This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

Materializing the Promises of Cognitive IoT: How Cognitive Buildings are Shaping the Way

Joern Ploennigs, *Senior Member, IEEE,* Amadou Ba, and Michael Barry

***Abstract*—Relatively tiny examples have demonstrated the potential of Cognitive IoT (CIoT) in its full-stack, namely semantic modelling, learning and reasoning over sensors data, and machine learning, to uncover and expose actionable insights via advanced user interfaces. In this paper, we make the case for the feasibility of CIoT in all of its dimensions. We devise a CIoT architecture that integrates thousands of sensors present in our buildings in order to learn the buildings' behaviour and intuitively assist users in diagnosing and mitigating undesired events. With our architecture, we place emphasis on the scalability and flexibility that reduce the configuration effort. The solution shows the potential of CIoT to create highly scalable, adaptable and interactive IoT systems functioning for buildings and capable of addressing the challenges encountered in the realm of Homes, Smart Cities and Industry 4.0.**

***Index Terms*—Cognitive IoT, fog and cloud computing, automated analytics, semantic modeling, learning and reasoning, augmented reality.**

I. Introduction

To materialize the benefits of the Internet of Things (IoT), improved infrastructures, architectures, and advanced analytics are needed to cope with the increasing volume, dynamicity, heterogeneity and distribution of IoT systems and data. In particular, existing architectures and analytics for IoT systems face stringent requirements to efficiently deal with the variety of IoT devices and the subsequent big data volumes. To address these requirements, IoT architectures need to adapt to the changing modus operandi of IoT systems, and support the easy deployment of advanced analytics—when for example on-line requirements are desired to extract deep insights. This has proven to be increasingly challenging. The exploitation of cognitive computing [1] in conjunction with IoT—also known as *Cognitive IoT* (CIoT) [2], [3]—enhances existing IoT systems by providing them with self-learning and self-adaption capabilities that faciliate scalability, and insight extraction in large scale systems with many thousands of devices. This provides CIoT systems with the mechanisms to usher in an era of unprecedented changes by empowering things with embedded intelligence that leads to improved optimization of processes, and ultimately to substantial benefits in all application areas—from manufacturing, to healthcare, to buildings etc.

The application of CIoT paradigm to buildings is pertinent for the objective of validating both its feasibility and viability. Buildings are large and complex IoT systems equipped with a multitude of diverse devices, in the order of tens of thousands. Their complexity is further exacerbated by their peculiarities as each building has a specific configuration of its devices, which challenges the deployment of analytics at large scale.

J. Ploennigs, A. Ba, and M. Barry are with IBM Research - Ireland

We take advantage of the advanced instrumentation to extend current approaches for improved operation of buildings, and demonstrate the potential of CIoT for energy management by optimizing energy consumption with the ultimate objective of reducing the CO_2 [4], whilst providing the users with better comfort. This equates *Cognitive Buildings* to the next generation of Smart Buildings that are capable of automatically processing, and extracting actionable insights from data generated by diverse buildings.

In this paper, we propose a CIoT architecture that combines the strength in scalability provided by recently developed IoT architectures with the self-learning and self-adaptation capabilities obtained from cognitive systems. The application of these architectures to buildings further represents an interesting test bed that enables the replication of the developed solutions to other areas. This is particularly evidenced by the fact that buildings encompass all the large-scale characteristics present in many IoT systems.

We review the state of the art in section II for CIoT in general, and for common IoT and CIoT architectures, in particular. Subsequent to the architectures, section III provides the trends and challenges relative to buildings, and make the case for their possible adaptation to the context of IoT systems, more broadly. Then, in section III-B we tackle scalability in big data analytics, and in section III-C we motivate the dynamic operations of IoT systems from buildings perspectives. In section IV-A we address semantic integration, self-learning and adaption of IoT systems, particularly of buildings. Finally, we apply our methodology to our use case for thermal comfort monitoring, where we address several buildings, the implementation details, along with the obtained performance.

II. State of the Art

A. Cognitive IoT

Cognitive IoT is the extension of IoT, where IoT systems are equipped with cognitive computing approaches allowing them to learn and reason over data, and extract deep actionable insights, while building a network where the physical world and the digital world are blended. Through this mechanism, the cognition of IoT systems endows them with the freedom to intelligently and autonomously operate. Subsequently, the IoT systems not only learn and reason based on the experiences that they gain from their interactions with their counterparts and their environments in general, but their learning and reasoning abilities are improved at the cadence of the new information that they acquire.

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

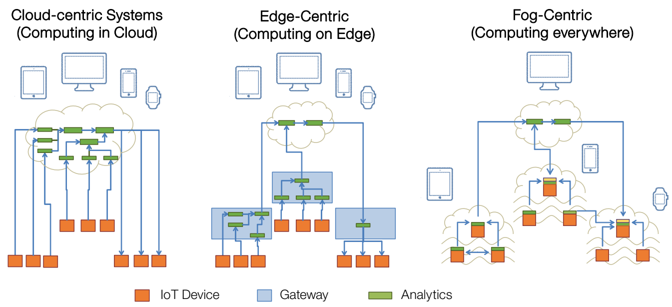

Fig. 1: Cloud-, edge-, and fog-centric architectures

In spite of its evident potential, research in CIoT is still at its earliest stages. The work in [5] led to a cognitive management framework for smarter cities. Their objective is to improve sustainability of cities by operating IoT systems in tandem with cognitive capabilities. Their cognitive approach identifies and connects the objects that are relevant for the application in question. In [6], the authors develop and present concepts and examples of applications pertaining to IoT. They, particularly, propose a cognitive system allowing for cooperation between various devices. By drawing inspiration from human cognition, the authors in [7] integrate the operational process of human cognition into the system design for better IoT systems. The exploitation of IoT in the context of Smart Home has been demonstrated in [8], where the authors present an innovative smart home system that enables emotion detection.

The use of wireless sensor networks allowed [9] to demonstrate the feasibility of CIoT. Their development expands on the requirements, and the issues related to adaptive systems and their architectures, and they present the corresponding intra-cognitive and inter-cognitive communications. The confluence of semantic computing, cognitive computing, and perceptual computing can help power future systems, particularly in the context of IoT is examined in [10].

These works illustrate several aspects of CIoT in early stages. However, large scale application and experimentation that demonstrate the full potential remain to be seen.

B. IoT Architectures

Several reference architectures for IoT have been developed both from academic and industrial perspectives. For example, the Industrial Internet Consortium Reference Architecture (IIRA) [11], IOT-A [12], ETSI oneM2M [13], AIOTI-HLA [14] and ITU-T IoT Architecture [15] are the leading concepts for the development of a reference model. Their objective is on the one hand to support the existence of a common understanding of the IoT, and on the other hand to provide a reference architecture that seeks to establish a common foundation for the development of interoperable IoT systems.

A common trait among the reference architectures is a-three tier pattern, which consists of:

- **The edge tier** abstracts data from edge nodes, where the characteristics depend on the specific use cases.
- **The platform tier** is responsible for consolidated processes and analysis of the data coming from the edge.

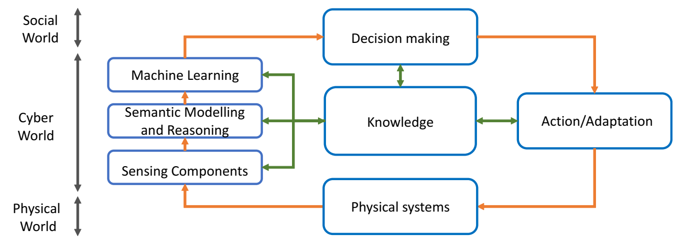

Fig. 2: Knowledge module in the CIoT architecture [7]

- **The enterprise tier** implements domain-specific applications, decision support tools and provides interfaces to end-users including operation specialists.

A differentiator for IoT architectures is the location of the analytics in the IoT system as shown in Fig. 1. IoT architectures accommodate the requirements of big and fast data processing in order to extract deep insights from the data by using cognitive computing capabilities into the IoT architectures.

- **Cloud-centric architectures** are useful when large amounts of data need to be joined to extract insights, and when the real-time and bandwidth constrained are not imposed.
- **Edge-centric architectures** are particularly suitable to real-time requirements and to satisfy privacy concerns. But, their local knowledge limits the derivable insights.
- **Fog-centric architectures** address the challenges posed by on-the-fly decision making requirements on heterogeneous and distributed systems, and when data integration between these systems is particularly needed.

C. Cognitive IoT Architectures

Different cognitive system architectures exist, including SOAR [16] and LIDA [17]—prominent agent-based frameworks. A cognitive architecture is sketched in [18] for buildings that utilises these frameworks. The common element of these architectures is a combination of reasoning, reinforced learning and emotional concepts (joy, fear, anger etc.) to enable the system to learn behaviours with rewards. It encodes the taken actions as rules that will be used in subsequent actions.

We argue that the strict focus on recreating human behaviour has several limitations in IoT. First, mapping emotional concepts to IoT is not simple, nor is its pertinence for decision making proven. Second, reasoning assumes that the systems behave in discrete and deterministic modes reducing the decision making process to binary decision rules. Third, reinforced learning neglects *a priori* knowledge available on the physical processes and the relevant external variables. IoT systems are usually driven by control strategies and many of the physical processes are well known. Common model-predictive control approaches for example use these physical models to predict the behaviour of these systems [19].

Also some cognitive IoT architectures exist like Fig. 2 [7]. They are *knowledge-centric* and their knowledge corpus is the central element of the cognitive computing system in the cyber

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

world that connects the physical world with the social world. The cognitive computing system is made of three components:

- **Sensing components:** acquire the critical information, relevant to the context of the physical systems, and allows the elaboration of the semantic meta model of the physical world.
- **Semantic modelling and reasoning:** uses the sensed information to build semantic models, which will serve afterward to the elaboration of physical and semantic reasoning.
- **Machine Learning:** advanced learning algorithms build on the existing semantic models to provide systems with self-learning capabilities.

The derived insights from machine learning algorithms are used to optimize the operations in the physical world by acting on the systems to automatically adapt their behaviours. This task takes place at the Actuation/Adaptation level.

D. Smart Building Architectures in the context of IoT

Modern IoT architectures have not been yet established in the building sector. Recent developments aim at facilitating the creation of open platforms that simplify data integration and processing. Dawson et al. [20] defined a layered software architecture that has a hardware abstraction layer to integrate various systems, a time series layer, and a software layer. Extension to this is given in [21] where an extensible architecture that includes more systems is provided. A similar architecture is proposed in [22], where an XMPP message bus as transport layer is utilized. While these architectures facilitate a hardware abstraction, they do not consider new IoT concepts nor cognitive elements.

III. TRENDS AND CHALLENGES RELATIVE TO BUILDINGS

Buildings are an example of a complex IoT system that facilitate the understanding of several technological and economic trends and challenges, since they are common to many IoT systems. We therefore analyze these trends and challenges in the buildings sector and, in the process, offer our insights into the trends and challenges associated to CIoT, in general.

A. Large scale, diversified instrumentation

a) History: The digitization of buildings started quite early to automate climate control. Fig. 3 provides an overview of the historical development, which started in the '70s with the automation of central systems such as boilers, chillers and air handling units (AHU). This was usually done through programmable logic controllers (PLC) to which all sensors and actuators were individually wired. The development of fieldbus networks at the end of the '80s allowed to connect all devices to one network. This simplified the installation and the control stretched to individual rooms.

b) State of Practice: It is surprising that the integration of buildings into the internet is still at its infancy, and most buildings still operate in isolation. The main reason for this is due to the fact that most operations for buildings have focused on the control of systems, and, from control perspectives, no direct benefit was derived in connecting buildings.

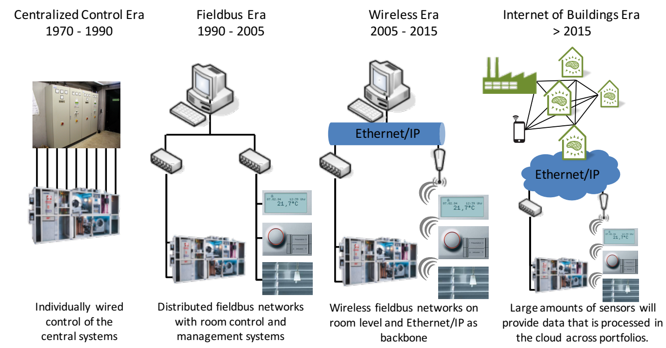

Fig. 3: Evolution from Centralised Control to the Internet of Buildings

IoT is changing this in two ways: (1) The increasing affordability of wireless IoT devices the instrumentation of legacy buildings with thousands of sensors down to individual workplace levels. This is accompanied by a diversification of devices with completely new class of devices such as smart mouse traps. (2) Data created by IoT systems create new economic value and its analysis opens completely new ways of optimizing building operations from comfort optimization to demand driven facility maintenance.

c) CIoT Challenge: IoT will lead to a substantial increase of diverse devices (multi-sensors, mobiles, wearable) in our environment. This creates a large integration challenge of devices into infrastructures as well as the backend analytics systems [23]. To address these challenges, approaches to *automatically integrate heterogeneous devices* into the analytics infrastructures are required.

B. Scalable Big Data Analytics

a) History: Fig. 4 shows the development of control and analytics in buildings. Traditional buildings were primarily automated. Since 2000, energy management systems have been introduced and take into account the analysis of the central systems. To enable further energy savings, it is essential to analyze all the available data, and to fully understand buildings behavior from supply to demand to occupant side.

b) State of Practice: It has been reported [24] that most building systems are not optimised for the designed energy consumption. Energy management systems are aiming at bridging this gap. To this end, they usually use rule based systems that require large manual effort to adapt them to the individual buildings. To limit effort, they only monitor central systems. But, they neglect the actual demand side of energy in buildings and the individual rooms where the energy is spent.

c) CIoT Challenge: IoT will lead to big data that cannot be analysed with the current manual approaches. Machine learning is well suited to *learn from large amount of data* and provide relevant insights that increase the operation performances. Technically, advanced analytics and big data platforms to analyze all data are available [25]. But, they also require major efforts of highly skilled data scientists in

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

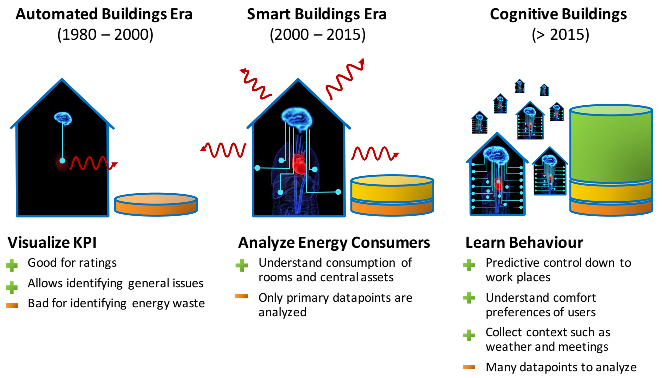

Fig. 4: Development of control and analytics in buildings

configurations to exploit their potential [23]. This process must be automated [26].

C. Dynamic and Mobile Operation Scenarios

a) History: Traditional systems for buildings were isolated and specialised on controls according to a set of defined setpoints and schedules. They were commissioned once and then left at these settings as there was no business need to adapt them due to cheap energy.

b) State of Practise: Modern operational conditions require a more dynamic operation and foster the development of the Internet of Buildings. The development of renewable energy sources led to a transformation of the grid to a smart grid [27] that supports a more flexible operation. Buildings consume 40 % of the energy [4] and are thus a central target of *demand response*, to better control the demand. The collection of data in the cloud eases its access from mobile devices for operators and occupants alike. This leads to a fundamental change in the operation of buildings towards dynamic adaption driven by user inputs.

c) CIoT Challenge: IoT enables a more dynamic and demand oriented operation of businesses. Future CIoT have to be able to *adapt to changing requirements and conditions*. They have to understand the reasoning of users and act on their behalf in a predictive manner. Moreover, the large amount of data requires new user interfaces that remove the complexity and allow the users to *naturally interact with systems* such that the systems become natural elements of our environment.

IV. Cognitive IoT Features

Based on the analysis made in section III, we derive in section IV-A some key features of a Cognitive IoT platform.

A. Semantic Interoperability & Automated Analytics

IoT lowers the barrier for installing more, diverse sensors in our environment. *Semantic Interoperability* is key to easily integrate these devices into cognitive systems and automate workflows that enable CIoT. Only if the diversity of devices can be mapped to a common knowledge model, it becomes possible to learn across the devices and systems automatically.

The analysis in [23] of 89 publications in the Smart Buildings area revealed that *automating analytic applications requires additional metadata* about the devices and relationships between the metadata. The most relevant relations cover the implemented semantic functions, the locations, and the observed cyber-physical systems (e. g. a heating system). Different approaches are reviewed in [28] to *automate the extraction of metadata* from available sources. The most promising are semi automated data mining approaches that learn common metadata schemata.

B. Self-Learning and Adaptation

CIoT systems need to be able to automatically process IoT data and learn from it. The analysis of analytic application in buildings [23] showed that various machine learning approaches are used depending on the application. So, there is no silver bullet ML approach, therefore, a CIoT system should be able to facilitate the existence of multiple approaches as *prescriptive micro-services* and link them together.

C. Natural Intuitive Interface

CIoT will only be successful if it is usable by everybody. One of the main reasons preventing the application of analytics in buildings is the requirements of skills in analytics. The operators and occupants of buildings are not data scientists. Therefore, interfaces should enable a building operator to intuitively understand the behaviour of a building. The interfaces should be easy to use *intuitive*, *responsive*, *mobile* and abstract the complexity of the underlying systems.

CIoT should provide *multi-modal user interfaces* that allow alternative interactions with the systems beyond graphical user interfaces. The integration of *speech interfaces* such as Amazon Echo shows great potential to interact in natural language with the operators and occupants. Also *augmented reality* interfaces are paving new ways to seamlessly access to sensors and systems data.

V. Cognitive IoT Architecture adapted to Buildings

Our Cognitive IoT architecture is depicted in Fig. 5. It implements the basic tiers of an IoT architecture extended by the elements of a cognitive system. It supports an enterprise tier, a platform tier and an edge tier. We split the last one into three sublayers, namely a centric Semantic Meta-Data Layer, a Data Integration Layer and an on-site Edge Gateway. The platform tier provides a sublayer for APIs to automatically execute reasoning, machine learning and support services for user interactions. The architecture is fog-enabled. Each layer in the architecture is containerised (as a docker) and can potentially be deployed in the cloud or at the edge level. The top layers usually reside on a cloud and act as a platform as a service (PaaS), and allow easy access from various clients to various applications that can be deployed either on-site, remotely, or on the PaaS. Information is exchanged between these layers via web service interfaces. The architecture provides several tools to automate the workflows as shown on the left hand side in Fig. 5. We discuss them along with the individual layers in the subsection V-A.

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

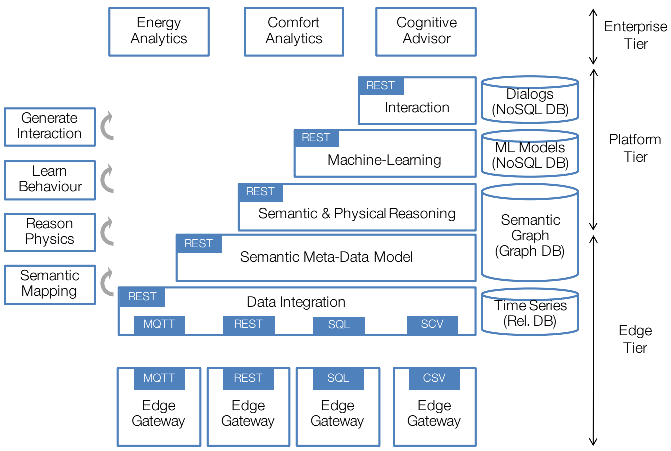

Fig. 5: High-level Cognitive IoT Architecture

A. Edge Tier

1) Edge Gateway:

a) Functionality: We use edge gateways to integrate data obtained from various sites. The gateways are connected to various systems at the sites and transfer the collected data to the data integration layer.

b) Implementation: We use commercial IoT gateways with Linux and a docker layer. The docker layer allows us to manage the containers remotely, to easily deploy code updates, and also to move upper layer functionalities to the edge. We have different containers for common building systems and support wireless networks like ZigBee, EnOcean, Yanzi as well as traditional building systems like ModBus and BACnet.

2) Data Integration:

a) Functionality: The data integration layer offers different interfaces to inject data via MQTT, REST APIs, and traditional CSV files. The incoming data is stored as time series in a dedicated database.

b) Implementation: We utilize the commercial MQTT client provided by the cloud platform that we use. In some on-premise scenarios, we have deployed containers of the open source Mosquitto broker on the edge. We extended the MQTT client with additional REST APIs for device and data management. The time series data is stored in a common relational database as they are efficient for time series and provide basic aggregation functions like average, sum etc. that are used by upper layers and embedded in the API.

3) Semantic Meta-Data Model:

a) Functionality: The semantic layer allows to store and query semantic meta-data associated to the devices, locations, and related equipments. It also allows upper layers to query the data integration layer via the semantic model as well as associate their data (e. g. models). Thus, users do not need to know each ID associated to a data point, but, can employ high level queries to abstract the underlying IoT layers.

b) Implementation: The semantic layer is built on Apache TinkerPop to abstract various Graph databases that scale from embedded edge devices (TinkerGraph) to big data databases (Titan). The semantic layer uses internally the Brick ontology [26]. Brick structures the meta-data of IoT systems in information dimensions. The core dimensions are the data *Point*, its functional *Measurement*, the monitored or physical *Location*, and the monitored *Equipment*. Each dimension is detailed with a domain taxonomy that defines its specific concepts.

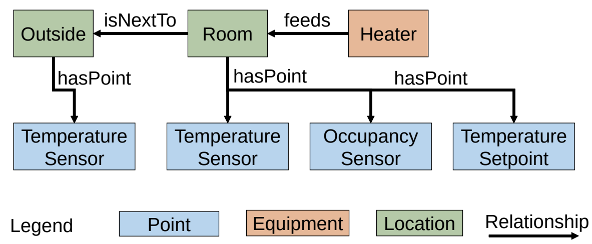

Fig. 6: Example for the Brick ontology

Brick also defines common relationships between these concepts. Fig. 6 shows an example of a Brick model for a single room. The *Room* is a subconcept of *Location*. It contains a *Temperature_Sensor* and an *Occupancy_Sensor* with the adequate *Point*. It is fed by a *Heater* with a *Temperature_Setpoint*. The room is located next to the *Outside*. Various semantic relationships like *hasPoint* and *feeds* describe their interaction in Brick.

Users can utilize this model to query, filter and aggregate data. For example the REST call: '/getHistory?aggregate=mean&point=temperature_sensor&location=room102&time=2weeks'. The semantic layer evaluates the query on the knowledge model to select the relevant data points and runs the aggregation query on the data layer.

c) Automation: The semantic model does not need to be manually specified. Various semi-automatic approaches and tools are available [28]. We use a web tool that maps the IoT data to the semantic model by a machine learning approach that learns common mappings. It recovers the dimensions of functionality, location, and asset.

B. Platform Tier

The platform tier provides advanced services for reasoning and machine learning.

1) Semantic & Physical Reasoning:

a) Functionality: We use reasoning to inject domain knowledge into the semantic model to reconstruct relationships that are not explicitly modelled in the semantic meta-data model. For example, the semantic model describes the sensors, the locations and the assets, but it does not capture the causal relationships of these sensors. They are critical to several analytics approaches, for example to identify relevant features for machine learning or to run diagnostic algorithms. The physical processes used in IoT systems are generally well known and understood. We capture these physical relationships in our domain knowledge model and use reasoning to automatically inject them into our knowledge model.

For the reasoning, we utilize a common design pattern in sensor networks [29] that distinguishes the actual *properties* (the air temperature at one point in a room) from the *observations* given by the data points (a temperature time series).

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

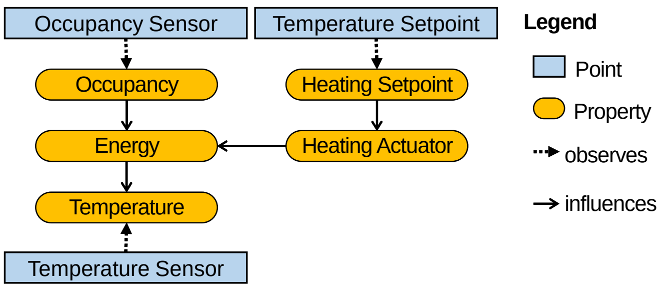

Fig. 7: Semantic knowledge graph of physical relationships

Based on their properties, we can deconstruct the physical relationships that describe the interactions into several atomic implications.

For example, it is known that the air temperature is actually a derivative of the inner energy in a room. Each room has such an inner energy and temperature regardless if it is observed by a sensor. We can express this in SWRL notation [30] with the implication

$$Room(?r) \implies hasProperty(?r, ?e) \land Energy(?e).$$

It states that each instance $?r$ of the concept $Room$ in our model has to have a relationship $hasProperty$ to another instance $?e$ of the concept $Energy$. If the relationship and the energy instance does not exists, then the reasoner will create them. In the same way, we can express that the room also has always a temperature that may be observed by a sensor

$$Room(?r) \implies hasProperty(?r, ?t) \land Temperature(?t).$$

$$Room(?r) \land hasPoint(?r, ?s) \land Temperature\_Sensor(?s) \implies hasProperty(?r, ?t) \land Temperature(?t) \land observes(?s, ?t).$$

These two implications created a temperature property for the room and linked it to the temperature sensor that is in the same room. We can describe now that the temperature derives from the energy if they share the same location, by writing

$$Room(?r) \land hasProperty(?r, ?e) \land Energy(?e) \land hasProperty(?r, ?t) \land Temperature(?t) \implies influences(?e, ?t),$$

which links the temperature to the energy property.

These implications are part of our domain model. So, we do not specify them anew for each building neither expect users to do that. They are just loaded with each new project and automatically applied wherever they fit.

Fig. 7 shows the derived physical processes as semantic graph for the example in Fig. 6. It shows each *Property* as an orange block and their causal relationship as directed edges. As a result, all sensors are now linked in a graph that explains their physical interaction and enables multiple machine learning scenarios.

b) Implementation: The reasoning engine is implemented in TinkerPops Gremlin language to directly run on the semantic graph DB. The API provides functions to expose the configuration and execution of the reasoning engine. The user can specify implications at the REST layer that are then translated to Gremlin code and run on the Graph DB.

c) Automation: We provide a web tool that allows to define and manage the atomic patterns of the physical reasoning.

2) Machine-Learning:

a) Functionality: The machine learning (ML) layer hosts analytic applications consisting of data cleaning, pattern analysis, predictive analytics, and diagnosis. It follows a modular concept in favour of an agile development of specialized analytics. They can be hosted on the cloud or on the edge in docker containers. The data exchange between the applications happens via the data and semantic layer. This loosely coupling of the analytics applications provides high flexibility facilitating the development and the connection of new analytics and is one of the core concepts of the architecture.

b) Implementation: The ML layer supports different analytics scripted in R or Python. We provide connector libraries to connect these analytics to the semantic and the data layer in a base docker file. Final analytics are then dockerized and deployed in a datacenter for parallel batch computation. The model output is stored as JSON-PMML models in a NoSQL database for easy access from the ML layer, and from the user interfaces. The NoSQL keys are stored in the semantic meta-data model for access trough the APIs.

c) Automation: Based on the physical graph model, machine learning approaches can be automatically deployed. The graph contains the information describing the influence that a sensor has on another. These causal relationships are relevant for several machine learning approaches as it specifies the relevant input features for training models. We will discuss an example in the use case section.

3) Interaction:

a) Functionality: The growing number of IoT devices increases the effort for configuring the user interfaces. Particularly, if one considers that modern user interfaces should be multi-modal and support graphical as well as conversational interactions with the users.

The interaction layer provides dedicated APIs for speech- or chat-based conversations that wraps the underlying ML and semantic layers. For example, we provide conversational APIs that support queries like: ”What was the mean temperature in room 102 over the last 2 weeks”, which is automatically mapped to the equivalent query on the semantic layer.

b) Implementation: The APIs are separated into visualizations and conversations. The visualization API wraps information from the semantic meta-data model into JSON objects that directly feed for example into a heatmap plot on the client side. It also provides mappings from Augmented Reality Markers (see Fig. 10) to the relevant data points and systems in the knowledge model. For the conversational API, we are utilizing commercial Speech-to-Text and Conversation APIs that are provided by our cloud platform. The API is linked to the semantic meta-data model that is executed to answer specific queries like the provided example.

c) Automation: The dialog flows are automatically configured with the knowledge stored in the semantic layer. The dialogs require entities for location, equipment type, and point types. We know for example: ’mean’ is an aggregate, ’temperature’ is a point type; ’room 102’ is a location; ’2 weeks’ is a time. So, the dialog system identifies: ”What was

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

the #aggregate #point in all #location over the #time" for the last query example.

VI. Use Case

A. Knowledge Model Size

We demonstrate our approach using a campus consisting of 6 buildings with 3,300 sensors. The sensors produce approximately 1 million samples a day that are fed to the Cognitive IoT platform from two gateway devices.

The size of the underlying knowledge model in the graph DB is shown in Fig. 8. The Brick domain ontology contains 2,396 vertexes and 16k edges. The 3,300 sensors are mapped to 220 semantic concepts from the Brick ontology (cf. Sec. V-A3c) bringing a similar size of vertexes and duplicating the number of edges. After the physical reasoning process, we derive 14,830 physical relationships between the sensors with the same number of vertexes and 27k edges. The total computation time for deriving all graphs is 30 minutes. We use the graph to train 14,043 models for anomaly detection and diagnosis of abnormal energy consumption and thermal comfort. The models are linked in the graph trough 22k edges. The platform further creates approximately 30k different conversational variants on the interaction layer, ranging from simple queries to ask for the location of an asset to complex ones like the previously discussed examples.

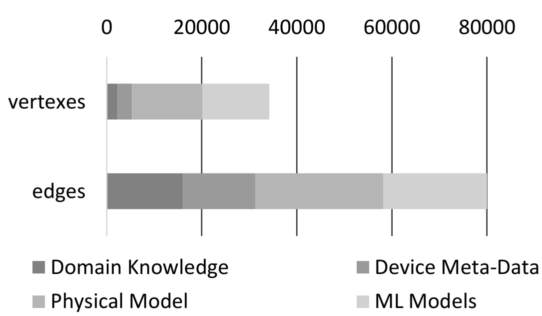

Fig. 8: Numbers of elements in the knowledge graph

B. Diagnosing High Temperature

Temperature sensors and heating systems are present in several rooms of the building. We define as an abnormal situation, the case where the temperature falls two degrees below the setpoint. In this circumstance, the monitoring system detects and diagnoses the abnormal temperature. We easily implement the analytic function capturing the detection and the diagnosis scenario with our semantic model, and for all the rooms. Let's assume that we have a function

$$High(A, B) : y(t) = 1(A(t) > B(t) + 2)$$

that returns 1 if the value of input $A(t)$ is higher than $B(t)+2$ at time t, and 0 otherwise. We want to deploy this function in all the rooms that have a temperature sensor and a setpoint. We can specify in our reasoning engine that

$$\begin{aligned} Room(?r) \wedge hasPoint(?r, ?t) \wedge Temperature\_Sensor(?t) \wedge \\ hasPoint(?r, ?p) \wedge Temperature\_Setpoint(?p) \implies \\ High(?h) \wedge asA(?h, ?t) \wedge asB(?h, ?p). \end{aligned}$$

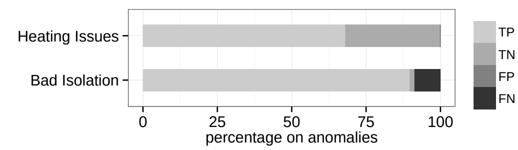

Fig. 9: Results for a building: TP - true positives, TN - true negatives, FP - false positives, FN - false negatives in percentage of anomalies.

The reasoner creates instances of $High$ for all appropriate locations and returns this list with the individual inputs and outputs. It is now easy to run the script for these instances[1].

We want to diagnose these anomalously high temperature samples in the next step. Let's define the function $Diagnose(A, C_1, \ldots, C_n)$ that diagnoses sensor A from the set of potential causes $C_1, \ldots, C_n$. We can identify the potential causes in the graph and deploy the algorithm with

$$\begin{aligned} Temperature\_Sensor(?t) \wedge observes(?t, ?p) \wedge \\ influences^*(?i, ?p) \wedge observes(?c, ?i) \implies \\ Diagnose(?d) \wedge asA(?d, ?t) \wedge asC(?c, ?t). \end{aligned}$$

This query looks for all temperature sensors $?t$ and the property $?p$ they observe. Then it collects all influencing[2] properties $?i$ and returns them if they are observed by another sensor $?c$. The list of $?c$ are then the observable potential causes for which a diagnostic model is deployed.

For the given example we can identify in Fig. 7 the potential causes for the anomaly, which are a low outside temperature, neighbouring rooms with a low temperature, and a low setpoint of the heating system.

For our building, 4 % of the room temperature samples were abnormal due to an inactive heating system and/or unusually low outside temperature. Fig. 9 summarizes the diagnosis results from a machine-learning classification model [31]. The model learns from historical anomaly-free time series data what the data range of the cause is under normal circumstances. We flag a cause as the real cause of an anomaly, if it is outside its predicted range. It bases on the intuition that a cause of an anomaly is also characterized by abnormal values in comparison to the anomaly-free data. Our approach retrieved that 67.95 % of the abnormal room temperature readings were related to an inactive heating system which could also be validated by the building operator. Furthermore our approach computed that 89.58 % of the abnormal cases was related to the outside air temperature which the operator largely confirmed to be the case.

Most importantly, our approach revealed that most abnormal temperature readings that were related to the outside temperature occurred in 11 rooms that had severe isolation problems. It turned out that they consumed an estimated 50 % of the buildings heating energy.

After detection and diagnosis of anomalies in the building, an operator will usually inspect the situation on site. The use

[1]The analytic platform is performing further steps to retrieve the data from the data platform and align it temporally.

[2]'$influences^*$' in the query indicates multiple steps in the graph. We normally use a depth of 4.

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2017.2755376, IEEE Internet of Things Journal

of traditional and unsophisticated means to scrutinize the data at the system level is complex and sometimes unpractical for identifying the related issues. A solution to this issue is the use of Augmented Reality. Fig. 10 shows an augmented view on the vent in the ceiling of our floor. We labelled the vents in the building with black-white markers encoding a unique ID that is mapped by the interaction layer to the corresponding asset. By pointing a smart device on the marker, the operator can now see the full asset behind the ceiling as well as the real-time values of the associated data points, which is delivered by the semantic meta-data API.

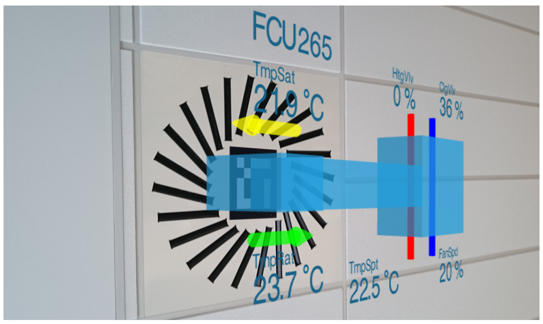

Fig. 10: Systems AR View showing a FCU in the ceiling.

VII. Conclusion

In this paper we conceived, developed, and implemented a CIoT architecture, and validated it with a complex use case: buildings. We addressed the lack of large scale applications of CIoT, as identified in the state-of-the-art analysis of CIoT and IoT, with a prime emphasis on their architectures. Based on this, we developed a CIoT architecture that combines the IoT strength in scalability with cognitive computing tools to integrate knowledge models and self-learning into the platform. The application of the architecture to buildings demonstrated the efficiency to diagnose temperature anomalies. Our solutions particularly showed the elements associated to data integration, semantic meta-data modelling, automated analytics, and the implementation. The example shows the strength of CIoT in enabling easily deployable, scalable, and self-learning complex IoT systems that are convenient to use.

The proposed architecture can be used to implement CIoT architectures in other IoT domains. As outlined in the trend analysis, the integration of semantics, machine learning, and natural user interfaces into CIoT are essential for extensive adoption of IoT systems. Our future research lies in semantic identification and matching, automated machine learning as well as natural user interfaces in order to further facilitate and democratize the use of these technologies.

References

[1] D. S. Modha, R. Ananthanarayanan, S. K. Esser, A. Ndirango, A. J. Sherbondy, and R. Singh, "Cognitive computing," *Comm. of the ACM*, vol. 54, no. 8, pp. 62–71, 2011.

[2] A. Afzal, S. A. R. Zaidi, M. Z. Shakir, M. A. Imran, M. Ghogho, A. V. Vasilakos, D. C. McLernon, and K. Qaraqe, "The cognitive internet of things: a unified perspective," *Mobile Networks and Applications*, vol. 20, no. 1, pp. 72–85, 2015.

[3] N. Mishra, C.-C. Lin, and H.-T. Chang, "A cognitive adopted framework for IoT big-data management and knowledge discovery prospective," *International Journal of Distributed Sensor Networks*, 2015.

[4] International Energy Agency, "World energy outlook 2012," 2012.

[5] P. Vlacheas, R. Giaffreda, V. Stavroulaki, D. Kelaidonis, V. Foteinos, G. Poulios, P. Demestichas, A. Somov, A. R. Biswas, and K. Moessner, "Enabling smart cities through a cognitive management framework for the internet of things," *IEEE Communications Magazine*, vol. 51, no. 6, pp. 102–111, 2013.

[6] M. Zhang, H. Zhao, R. Zheng, Q. Wu, and W. Wei, "Cognitive internet of things: Concepts and application example," 2012.

[7] Q. Wu, G. Ding, Y. Xu, S. Feng, Z. Du, J. Wang, and K. Long, "Cognitive internet of things: a new paradigm beyond connection," *IEEE Internet of Things Journal*, vol. 1, no. 2, pp. 129–143, 2014.

[8] M. Chen, J. Yang, X. Zhu, X. Wang, M. Liu, and J. Song, "Smart home 2.0: Innovative smart home system powered by botanical iot and emotion detection," *Mobile Networks and Applications*, pp. 1–11, 2017.

[9] J. Tervonen, K. Mikhaylov, S. Pieskä, J. Jämsä, and M. Heikkilä, "Cognitive internet-of-things solutions enabled by wireless sensor and actuator networks," in *CogInfoCom*. IEEE, 2014, pp. 97–102.

[10] A. Sheth, "Internet of things to smart IoT through semantic, cognitive, and perceptual computing," *IEEE Intelligent Systems*, vol. 31, no. 2, pp. 108–112, 2016.

[11] Industrial Internet Consortium, "IIRA - industrial internet reference architecture," 2015.

[12] IoT-A Consortium, "IoT-A - internet of things architecture," 2013.

[13] ETSI, "oneM2M - standards for M2M and the internet of things," 2012.

[14] AIOTI, "AIOTI - alliance for internet of things innovation - high level architecture," 2015.

[15] ITU, "ITU-T - internet of things global standards initiative," 2015.

[16] R. Reisenzein, E. Hudlicka, M. Dastani, J. Gratch, K. Hindriks, E. Lorini, and J.-J. C. Meyer, "Computational modeling of emotion: Toward improving the inter-and intradisciplinary exchange," *IEEE Trans. Affective Computing*, vol. 4, no. 3, pp. 246–266, 2013.

[17] S. Franklin, T. Madl, S. D'mello, and J. Snaider, "Lida: A systems-level architecture for cognition, emotion, and learning," *IEEE Trans. Auton. Ment. Dev.*, vol. 6, no. 1, pp. 19–41, 2014.

[18] G. Zucker, U. Habib, M. Blochle, A. Wendt, S. Schaat, and L. C. Siafara, "Building energy management and data analytics," in *EDST - Smart Elect. Distr. Syst. and Tech.*, 2015, pp. 462–467.

[19] T. M. Lawrence, R. T. Watson, M.-C. Boudreau, and J. Mohammadpour, "Data flow requirements for integrating smart buildings and a smart grid through model predictive control," *Procedia Engineering*, vol. 180, 2017.

[20] S. Dawson-Haggerty, A. Krioukov, J. Taneja, S. Karandikar, G. Fierro, N. Kitaev, and D. Culler, "Boss: building operating system services," in *10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*, 2013, pp. 443–457.

[21] G. Fierro and D. E. Culler, "XBOS: an extensible building operating system," in *BuildSys*, 2015, pp. 119–120.

[22] C. Palmer, P. Lazik, M. Buevich, J. Gao, M. Berges, and A. Rowe, "Mortar.io: Open source building automation system," in *BuildSys*, 2014, pp. 204–205.

[23] A. Bhattacharya, J. Ploennigs, and D. Culler, "Short paper: Analyzing metadata schemas for buildings: The good, the bad, and the ugly," in *BuildSys*, Nov. 2015, pp. 33–34.

[24] R. Galvin, "Making the rebound effectmore useful for performance evaluation of thermal retrofits of existing homes: defining the energy savings deficitand the energy performance gap," *Energy and buildings*, vol. 69, pp. 515–524, 2014.

[25] J. Ortiz, "A platform architecture for sensor data processing and verification in buildings," Ph.D. dissertation, EECS Dep. Univ. of California, Berkeley, 2013.

[26] B. Balaji, A. Bhattacharya, G. Fierro, J. Gao, J. Gluck, D. Hong, A. Johansen, J. Koh, J. Ploennigs, Y. Agarwal, M. Berges, D. Culler, R. Gupta, M. B. Kjærgaard, and K. Whitehouse, "Brick: Towards a unified metadata schema for buildings," in *BuildSys*, 2016.

[27] V. C. Gungor, D. Sahin, T. Kocak, S. Ergut, C. Buccella, C. Cecati, and G. P. Hancke, "A survey on smart grid potential applications and communication requirements," *IEEE Trans. Ind. Inf.*, vol. 9, no. 1, pp. 28–42, 2013.

[28] J. Ploennigs, "Automating analytics: How to learn metadata such that our buildings can learn from us," in *IoTPDA*, 2016.

[29] K. Janowicz and M. Compton, "The stimulus-sensor-observation ontology design pattern and its integration into the semantic sensor network ontology," in *Int. Workshop on Semantic Sensor Networks*, 2010.

[30] W3C, "SWRL: A semantic web rule language combining OWL and RuleML," 2016.

[31] J. Ploennigs, B. Chen, A. Schumann, and N. Brady, "Exploiting generalized additive models for diagnosing abnormal energy use in buildings," in *BuildSys*, 2013, pp. 1–8.

2327-4662 (c) 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.