10th International Food Operations & Processing Simulation Workshop, 016
21st International Multidisciplinary Modeling & Simulation Multiconference

2724-0355 © 2024 The Authors.
doi: 10.46354/i3m.2024.foodops.016

CoAP-based Digital Twin Modelling of Heterogeneous IoT Scenarios

Luca Davoli[1,*], Hafiz Humza Mahmood Ramzan[1], Laura Belli[1] and Gianluigi Ferrari[1]

[1]Internet of Things (IoT) Lab, Department of Engineering and Architecture, University of Parma, Parco Area delle Scienze 181/A, 43124 Parma, Italy

[*]Corresponding author. Email address: luca.davoli@unipr.it

Abstract

Modern societies nowadays require more and more abstraction efforts to hide the complexity of underlying systems and infrastructures. To this end, the concept of Digital Twin (DT) has recently emerged as a key enabler for the digital transformation of well-established architectures toward their virtual representation, opening to intelligent processing capabilities (e.g., monitoring, simulation, prediction, optimization). Aside from defining DTs to enhance these services, another key paradigm that is noteworthy of attention is the Internet of Things (IoT), enabling data and information collection through heterogeneous *smart devices* (often equipped with sensors and actuators). Thus, combining DTs and IoT together with the Constrained Application Protocol (CoAP) as communication protocol (with its native features), will allow to define scalable and lightweight replicas of real systems, and exploit key features (e.g., service and resource discovery) to provide end users with smart solutions. In this paper, a modelling paradigm for heterogeneous IoT scenarios, based on the definition of a DT for each entity involved in a specific context to be mapped, is detailed. This will allow to *a-priori* estimate the behaviour of an IoT ecosystem and provide well-known interaction endpoints to request data from/pushing information to the hidden lower layers of the same ecosystem.

Keywords: Internet of Things; Digital Twin; CoAP protocol; Industrial IoT; Smart Agriculture; Smart Cities

1. Introduction

Nowadays, everywhere people are experiencing how their lifestyle and well-being level can be improved in the presence of scalable and intelligent systems deployed in the environment daily habited and visited (e.g., workplace, leisure areas, the city where they live and those visited as tourists). In this context, one of the main technological breaking points allowing this lifestyle improvement is represented by the Internet of Things (IoT), to be be applied to heterogeneous environments (e.g., smart farms, smart industries, smart cities, etc.) deploying and connecting a massive amount of (typically) *constrained* devices, and obtaining a cooperating ecosystem (beneficial for multiple purposes). As a consequence, this opens to wide and scalable data collection and actuation, leveraging the adoption of Human-to-Machine (H2M) and Machine-to-Machine (M2M) paradigms in a seamless way—at least for people being aware of this technological "backbone" supporting this enhanced way to live and work. In fact, in the near future the IoT is expected to play a significant role in people's life, leading to the design and development of new paradigms and architectures, together with new challenges and services—as an example, the security would be one of the main aspect to be primarily taken care of.

Focusing on this variety of environments and contexts to be covered (and possibly connected together), one of

the main challenges arising for systems and networks administrators is the need to provide enhanced (yet easy-to-use and secure) services to end users trusting these IoT-enabled paradigms, as detailed by Mangia et al. (2018). In particular, these needs target interactions between heterogeneous (often "vertical") infrastructures at higher layers, where a *plethora* of data and information should be collected through several network protocols and communication technologies, as highlighted in Davoli et al. (2018). Since at higher layers there is often incompatibility between transmission mechanisms adhering to well-known protocols (e.g., the IP protocol) and communication technologies that, due to their inner nature, cannot rely on these widely-adopted protocols, there is the need to "normalize" these information. To this end, one way can involve the definition of *Digital Twins* (DTs) for these systems, as proposed by Jacoby and Usländer (2020): abstracting from underlying specific implementations (both in terms of hardware and software), and exploiting high layer interaction protocols, DTs allow to model communication among heterogeneous platforms, modelling the required interaction patterns in a seamless way (and hiding to the end users the effort required to translate data between the different systems).

In this work, the design of a modelling paradigm for heterogeneous IoT scenarios, exploiting the Constrained Application Protocol (CoAP, defined in Shelby et al. (2014)) and its main features, and relying on the definition of a DT for each entity involved in a specific context to be mapped, is presented. This allows to (i) *a-priori* estimate the behaviour of an IoT ecosystem, (ii) "hide" to the end user the complexity and inner mechanisms of each (maybe vertical) subsystem, and (iii) provide well-known interaction endpoints—such as RESTful Application Programming Interfaces (APIs)—to request data from/pushing information to the hidden lower layers of the specific IoT scenario (in this way modifying their internal state and behaviour).

The remainder of this paper is organized as follows. In Section 2, an overview on relevant literature works is presented. In Section 3, the proposed modelling paradigm, where CoAP is exploited as communication protocol among virtual DTs, together with a few illustrative IoT-oriented scenarios, is presented. Finally, in Section 4 conclusions are drawn.

2. Related Works

The concept of DT was first introduced in Githens (2007), discussing on how rapidly it can be adopted in heterogeneous scenarios with different requirements and behaviors, ranging from industry and manufactures to big data. To this end, authors detail how one of the main benefits of DTs is their ability to provide a solid and scalable layer on top of physical resources, allowing the generation of applications that can easily and securely interact with each other. Then, considering the modelling of IoT-based DTs, the performance of most common application layer protocols (namely: CoAP, Advanced Message Queuing Protocol, AMQP, and Message Queuing Telemetry Transport, MQTT, presented in Naik (2017)), together with different options for data serialization, are discussed in Proos and Carlsson (2020), mainly addressing suitable solutions enabling emerging delay-sensitive Intelligent Transport Systems (ITSs).

With regards to CoAP, in Seoane et al. (2020) the efficiency of this constrained communication protocol is analyzed, highlighting its benefits and potential use by *resource-constrained* IoT devices for various tasks, including securing communications. In detail, a performance evaluation is conducted on a real network and considers various metrics, such as: bandwidth usage; processing demand, in terms of CPU cycles; interconnect delays.

For the sake of comparison, in Nwankwo et al. (2024) the integration of CoAP with another *publish/subscribe* protocol, namely MQTT, via an abstraction layer, is shown, examining the performance of MQTT-based IoT nodes whose resource management is altered via CoAP requests, and revealing that latency and energy depletion stay within reasonable limits (with reference to IoT tasks). Moreover, the need of congestion control for CoAP is analyzed in Donta et al. (2022), where CoAP is shown to adopt a straightforward binary exponential back-off technique, at the cost of increasing retransmission delay, energy consumption and packet loss, while decreasing the packet delivery ratio.

Focusing on security aspects to be guaranteed in IoT environments, in Sáez-de-Cámara et al. (2024) the effects of botnets and hacking tackles, performing attacks such as Denial of Service (DoS), periodic scanning, and IoT protocol exploitation, are evaluated. A similar analysis is conducted in Hussain et al. (2021), where a dataset (resulting from a simulated testbed composed of nine sensors, a MQTT broker, and a CoAP server) is collected, while the analyzed attacks include MQTT packets' flooding, construction of packet, and CoAP replay attacks. Finally, in Belli et al. (2016) security aspects for IoT-oriented big stream applications—efficiently handling information through a graph-based platform and delivering processed data to consumers with a low latency—are analyzed.

3. CoAP-based DT Modelling

As introduced in Section 1, our goal is to propose a modelling paradigm allowing end users—living and working in specific IoT-enhanced environments (e.g., workers inside smart industries; citizens in smart cities; farmers in smart farms)—to seamlessly access enhanced information and to benefit from improved services, at the same time remaining unaware of the complexity of each single infrastructure composing the context of interest.

More in detail, for each system composing the candidate IoT environment to be "twinned," the proposed modelling mechanism foresees the definition and deployment (at software layer) of a corresponding DT. Therefore, CoAP

seems to be one of the best communication protocols to be used to accomplish this task, thanks to its fundamental property of being (at the same time) a *client/server*-like protocol—following a RESTful-like paradigm allowing interactions through well-known request and response patterns—but also providing the *observing* functionality, defined in Hartke (2015), which in the end enables CoAP to support interactions adhering to the *publish/subscribe* paradigm.

On the basis of these characteristics, in the proposed modelling paradigm each real system, belonging to the reference IoT environment to be twinned and hosting information to be possibly of interest for other entities, has a corresponding DT embodied by a CoAP server—following an IoT-like network structure—while the services to be exposed for interested requesting entities are modelled as CoAP resources. This will allow CoAP-enabled entities to reach the service sought on the basis of the specific request that should be performed.

In order to support enhanced requests paradigms, CoAP-based DTs can exploit the CoRE Link Format (defined by Shelby (2012)) to describe the resources hosted by each CoAP server, their attributes, as well as specific choice parameters to be used in the requests. As an example, a generic DT (denoted as $\mathbb{D}_x$) can be modelled as follows:

- each time an entity (either internal or external, and denoted as $\mathbb{E}_y$) requests data to $\mathbb{D}_x$, the corresponding function managing this request (inside $\mathbb{D}_x$) is handled through a CoAP `GET` request;
- if $\mathbb{E}_y$ is interested in creating a new instance of a particular service in $\mathbb{D}_x$, then $\mathbb{D}_x$ should handle this request *if and only if* it arrives through a CoAP `POST` request;
- in the case $\mathbb{E}_y$ is interested in modifying the internal state of $\mathbb{D}_x$, then $\mathbb{D}_x$ should manage this operation defining a proper handler function listening for incoming CoAP `PUT` requests;
- when $\mathbb{E}_y$ is no longer interested in data generated by a specific service running in $\mathbb{D}_x$, then $\mathbb{D}_x$ should handle this request *if and only if* it is arriving through a CoAP `DELETE` request.

This allows a generic DT to cope with the standard mechanisms defined by CoAP (e.g., service and resource discovery mechanism through the native `/.well-known/core` CoAP resource, as detailed in Cirani et al. (2014)), as well as exploiting the CoRE Link Format that allows to define one handler endpoint for each CoAP operation (namely: `GET`, `POST`, `PUT`, `DELETE`) and also to manage each specific request on the basis of its accompanying query parameters and descriptors.

In order to discuss on how the proposed CoAP-based DT-oriented modelling paradigm can adapt to heterogeneous contexts, in the following three realistic scenarios are detailed.

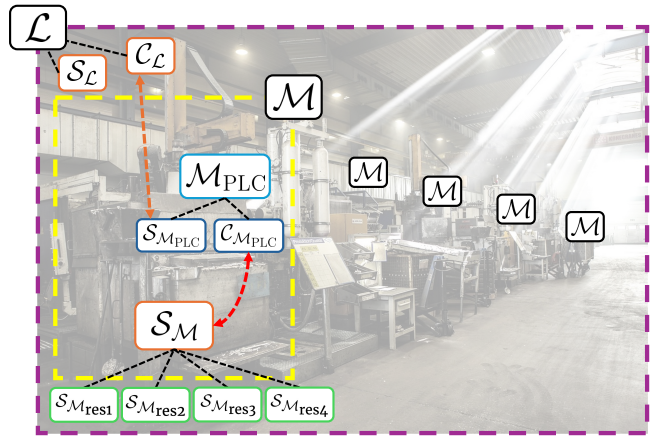

Figure 1. DT-enabled smart factory.

3.1. Smart Factory

Focusing on a smart factory relying on the Industrial IoT (IIoT) paradigm and interested in improving its work cycles before deploying real production lines and machinery, one way could be to model these complex entities as the union of multiple DTs interacting with each other through CoAP, as detailed in Section 3. To this end, as shown in Figure 1, following a *bottom-up* approach, each production machinery (denoted as $\mathcal{M}$) can be mapped as a CoAP server (denoted as $\mathcal{S}_{\mathcal{M}}$). Then, its internal components (e.g., conveyor belt, gears, engines, status monitor, etc.) are twinned as CoAP resources linked (*by-design*, as per the CoAP specifications) to $\mathcal{S}_{\mathcal{M}}$, each one handling incoming requests as detailed in Section 3: as an example, a status monitor might support both `GET` and `OBS` methods, being a feedback system, while coveyor belt and engines might also support `PUT` operations, being actuating components.

At this point, since it is expected that $\mathcal{M}$ is controlled by a Programmable Logic Controller (PLC, denoted as $\mathcal{M}_{\text{PLC}}$), the PLC can be modelled (in terms of DT) as the union of two elements: (i) a CoAP client (denoted as $\mathcal{C}_{\mathcal{M}_{\text{PLC}}}$) and (ii) a CoAP server (denoted as $\mathcal{S}_{\mathcal{M}_{\text{PLC}}}$). On the operational side, $\mathcal{C}_{\mathcal{M}_{\text{PLC}}}$ is in charge of interacting with the different CoAP resources of $\mathcal{S}_{\mathcal{M}}$ (mapping the components of $\mathcal{M}$) through the CoAP-enabled operations detailed in Section 3, while $\mathcal{S}_{\mathcal{M}_{\text{PLC}}}$ represents the endpoint to be queried by external CoAP-enabled entities interested in knowing the status of the machinery.

Finally, assuming that all the production machinery compose a production line (denoted as $\mathcal{L}$), then $\mathcal{L}$ could be seen as an entity needing to interact with its production machinery. This is the reason why $\mathcal{L}$ could be mapped with a corresponding DT that, similarly to $\mathcal{M}_{\text{PLC}}$, would be a complex software entity composed by a CoAP client and a CoAP server, denoted as $\mathcal{C}_{\mathcal{L}}$ and $\mathcal{S}_{\mathcal{L}}$, respectively.

This approach allows a company's Supervisory Control And Data Acquisition (SCADA) to monitor the trend of each production line, being notified (thank to the CoAP's

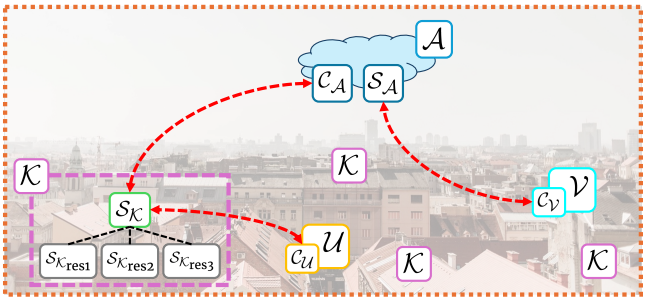

Figure 2. Smart city enabled by CoAP-based DTs.

observable mode) in the presence of dangerous situations.

3.2. Smart City

Considering the context of a smart city willing to offer improved and distributed services to its citizenship, it could be reasonable to create its DT following a *layered* approach similar to that discussed in Subsection 3.1. In particular, the city might be seen as the union of different heterogeneous "pillars" (as discussed in Belli et al. (2023)), each one composed by sensing and actuating elements sharing data with heterogeneous communication technologies. Therefore, as shown in Figure 2, each *on-field* device (denoted as $\mathcal{K}$) can be modelled as a CoAP server (denoted as $\mathcal{S}_\mathcal{K}$), while its physically-connected sensors and actuators could be mapped as CoAP resources hosted by $\mathcal{S}_\mathcal{K}$.

Should a high layer aggregation and processing unit (denoted as $\mathcal{A}$) be interested in the data collected by *on-field* devices, then $\mathcal{A}$—which can be modelled (similarly to the production line $\mathcal{L}$ detailed in Subsection 3.1) as the union of a CoAP client and a CoAP server, denoted as $\mathcal{C}_\mathcal{A}$ and $\mathcal{S}_\mathcal{A}$, respectively—sends CoAP requests targeting $\mathcal{S}_\mathcal{K}$.

Finally, citizens and visitors (denoted as $\mathcal{U}$ and $\mathcal{V}$, respectively) could express their interest in being notified for a particular service offered by the smart city, as well as for its overall status. Therefore, each citizen and visitor can be twinned as a CoAP client (denoted as $\mathcal{C}_\mathcal{U}$ and $\mathcal{C}_\mathcal{V}$, respectively) and can send CoAP requests targeting $\mathcal{S}_\mathcal{A}$ (or directly querying $\mathcal{S}_\mathcal{K}$).

3.3. Smart Agriculture

Finally, with regard to smart agriculture scenarios, the proposed modelling paradigm can be applied to the case of a distributed system deployed to monitor specific crops (e.g., tomatoes) through IoT devices—i.e., aiming at optimizing the usage of resources on the field (e.g., water, fertilizers, additives, etc., as discussed in Stefanini et al. (2023)). To this end, as shown in Figure 3, it can be assumed that the agricultural crops to be monitored foresee the installation of (i) a certain amount of smart water meters (denoted as $\mathcal{W}$) and a water pump (denoted as $\mathcal{P}$), composing the irrigation system, and (ii) several soil sensors (denoted as $\mathcal{Z}$), to check the status of the crop's soil. In this case, each IoT device can be twinned with a CoAP server (denoted as $\mathcal{S}_\mathcal{W}$, $\mathcal{S}_\mathcal{P}$ and $\mathcal{S}_\mathcal{Z}$, respectively), while each information collected by the sensors can be represented as a CoAP resource (e.g., soil temperature and humidity collected by $\mathcal{Z}$; current water volume returned by $\mathcal{W}$). In this way, should the water consumption be saved in the agricultural field, the irrigation system would be activated *if and only if* specific conditions applied (e.g., in the case the soil humidity drops below a certain threshold, as well as on the basis of decisions taken by Artificial Intelligence, AI, algorithms, as in Codeluppi et al. (2020)): this behaviour can be twinned by creating a high layer processing unit (denoted as $\mathcal{H}$) that, through its internal CoAP client (denoted as $\mathcal{C}_\mathcal{H}$) *observes* the different $\mathcal{W}$ and $\mathcal{Z}$: when the watering threshold is exceeded, activates the irrigation system POSTing a specific command targeting $\mathcal{S}_\mathcal{P}$.

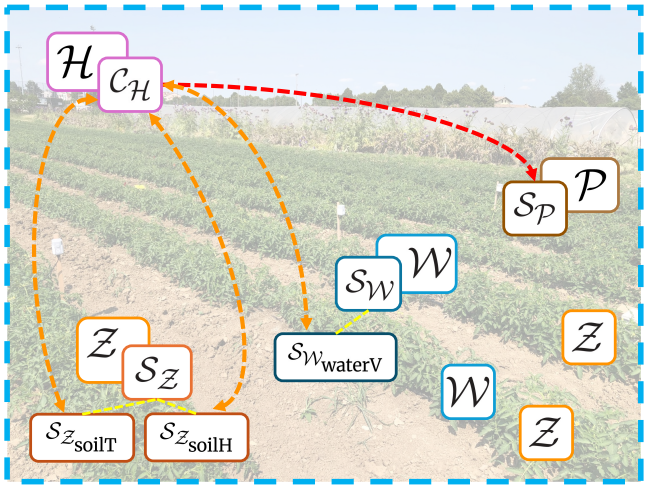

Figure 3. Smart farming scenario managed through DTs.

4. Conclusions

In this paper, the design of a modelling paradigm, exploiting the native features of CoAP and targeting heterogeneous IoT scenarios, has been discussed. In particular, the creation of a CoAP-based DT for each real entity composing the IoT ecosystem is foreseen, to enable pre-validation phases and open to the possibility of evaluating the scalability and the security of a particular solution before its real deployment. As a consequence, from the end user's point of view, this approach allows to "hide" the complexity and the internal characteristics of each (maybe vertical) system, while providing standard (yet widely adopted) APIs and data access mechanisms. In the end, even if the proposed DT-oriented modelling approach is not exempt from limitations—e.g., time and processing resource consuming activities, devoted to the design and deployment of virtual DTs—this simplifies to request data from/pushing information to the hidden lower layers of the specific IoT scenarios. Future research activities will involve a prototypical development of the proposed modelling paradigm, a performance evaluation with experimental deployments

in the considered heterogeneous scenarios, as well as the evaluation of alternative communication protocols to be used in the interactions between the different components (e.g., MQTT, Zenoh, HTTP2) in additional use cases that might require similar DT-oriented architectures.

5. Funding

This work received funding from the SMALLDERS research project - "Smart Models for Agrifood Local vaLue chain based on Digital technologies for Enabling covid-19 Resilience and Sustainability," co-funded by the PRIMA Program - Section 2 Call multi-topics 2021, through the following National Authorities: Ministry of Universities and Research (MUR, Italy), State Research Agency (AEI, Spain), Agence Nationale de la Recherche (ANR, France), Ministry of Higher Education and Scientific Research (Tunisia). It has also received funding from the Agritech project - "National Research Centre for Agricultural Technologies," project code CN00000022, funded under the National Recovery and Resilience Plan (NRRP), Mission 4 Component 2 Investment 1.4 - Call for tender no. 3138 of 16/12/2021 of Italian Ministry of University and Research funded by the European Union – NextGenerationEU, Concession Decree no. 1032 of 17/06/2022 adopted by the Italian Ministry of University and Research. The work reflects only the authors' views; the European Commission is not responsible for any use that may be made of the information it contains.

References

Belli, L., Davoli, L., and Ferrari, G. (2023). Smart City as an Urban Intelligent Digital System: The Case of Parma. *Computer*, 56(7):106–109. doi:10.1109/MC.2023.3267245.

Belli, L. et al. (2016). Applying Security to a Big Stream Cloud Architecture for the Internet of Things. *International Journal of Distributed Systems and Technologies (IJDST)*, 7(1):37–58. doi:10.4018/IJDST.2016010103.

Cirani, S. et al. (2014). A Scalable and Self-Configuring Architecture for Service Discovery in the Internet of Things. *IEEE Internet of Things Journal*, 1(5):508–521. doi:10.1109/JIOT.2014.2358296.

Codeluppi, G. et al. (2020). AI at the Edge: a Smart Gateway for Greenhouse Air Temperature Forecasting. In *2020 IEEE International Workshop on Metrology for Agriculture and Forestry (MetroAgriFor)*, pages 348–353, Trento, Italy. doi:10.1109/MetroAgriFor50201.2020.9277553.

Davoli, L. et al. (2018). From Micro to Macro IoT: Challenges and Solutions in the Integration of IEEE 802.15.4/802.11 and Sub-GHz Technologies. *IEEE Internet of Things Journal*, 5(2):784–793. doi:10.1109/JIOT.2017.2747900.

Donta, P. et al. (2022). Survey on recent advances in IoT application layer protocols and machine learning scope for research directions. *Digital Communications and Networks Journal*, 8(5):727–744. doi:10.1016/j.dcan.2021.10.004.

Githens, G. (2007). Product Lifecycle Management: Driving the Next Generation of Lean Thinking. *Journal of Product Innovation Management*, 24(3):278–280. doi:10.1111/j.1540-5885.2007.00250_2.x.

Hartke, K. (2015). Observing Resources in the Constrained Application Protocol (CoAP). RFC 7641, IETF. doi:10.17487/RFC7641.

Hussain, F. et al. (2021). A framework for malicious traffic detection in IoT healthcare environment. *IEEE Transactions on Dependable and Secure Computing Journal*, 21(9):186 – 203. doi:10.3390/s21093025.

Jacoby, M. and Usländer, T. (2020). Digital Twin and Internet of Things—Current Standards Landscape. *Applied Sciences*, 10(18). doi:10.3390/app10186519.

Mangia, M. et al. (2018). Rakeness-based compressed sensing and hub spreading to administer short/long-range communication tradeoff in iot settings. *IEEE Internet of Things Journal*, 5(3):2220–2233. doi:10.1109/JIOT.2018.2828647.

Naik, N. (2017). Choice of Effective Messaging Protocols for IoT Systems: MQTT, CoAP, AMQP and HTTP. In *2017 IEEE International Systems Engineering Symposium (ISSE)*, pages 1–.7, Vienna, Austria. doi:10.1109/SysEng.2017.8088251.

Nwankwo, E. et al. (2024). Integration of MQTT-SN and CoAP protocol for enhanced data communications and resource management in WSNs. *Bulletin of Electrical Engineering and Informatics Journal*, 13(3):1613–1620. doi:10.11591/eei.v13i3.5158.

Proos, D. P. and Carlsson, N. (2020). Performance Comparison of Messaging Protocols and Serialization Formats for Digital Twins in IoV. In *2020 IFIP Networking Conference (Networking)*, pages 10–18, Paris, France.

Seoane, V. et al. (2020). Performance Evaluation of the CoAP Protocol with Security Support for IoT Environments. In *Proceedings of the 17th ACM Symposium on Performance Evaluation of Wireless Ad Hoc, Sensor, & Ubiquitous Networks*, PE-WASUN '20, pages 41–48, Alicante, Spain. doi:10.1145/3416011.3424754.

Shelby, Z. (2012). Constrained RESTful Environments (CoRE) Link Format. RFC 6690, IETF. doi:10.17487/RFC6690.

Shelby, Z., Hartke, K., and Bormann, C. (2014). The Constrained Application Protocol (CoAP). RFC 7252, IETF. doi:10.17487/RFC7252.

Stefanini, R. et al. (2023). Selection of 4.0 sensors for small holders: the compromise between the advantages and the costs of the implementation. In *Proceedings of the 9th International Food Operations and Processing Simulation Workshop (FoodOPS 2023)*, pages 1–7, Athens, Greece. doi:10.46354/i3m.2023.foodops.007.

Sáez-de-Cámara, X. et al. (2024). Gotham Testbed: A Reproducible IoT Testbed for Security Experiments and Dataset Generation. *IEEE Transactions on Dependable and Secure Computing Journal*, 21(1):186–203. doi:10.1109/TDSC.2023.3247166.