Bridging Signals and Human Intelligence

Log Mining-Driven and Meta Model-Guided Ontology Population in Large-Scale IoT

David Graf[1,2](✉), Werner Retschitzegger[1], Wieland Schwinger[1], Elisabeth Kapsammer[1], and Norbert Baumgartner[2]

[1] Johannes Kepler University, Linz, Austria
{werner.retschitzegger,wieland.schwinger,elisabeth.kapsammer}@jku.at
[2] team Technology Management GmbH, Vienna, Austria
{david.graf,norbert.baumgartner}@te-am.net

Abstract. Large-scale Internet-of-Things (IoT) environments such as Intelligent Transportation Systems are facing tremendous challenges wrt. monitoring their operational technology (OT) not least due to its inherent heterogeneous and evolutionary nature. This situation is often aggravated by the lack of machine-interpretable information about the interdependencies between OT objects in terms of "semantic relationships", thus considerably impeding the detection of root causes of cross-system errors or interrelated impacts. Therefore, we propose a novel hybrid approach for identifying semantic relationships based on both, mined functional correlations between OT objects based on log files and domain knowledge in terms of an IoT meta model. For this, we firstly contribute a systematic discussion of associated challenges faced in large-scale IoT environments, secondly, we put forward an IoT meta model based on both, industry standards and academic proposals, and finally, we employ this meta model as guidance and target template for the automatic population of semantic relationships into an OT ontology.

Keywords: IoT · Operational Technology Monitoring · Hybrid Approach · OT Ontology Population · Intelligent Transportation Systems

1 Introduction

Operational Technology Monitoring. Large-scale *Internet-of-Things (IoT)* environments such as *Intelligent Transportation Systems (ITS)* are characterized by massive heterogeneities [11,21] of the underlying IoT-based Operational Technology (OT), due to different manufacturers, evolving standards, diverse capabilities and partly legacy components. Consequently, the monitoring of OT objects (e.g., video camera, traffic sensor), aka. *Operational Technology Monitoring (OTM)*,

This work is supported by: the Austrian Research Promotion Agency (FFG) under grant FFG Forschungspartnerschaften 874490 and by Erasmus+ under grant agreement No 2021-1-SI01-KA220-HED-000032218, project ID KA220-HED-15/21.

G. Memmi et al. (Eds.): KSEM 2022, LNAI 13369, pp. 571–585, 2022.
https://doi.org/10.1007/978-3-031-10986-7_46

across different subsystems of an ITS (e.g., video system, tunnel control systems) is challenging. This is aggravated by the fact that the majority of historically grown and therefore rather isolated subsystems lack machine-interpretable information about the semantic of interdependencies between OT objects. This is, however, an indispensable prerequisite for efficient OTM allowing to automatically identify *root causes* of cross-system errors (e.g., communication hub failure leads to unreachable devices) or *interrelated impacts* (e.g., CO2 sensor warning goes along with ventilation system start up) as well as for effective, e.g., predictive, maintenance strategies.

Hybrid Approach for OT Ontology Population. Defining all the semantic interdependencies on instance-level *manually*, is, however, not feasible for such large-scale systems due to the *sheer amount* of objects and its *omnipresent evolution* since objects and interdependencies are added, removed, and changed on a daily basis. Therefore, in our previous work [12,13] as a first step towards identifying these interdependencies, we proposed an automatic mining method focusing on identifying time-based functional correlations between pairs of OT objects (e.g., a sensor object is *functionally correlated* with a controller object) from log data comprising message streams of OT objects. Building on this work, we now put forward a dedicated approach for automatically instantiating semantic relationships (e.g., *isSensorFor* relationship between a sensor object and a controller object) based on mined functional correlations between OT objects by populating a corresponding OT ontology. Thereby, we adhere to a hybrid approach as also proposed in other domains like process mining [23], synergistically combining *"signal intelligence"* concealed in OT object's log data and *"human intelligence"* in terms of domain experts' apriori ontological knowledge about the environment under control.

Contribution and Paper Structure. paginationThe contribution of our paper is three-fold: Firstly, after elaborating on related work in Sect. 2, we discuss the complexity of real-world IoT environments and identify thereupon prevalent *challenges* throughout the process of mining interdependencies and the population of an OT ontology in Sect. 3. Secondly, in order to tackle these challenges, we elaborate a *meta model*, based on commonly used industry standards (*OPC-UA* [2]) and academic proposals (*IoT-O* [24]), comprising generic concepts for the IoT domain while at the same time allowing to plug-in further domain specific concepts in a framework-like manner in Sect. 4. Thirdly, we propose an *approach* exploiting the proposed meta model for the *automatic population of an OT ontology*, by instantiating semantic relationships between OT objects based on mined interdependencies and demonstrate the approach's applicability by means of *canonical semantic relationship patterns* prevalent in large-scale IoT systems in Sect. 5. Finally, we present evaluation details in Sect. 6 before discussing future work in Sect. 7.

2 Related Work

Addressing our primary goal namely populating an ontology with OT objects and their semantic relationships, closely related work can be found in the areas of (i) *IoT event log mining*, (ii) *organizational process mining*, (iii) *complex event (stream) processing*, as well as of course (iv) *ontology population* in various domains, which will be discussed in the following.

IoT Event Log Mining. Putting emphasize on the data characteristics of our work, [6] apply data mining techniques in order to enrich event log data, whereas [26] aim to mine patterns from semi-structured event logs. Both, however, focus on the enrichment of the log itself, rather than using logs for ontology population and further failure reasoning. Hromic et al. [15] transform air quality sensor data to a semantic representation, focusing in contrast to our approach, however, on the event data itself, rather than on the underlying resources, i.e., the individual sensors. Based on the correlation mined between events of a log, [25] determine dependencies between events among sensor data, whereas we focus on semantic relationships between OT objects.

Organizational Process Mining. Aiming to extract knowledge about underlying resources from event logs, promising mining approaches are reviewed by [19]. While most of them are widely similar to our approach, [8] discovers unknown dependencies in the medical domain, but in contrast to us, enriches the event logs themselves instead of populating a dedicated ontology.

Complex Event Processing. Regarding the streaming aspect in our application domain, the event-stream-based pattern matching approach [18] based on Allens interval-algebra [5] identifies temporal patterns rather than addressing the semantics of relationships. Endler et al. [9] transfer real-time sensor event data to a semantic model in terms of RDF triples, but, in contrast to our work, focus on context information of events, rather than deriving knowledge of underlying resources (OT objects and their semantic relationships).

Ontology Population in Various Domains. Regarding ontology population in various domains, [22] aims at populating a web service ontology using data-driven techniques such as clustering, however, primarily based on unstructured text documents rather than on semi-structured streaming data originating from event logs. This also applies to the work of [17] using semi-supervised classification to populate the content of text documents into an ontology. The approach of [7] populates an event ontology for monitoring vineyards grounded on a IoT sensor network aiming to mine causality relationships between events, instead of semantic relationships between IoT objects.

3 Ontology Population Challenges in Real-World IoT

This section systematically discusses ontology population challenges in real-world IoT from two different angles, firstly from the viewpoint of IoT data characteristics and - secondly - based on a brief summary of our functional correlation mining approach [12,13], from the perspective of deriving semantic relationships out of these rather basic functional correlations.

Challenges Arising from IoT Data Characteristics. The IoT-based ITS environment focused by our work, a national highway network, comprises *more than 1.000.000 OT objects* of more than 200 different objectTypes, ranging from simple sensors (e.g. CO2-sensor) and actuators (e.g., traffic light) to more complex systems (e.g., a video system) consisting of many objects of various types, being geographically distributed over 2.220 highway kilometers and 165 tunnels. Information about the object's *states* (e.g., warnings & failures) and *services* (e.g., traffic jam detections) is provided in terms of several logs, recording a stream of ten thousands messages per hour, the amount being strongly dependent on the kind of object - some reporting regularly, some seldom, some never at all in the considered period of time. Thus, *available information in log files about certain objects is limited*, especially considering a rather short period of time. Another crucial challenge are potentially different OT objects reporting with different messages (often having different message texts) about the same event *redundantly* being the result of partly isolated sub-systems where OT objects themselves report about their states or getting reported about by monitoring agents. In addition challenging are potentially *unreliable timestamps* exhibiting different semantics and expressing a partial order, only, due to differences in recording time or transmission delays. Last but not least, OT objects repeatedly report via *duplicate* messages about the same event or even report *irrelevant messages*, which is due to their intended usage regarding (i) *which kind of* information is recorded about objects and underlying events since often model- or vendor-dependent, and (ii) *how* often such information is recorded.

Mining Functional Correlations at a Glance. In order to address these challenges, in our previous work [12,13], we put forward a message-driven approach for mining interdependencies in terms of so-called *functional correlations*, between OT objects based on log files as a first step towards automatically populating an OT ontology. Thereby, a functional correlation arises when (i) an OT object's proper functioning is impeded by the (failure) *state* of another OT object (e.g., network failure) or (ii) an OT object realizes together with another OT object a certain *service* (e.g., traffic monitoring). Functional correlations, in a nutshell, are calculated based on the so-called *z-score* [20] representing the statistical dependence of two objects within a certain period of time. This dependence results from the temporal co-occurrence of messages (based on a configurable time lag) reported about these two objects (e.g., a videoServer object failure will occur nearly simultaneously with a failure reported by a camera object connected thereto). Random co-occurrences are explicitly considered by the z-score allowing to quantify the likelihood of OT objects being functionally correlated (values between 0.0 and 1.0). Those functional correlations with a likelihood below a certain (use-case specific and configurable) threshold (e.g., likelihood threshold of 0.5) are excluded for further processing. Which crucial challenges arise from these functional correlations between objects, when being used to derive meaningful semantic relationships, is outlined in detail in the following.

Challenges Arising from Functional Correlations. paginationDue to their nature of solely being calculated on mere temporal co-incidence, functional correlations lack any further semantics. Thus, the challenges for a proper "semantification" of functional correlations are manifold, comprising the determination of the *underlying semantic relationshipType* (e.g., isSensorFor), a *possible direction* (e.g., isEnergyDependentOn from sensor to energySupplyObject) and whether there is just a *single* semantic relationship underlying a functional correlation or even *multiple ones* (e.g., isSensorFor and plausibilizes). Another crucial challenge is that *knowledge conflicts* have to be dealt with in case that certain mined functional correlations do not correspond to domain knowledge, representing either *false positives* or indicating unexpected system behavior, thus being non-conform to domain knowledge. Finally, since functional correlations consider pairs of OT objects, only, handling more coarse-grained relationship-patterns like, e.g., *transitivity* of semantic relationships (e.g., transitive relationship between sensor, actuator and controller in-between) also quite relevant in practice, is challenging.

Since objects are able to play different roles when interacting with other objects (e.g., an object's role naturally differs between providing sensor values and notifying about its energySupply status), a vital challenge is to correctly interpret the object's role within co-occurring events by determining the events' type. Thereby, the type of an event can be exploited indicating which specific semantic relationshipType is represented by a certain functional correlation.

As already mentioned, the information about interdependencies between OT objects which can be mined from log files is limited due to several reasons, ranging from the infrequent reporting nature of certain OTs (i.e., sparsity of messages) to a lack of integration of individual parts of OT (i.e., unmonitored areas). Thus, although there are functional correlations in real-world, they are not mineable from log files. Thus, the challenge is to deal with such "blind spots" by properly exploiting available domain knowledge in order to establish appropriate semantic relationships without having inductive knowledge about functional correlations.

Last but not least, two further challenges arise with respect to providing for trustworthiness in terms of traceability of functional correlations and their derived semantic relationships. Firstly, it is challenging to estimate the *quality of functional correlations* in the sense of trustworthiness of the instantiation of a certain semantic relationship. By its very nature that, the more often a certain functional correlation between two dedicated objects can be mined in a log stream, the higher is its quality and the resulting trustworthiness of the established semantic relationship, in contrast to a functional correlation being based on rare and divergent minings. Secondly, omnipresent evolution of (i) the real-world, i.e., underlying OT is changed on a daily basis, and of (ii) the virtual-world, i.e., learning progress about underlying OT objects and relationships while more data is processed, is challenging.

4 A Meta Model for OT Ontology Population

As a *central prerequisite* in order to address these challenges in terms of our OT ontology population approach (cf. Sect. 5), we put forward an IoT meta model (cf. Fig. 1) allowing the incorporation of domain knowledge in the entire ontology population process. In other words, this meta model enables to bridge the gap between *"signal intelligence"*, i.e., knowledge gained from data in terms of functional correlations and *"human intelligence"*, i.e., knowledge gained from domain experts in terms of T-Box information. In particular, it describes core IoT concepts and their interdependencies in a generic way on both, instance-level and type-level, using UML class diagrams as basic formalism.

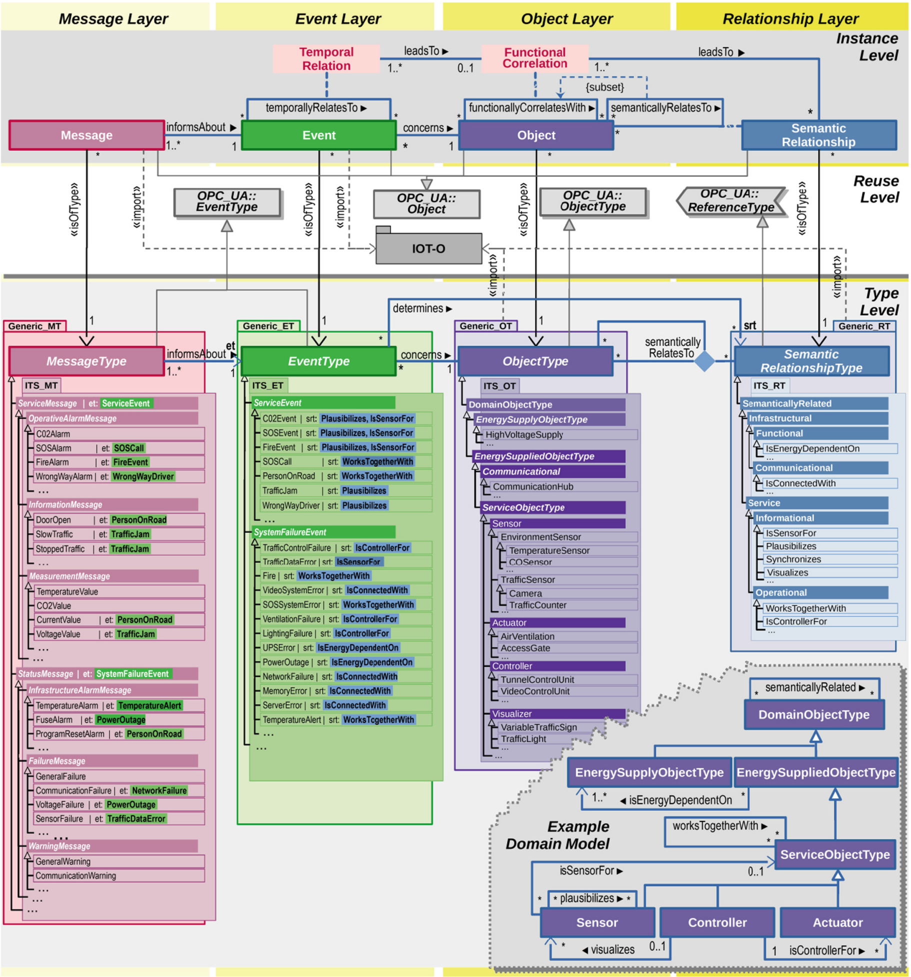

Fig. 1. Proposed meta model for OT ontology population

It serves as the central backbone of our ontology population approach and is used for the following purposes: Firstly, the instance-level concepts are, in terms of an ontology's A-Box, used as the target of our log-mining driven population approach as already mentioned in Sect. 3. Secondly, the type-level concepts are, in terms of an ontology's T-Box, used as a kind of template, further guiding the population approach by providing not only an extensive *generic type system*, but also packages of *domain-specific types* plugged in as subclasses. Based on the concepts of our meta model, a simple running example has been build up covering four IoT objectTypes as well as their basic semantic relationshipTypes (cf. bottom right of Fig. 1). The rationale behind the concepts contained in this meta model are manifold:

(1) The core *instance- and type-level* concepts covering *Message(Types)*, *Event(Types)*, *Object(Types)*, and *Relationship(Types)* are derived via subclassing from the *Open Platform Communications Unified Architecture (OPC UA)* [2], a widely used service-oriented, manufacturer and platform independent standard for industrial IoT applications, thus providing a solid basis for broad applicability in different domains (cf. Fig. 1 reuse-level).
(2) The *generic type system* represented as subclasses of the core concepts builds upon one of the most promising IoT ontologies (*IoT-O* [24]), as factored out in our previous survey on the 'IoT ontology jungle" [10], being a modular core-domain IoT ontology to represent connected devices reusing several other prominent ontologies like the *W3C SSN standard* [14]. It has to be noted that Fig. 1 just exemplary illustrates some of the available generic types, whereby the reuse of IoT-O has been denoted by package import dependencies.
(3) The *domain-specific types* are derived from the *DATEX-standard* [1] commonly used in the domain of ITS, again showing only a small fraction of several hundred objectTypes fully representing our application domain.
(4) The distinction between *Message(Types)* and *Event(Types)* was motivated by the need to bridge the gap between low-level log entries in terms of messages and high-level events representing occurrences of interest for the ITS operator, as also proposed in other domains like process mining [3,16]. This allows to aggregate messages sent by eventually different objects about the same real-world occurrence to a dedicated event. In Fig. 1, this aggregation is denoted by the corresponding "et" attribute within messageType subclasses being used in favor of explicit associations specializations for the sake of readability.
(5) The explicit representation and thus *"reification" of semantic relationshipTypes* in terms of a *class hierarchy* provides the benefit of type reusability between arbitrary subclasses of the eventType-hierarchy, depending on the requirements of the application domain. Analogous to message aggregations described before, the determination of these semantic relationshipTypes is denoted by an attribute "srt" within the eventType subclasses.
(6) The explicit representation of *TemporalRelation* and *FunctionalCorrelation* in terms of *UML association classes* represents the basis for recording *provenance information*, thereby providing the rationale for the population of certain semantic relationshipTypes (cf. Sect. 5).

5 OT Ontology Population Approach

By applying the proposed meta model, we aim to incorporate domain knowledge along the entire ontology population process, from the early message-level, via the mining of functional correlations, to the instantiation of semantic relationships thereof. Overall, we stick to the "closed-world-assumption", more precisely, each functional correlation between a pair of objects being mined from the log data, is aligned with domain knowledge provided by the meta model in form of explicated relationshipTypes and their properties (e.g., multiplicities) to derive semantic relationships. In case the domain knowledge is not sufficient to unambiguously interpret the functional correlation, additional *heuristic guidelines* (e.g., how to deal with optional relationships) are used for populating an appropriate semantic relationship into the ontology. Based on that overall rationale, our ontology population approach can be characterized along our challenges described in Sect. 3 leading to the following main cases, namely (1) *semantification of functional correlations*, (2) *semantic differentiation of relationships based on eventTypes*, (3) *instantiation of silent objects and silent relationships*, and (4) *allocation of provenance information.*

Canonical Relationship Constellations. In order to illustrate the details of these cases as well as to get a grip on the complexity of coarse-grained real-world relationships between numerous OT objects while preserving, at the same time, the general applicability of our approach, we *focus on canonical constellations of relationships* between a minimal number of necessary OT objects as a baseline. The rational behind these canonical constellations is mainly derived from the expressiveness of our T-Box formalism provided by the meta model, covering relationship properties like name, direction, multiplicity, multi-dependencies, transitivity, inheritance, single/multiple relationships, and relationship roles. By focusing on these relationship properties, Figs. 2, 3, 4, 5 and 6 visualize the instantiation of semantic relationships between OT objects derived from functional correlations thereby basing on the exemplary domain model shown in Fig. 1.

(1) *Semantification of Functional Correlations.* In its most basic case, semantification of a functional correlation entails the instantiation of a relationshipType between the functionally correlated objects. Thereby, fundamentals like a meaningful name and a navigation direction are manifested (see Fig. 2(a))[1]. However, the instantiation is naturally constrained by the relationship's corresponding objectTypes' multiplicity, thereby expressing mandatory and optional semantic relationships (see Fig. 2 (b) and (c)).

Furthermore, supposedly mined functional correlations can be simply discarded if no relatonshipType counterpart can be found in the T-Box. This is specifically relevant wrt. real-world relationships involving numerous objects, i.e., forming a "functional correlation cluster" where each object is functionally

[1] Please note that the notation sticks to UML class and object diagrams, whereas the red lines between objects represent a functional correlation and yellow arrows represent the impact one object has on another object in real-world.

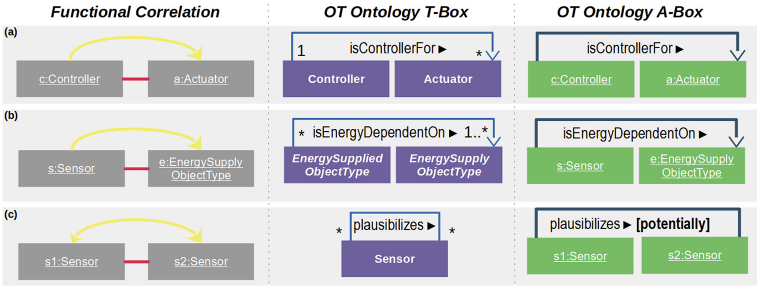

Fig. 2. Providing a meaningful name and direction (a) as well as considering mandatory (b) and optional (c) multiplicity constraints

correlated with all other objects. However, not necessarily each object has a direct relationship to each other object (i.e., false-positives), thus not all functional correlations lead to an instantiation of a semantic relationship (cf. (a) and (b) in Fig. 3). This goes along with real-world transitive relationships being not expressed by pairwise functional correlations. Thus, T-Box information of semantic relationshipTypes between intermediate objectTypes allows to put functional correlation between pairs of objects and the semantic relationships in-between "in the correct order" (see sensor - controller - actuator constellation of Fig. 3 (c)). Finally, if semantic relationshipTypes between objectTypes are not foreseen in the T-Box, this also allows to eliminate functional correlations being irrelevant (i.e., again false-positives).

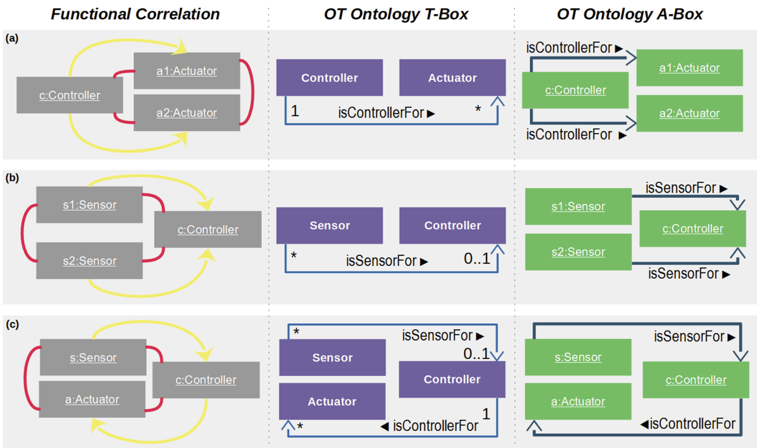

Fig. 3. Discarding false-positive functional correlations resulting from complex real-world relationship Constellation

Since domain knowledge might be expressed at more general levels, i.e., exploiting modeling "languages" generalization mechanisms, the ontology population approach considers all relationshipTypes along the inheritance hierarchy of the involved objectTypes (e.g., like the isSensorFor relationship visualized in Fig. 4).

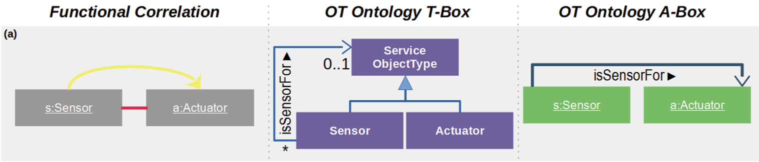

Fig. 4. Exploiting modeling "languages" generalization mechanisms

(2) *Semantic Differentiation of Relationships Based on EventTypes.* Multiple semantic relationshipTypes between two objectTypes might not necessarily all hold between two specific objects of those objectTypes simultaneously. This is because objects might relate in different roles to individual objects at different times. For example, while sensors and controllers may be in an *isSensorFor* and in a *visualizes* relationship, a sensor's controller might not necessarily visualize the sensor, thus the sensor and the controller might not always play a role in both relationships at the same time. To address this challenge, we extended our mining approach (discussed in Sect. 3) in order to exploit also eventTypes. For this, we calculate functional correlations between pairs of eventTypes about distinct objects rather than between pairs of objects, only (see colored circles next to functional correlations of Fig. 5). Based on that, the domain model

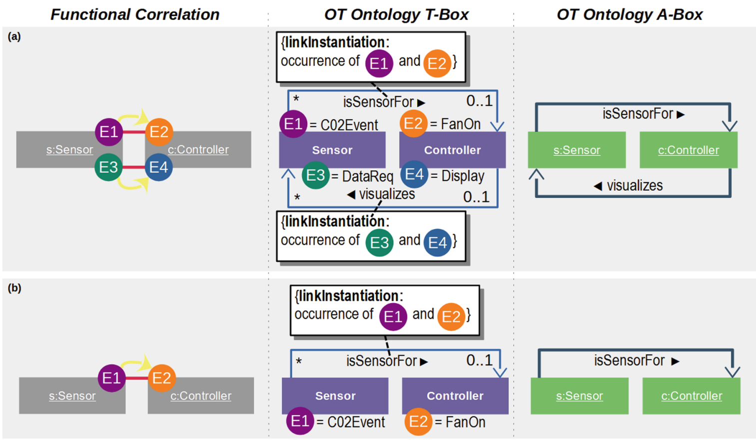

Fig. 5. Semantic differentiation of relationships based on event types

defines which specific semantic relationship shall be instantiated in the light of an eventType-specific functional correlation (see Fig. 1 association between eventType and semantic relationshipType).

Capitalizing on that, it also allows, even for a single possible semantic relationship, a more fine-grained differentiation in which cases, based on the eventType, a specific semantic relationship is instantiated.

(3) *Instantiation of Silent Objects and Silent Relationships.* The "prescriptive nature" of the domain knowledge can be utilized to address semantic relationships to certain objects which are required, although underlying functional correlations are not yet mineable from event log data, representing "blind spots". Concretely, we exploit domain knowledge in terms of semantic relationshipType multiplicity constraints (specifying that certain objects need to be in a relationship with another object) to populate those semantic relationships to not (yet) perceived objects as *"silent"* ones into the ontology (see Fig. 6). In contrast to the previous population cases, which are data-driven by functional correlations, the instantiation of silent objects and silent relationships is driven by domain knowledge. In the light of OTM, this mechanism is beneficial, since it enables identifying objects being the potential cause of or being affected by a failure of an object not (yet) recorded in the log. On the downside, this, however, comes at the costs that, as mining continues, already existing silent objects and relationships eventually have to be replaced by objects now perceived in the log (cf. Sect. 7). Thus, the existence of a particular silent object and its silent relationships in the A-Box might be temporary, only.

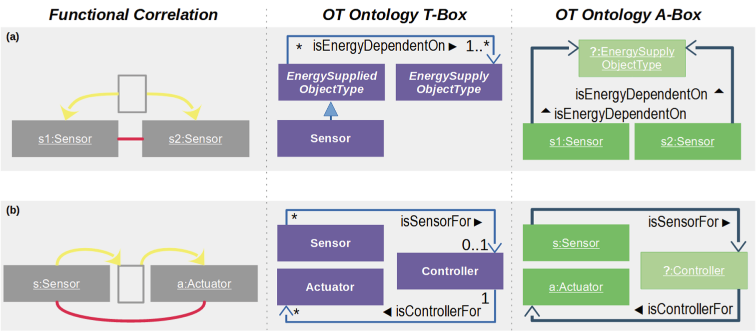

Fig. 6. Silent objects and silent relationships

(4) *Allocation of Provenance Information.* In order to improve trustworthiness and support traceability as well as to trace evolution aspects, our ontology population approach enables adding further semantics in terms of provenance information of functional correlations directly in the OT ontology's A-Box. Currently, provenance information manifested includes (i) a history of the probability values of functional correlations to trace their evolution over time, (ii) meta-information

about underlying number of messages over time, functional correlation values and their underlying z-score values, thus being an indicator for the trustworthiness of functional correlations, and (iii) the concrete eventTypes underlying functional correlations, since those may give an indication about previously not explicated semantic relationships of the real-world, thus pointing the way for a (semi-)automatic evolving T-Box (cf. Sect. 7).

6 Scenario-Based Evaluation

Evaluation Based on Simulated Domain-Derived Scenarios. As a first step towards evaluating the appropriateness of our approach in the light of constellations of relationships prevalent in large-scale IoT systems (cf. example domain model in Fig. 1), our evaluation is based on a set of domain-derived scenarios. Thereby, scenarios comprise *abnormal operation cases* (the improper function of OT objects is impeded by other OT objects), as well as *normal operation cases* (inter-operating objects in the light of certain services). Since ground-truth data (gold standard A-Box) needed for evaluating those scenarios and accompanying relationship information is not readily available, we resort to simulated data based on real-world datasets. Thereby, characteristics of simulated data, such as the distribution of messages, the frequency of failures of certain OT types, correspond to the characteristics of real-world data.

Synthetic Log Data Generation Framework. In order to generate the simulated data, i.e., synthetic log data used for evaluation, we developed a log data generation framework able to simulate the message behavior of real-world OT objects in an artificial IoT environment. Thereby, we define a gold standard A-Box considering the domain-derived scenarios, i.e., OT objects, relationships in-between as well as the impacts of their corresponding types on other OT objects being finally used as ground-truth for evaluation. The gold standard itself is then used as input for the synthetic log data generator in order to simulate the contained OT objects and outputs a synthetic log file. Parameterization of the (i) message rate, which specifies the probability of a randomly occurring message, allows to control the message behavior of certain objectTypes, as well as the (ii) amount and time of manually injected messages allow to control the domain scenarios (e.g., simulating a video server failure impacting several cameras connected to that video server).

Scenario-Specific Experiments Evaluating Performance and Robustness. Based on the synthetic log data of six different domain scenarios, we applied our hybrid approach and systematically evaluated two dimensions, namely (1) *performance* of our approach in terms of (i) *accuracy*, (ii) *efficiency*, and (iii) *effectivity*, as well as (2) *robustness* of our approach regarding (a) *amount of data* by varying the temporal extent of simulated log data, (b) *parameterization* by varying functional correlation mining configurations, and (c) *number of objects* by varying simultaneity of co-occurring scenario instances. As a benchmark for our evaluation, we use performance and robustness results of experiments realized without exploiting domain knowledge provided by the meta model of Fig. 1.

Lessons Learned. To briefly sum up our lessons learned of 54 experiments (combinations of two evaluation dimensions, i.e., performance and robustness, for each of the 6 domain-derived scenarios), first of all, regarding *accuracy*, we are able to achieve noticeable improvements in terms of precision compared to the benchmark, which is due to the employed semantic type-model (message-, event-, object-, and relationshipTypes), whereas at the same time, there is no degradation with respect to runtime recognizable. Regarding *efficiency*, we are able to achieve noticeable improvements in runtime while accuracy remained stable, which is due to applying objectType-specific configurations (considering their message behavior) when mining functional correlations. Finally, analyzing *effectivity*, semantic relationships are recognized faster, which is due to considering the sparsity of certain message- and eventTypes.

7 Future Work

One major issue for future work results from the fact that our ontology population approach does not yet sufficiently address the evolution aspect being omnipresent in real-world large-scale ITS. We therefore intend to provide mechanisms for identifying different kinds of *real-world evolution* meaning that objects and their corresponding semantic relationships are added, removed, and changed on a daily basis and based on that, to seamlessly co-evolve the *virtual-world* in terms of learning/unlearning real-world objects and semantic relationships thereby resembling the digital shadow paradigm [4]. This especially includes mechanisms of merging silent objects and relationships with real ones, identified at a later time. Furthermore, our approach relies on the existence and the quality of incorporated domain knowledge, which might lead to improper results in case of insufficient domain knowledge expressed by the T-Box. Regarding evaluation, we have to mention that, because of the canonical character of our domain-derived scenarios, completeness of possible cases was favored instead of covering real-world complexity. Therefore, for providing a more in-depth evaluation, a thorough and systematic combination of our canonical relationship constellations would be a crucial prerequisite, complemented by a chart-based visualization using rapid prototyping environments like Oracle APEX.[2]

References

1. DATEX II. https://www.datex2.eu
2. Open Platform Communications Unified Architecture (OPC UA). https://opcfoundation.org

[2] https://apex.oracle.com/.

3. Pecchia, A., Weber, I., Cinque, M., Ma, Y.: Discovering process models for the analysis of application failures under uncertainty of event logs. Knowl.-Based Syst. **189**, 105054 (2020)
4. Brauner, P., et al.: A computer science perspective on digital transformation in production. ACM Trans. Internet Things **3**(2), 1–32 (2022)
5. Allen, J.F.: Maintaining knowledge about temporal intervals. Commun. ACM **26**(11), 832–843 (1983)
6. Amato, F., et al.: Detect and correlate information system events through verbose logging messages analysis. Computing **101**(7), 819–830 (2019)
7. Belkaroui, R., et al.: Towards events ontology based on data sensors network for viticulture domain. In: Proceedings of the 8th International Conference on the Internet of Things, pp. 1–7. ACM (2018)
8. Detro, S., et al.: Enhancing semantic interoperability in healthcare using semantic process mining. In: Proceedings of International Conference on Information Society and Technology, pp. 80–85 (2016)
9. Endler, M., et al.: Towards stream-based reasoning and machine learning for IoT applications. In: Intelligent System Conference, pp. 202–209. IEEE (2017)
10. Graf, D., et al.: Cutting a path through the IoT ontology jungle - a meta survey. In: International Conference on Internet of Things and Intelligence Systems. IEEE (2019)
11. Graf, D., Schwinger, W., Retschitzegger, W., Kapsammer, E., Baumgartner, N.: Event-driven ontology population - from research to practice in critical infrastructure systems. In: Rocha, Á., Adeli, H., Dzemyda, G., Moreira, F., Ramalho Correia, A.M. (eds.) WorldCIST 2021. AISC, vol. 1366, pp. 405–415. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-72651-5_39
12. Graf, D., et al.: Dependency mining in IoT - from research to practice in intelligent transportation systems. In: Rocha, A., Adeli, H., Dzemyda, G., Moreira, F. (eds.) Information Systems and Technologies. WorldCIST 2022. LNCS, vol. 469. Springer, Cham (2022). https://doi.org/10.1007/978-3-031-04819-7_26
13. Graf, D., et al.: Semantic-driven mining of functional dependencies in large-scale systems-of-systems. In: Rocha, Á., Ferrás, C., Méndez Porras, A., Jimenez Delgado, E. (eds.) Information Technology and Systems. ICITS 2022. LNCS, vol. 414. Springer, Cham (2022). https://doi.org/10.1007/978-3-030-96293-7_31
14. Haller, A., et al.: The SOSA/SSN ontology: a joint WEC and OGC standard specifying the semantics of sensors observations actuation and sampling. In: Semantic Web, vol. 1, pp. 1–19. IOS Press (2018)
15. Hromic, H., et al.: Real time analysis of sensor data for the IoT by means of clustering and event processing. In: Proceedings of International Conference on Communications, pp. 685–691. IEEE (2015)
16. Janiesch, C., el al.: The Internet of Things meets business process management: a manifesto. IEEE Syst. Man Cybern. Mag. **6**(4), 34–44 (2020)
17. Jayawardana, V., et al.: Semi-supervised instance population of an ontology using word vector embeddings. In: Proceedings of International Conference on Advances in ICT for Emerging Regions, pp. 217–223. IEEE (2017)
18. Körber, M., Glombiewski, N., Morgen, A., Seeger, B.: TPStream: low-latency and high-throughput temporal pattern matching on event streams. Distrib. Parallel Databases **39**(2), 361–412 (2019)
19. Matzner, M., Scholta, H.: Process mining approaches to detect organizational properties in CPS. In: European Conference on Information Systems (2014)
20. Messager, A., et al.: Inferring functional connectivity from time-series of events in large scale network deployments. Trans. Netw. Serv. Manag. **16**(3), 857–870 (2019)

21. Noura, M., Atiquzzaman, M., Gaedke, M.: Interoperability in Internet of Things infrastructure: classification, challenges, and future work. In: Lin, Y.-B., Deng, D.-J., You, I., Lin, C.-C. (eds.) IoTaaS 2017. LNICST, vol. 246, pp. 11–18. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00410-1_2
22. Reyes-Ortiz, J., et al.: Web services ontology population through text classification. In: Proceedings of Conference on Computer Science and Information Systems, pp. 491–495. IEEE (2016)
23. Schuster, D., et al.: Utilizing domain knowledge in data-driven process discovery: a literature review. Comput. Ind. **137**, 103612 (2022)
24. Seydoux, N., Drira, K., Hernandez, N., Monteil, T.: IoT-O, a core-domain IoT ontology to represent connected devices networks. In: Blomqvist, E., Ciancarini, P., Poggi, F., Vitali, F. (eds.) EKAW 2016. LNCS (LNAI), vol. 10024, pp. 561–576. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-49004-5_36
25. Zhu, M., et al.: Service hyperlink: modeling and reusing partial process knowledge by mining event dependencies among sensor data services. In: Proceedings of International Conference on Web Services, pp. 902–905. IEEE (2017)
26. Zhuge, C., Vaarandi, R.: Efficient event log mining with LogClusterC. In: Proceedings of International Conference on Big Data Security on Cloud, pp. 261–266. IEEE (2017)