WoTDT: an Extension of the WoT Thing Description Ontology for Digital Twins in the Construction Domain

Salvador González-Gerpe[1,*], Andrea Cimmino[1], Socorro Bernardos[1], María Poveda-Villalón[1] and Raúl García-Castro[1]

[1]Ontology Engineering Group, Universidad Politécnica de Madrid, Spain

Abstract

Digital Twins (DTws) are a new category of technologies that enable construction industry to improve the precision of its predictions, make more logical decisions, and develop well-informed plans. To address these problems, it is essential to develop a method for semantically describing the various aspects that model the architecture of a DTw, i.e., representing DTws with formal semantics. This article introduces an extension of the W3C Web of Things (WoT) Thing Descriptions (TD) ontology, called "WoTDT " (WoT Digital Twin ontology). The WoTDT ontology is based on the most widely adopted architecture that models DTws into five dimensions. Furthermore, since the ontology has been created as a WoT extension, it will offer benefits such as enabling the discovery of services [1] across dimensions or enhancing the accessibility [2] of the information from a specific dimension, thereby promoting data interoperability. In addition, an evaluation of the ontology has been performed to ensure its quality by verifying that it is pitfall-free and covers all identified requirements. Finally, an example of the WoTDT ontology applicability is shown in the context of the European H2020 construction-related project COGITO.

Keywords

Digital Twins, Ontology, Web of Things, Thing Descriptions

1. Introduction

Construction is one of the most important industries worldwide, and its impact is significant in various aspects, such as environmental [3] or production costs [4]. The complexity of construction projects, the participation of numerous stakeholders, and the lack of transparency with data related to these projects contribute to increased costs and delays [5]. To address this matter, a potential approach consists of collecting valuable data associated with construction sites and analysing it to extract meaningful information that allows informative decision making that may reduce costs or delays [6].

Digital Twins (DTws) aim to address and solve these problems rooted in construction projects. DTws are a planning and optimisation approach that is based on simulations, allowing construction sites to improve the accuracy of their predictions, make more logical decisions, and develop well-informed planning processes [7]. Moreover, DTws possess two-fold characteristic:

LDAC 2024: 12th Linked Data in Architecture and Construction Workshop, June 13–14, 2024, Bochum, Germany

salvador.gonzalez.gerpe@upm.es (S. González-Gerpe)

0000-0003-1550-0430 (S. González-Gerpe); 0000-0002-1823-4484 (A. Cimmino); 0000-0003-1790-5941 (S. Bernardos); 0000-0003-3587-0367 (M. Poveda-Villalón); 0000-0002-0421-452X (R. García-Castro)

a physical aspect, which includes data pertaining to the actual state of affairs in the physical world, and a virtual aspect, encompassing the present operational status. The use of DTws in construction projects has been proved to increase operating efficiency by 35%, sustainability by 50%, reduce labour costs by 50%, increase productivity by 20%, and extend the space that can be used by 15% [8].

Even though DTws offer substantial advantages to construction projects, nowadays there is a lack of a standardised architecture for DTws that describes their design and guides their development and implementation. However, there are numerous DTw architectures in the literature [9], one of the most recognised and adopted is one that models DTws into three dimensions [10]: physical, virtual, and the connections between them. Several research proposals have refined these dimensions [11, 12, 13, 14], nevertheless, only one has extended these dimensions, adding two new and promoting a five-dimensional architecture [15], which has become the reference architecture for DTws. However, such approach lacks formal semantics to describe the different dimensions and their features. This may lead to a set of problems to achieve semantic interoperability among DTws: i) there is no formal approach to discover the services offered by DTws and also to know to which dimension they belong; ii) there is no description of which model follows the data of a DTw; iii) there is no security specification for the dimensions of a DTw to address these problems. As a result, it is essential to develop a method for semantically describing the various aspects that model the five-dimensional architecture in a DTw.

In this article, an extension of the W3C standard Thing Descriptions (TD) ontology [16] proposed by the Web of Things group (WoT), called WoTDT (WoT Digital Twin ontology), is presented. The goal of WoTDT is to provide formal semantics to represent the five dimensions of a DTw and its features, allowing semantic discovery by reusing the TD ontology, and a common approach to describe the capabilities of a DTw allocated by dimensions. The WoTDT ontology was developed using the LOT methodology [17], following ontology engineering good practices. Finally, an evaluation of the ontology has been performed to ensure its quality by checking that it is pitfall-free and that it covers all the identified requirements.

The WoTDT ontology has been developed and used in the context of the European project COGITO. In this project WoTDT has been used to model the DTw of railway network construction site (with information such as workers, machines, IoT, . . .). This ontology is public available as machine-readable format and as a human-readable document[1]. Furthermore, since the ontology is based on WoT, it allows i) to discover the services provided by the different dimensions [1]; ii) to facilitate the accessibility of information of a specific dimension [2]; iii) to define the security specification of each dimension using WoT capabilities by declaring a security scheme. Furthermore, WoTDT ontology can represent domain independent DTw, Digital Shadows or Digital Models, and it is meant to be extended, when required, with domain specific terms (such as sensors, building parts or machinery). Practitioners shall use the WoTDT ontology as a common data model for interoperability within their infrastructure, and extend it with specific terms from their infrastructure domain to increase the expressiveness of the ontology.

The rest of this article is structured as follows. Section 2, provides an overview of the literature

[1] https://w3id.org/def/digitaltwin

and research proposals related to DTws in terms of architectures, ontologies, and WoT. Section 3 includes the development of the WoTDT ontology using the LOT methodology. Section 4 provides a use case of this ontology in the COGITO project. Finally, Section 5 recaps the conclusions of this research.

2. State of the Art

The concept of "Digital Twin" was coined by Michael Grieves in the field of *Product Lifecycle Management* (PLM) [10], defining it as a virtual representation of a physical object or system that can be used to analyse its behaviour and performance. The proposal models an architecture of a DTw that consists of three dimensions: the physical, which represents the actual asset in the real world; the virtual, which represents the software representation of the physical asset; and the connection parts, which exist between the physical and virtual dimensions.

After the three-dimensional approach, the researchers focused on optimising aspects of these dimensions to tackle new challenges [11, 12, 13, 14]. For example, NASA [12] improved existing simulations performed in the virtual dimension with the goal of improving the management of US Air Force aircraft throughout their lifetime by creating individualised structural management plans, and in the physical dimension by including real-time data collection.

In 2017, Tao and Zhang [15] proposed a five-dimensional domain independent architecture model of a DT, extended from the three-dimensional approach, that provided theoretical guidance for the digitalisation and intellectualisation of the manufacturing industry. Also, they adapted the five-dimensional approach to specific domains, such as prognostics and health management domain [18]. In this proposal, the physical dimension of Grieves' proposal is called the Physical Entity Dimension, and the connections are called Digital Twin Connection Dimension. The virtual dimension of Grieves' proposal is divided into three dimensions: the Virtual Entity Dimension, which includes the description of the DTw models (such as rules, behavioural, physical, and geometric models); the Digital Twin Data Dimension, which stores all the data used by the DT; and the Digital Twin Service Dimension, which includes all the services used by the DTw.

The goal of the WoTDT ontology is to model this five-dimensional architecture. Therefore, DTw proposals in the literature that rely on ontologies to model this architecture, or similar ones, and/or apply WoT in the context of DTws are analysed.

Ontologies in DTws: The application of Semantic Web technologies in the creation, management, and modelling of DTws in industry has grown in recent years. In particular, guidelines have been proposed on how to incorporate ontologies and their benefits for DTws [19]. Using these ontologies has one of these two main goals: i) to improve the visualisation and understanding of DTw data [20], or ii) to describe the architecture model of a DT [21], which is the goal of the WoTDT ontology.

In the context of the second goal there is a study proposed by Sumit Singh et al. [22], where an ontology is developed to capture the domain knowledge and maintain the semantics of the asset functions and basic characteristics during its operational phase. Another proposal by Charles Steinmetz et al. [21] conducted a study that expands the IoT-Lite ontology [23] by incorporating classes associated with the virtual dimension of the three-dimensional model, allowing external applications and systems to access the virtual dimension with a particular protocol. In addition, Marah and Challenger [24], introduce a definition of an ontology for agent-based digital twins for Cyber-Physical Systems. Furthermore, Meijers from Microsoft Azure introduced the Digital Twins Definition Language (DTDL) [25] based on an RDF metamodel to represent DTws within their proprietary tools.

WoT in DTws: In the literature, WoT has not been used to model the dimensions of a DTw. However, the closest proposal is the research presented by Ricci et al. [26], where a description of an ecosystem of DTws has been developed using semantic web technologies in the healthcare domain. In their research, the combination of DTw and WoT allows greater precision in monitoring construction assets in real time, interoperability with other systems, and better behaviour and performance over time. Another research, presented by Pittaras et al. analysed how to combine the WoT standard with smart contracts to describe DTws of physical devices [27]. In addition, in the research conducted by Luca Bedogni and Federico Chiariotti [28], an architecture based on the WoT standard is developed to provide a DTw of a smart environment, such as sensors used in a construction-related project. Finally, Xiaochen Zheng et al. in [29] bring together these technologies, where WoT is applied in Cognitive DTws (CDTws) with the intention of standardising and achieving data interoperability.

This article introduces the first proposal, as far as we know, that covers all five dimensions of a DTw and their characteristics. Moreover, since the ontology has been developed as an extension of WoT, it will provide advantages such as enabling the discovery of services across dimensions, or facilitating the access to the information from a particular dimension, thereby promoting data interoperability.

3. WoTDT Ontology Development

In this article, a novel approach is presented to semantically represent Digital Twins (DTw) that follow the five-dimensional architecture along with their characteristics and functionalities. To this end, an extension of the W3C standard Thing Description (TD) ontology published by the Web of Things (WoT) group has been developed. This novel ontology, named WoTDT provides an approach that describes all the dimensions and internal capabilities of a DTw, and ranges from related metadata to the functions it offers using TDs enriched with this ontology.

The WoTDT ontology has been developed following the LOT (Linked Open Terms) methodology [17]. This methodology is based on the ontological engineering activities described in the NeOn methodology [30]. The LOT methodology outlines a process that includes the following activities: i) specifying ontological requirements; ii) implementing the ontology; iii) publishing the ontology; and iv) maintaining the ontology. The following subsections provide further details about these activities and their implementation to deliver the WoTDT ontology.

3.1. WoTDT Ontology Requirements

The specification of the ontology requirements is derived from the analysis of the proposed five-dimensional model and refined during its adoption and use in the COGITO H2020 project. From these inputs, a collection of ontological requirements was generated in the form of competency questions and sentences in natural language. Table 1 shows the fifteen requirements obtained from the specification process.

Table 1
Requirements for WoTDT ontology.

ID	Competency Question / Statement - Possible answer
WOTDT-1	A Digital Twin is a Thing.
WOTDT-2	A Digital Twin contains 5 dimensions.
WOTDT-3	Physical Entity is a dimension that represents the real world asset of the Digital Twin.
WOTDT-4	Virtual Entity is a dimension that represents the different models used in the Digital Twin.
WOTDT-5	Digital Twin Data is a dimension where are stored all the data used in the Digital Twin.
WOTDT-6	Digital Twin Services is a dimension where all the services of the Digital Twin are described.
WOTDT-7	Digital Twin Connection is a dimension where all the connections between other dimensions in the Digital Twin are described.
WOTDT-8	Physical Entity dimension can have components.
WOTDT-9	Which kind of components can be described in the Physical Entity dimension? - The component can be from the physical asset that the Digital Twin is modelling, to the different devices like sensors or actuators that read or act over the specific physical asset.
WOTDT-10	Virtual Entity dimension can have models.
WOTDT-11	Which kind of models can be described in the Virtual Entity dimension? - The models can be from rules, behavioral, physical and geometric models to semantic models like ontologies.
WOTDT-12	Digital Twin Data dimension can have resources that can be used to represent the different type of data stored at the Digital Twin.
WOTDT-13	Digital Twin Service dimension can have Interaction Affordances from the WoT Thing Descriptions ontology to represent the different services used at the Digital Twin.
WOTDT-14	Digital Twin Connection dimension can have different connections.
WOTDT-15	Which type of connections the Digital Twin Connection dimension can describe? - The connections defined in the Digital Twin Connection dimension are described with the different existing elements of other dimensions of the Digital Twin such as models, resources and interaction affordances; and the connections with external Things such as other Digital Twins.

3.2. WoTDT Implementation

To streamline the implementation process, a conceptualisation of the ontology was proposed using the requirements identified in the previous step as input. This conceptualisation incorporates the key concepts to represent the five dimensions when applied to the construction domain. Figure 1 shows the representation of this conceptualisation, which follows the Chowlk Visual Notation [31] that establishes a connection between the stereotypes used in the profile with OWL, RDF (S) constructs and certain OWL 2 constructs. The use of prefixes indicates the ontologies in which each concept or relation is defined. For example, `dt:DigitalTwin` is defined in the *"https://w3id.org/def/digitaltwin#"* namespace. Furthermore, this ontology reuses the core ontology of SAREF, such as `saref:Device`; the DCAT ontology (Data Catalog Vocabulary), such as `dcat:Resource`; the HCTL ontology (WoT Hypermedia Controls Ontology), such as `hctl:hasTarget`; apart from the WoT Thing Description ontology.

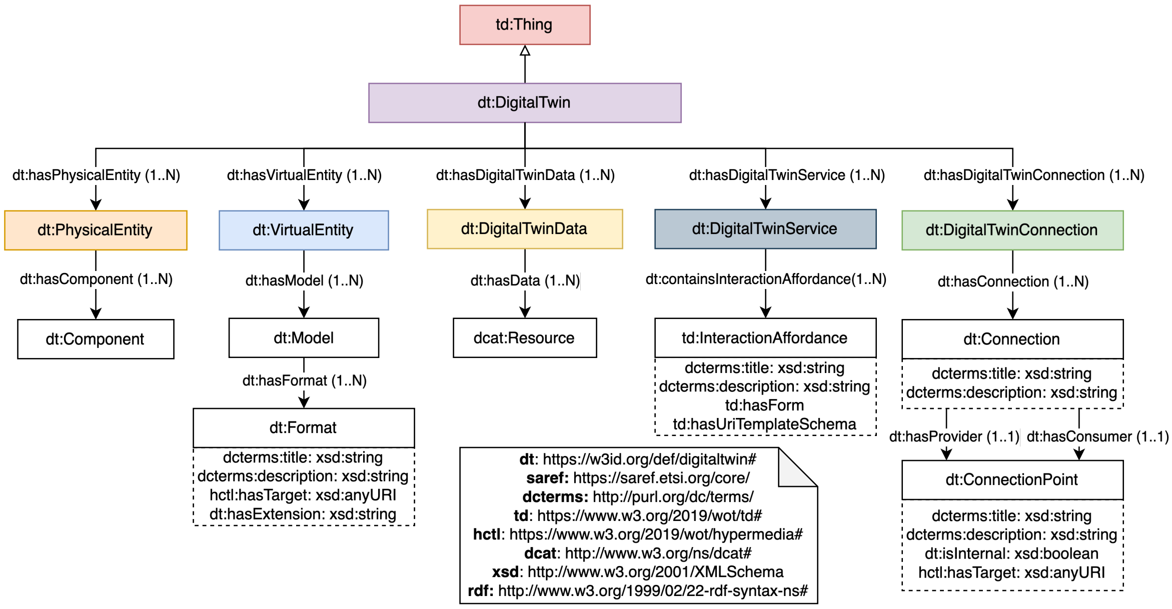

Figure 1: WoTDT conceptualization overview.

As shown in Figure 1, the class of `dt:DigitalTwin` is represented as a `rdfs:subClassOf` of `td:Thing`, allowing inheritance of properties contained in the `td:Thing` class. In addition, the different dimensions are conceptualised using the classes: `dt:PhysicalEntity`, `dt:VirtualEntity`, `dt:DigitalTwinData`, `dt:DigitalTwinService` and `dt:DigitalTwinConnection`. These five-dimensional conceptualisations will be explained separately in the following subsections.

Physical Entity Dimension: In the WoTDT ontology, the `dt:PhysicalEntity` class represents the dimension of the physical entity, defined as a set of various subsystems, objects, and sensors. devices. These subsystems can include dynamic systems, control systems, maintenance systems, among others, and can be combined for a specific task [15]. To represent all these subsystems, objects, and sensor devices, the class `dt:Component` has been created. The `dt:Component` class can be shown in Figure 2 where it contains subclasses such as `dt:MainComponent`, that represents the main component of the dimension of the physical entity in the case of multiple; and `saref:Device`, that represents devices such as sensors (`saref:Sensor`) and actuators (`saref:Actuator`), allowing this dimension to read and act on the physical asset of the DTw.

Virtual Entity Dimension: For the dimension of the virtual entity, in the WoTDT ontology, this dimension is represented in the class `dt:VirtualEntity`, defined as a set of various data models that represent all the information allocated in the digital twin [15]. To represent the models allocated in this dimension, the class `dt:Model` has been created, defined as a

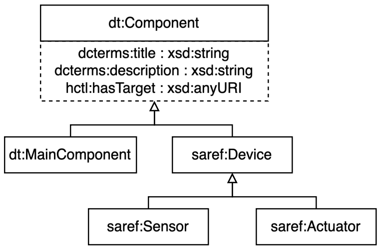

Figure 2: Class `dt:Component` and its subclasses.

representation or conceptualisation of the data registered in the dimension of the virtual entity. Following the same approach as in the previous subsection, in Figure 3 are represented the different subclasses of the class `dt:Model`, containing from non-semantic models such as `dt:RulesModel`, `dt:BehavioralModel`, `dt:PhysicalModel` and `dt:GeometricModel` described in [15]; and semantic models such as `dt:OntologyModel`, `dt:MappingModel` and `ShapesModel`. Furthermore, as shown in Figure 1, each of these `dt:Model`, has a specific format, represented by the class `dt:Format`, containing attributes (`owl:DatatypeProperty`) such as the route where the model is stored (`hctl:hasTarget`), and the extension of the model (`dt:hasExtension`), that can be for example in the case of a `dt:OntologyModel` the *"ttl"* extension that refers to Turtle serialisation.

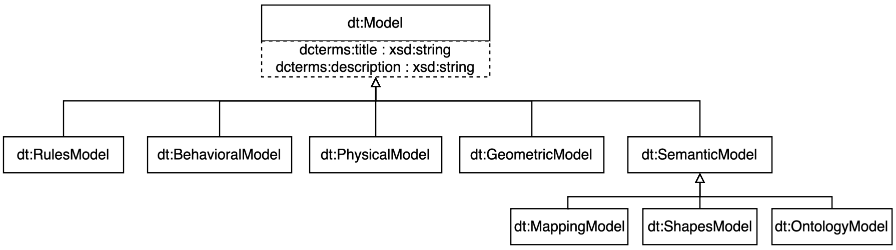

Figure 3: Class `dt:Model` and its subclasses.

Digital Twin Data Dimension: Once the physical and virtual entities have been explained, the dimension of digital twin data will follow. In the WoTDT ontology, the dimension of the digital twin data is represented by the class `dt:DigitalTwinData`, defined as the dimension of the digital twin where all data are contained and used by the DTw [15]. As shown in Figure 1, these data are represented with the class `dcat:Resource`. In this case, we reuse the DCAT ontology instead of the WoT TD ontology to represent the data, because DCAT is more complete and allows describing a larger volume of data (as used in DTws) with a larger number of features, such as `dcat:Dataset`, `dcat:DataService` or `dcat:Catalog`. This can be useful in the case that there are multiple volumes of data categorised by semantic and non-semantic.

Digital Twin Services Dimension: The following dimension depicted is the dimension of digital twin services. This dimension is represented with the class `dt:DigitalTwinService`, defined as the dimension of the DTw that contains all the services that are used to perform all the processes of the digital twin [15]. As explained earlier, WoT allows us to describe the services of a DTw, this is the reason why we reuse in this dimension the class `td:interactionAffordance` to represent the different existing services in the DTw, under the class of `dt:DigitalTwinService`. Furthermore, this class allows to show the possible choices to the DTw consumers, suggesting how they can interact with the DTw services.

Digital Twin Connection Dimension: The last dimension is the digital twin connections, which are represented by the class `dt:DigitalTwinConnection`. This class is defined as the dimension of the DTw that contains all the connections that exist between the different dimensions of the DTws and the information used [15]. To represent these connections, the class `dt:Connection` has been created and defines the existing connection between different dimensions. The `dt:ConnectionPoint` class is responsible for representing a particular endpoint of a dimension that functions as a provider or consumer of information (represented by `dt:hasProvider` and `dt:hasConsumer`) within the connection. With this approach, there will always be a connection with only two connection points. The different types of `dt:ConnectionPoint` are represented in Figure 4. The authors decided to introduce the class `dt:ConnectionPoint` instead of using the existing SEAS ontology [32]. This decision was made because SEAS only considers connections between systems, whereas in the context of DTws, connections may not always be limited to systems.

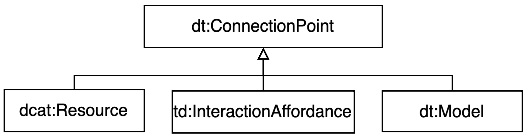

Figure 4: Class `dt:Connection` and relation with `dt:ConnectionPoint`.

In addition, even though the ontology has been developed within the construction field, it is extensible to other domains, as it allows for the inclusion of new subclasses within the various classes of dimensions to address the requirements of a specific domain. Once the WoTDT conceptualisation was defined, it was encoded in OWL using Chowlk [33] and Protégé [2], and stored in a GitHub repository[3].

3.3. WoTDT Evaluation

The evaluation to identify typical errors during ontology implementation has been carried out using the OOPS! tool [34]. Several important and minor pitfalls have been identified. However, these important pitfalls do not affect the consistency, reasoning, or applicability of the ontology. In addition, when it comes to minor pitfalls, they mainly involve identified unconnected elements. The remaining errors identified by the tool were corrected accordingly.

Furthermore, the authors performed a systematic analysis to ensure that the ontology satisfies the requirements elicited in Table 1. To assist in this validation, a set of tests[4] was defined and executed using the Themis tool [35], which implements the testing methodology described in

[2]https://protege.stanford.edu
[3]https://github.com/oeg-upm/WoT-DT-ontology
[4]https://raw.githubusercontent.com/oeg-upm/WoT-DT-ontology/main/testsuite.ttl

[36]. After executing the tests, it was discovered that all of them were successfully completed, indicating that the developed ontology meets all the defined requirements.

3.4. WoTDT publication and maintenance

Once the ontology is developed and evaluated, it has to be published online. To achieve this goal, OnToology [37], a web-based system, is used. OnToology is designed to work seamlessly with Git-based environments and incorporates various pre-existing documentation tools. In addition, OnToology offers two options for publication using content negotiation mechanisms. The first option is to publish the ontology using a permanent id through the services provided by *"https://w3id.org"*. The second option is to download a bundle containing all the necessary files to publish the ontology on a server. For the WoTDT ontology, the first option was selected, publishing it under the URI *"https://w3id.org/def/digitaltwin"*, where the ontology is available in machine-readable format and as a human-readable document.

Finally, to support the maintenance activity in the developed ontology, an issue tracker[5] is used. Hence, for users, domain experts, or ontology developers to suggest new requirements, identify bugs, or provide any suggestions, they are required to create an issue in the repository. This issue tracker enables the monitoring of all the issues suggested, and once an issue is open, the ontology development team needs to discuss and make a decision on whether the proposal presented in the issue should be implemented in the ontology or rejected.

4. Applicability of WoTDT

This section provides an example of an application of the WoTDT ontology in the European H2020 construction-related project COGITO. In particular, the section aims to show a DTw description using this ontology, i.e., data instantiation, for a wall in a pilot COGITO construction site. The instantiation of the data in the given example is obtained from the URI used to publish the different resources used in COGITO (*https://data.cogito.iot.linkeddata.es/resources/<prefix>/<id>*). For instance for accessing the resource `sdt:01U2O` the public url is the following: *https://data.cogito.iot.linkeddata.es/resources/sdt/01U2O*. In Figure 5, an instantiation of a DTw and its respective dimensions is shown. After the DTw has been instantiated, an example of how each dimension can be instantiated has been provided.

First, in Figure 6, an example of the Physical Entity dimension instantiation is presented. In this example, a "Wall" is represented as a `dt:Component`, that also is a `dt:MainComponent` and a `facility:Element`.

Second, in Figure 7, an example of Virtual Entity dimension instantiation is given. In this case, the dimension of the virtual entity has a model that is a `dt:OntologyModel`, found in four different `dt:Format`, containing each of them the specific target URI to get the model.

Third, in Figure 8, an example of Digital Twin Data dimension is shown. In this dimension, the instantiation of the data resources is presented, where a `dcat:Dataset` is representing

[5]https://github.com/oeg-upm/WoT-DT-ontology/issues

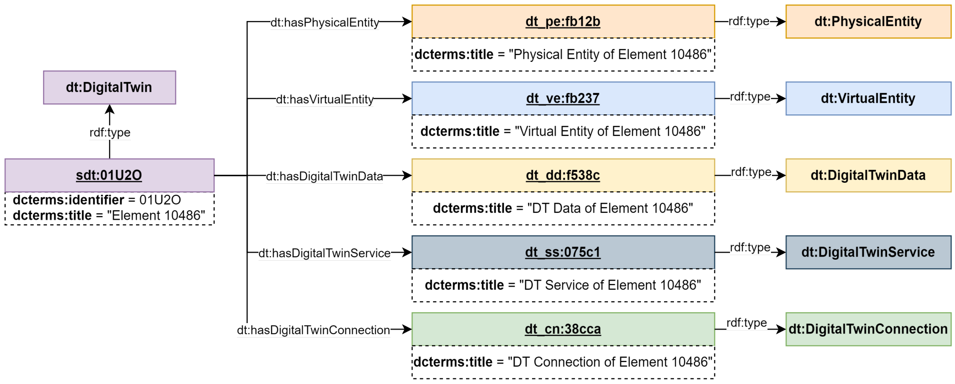

Figure 5: Example of WoTDT DTw from COGITO.

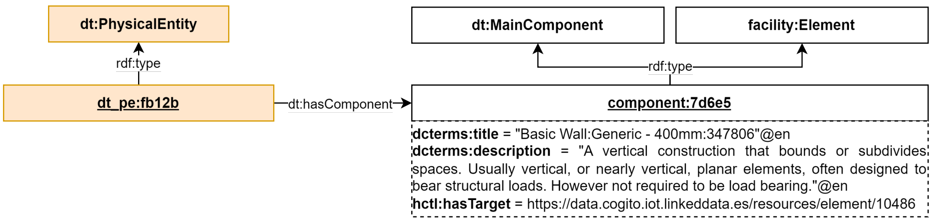

Figure 6: Example of WoTDT Physical Entity dimension from COGITO.

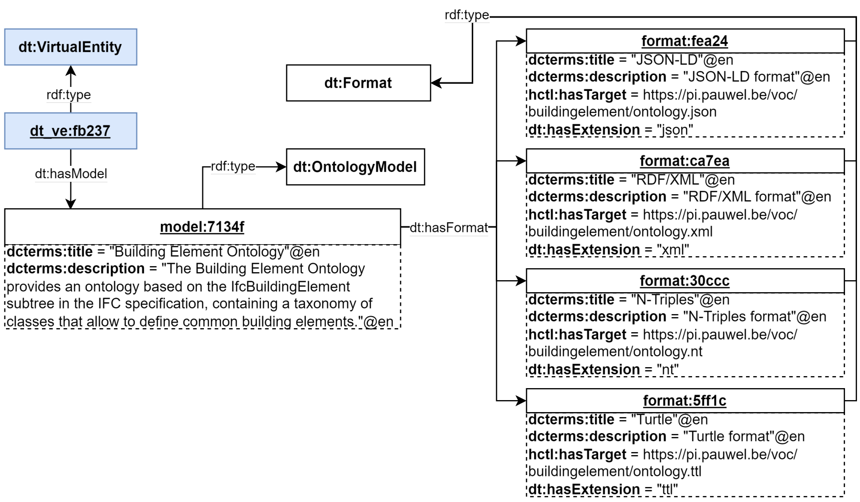

Figure 7: Example of WoTDT Virtual Entity dimension from COGITO.

the *Knowedge Graph* (KG) [38] of the DTw. Furthermore, the representation of access to the `dcat:Distribution` and `dcat:DataService` are provided, in order to retrieve the KG data.

In fourth place, in Figure 9, an example of instantiation of the dimension of the Digital Twin Service is presented. In this particular case, the various services that offer the DTw are

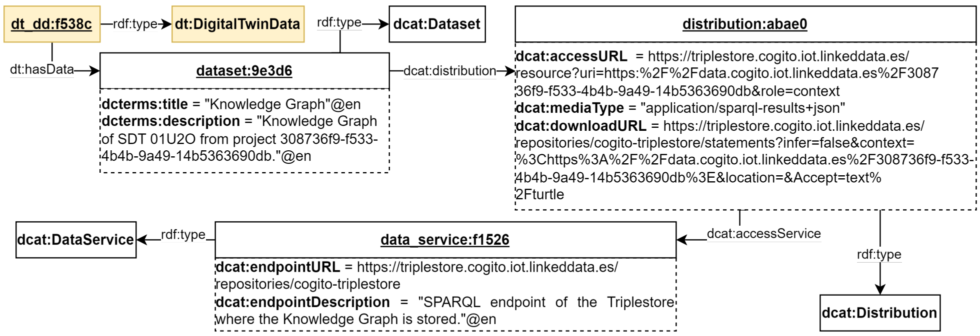

Figure 8: Example of WoTDT DTw Data dimension from COGITO.

depicted as `td:PropertyAffordance` and `td:ActionAffordance`. These services provide their respective `hctl:Form`, which specifies the URL that the consumer needs to access the service.

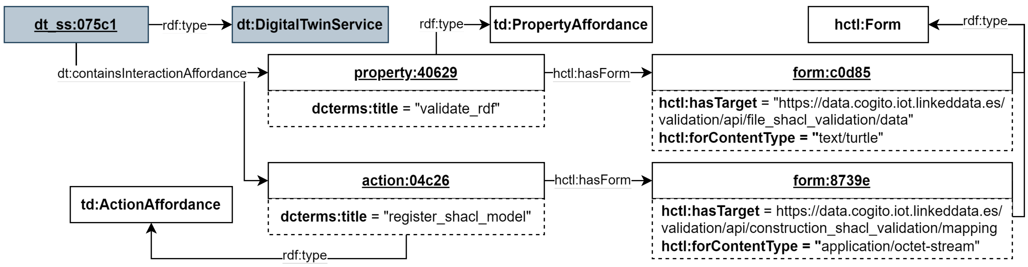

Figure 9: Example of WoTDT DTw Service dimension from COGITO.

Finally, in Figure 10, an example of the dimension of the Digital Twin Connection is shown. In this example, a representation of a `dt:Connection` is presented, where two different dimensions are linked with a specific functionality, specifically, the validation of the *Knowledge Graph* from the DTw Data dimension, using the validation service present in the DTw Service dimension.

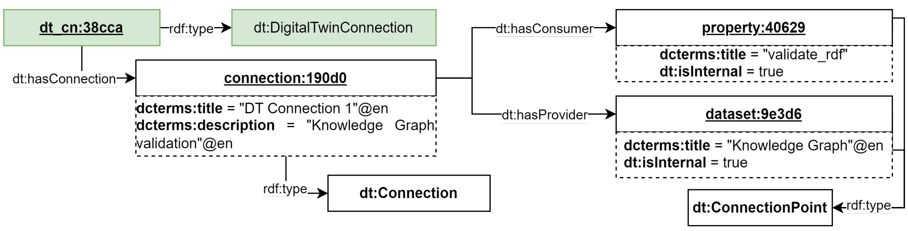

Figure 10: Example of WoTDT DTw Connection dimension from COGITO.

It is important to remark that WoT allows describing data resources using the TD standard [2]. Also, promotes a standardised discovery of these resources by registering their TDs into WoT directories and finding them using syntactic and semantic queries [1]. The expressiveness of TDs allows describing different features of the data resources; how they are accessed, the protocols they use, their security methods, and metadata identifying the real world entity they represent. For instance, a TD could contain metadata describing a building, specifing its HTTP data endpoints and how to invoke them, also, the kind of data they provide (such as columns position, light bulbs status, etc), and a basic auth mechanisms to access.

Due to the fact that WoTDT ontology inherits from TD it also acquires all its benefits, including expressiveness and the discovery capability in WoT directories. This ontology aims at enhancing the expressiveness of TDs with domain specific terms for DTws, an expressivness that TDs lack due to its domain neutrality.

As a running example, Figures 5-10 show a TD extended with the WoTDT ontology. As a result the TD describes the five dimensions of a DTw (domain specific terms) and also describes the data resources that conform the DTw (neutral domain terms). Note that these examples do not show explicitly security schemes but they could be specified. Also, the TD of these examples is compliant with the WoT discovery standard, and therefore, it could be registered and found in a WoT directory.

5. Conclusions

DTws in the construction domain allow for the accuracy of needed predictions to be improved, making more logical decisions, and developing well-informed plans. However, there is no standardised architecture that provides a precise definition of a DTw and offers guidance for its development. In the literature, there are research proposals that model their architecture in a three-dimensional model [14]. Subsequently, a novel proposal presented an extended version that models the DTw in five dimensions [15], providing a more complete understanding of the architectures of the DTws. However, DTws lack formal semantics, which hinders interoperability with third-party DTws or external services.

To address this problem, the WoTDT ontology is developed as an extension of the Thing Description ontology proposed by the W3C Web of Things group, which is a standard that allows the description of services, promoting interoperability, discoverability, accessibility, security and scalability. As a result, this ontology will enable a more precise comprehension of the DTws and will provide direct access to all the system functionalities for the described DTw.

Moreover, despite being initially designed for the construction domain, the ontology is flexible enough to accommodate the incorporation of additional subclasses into classes related to different dimensions. This capability enables the specific requirements of a particular domain to be addressed. However, some issues may arise when adding these new requirements that may trigger changes in the ontology. Therefore, a potential area for future research involves expanding the ontology to encompass a wide range of information from various domains. Furthermore, there might be value in representing the aggregation of an ecosystem of DTws, as it allows for the conceptualisation of the interconnections among various DTws.

Acknowledgments

This work has been supported by COGITO funded by the European Union's Horizon 2020 research and innovation programme under grant agreement no. 958310 and by the Madrid Government (Comunidad de Madrid-Spain) under the Multiannual Agreement with the Universidad Politécnica de Madrid in the Excellence Programme for University Teaching Staff, in the context of the V PRICIT (Regional Programme of Research and Technological Innovation).

References

[1] A. Cimmino, M. McCool, F. Tavakolizadeh, K. Toumura, Web of Things (WoT) Discovery (2023). https://www.w3.org/TR/wot-discovery/.

[2] S. Kaebisch, M. McCool, E. Korkan, Web of Things (WoT) Thing Description 1.1 (2023). https://www.w3.org/TR/wot-thing-description11/.

[3] Z. Mingqiang, H. Yabo, The Application Proposal of Green Supply Chain Management in Construction Industry (2009).

[4] R. Janani, P. Rangarajan, S. Yazhini, A systematic study on site overhead costs in construction industry (2015).

[5] A. Dos Santos, J. Powell, J. Sharp, C. Formoso, Principle of transparency applied in construction (1998).

[6] M. Bilal, L. O. Oyedele, J. Qadir, K. Munir, S. Ajayi, O. O. Akinadé, H. Owolabi, H. Alaka, M. Pasha, Big Data in the construction industry: A review of present status, opportunities, and future trends, Adv. Eng. Informatics 30 (2016) 500–521.

[7] F. Tao, H. Zhang, A. Liu, A. Y. C. Nee, Digital Twin in Industry: State-of-the-Art, IEEE Trans. Ind. Informatics 15 (2019) 2405–2415.

[8] N. B. Ottinger, E. Jordan Stein, M. G. Crandon, A. Jain, Digital twin: the Age of Aquarius in construction and real estate, London: Ernst & Young Global Limited (2021).

[9] B. R. Barricelli, E. Casiraghi, D. Fogli, A Survey on Digital Twin: Definitions, Characteristics, Applications, and Design Implications, IEEE Access 7 (2019) 167653–167671.

[10] M. W. Grieves, PLM–beyond lean manufacturing, Manufacturing Engineering 130 (2003).

[11] K. Främling, J. Holmström, T. Ala-Risku, M. Kärkkäinen, Product agents for handling information about physical objects, Report of Laboratory of information processing science series B, TKO-B 153 (2003).

[12] E. Tuegel, The airframe digital twin: some challenges to realization (2012).

[13] J. Ríos, J. C. Hernández, M. Oliva, F. Mas, Product Avatar as Digital Counterpart of a Physical Individual Product: Literature Review and Implications in an Aircraft 2 (2015) 657–666.

[14] M. Grieves, J. Vickers, Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems, Transdisciplinary perspectives on complex systems: New findings and approaches (2017).

[15] F. Tao, M. Zhang, Digital Twin Shop-Floor: A New Shop-Floor Paradigm Towards Smart Manufacturing, IEEE Access 5 (2017) 20418–20427.

[16] S. Kaebisch, D. Anicic, Thing description as enabler of semantic interoperability on the Web of Things (2016).
[17] M. Poveda-Villalón, A. Fernández-Izquierdo, M. Fernández-López, R. García-Castro, LOT: an industrial oriented ontology engineering framework, Eng. Appl. Artif. Intell. 111 (2022) 104755.
[18] F. Tao, M. Zhang, Y. Liu, A. Y. Nee, Digital twin driven prognostics and health management for complex equipment, Cirp Annals 67 (2018).
[19] C. Boje, A. Guerriero, S. Kubicki, Y. Rezgui, Towards a semantic Construction Digital Twin: Directions for future research, Automation in construction 114 (2020).
[20] S. Mayer, J. Hodges, D. Yu, M. Kritzler, F. Michahelles, An Open Semantic Framework for the Industrial Internet of Things, IEEE Intell. Syst. 32 (2017) 96–101.
[21] C. Steinmetz, A. Rettberg, F. G. C. Ribeiro, G. Schroeder, C. E. Pereira, Internet of Things Ontology for Digital Twin in Cyber Physical Systems (2018) 154–159.
[22] S. Singh, E. Shehab, N. Higgins, K. Fowler, D. Reynolds, J. A. Erkoyuncu, P. Gadd, Data management for developing digital twin ontology model, Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 235 (2021).
[23] M. Bermúdez-Edo, T. Elsaleh, P. M. Barnaghi, K. L. Taylor, IoT-Lite: A Lightweight Semantic Model for the Internet of Things (2016) 90–97.
[24] H. Marah, M. Challenger, An architecture for intelligent agent-based digital twin for cyber-physical systems (2023).
[25] A. Meijers, Hands-On Azure Digital Twins: A practical guide to building distributed IoT solutions (2022).
[26] A. Ricci, A. Croatti, S. Mariani, S. Montagna, M. Picone, Web of Digital Twins, ACM Trans. Internet Techn. 22 (2022) 101:1–101:30.
[27] I. Pittaras, N. Fotiou, C. Karapapas, V. A. Siris, G. C. Polyzos, Secure, Mass Web of Things Actuation Using Smart Contracts-Based Digital Twins (2022) 1–6.
[28] L. Bedogni, F. Chiariotti, A Web of Things Architecture for Digital Twin Creation and Model-Based Reinforcement Control, arXiv preprint arXiv:2301.12761 (2023).
[29] X. Zheng, J. Lu, D. Kiritsis, The emergence of cognitive digital twin: vision, challenges and opportunities, Int. J. Prod. Res. 60 (2022) 7610–7632.
[30] M. C. Suárez-Figueroa, A. Gómez-Pérez, M. Fernández-López, The NeOn Methodology framework: A scenario-based methodology for ontology development, Appl. Ontology 10 (2015) 107–145.
[31] M. Poveda-Villalón, S. Chávez-Feria, S. Carulli-Pérez, R. García-Castro, Towards a UML-based notation for OWL ontologies 3508 (2023) 18–27.
[32] M. Lefrançois, Planned ETSI SAREF extensions based on the W3C&OGC SOSA/SSN-compatible SEAS ontology patterns (2017).
[33] S. Chávez-Feria, R. García-Castro, M. Poveda-Villalón, Chowlk: from UML-Based Ontology Conceptualizations to OWL 13261 (2022) 338–352.
[34] M. Poveda-Villalón, A. Gómez-Pérez, M. C. Suárez-Figueroa, OOPS! (OntOlogy Pitfall Scanner!): An On-line Tool for Ontology Evaluation, Int. J. Semantic Web Inf. Syst. 10 (2014) 7–34.
[35] A. Fernández-Izquierdo, R. García-Castro, Themis: a tool for validating ontologies through requirements (2019) 573–753.

[36] A. Fernández-Izquierdo, R. García-Castro, Requirements Behaviour Analysis for Ontology Testing 11313 (2018) 114–130.
[37] A. Alobaid, D. Garijo, M. Poveda-Villalón, I. Santana-Pérez, A. Fernández-Izquierdo, Ó. Corcho, Automating ontology engineering support activities with OnToology, J. Web Semant. 57 (2019).
[38] A. Hogan, E. Blomqvist, M. Cochez, C. d'Amato, G. D. Melo, C. Gutierrez, S. Kirrane, J. E. L. Gayo, R. Navigli, S. Neumaier, et al., Knowledge graphs, ACM Computing Surveys (CSUR) 54 (2021).