Decentralized Management Framework for heterogeneous Devices in FM

T. Preindl, J. Pannosch, W. Kastner
Institute of Computer Engineering, TU Wien, Vienna, Austria

A. Redlein, C. Baretschneider
IFM – Real Estate and Facility Management, Vienna University of Technology, 1040 Vienna, Austria

Abstract

The rapid technological development of smart building appliances leads to changing system interfaces with each iteration of the product development cycle. Combined with the huge amount of device manufacturers and different submarkets for smart devices this leads to difficulties for the integration of different systems into an interconnected Building Automation System (BAS) and further into Building Management Systems (BMS). Existing solutions for device interoperability are lacking in certain aspects. The Web of Things (WoT) protocol is the most promising approach for communication. Based on the relevant literature, this paper presents a framework architecture that augments the WoT protocol with decentralized authentication and authorization capabilities based on biscuit tokens. Furthermore, baseline protocol workflows enable the integration with existing management systems while creating an immutable configuration and workflow history of the building infrastructure. We further discuss how this framework enables new use cases and argue for its potential to decrease operational cost and thus increase building value.

Keywords: IoT, Blockchain, Facility Management

1. Introduction

One of the aspects of Facility Management (FM) is concerned with the provisioning and maintenance of all the systems that are integrated into a building. From the heating, ventilation, and air conditioning (HVAC) system, the lighting and shading systems, infotainment systems to access control and security systems many devices can be part of a building's infrastructure and hence have to be provisioned, configured, and maintained. Additionally, the usage of the individual occupants may have to be tracked to allow for billing. Access to certain systems has to be managed and traced as well. With the introduction of Internet of Things (IoT) devices, many parts of these management tasks could be handled digitally by integrating the devices into a management system.

There are however several hurdles when it comes to the integration of the devices into a general management system. Due to the variety of device types and device manufacturers, there is no common interface to connect these systems. Even more, also the architecture of the individual systems can be diverse. Some work locally, as in the case of traditional home automation systems such as KNX or local automation hubs such as OpenHAB and HomeAssistent. Other systems are dependent on the cloud such as in the case of Nest, Tuya, or similar systems. For this reason, there is the need for a framework, that can bridge the different silos and their different architectures and interfaces while preventing the formation of a new silo with an open, decentralized architecture that facilitates the integration of any technology.

To allow the creation of a management system using such a framework, several capabilities have to be present. The primary goal of the framework is the connection of different device types. This allows for cross-domain interactions. For this reason, a standard communication protocol is necessary to facilitate these interactions. However, the communication and interaction between subsystems have to be governed to prevent unintended access to critical infrastructure. Therefore, the framework has to keep a record of which subsystems are known and what kind of interaction between them is allowed. Furthermore, subsystems could be used by several users with different access requirements and authorization. Hence the framework needs a concept for the representation of devices, users, and access policies. Due to the fact, that the devices need the data on devices, users, and access rights but also other application-specific data to function properly, the framework has to provide a concept for the decentralized storage as well as methods for the controlled update of this information.

This framework can be the foundation of a decentralized, local-first, off-line capable device management system. Such a system would allow many novel use-cases and capabilities. First

of all, device on-boarding and configuration can be managed using a single conceptual approach. The overall system state is distributed over the individual devices and thus independent of external infrastructure. This creates a kind of decentralized digital twin, that lives between the local system nodes and is bundled with the building, hence increasing its value. The interaction with centralized systems of record such as Enterprise Resource and Planning (ERP) is possible as those systems can simply act as subsystems themselves.

The interoperability of devices is an important prerequisite to enable the full potential of the IoT (Noura et al 2018). The Web of Things (WoT) is one standard that aims to enable interoperability across IoT Platforms and application domains. The key essence is to use already well-known and established Web technologies like Representational State Transfer (REST), Hypertext Transfer Protocol (HTTP), Constrained Application Protocol (CoAP), and many more to establish a broad basis for interoperability and connectivity over a heterogeneous environment.

The WoT focuses on the standardization of the communication interfaces while leaving aspects such as authentication and security open. This is not an issue for homogeneous systems that can use a centralized system to establish communication authentication. However, for heterogeneous systems this approach is unsuitable. For this reason, a system to manage the identity of the different devices and additionally also the users is required for a secure integration. In this paper, we first present a study of the relevant literature in the following section. The focus of this study lays on the technological gaps in the proposed solutions as well as technologies mitigating the identified shortcomings. Based on these findings, we propose a framework architecture for the integration of IoT devices and systems. The framework consists of WoT gateways that act as bridges to the common data and communication environment. The gateways store the state of the overall system from the perspective of the underlying subsystem i.e. configuration of the gateway itself, the configuration of the subsystem, the identity of the gateway as well as access permissions for devices and users. The data sets contained inside a gateway are synchronized using the baseline protocol, by storing cryptography hashes of the current and previous versions of the shared data on the blockchain. In Section 4, we discuss how this framework enables new use cases and argue for its potential to decrease operational cost and thus increase building value. In the last section, we conclude that the proposed framework has a high potential to solve the described problems but needs to be implemented as future work for specific use cases and subjected to an extensive evaluation.

2. Background

Over the years many Building Automation Systems (BAS) from different vendors were developed. Many of these systems were developed without the focus on interoperability to other vendors or technologies. Therefore, many BAS form an isolated silo, which raises the cost because of vendor or technology lock-in. Furthermore, it must be distinguished between self-operational systems and Web-enabled ones. For example, KNX or BACnet can run isolated from further components like cloud or management tools and provide a standardized communication interface. The integration and commissioning are done by professionals and require a cost-intensive configuration tool and a lot of know-how. On the other side, there are many so-called web-enabled smart home devices like Tuya, Google Nest, Apple Home Kit, or many more, which require Internet access to provide their full potential. Depending on the vendor, they provide limited compatibility with other technologies. Therefore, community-driven BAS like Home Assistant, OpenHAB, or Domoticz emerged which act as an integrator between various vendors and technologies. These community-driven BAS act as a central point where all devices and also automated scenes are stored and configured. This is also the major drawback of these systems, as they represent a single point of failure.

Web of Things (WoT)

WoT was presented in Guinard (2011) and later on adopted by the W3C in its working group. The working group at W3C is divided into different task forces which are working on parts of WoT. In the current state there are five task forces:

- WoT Architecture
- WoT Thing Description
- WoT Discovery
- WoT Security
- WoT Scripting API

These task forces also represent the key building blocks of WoT. In April 2020, the W3C published the first version of the Architecture and Thing Description (TD) as a recommendation and has thus formed a solid foundation for WoT. The key properties of WoT are flexibility, compatibility, scalability, and interoperability. Figure 1 shows the abstract architecture of WoT designed by W3C using these key properties.

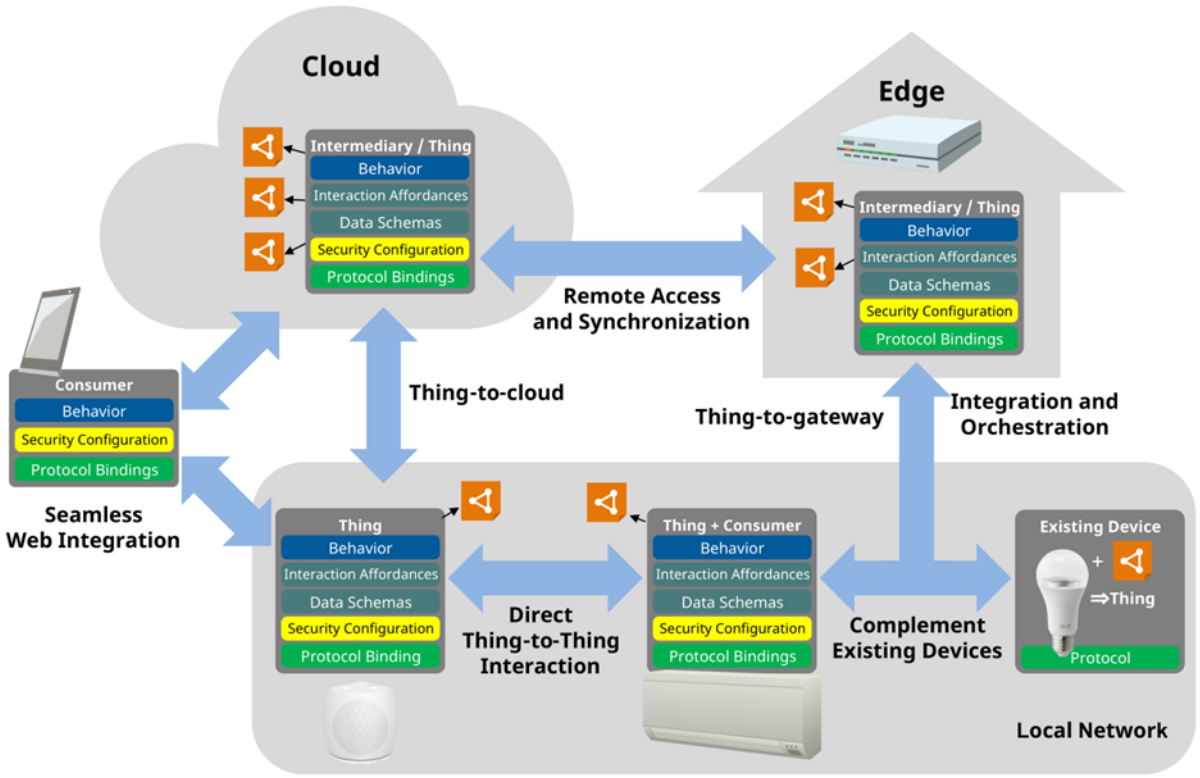

Fig.1: Abstract architecture of WoT (Lagally et al, 2020) (© W3C Software and Document Notice and License (W3C, 2015))

The architecture of WoT is open to integrate every possible device regardless of whether it is WoT enabled by default or uses legacy communication standards. Looking at Figure 1 on the system topological level, one can see how WoT devices interact with controllers, devices, agents, and further Web services. Furthermore, it shows, that WoT allows communication over different domains, such that different buildings can form a group, or different services can be aggregated to provide additional information to the system. This can even be enhanced to form groups that are representing physical cities or parts of them.

Getting into more detail, the central building block of every device is the Thing Description (TD) which describes the metadata and interfaces of things, where a "Thing" is an abstraction of a physical or virtual entity that provides interactions to and participates in the WoT (Kaebisch et al., 2021). Such interactions are called Interaction Affordances and can be properties, actions, and events. Figure 2 shows an example of a TD, consisting of the metadata about the device itself and its Interaction Affordances. The TD itself is a JSON-LD file, which is both machine and human-readable.

```
{
    "@context": "http://www.w3.org/ns/td",
    "id": "urn:dev:ops:32473-WoTLamp-1234",
    "title": "MyLampThing",
    "securityDefinitions": {
        "basic_sc": {"scheme": "basic", "in": "header"}
    },
    "security": "basic_sc",
    "properties": {
        "status": {
            "type": "string",
            "forms": [{"href": "https://mylamp.example.com/status"}]
        }
    },
    "actions": {
        "toggle": {
            "forms": [{"href": "https://mylamp.example.com/toggle"}]
        }
    },
    "events":{
        "overheating":{
            "data": {"type": "string"},
            "forms": [{
                "href": "https://mylamp.example.com/oh",
                "subprotocol": "longpoll"
            }]
        }
    }
}
```

Fig.2: Thing Description sample (Kaebisch et al., 2021)

Every WoT enabled device has five building blocks, Behavior, Interaction Affordances, Data Schemas, Security Configuration, and Protocol Binding(s), as can be seen in Figure 1. These blocks are also represented in Figure 2, where the thing named `MyLampThing` uses basic security, has one property called status which returns plain text, an endpoint to toggle the lamp power state, and an event interface which can inform other systems in case of an overheat of the lamp.

After generating such a TD, it needs to be distributed to the other devices on the network, and it must be guaranteed that only permitted devices can communicate with each other. This problem is being worked out in the WoT Discovery Task Force. At the time of writing, the discovery part of WoT is a working draft (Cimmino et al, 2021). But even in this state, there are five possible mechanisms to find other devices on the network. These can be separated into local and non-local discovery mechanisms, whereas the first targets local network structures and the second aims at discovery over the Web. Figure 3 shows the discovery mechanisms of WoT.

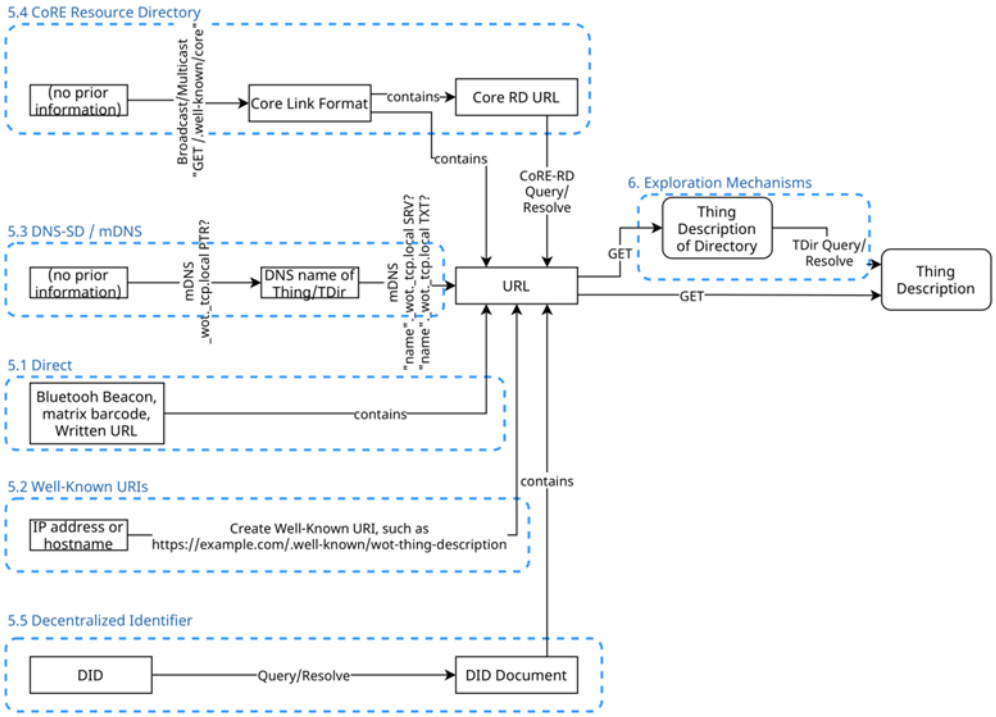

Fig.3: WoT discovery process overview. (Cimmino et al., 2021) (© W3C Software and Document Notice and License (W3C, 2015))

To get a list of devices on the network, there is an exploration mechanism which is a TD Directory. This directory provides the ability to get, search, modify, and delete TD. So that, it is possible to search devices by category or ID. Furthermore, the TD Directory also provides information about authentication mechanisms, this can be Open Authorization (OAuth) or any other provider.

WoT heavily relies on Web technologies so that they also thought on having the ability to dynamically adjust the behavior of devices. Therefore, a similar approach like client-side scripting of the browser was considered by the WoT Scripting API task force. The task force defined a runtime that is executed by a WoT device, like a gateway.

Decentralized State

A major concern in distributed system design represents the handling of shared states. More specifically, communication of state changes and the consensus on the current state are difficult problems in computer science. Depending on the fault and security models, different approaches can be implemented to achieve a shared state. In a centralized system, a single service acts as a reference for all other services. Such a service is often called a system of record. In a decentralized system, on the other hand, multiple services take part in a consensus process to establish a common view of the system state. Many approaches lie in between these two

extreme cases. Due to the desired properties of the framework, a decentralized architecture is required.

Blockchains or more specifically Smart Contracts (SC)-enabled blockchains such as Ethereum offer a decentralized consensus together with a strong guarantee of immutability (Buterin, 2014). While changes to the state are proposed via transactions, SCs define the rules for the state changes and determine the acceptance of transactions. Miners group and order transactions compute the new state and publish it in a new block. Each following block confirms the validity of the blocks before. The blocks can only be changed by recalculating the Proof of Work (PoW), which is prohibitively expensive. This makes blocks immutable after a short amount of time (Wood et al., 2014). However, this distributed consensus has certain drawbacks as the state and its changes need to be shared publicly. This has implications for privacy and operational costs. For this reason, only data that is necessary for the evaluation of the SC should be stored on-chain (Eberhardt and Tai, 2017).

There exist many trade-offs for the usage of on-chain vs. off-chain storage and computation. Eberhardt and Heiss give possible strategies for moving data and computation off-chain without losing desired properties of the blockchain (Eberhardt and Heiss, 2018). Baseline protocol, an emerging standard for blockchain-secured data exchange, follows in the same direction. In its draft version, available at the time of this writing, baseline-enabled services exchange data via peer-to-peer communication and submit only references of the data bundled with a compliance-proof to a so-called 'Shield'-SC. A so-called 'Verifier' SC checks the validity of the new state commitment, after which only the new reference is stored on the blockchain. While this approach minimizes the amount of data that needs to be stored on-chain, computation of the verification is off-chained using a zero-knowledge proof system, which additionally improves privacy. Data sharing and cooperation on the baseline protocol are organized in workgroups consisting of several participants. The participants create workflows for the individual business processes. These workflows consist of work steps that define which participant has to provide data under respective conditions to complete the step (Baseline Community, 2020).

Authentication and Permission Management

The WoT standard leaves the authentication and management of access permissions open to individual implementation (Lagally et al., 2020). This is necessary, as different deployments have different requirements. One proposed approach is based on the OAuth 2.0, an open standard that allows the delegation of access rights to services (Hardt, 2012). In its workflow, an identity

provider provides tokens to an application that can be used to access services representing the user. Biscuit is a standard for authentication tokens that can be used on OAuth deployments. It allows for offline validation of fine-grained access policies that are encoded in a special logic language. The tokens can be attenuated by the token holder to restrict access even further. This property is very useful when passing tokens from service to service during user requests as it allows to narrow down access policies to the scope of the request (CleverCloud, 2021).

3. Framework

In this section, we present our framework for a decentralized, distributed device management system for diverse IoT landscapes. The framework uses the WoT standard to handle communication between different IoT devices and -networks. Gateway components bridge the gap to the underlying systems if there is no WoT compatibility. These gateways additionally store the overall system state i.e. known devices, users, and access rights, as well as configuration data. State updates are distributed using workflows in the baseline protocol, which creates a common reference point for the current and previous versions of the system configuration.

Nowadays, there are many smart devices already in the market and it is necessary to integrate these devices into our approach, therefore gateways are used which can on one side talk to the non-WoT-enabled devices and on the other side provide a bridge to the world of WoT. These gateways not only care about the right translation between the communication protocols but also handle device configuration and enforce access permissions for communication between the devices.

As demonstrated in Figure 4, gateways in our framework provide access to interfaces of the underlying systems via WoT-properties, -actions, and -events. The access to these interfaces is guarded and they can only be used with authorization tokens using the OAuth 2.0 protocol. This allows for different security policies for any of the device interface points. Administrative users generate biscuit tokens that encode user and device identities and roles. The biscuit logic language is used to enforce authorization rules in the devices. While interfaces related to the control of the devices can be used by simple delegation to the underlying systems, for interfaces that change the configuration of a device, special care has to be taken. Configuration changes should only occur via baseline workflows to ensure, that the history of changes is preserved and changes can only occur according to the proper workflows.

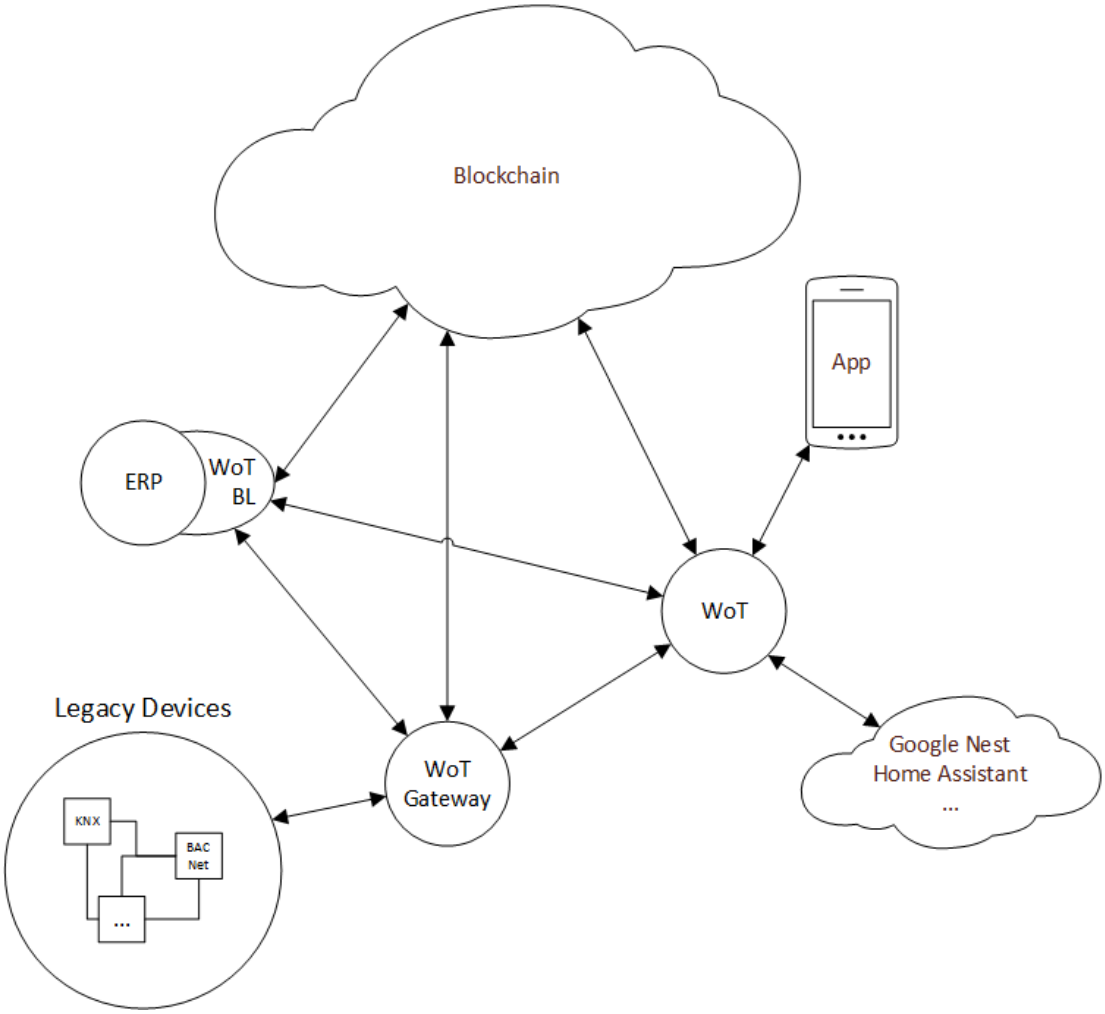

Fig.4: The framework architecture

To enforce that the workflows are followed and that the state reference is stored on the blockchain, the gateway not only checks for the authorization for the interface access but additionally verifies the baseline workflow. If everything succeeds the underlying system is updated and the new configuration is stored in the gateway and depending on the workflows also shared with other systems such as databases. Similarly, state updates related to the configuration of the framework, such as users, devices, or security policies, can only be updated via baseline workflows. These workflows can be arbitrary complex and for example, require the consent of several users or the submission of additional information such as work reports.

4. Discussion

The deployment of the framework allows for many new use cases that are not possible without it. The user, device, and security management establish a foundation for secure user access and device interaction according to the WoT vision. This foundation can be used to implement smart building applications that span systems that could not communicate without a dedicated

integration. This has many advantages, as it enables the possibility to add new features to the BAS without a redesign of the system. For example, when a solar power plant is added to a building roof, the availability of local power can be communicated to other systems to trigger energy-intensive processes. Another possibility is the replacement of broken components of an installed system with components from another manufacturer and the integration via the framework. In this way, the established infrastructure can be replaced piece by piece without requiring a complete reinstallation.

The powerful authorization system opens up another set of use cases. The usage of biscuit tokens allows the segmentation of a subset of the interfaces and hence allows the definition of intricate access patterns. This opens up the possibility to give access to certain parts of the system via standardized interfaces. For example, in the case of shared spaces, such as rental apartments or hotels, guests could be given access to the smart home appliances of their rooms. Access to special features such as air conditioning or wellness appliances could also be controlled via the framework and the connection to the blockchain would even allow automating the payment for these services.

However, the biggest advantage of the framework comes from the usage of the baseline protocol for the synchronization of configuration data. Any change to the configuration can be managed using individual workflows. Due to the open architecture of the baseline protocol, other systems of record can be integrated. For example, a repair workflow could be started in the ERP system of the FM, which triggers a repair assignment in a contractor's Customer Relationship Management (CRM) system and additionally grants access to the interfaces of the affected subsystems. After a change of some hardware component, the workflow would then be finalized with the deployment of the new configuration and submission of the contract report together with a payable invoice. While every step of the workflow is committed to the blockchain with a reference to the data, the data itself is stored by the individual stakeholders, ensuring that only minimal data is exchanged and shared.

By using blockchain technology combined with WoT it is possible to generate a historical log of all changes done to the system and of all changes of device configurations in a BAS. This means that any modifications to the BAS are irrevocably stored and can be traced back at any time. From these detailed logs, the condition of the components in the building can be inferred and thus can attest to the actual value of the components and ultimately of the building. Due to the decentralized nature of the framework architecture, it is possible to migrate the data and

access rights to a new owner in the case of a sale. No additional infrastructure is needed to interact with the building.

5. Conclusion

The interoperability of IoT devices and their integration into higher-level FM systems is an important aspect necessary for the digital transformation in the sector. Proposed solutions are lacking in certain parts with the WoT protocol being the most promising candidate for device interoperability. However, for a heterogeneous and thus decentralized device landscape user, device and security management is a missing aspect of the standard. Furthermore, it lacks the capability for integration into higher-level systems of record such as ERP and CRM systems.

In this work, we presented a framework enabling the deployment of WoT in decentralized architectures and narrowing the gap between IoT systems and process management in FM. The framework security is based on biscuit tokens which allow for offline authentication of users and enforcement of device access. The known devices, users, the access policies as well as the configuration of WoT gateways and underlying legacy subsystems are managed in baseline workflows. Smart Contracts store cryptographic references to the system state and enforce workflow procedures, while systems only share the necessary data in a peer-to-peer fashion, ensuring privacy. The framework opens up new use cases in FM for configuration, maintenance, and operation that increase building value and help in the digital transformation of management processes. To demonstrate the promising features of the framework, proof-of-concept prototypes for specific use cases in the described area will be implemented and evaluated extensively in future work. Additionally, tooling and concepts for the management of configuration complexity and deployment automation will be investigated further.

Bibliography

Baseline Community (2020) Baseline protocol v0.1.0. Baseline Working Group, https://github.com/eea-oasis/baseline/tree/v0.1.0, accessed: 2021/08/21

Buterin V (2014) Ethereum White Paper. https://ethereum.org/en/whitepaper/, Ethereum Foundation, accessed: 2021/08/21

Cimmino A, McCool M, Tavakolizadeh F, Toumura K (2021) Web of things (wot) discovery. W3c working draft 2 June 2021, W3C, https://www.w3.org/TR/2021/WD-wot-discovery-20210602/, accessed: 2021/08/21

CleverCloud (2021) Biscuit. https://github.com/CleverCloud/biscuit, accessed: 2021/08/21

Eberhardt J, Heiss J (2018) Off-chaining models and approaches to off-chain computations. In: Proceedings of the 2nd Workshop on Scalable and Resilient Infrastructures for Distributed Ledgers (pp. 7-12) (2018), Association for Computing Machinery.

Eberhardt J, Tai S (2017) On or Off the Blockchain? Insights on Off-Chaining Computation and Data, In: Service-Oriented and Cloud Computing (pp. 3–15) (2017), Springer International Publishing.

Guinard D (2011) A web of things application architecture - integrating the real-world into the web. PhD thesis, ETH Zürich.

Hardt D (2012) The OAuth 2.0 Authorization Framework. RFC 6749. RFC Editor

Kaebisch S, Kamiya T, McCool M, Charpenay V (2021) Web of things (wot) thing description 1.1. W3c working draft 7 June 2021, W3C, https://www.w3.org/TR/2021/WD-wot-thing-description11-20210607/, accessed: 2021/08/21

Lagally M, Matsukura R, Kawaguchi T, Toumura K, Kajimoto K (2020) Web of things (wot) architecture 1.1. W3c working draft 24 November 2020, W3C, https://www.w3.org/TR/2020/WD-wot-architecture11-20201124, accessed: 2021/08/21

Noura M, Atiquzzaman M, Gaedke M (2018) Interoperability in internet of things: Taxonomies and open challenges. In: Mobile Networks and Applications 24(3) (pp. 796–809.), Springer International Publishing.

W3C (2015) W3C Software and Document Notice and License. W3C, http://www.w3.org/Consortium/Legal/2015/copyright-software-and-document accessed: 2021/08/21

Wood G, et al (2014) Ethereum: a secure decentralised generalised transaction ledger. Ethereum project yellow paper, Ethereum Foundation, https://ethereum.github.io/yellowpaper/paper.pdf, accessed: 2021/08/21