxena, V.; Jain, D.; Goyal, P.; Sikdar, B. A Survey on IoT Security: Application Areas, Security Threats, and Solution Architectures. *IEEE Access* **2019**, *7*, 82721–82743. [CrossRef]
63. Lu, Y.; Xu, L.D. Internet of Things (IoT) Cybersecurity Research: A Review of Current Research Topics. *IEEE Internet Things J.* **2019**, *6*, 2103–2115. [CrossRef]
64. Sowmya, S.; Deepika, P.; Naren, J. Layers of cloud—IaaS, PaaS and SaaS: A Survey. *Int. J. Comput. Sci. Inf. Technol.* **2014**, *5*, 4477–4480.
65. Marguglio, A. Reference Architecture for Cross-Domain Digital Transformation. 2020. Available online: https://www.opendei.eu/wp-content/uploads/2020/10/D2.1-REF-ARCH-FOR-CROSS-DOMAIN-DT-V1_UPDATED.pdf (accessed on 25 February 2022).
66. Marinakis, V.; Doukas, H.; Tsapelas, J.; Mouzakitis, S.; Sicilia, Á.; Madrazo, L.; Sgouridis, S. From big data to smart energy services: An application for intelligent energy management. *Future Gener. Comput. Syst.* **2020**, *110*, 572–586. [CrossRef]
67. Marinakis, V.; Doukas, H. An Advanced IoT-based System for Intelligent Energy Management in Buildings. *Sensors* **2018**, *18*, 610. [CrossRef]
68. Kapsalis, P.; Kormpakis, G.; Alexakis, K.; Askounis, D. Leveraging Graph Analytics for Energy Efficiency Certificates. *Energies* **2022**, *15*, 1500. [CrossRef]
69. Leprince, J.; Miller, C.; Zeiler, W. Data mining cubes for buildings, a generic framework for multidimensional analytics of building performance data. *Energy Build.* **2021**, *248*, 111195. [CrossRef]