Towards Semantic Web of Things: Reference Architecture and Gap Analysis

Xiang Su
Department of Computer Science
Norwegian University of Science and Technology
Gjøvik, Norway
xiang.su@ntnu.no

Ekaterina Gilman
Center for Ubiquitous Computing
University of Oulu
Oulu, Finland
ekaterina.gilman@oulu.fi

Xiaoli Liu
Department of Computer Science
University of Helsinki
Helsinki, Finland
xiaoli.liu@helsinki.fi

***Abstract*—Internet of Things (IoT) connects objects and allows data exchange between smart things and the Internet. Web of Things (WoT) further integrates smart things into the Web architecture by leveraging existing standardized Web technologies. In this paper, we envision Semantic Web of Things (SWoT), aiming to exchange semantically rich and machine understandable information with smart things. This article discusses the landscape of IoT and WoT through the past and its future prospects, presents a SWoT reference architecture, and conducts a gap analysis. Aiming to highlight the deficiencies of existing solutions, we carry out a gap analysis with respect to (i) extension of WoT, (ii) semantic knowledge base, (iii) semantic reasoning, (iv) semantic interoperability, and (v) semantic service composition. Based on the results of our analysis, we conclude this article with a list of recommendations in order to fill the gaps. The main contributions pertain to a SWoT reference architecture with key SWoT building blocks and an analysis of technologies to support SWoT.**

***Index Terms*—Web of Things, semantics, architecture, gap analysis**

I. INTRODUCTION

Internet of Things (IoT) transforms a set of connected smart objects to intelligent systems with well-designed Application Program Interfaces (API) and user-friendly interfaces. These systems understand and react to physical environments leveraging the connectivity of smart devices and data exchange over the Internet. According to market research, the number of linked IoT devices is expected to reach approximately 75 billion by 2025, paving the way for development of numerous novel IoT applications and services in different domains.

A significant challenge pertains to development of an intelligent and scalable architecture concerning connectivity and interoperability with IoT devices equipped with sensing, networking, and information processing capacities. The heterogeneity of "Things" challenges the collaboration of devices and hinders the widespread adoption of IoT [1]. Compatible communication protocols and data sharing mechanisms are crucial to address this challenge. Assigning unique Internet Protocol (IP) addresses to devices and connecting them to the Internet are the initial steps towards the IoT, allowing seamless data exchange among devices, although not necessarily understand the meaning behind the exchanged data.

Web of Things (WoT) contributes by establishing a common Web service stack based on RESTful architecture and connecting objects and their services to the Web with Web standards and techniques. Particularly, the World Wide Web Consortium (W3C) WoT initiative offers a set of standards aiming at improving the interoperability and usability of the IoT. Through standardized metadata and other re-usable technological building blocks, W3C WoT standards enable easy integration across IoT platforms and application domains. For example, HTTP provides a standardized protocol for retrieving images, text, and other media elements and Constrained Application Protocol (CoAP) is an efficient and lightweight alternative of HTTP, which links constrained devices to IoT. The WoT offers a set of standardized technologies to simplify IoT application development by adhering to the well-known Web paradigm.

Berners-Lee *et al.* [2] state that Semantic Web is an extension of the current web, where information is given well-defined meaning, thereby enhancing collaboration between computers and people. To enable the Semantic Web functionally, computers have to access to structured information and sets of inference rules that they can employ to conduct automated reasoning. For example, WoT allows understanding of the web pages structure and layout, but has no knowledge about their intended meaning. The Semantic Web has a layer of machine-interpretable metadata allowing computers to understand the web page's meaning. This requires a means to describe the concepts and their relations. Two primary languages capable of doing this are eXtensible Markup Language (XML) and the Resource Description Framework (RDF). XML allows users to add arbitrary structure to their documents but doesn't specify the meaning of the structure. RDF provides standard model for data interchange on the Web [2]. Su *et al.* [3] argue that Semantic Web technologies, especially RDF-based data, will become the *de facto* standard for the IoT in representing measurements and activities from physical world. However, there are certain challenges for utilising RDF for IoT, such as ability to validate RDF syntax, limitations on expression, performance issues, and lack of familiarity and the potential high learning curve [4].

In this paper, we introduce Semantic Web of Things (SWoT) as an extension of WoT. Semantic technologies empower WoT with the capability to utilize meaningful data models. The fundamental idea behind SWoT is to employ the same principles that underpin the Semantic Web and linked data, such as accessing semantic data with standard semantic data models (e.g., RDF) and Web protocols. This approach offers several advantages, including 1) enhancing interoperability between isolated systems; 2) enabling advanced data analysis and reasoning technologies; and 3) facilitating WoT application developments through semantic mashups. Many existing information models are closely tied to the underlying communication architecture. However, it is crucial to employ an information model that is independent of both encoding and communication protocols. Semantic modelling serves this purpose and facilitates reasoning by interpreting the meaning of data.

SWoT is an emerging vision combining WoT and Semantic Web technologies. This paper contributes a SWoT reference architecture, providing a foundation for the concrete architectures across diverse domains. SWoT reference architecture focuses on adapting existing standards to enhance interoperability among devices. Such a reference architecture adopts a component-based structure and serves a guideline for the creation of new systems showcasing the relationships between these components [5]. We further conduct a comprehensive gap analysis, outlining the current status, expectations, gaps, problems, and recommendations for development within various categories. Our contributions are twofold. First, a reference architecture is developed considering scalability, extensibility, modularity, and interoperability for heterogeneous IoT devices. Second, a gap analysis is presented where the state of the art is examined and recommendations for future research are proposed.

The remainder of the paper is organised as follows. Section II presents landscape of IoT, WoT, SWoT, and key supporting technologies for SWoT. Section III introduces the reference architecture and Section IV conducts a gap analysis to identify existing gaps in SWoT. Section V concludes the paper with discussing the future research.

II. FROM IOT AND WOT TO SWOT

Similar to Web servers as the global integration platform for distributed applications over the Internet, WoT facilitates the integration of all kinds of devices and applications that interact with them. WoT achieves this by reusing readily available and popular Web protocols and standards, thus enhancing the accessibility of data and services provided by objects to a broader range of applications. While IoT focuses on networking and WoT focuses on application level protocols and models, SWoT presents the evolution of the Web with the Semantic Web technologies. One major focus of SWoT is to provide Internet-scale interoperability that allows the sharing and reusing of connected things.

A. IoT and WoT

IoT focuses on the lower layers of the networking stack. IoT connects uniquely identified physical objects and allows communications with each other over the network, enabling a hyper-connected world for various application domains [6]. The challenge pertains to the heterogeneity of devices in terms of technology standards, data models, and communication protocols, thereby giving rise to integration and interoperability concerns. [7]. One example is the IPSO Application Framework that defines RESTful interfaces for resource definition and management.

WoT counters the fragmentation of the IoT by leveraging and extending standardized Web technologies. Specifically, WoT introduces high-level abstractions for things and their interactions and leverages web standards, approaches, and languages to facilitate interoperability [8], [9]. Generally, the WoT could be comprised of four layers, including: 1) *Access*: the Things turns into Web Things that can be interacted with. Web Things enables interaction with real-world objects leveraging a REST API, such as retrieving data from a temperature sensor; 2) *Discovery*: the Web Things can be discovered and automatically utilized by other WoT applications; 3) *Sharing*: the Web Things can be shared efficiently and securely over the web; and 4) *Composition*: the Web Things can be composed into web applications [10]. This implies that data and services from heterogeneous Things can be integrated into a vast ecosystem of online services and applications. In a nutshell, WoT extends IoT to enable the controlling of physical objects leveraging web standards [11].

B. Context modeling and reasoning

The proliferation of IoT devices and the diversity of their applications across various application domains pose significant challenges in terms of development, integration, and interoperability. One possible solution here is to have a proper formal description of the application domain, including concepts and their relations and dependencies, which is known as context modelling [12]. There exist different requirements for such formal descriptions, e.g. they should address data structure, integrity, and manipulation [13]. Data structure ensures the exchange of information and its storage, validation, modification, and reasoning. Integrity supports validation of structure and actual data of contextual information. Manipulation dictates the operations that can be applied to data structures for reasoning [12], [13]. Also, support for handling incompleteness and ambiguity is crucial, given that physical sensors are error-prone [14]. There exist a number of approaches for context modelling, such as key-value, ontology-based models, and markup scheme models [12]–[14].

With a proper context model in place, we are able to use reasoning – utilizing logical operations on logical statements within a model in order to draw conclusions and derive other statements [15]. Reasoning serves various purposes in IoT scenarios [16]. For example, it could be used for verification and validation, since IoT devices are error-prone and the contextual data they provide can be uncertain or even wrong. Another use is knowledge discovery from implicit context information or low-level context. Finally, reasoning could trigger adaptive actions from the system, based on context changes [12], [16].

There exist a number of reasoning approaches, and their suitability depends on the context model, as the formal representation model dictates the operations that can be applied to it [17], [12]. For instance, ontological reasoning deals with Description logics (DL)-based representations. Such representation facilitates describing the application domain at the conceptual level (its terminology, i.e., TBox). This terminology is then used to name individuals functioning in this domain (e.g., ABox) [18]. The basic TBox operations include determining whether a description is satisfiable (i.e., non-contradictory) and whether one description is more general than another. The main operation for ABox is finding out whether its set of assertions is consistent. In other words, entailment that a particular individual is an instance of a given concept [12], [18]. Moreover, hybrid reasoning schemes are used nowadays, where different approaches are combined, like ontological and rule-based reasoning [17].

C. Semantic Web Technologies

Semantic Web technologies have been noted as essential enablers for IoT as they facilitate proper domain description, as well as reasoning of actionable knowledge from multiple heterogeneous information sources and foster interoperability amongst a variety of applications and systems [19]. Semantic Web provides methodologies and tools to formally represent the knowledge, enable easy manipulation and reasoning. The Semantic Web languages can also be characterised by their expressive power, capability to represent semantics, constructs for knowledge representation, underlying logic [20]. Key Semantic Web technologies include: 1) RDF is a framework which encodes the knowledge in sets of subject, predicate and object triplets [2]; 2) RDF Schema (RDFS) provides a data modelling vocabulary for RDF, allowing describing resources and their relationships [21]; 3) Web Ontology Language (OWL) is a highly expressive and efficient knowledge representation language to model domains [22]. In general, OWL brings more reasoning power to the semantic web, and stating more advanced semantics to RDF statements; and 4) SPARQL is a RDF query language, developed to retrieve and manipulate the data stored in RDF format.

The capability to formalise the application domains presents great opportunities for reasoning. Semantic reasoners play a crucial role in various essential tasks, including classification, consistency checking, and satisfiability checking of concepts [23]. Some examples of reasoners available in Semantic Web technologies landscape include Pellet, SWRL-IQ, and RACER [24]. The ontological approach for context model design allows to use hybrid reasoning, which combines description logic based reasoning with rule-based reasoning (e.g. to infer adaptive functionality) [12]. A number of instruments and initiatives exist to achieve such functionality, such as Apache Jena [25] and Semantic Web Rule Language (SWRL) [26].

Most current research related to applying semantics to the WoT is fragmented. For example, Barnaghi *et al.* presents an early perspective on how semantics can be utilised at different levels in IoT [1]. Su *et al.* evaluated different semantic data formats from energy efficiency perspective, including RDF, N3, SenML, JSON-LD, and Entity Notation [27]. Kiljander *et al.* designed a semantic-based architecture for Pervasive Computing and IoT, with a focus on abstracting the heterogeneity of devices [28]. Finally, Gyrard *et al.* suggested a semantic engine for IoT and discuss their deployment at cloud systems and mobile devices [29]. We notice that most current research focuses on data interoperability at the data content level, i.e. figuring out how to make it easier to use different data sets together (from different sources and using different data models). Moreover, there are limited works addressing the creation of shared vocabularies and ontologies for describing data provenance.

D. Service composition

The development of IoT and WoT applications could be a demanding task, given the diversity of technologies and devices involved. Therefore, often such applications are created through composition. This involves the integration of various software components, Web services, and computational devices to provide required functionality that cannot be achieved by a single isolated component [12].

Generally, application composition requires service description, service discovery, composition, and execution engines [12]. To compose services or things effectively, they should be accurately described in terms of their functionality, interfaces for executing this functionality, and data flow dependencies. Such descriptions could be done by, e.g., Web Services Description Language (WSDL) and OWL-S Semantic Markup for Web Services. These descriptions are often uploaded to the centralized repository to be accessible via the discovery mechanism. Based on the requirements provided by the application developers or users, the composition engine generates the composite application specification. It identifies services satisfying requirements (with the help of discovery mechanism) and employs a composition algorithm to create the execution plan for the services. The execution engine receives the composite application specification and manages the composite service execution [12].

Various approaches exist for application composition, encompassing static and dynamic, automated and manual, and semantic and syntactic composition approaches [30], [31]. Furthermore, different solutions guide the composition engines in selecting services that fulfill the specified requirements [32].

III. A SWoT REFERENCE ARCHITECTURE

This section presents key design considerations and outlines a SWoT reference architecture. The reference architecture serves as a foundation for the concrete architectural design, which should be tailored to fulfill the specific requirements of different IoT domains and applications. During the design process of the SWoT reference architecture, multiple application domains have been considered to collect common patterns that need to be addressed [33].

A. Design considerations

Scalability and interoperability. Scalability refers to the system level capacity to efficiently handle an increasing workload or an increasing number of devices, users, and resources.

It stands as a fundamental characteristic to fulfil the performance requirements as system expands. To enable the scalability, the architecture avoids relying on single point of data processing and storage (or even single processing and storage technology). Another design consideration is interoperability, primarily due to the absence of a common standard for devices, as each system employs its own scheme [34]. To foster scalability and interoperability, a component-based approach is desirable to ensure a common and valuable decomposition of logical clusters [35]. For example, employing semantic descriptions of devices and their functionalities can hide the heterogeneity of real world objects and support interoperability [36].

Diversity of open protocols and data models. Connecting various IoT devices that employ diverse interfaces and protocols poses an arduous challenge. WoT partially mitigates this challenge by utilizing Web protocols to read and write data from/to devices. When all devices offer a Web API, an uniform programming model can be used for all devices. Furthermore, proprietary protocols and data models lead to vendor lock-in and some protocols and some data model standards are not publicly documented and cannot be simply used and implemented without paying a significant annual fee. Additionally, their specifications are not publicly available, which limits their adoption to large industrial organizations. On the contrary, the popularity of Web standards can be attributed, in part, to their complete openness and accessibility without cost. They ensure that data can be easily and rapidly exported from any system, making HTTP and REST a logical choice for providing public access to data.

JSON is widely favoured among IoT devices for its lightweight and simplicity, offering capabilities comparable to XML. Once the resource is encapsulated over a supported content type, it should be properly described in the semantic-enabled Web. For example, Web Linking can construct semantically annotated resource descriptions. SWoT should leverage common protocols (HTTP/CoAP), common methods (such as GET, PUT, POST, and DELETE from HTTP/CoAP), universal identifiers (URIs), and common solutions to annotate resources (e.g., Web Linking). These elements are essential for creating a shared reference (i.e., an ontology). However, there exist use cases where relying solely on Web protocols may not achieve the desired properties, such as the Real-Time Streaming Protocol (RTSP). Semantic data models play a crucial role in ensuring universal understanding of data, such as mapping Sensor Measurement Lists (SenML) to RDF, from minimal semantic descriptions towards elaborated models and general high level ontology (such as W3C SSN ontology). For example, the automation of detecting similar patterns within existing SSN ontologies becomes possible.

Loose coupling. The fundamental principle of loose coupling is to reduce communications both between layers and among subsystems within the same layer, and to balance centralized and local decision making [37]. In the context of IoT devices and applications, loose coupling allows room for ad-hoc, unplanned interactions, and purposing of services into new use-cases, which is an essential requirement in large scale open networks of devices. Web protocols, such as HTTP and CoAP, are loosely coupled in their design because the contract between actors on the Web is both straightforward and well defined. The efforts towards RESTful standlization aim to lower the entry barriers for the new service providers and users, improve the interoperability among different applications/systems, and enable products or services to perform more efficiently at a higher level.

B. A SWoT reference architecture

A distinct separation should exist between general reference architecture and domain specific reference architecture instances. When necessary, additional elements from each concrete WoT and IoT design should be considered and incorporated to the reference architecture. The evolution of the reference architecture sheds light on the design of new concrete IoT subsystems, because the reference architecture already encompasses lessons learned from previous assessments of such IoT subsystems. Functional requirements should only be incorporated into the reference architecture if they are directly linked to fundamental domain functionality.

Figure 1 presents our proposed SWoT reference architecture, designed to tackle the challenge of sharing, reusing, and integrating data between WoT applications and services. The core components of this architecture are scalable at a web-level. The proposed architecture acquires data from various sources, including physical sensors deployed at fixed location, mobile sensors, and even virtual sensors generating virtual data streams, such as social media data streams. IoT devices transmit data to Data Perception and Annotation component using a set of standard communication protocols like CoAP. The Data Perception and Annotation Component processes raw data and adds semantic annotations. Data Storage stores data in a semantic format (e.g. RDF) and provides functions, such as searching, modification, and deletion. Knowledge Storage stores ontology, related semantic information, and the new semantic information generated by reasoners. Knowledge Processing component classifies, stores, and provides discovery mechanism of ontologies. An ontology can be accessed, for example, through Linked Open Vocabularies (LOV), and stored in Knowledge Storage. The Semantic Reasoning component derives implicit knowledge and addresses complex user queries. Semantic Mashup provides functions to create novel services using multiple sources. WoT Apps access semantic services through a RESTful interface to aggregate, recommend, analyze, and visualize IoT data. Finally, establishing an unique identification is essential for Web scale systems. Global name space is needed when Web scale applications access services. A mapping mechanism is employed to establish connections from the local name space to the global name space. We describe details of each component as follows.

- **WoT Application**: WoT applications leverage WoT standards to integrate services from multiple device vendors and ecosystems. As illustrated in Figure 1, WoT applications can access semantic services and deliver

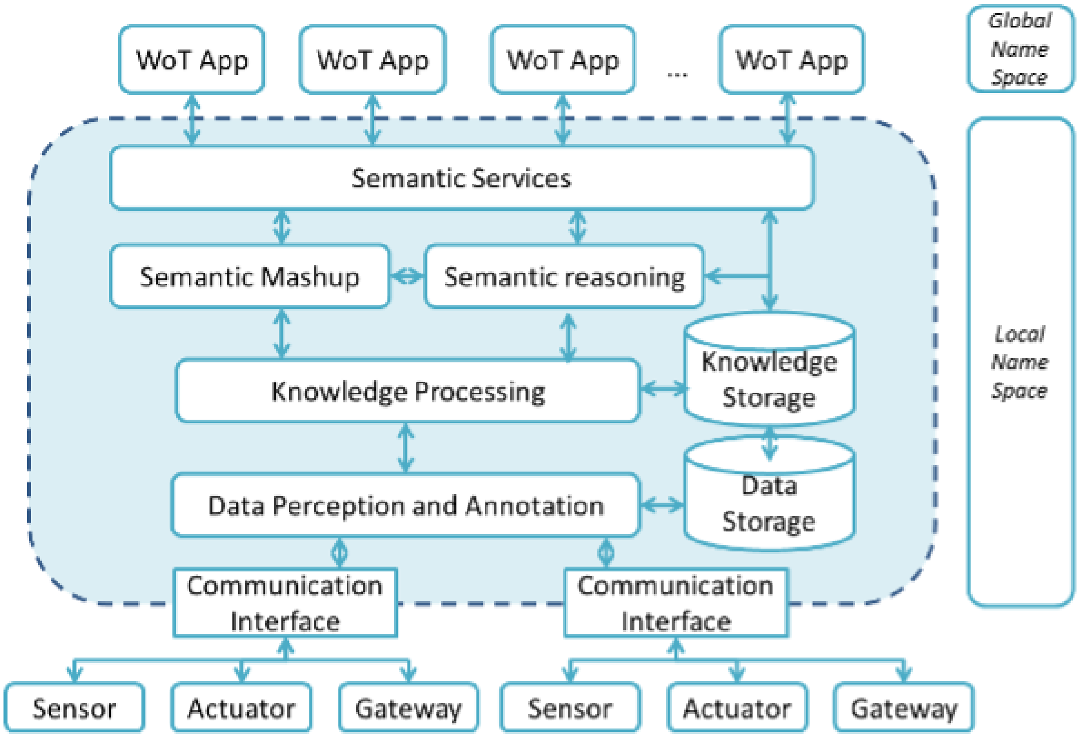

Fig. 1. A SWoT reference architecture.

information to users through well-defined user interfaces. For example, in smart city and smart home scenarios, WoT applications utilize semantic services to facilitate communication among devices, visualize sensor data, and control actuators.

- **Semantic services**: A variety of services can be provided for WoT applications, tailored to the user and system requirements. Semantic services can take the form of RESTful services or service-oriented architecture services. RESTful APIs support applications with publicly accessible Web APIs. RESTful services are built around resources that are linked together by hyperlinks and organized into collections, with retrieval and manipulation provided by a set of methods generally implemented with HTTP [38].
- **Semantic Mashup**: Mashup tools facilitate the development of WoT services with aggregating semantic services. Semantic Mashups are RESTful service-based composites with a semantic layer annotating their APIs for dynamic selection and composition [39]. We envision that SWoT system shall offer the capability of mashing up using semantic descriptions and reasoning methods and the capability of interpreting and applying service logic (e.g. rules of triggering operations upon other resources or attributes according the change of the monitoring resources) described with semantic annotation and ontologies.
- **Semantic Reasoning**: Given an application domain model in form of ontology, e.g., semantic reasoners can deduce new relevant information by relying on logical operations. As it is discussed in Section II, reasoning is useful for verification and validation, discovery of knowledge, and trigger adaptive actions [12], [16].
- **Knowledge Processing**: Knowledge processing is the foundational procedure for extracting knowledge, achieved through the utilization of ontologies that represent and manage knowledge in a structured and formalized manner. The ontologies gather data from the knowledge storage and perform pre-processing. An ontology can be developed by domain experts who are operators that are ruling within the parameters of its respective field. This process includes vocabulary extraction, where data are classified into sub classes, labelled, and assigned specific properties [6].
- **Knowledge Storage**: In collaboration with knowledge processing, the knowledge storage utilizes ontologies to classify data based on its T-box and A-box components. The T-box manages the terminology of the data, which is defined by the ontology's identification of elements within the data. The A-box manages the assertive data, identified by the same or other ontologies. In the context of IoT, the devices are catalogued within the knowledge storage, ensuring that data is efficiently organized for potential future retrieval.
- **Data Storage**: Data storage is responsible for storing and manipulating data from the sensors and actuators while

TABLE I
GAP ANALYSIS FOR SWOT

Category	Current Status	Expectations	Gaps	Recommendations
Extending WoT	Basic knowledge models and ontologies are developed for WoT	Well-structured meaningful data models can be integrated in WoT applications of diverse domains	- Proper WoT knowledge models and ontologies - Integration of relevant WoT related ontologies	- Development of comprehensive and domain-specific WoT ontologies - Alignments with other WoT-related ontologies
Semantic Knowledge Base	Extensive existing knowledge base solutions	Leveraging semantic knowledge bases to support comprehensive functions in WoT applications	Missing links between knowledge base solutions and WoT applications	- Integration of knowledge base with WoT - Improved performance for knowledge base and knowledge processing components
Semantic Reasoning	Standardized semantic reasoning solutions and tools	Integration of semantic reasoning to enable intelligent functions in WoT applications	Absence of semantic reasoning to facilitate intelligent WoT functions	- Allowance of semantic reasoning with diverse expressive power in WoT applications - Lightweight semantic reasoning for resource-constrained devices and systems
Semantic Interoperability	Existing WoT applications have very limited interoperability, because of inadequate adoption of standardized data models and proprietary data models	Massive WoT applications and devices can interoperate with each other and exchange data	- Well-defined standardized data formats and models - Ontologies to enable semantic interoperability	- Adoption of standardized data formats and models - A general ontology model with link domain-specific ontologies
Semantic Service Composition	Simple WoT service composition without or with very limited semantic knowledge	Dynamical service composition to create novel and integrated services guided by semantic descriptions of services	- Well structured semantic descriptions for individual services - Dynamic composition of services to fulfil specific needs from WoT applications - Security and trustworthy concerns in semantic service composition	- Services are annotated with semantic metadata, describing their capabilities in a machine-understandable format - Dynamically combined or orchestrated services based on their semantic descriptions to create new composited services for WoT applications - Addressing security and trustworthy concerns in service composition

securing its integrity and safety. The data storage can be deployed either on a local server or to the cloud.

- **Data Perception and Annotation**: The data perception and annotation component plays a vital role in detecting, labeling, and identifying raw data. Its primary function is to characterize input from sensors. The annotation techniques typically rely on the metadata model for annotating raw sensor observations. The annotated observations facilitate the integration of heterogeneous observations and the interpretation of these observations in a contextually meaningful manner [40].
- **Communication Interface**: Communication interface of IoT devices allows communicating with other objects and components using various communication protocols and patterns. The communication protocols can be categorized into low power wide area network (LWPAN) and short range network. LWPAN covers cellular and wireless connection with a range up to 50 kilometres [41]. Short range protocols, which are cost-effective, have limited bandwidth and include technologies, such as RFID-chips, NFC, and Bluetooth [41]. There exist four types of communication patterns, including telemetry, which works as a single data flow from a device to another system; inquiries, which are requests from devices looking to gather information; commands, which are sent from a system to a device to perform or execute a specific activity; and notifications, which include information flows in one direction from other systems to a device [42].
- **Sensors and actuators**: Sensors are physical devices that convert environmental information into a signal. Actuators perform an action or control physical entities in the real world based on signal retrieved from the system [36]. Additionally, one can consider "virtual" sensors which are software components, providing various kinds of information, such as social media data.
- **Gateway**: A gateway functions as the connection point between the sensors and actuators. Gateways vary from simple devices that merely forward messages to more complex systems that can perform protocol translation, encryption, and implement additional security measures before transmitting or receiving data.

IV. GAP ANALYSIS

In this section, we present a gap analysis, summarized in Table I, with the objective of evaluating the maturity of the current solutions by assessing their shortcomings from several perspectives. These perspectives encompassed in the analysis include (i) extension of WoT, (ii) semantic knowledge base, (iii) semantic reasoning, (iv) semantic interoperability, and (v) semantic service composition. The goal of the gap analysis is to identify the shortcomings of today's solutions in order to enable future SWoT. The current state of the art is assessed, and recommendations for further research are made. The evaluation is carried out to map the WoT landscape as it stands today in **Current Status**. Furthermore, the analysis focuses on what to be expected in each category in **Expectations**. Gaps of the current state and expectations are highlighted in **Gaps** and recommendations and action points are highlighted in **Recommendations**.

In the first category **Extending WoT**, the analysis promotes constructing comprehensive and domain-specific WoT ontologies and for alignments with other WoT-related ontologies. In **Semantic Knowledge Base**, the analysis promotes integration of knowledge base with WoT with improved performance. In the category of **Semantic Reasoning**, the analysis promotes semantic reasoning in SWoT with diverse expressive power, especially lightweight semantic reasoning for resource-constrained devices and systems. In the category of **Semantic Interoperability**, the analysis promotes adoption of standardized data formats and models and development of general ontology models to link domain-specific ontologies. In the category of **Semantic Service Composition**, the analysis promotes dynamical service composition to create novel and integrated services guided by semantic descriptions of services and security and trustworthy concerns in service composition.

V. DISCUSSION

IoT has facilitated the worldwide integration of data originated from various sources. However, a significant challenge within the realm of IoT is how to establish an intelligent and scalable infrastructure providing connectivity and interoperability on a web-scale. To address this challenge, WoT extends IoT by suggesting an interoperable infrastructure providing connectivity and interoperability. WoT contributes with a common Web service stack based on RESTful architecture and connects objects and their services to the Web.

This paper introduces SWoT, an extension of WoT, leveraging semantic technologies to enhance WoT with the capacity to employ meaningful data models. Our fundamental idea aligns with the principles upon which the Semantic Web and linked data are constructed, emphasizing the use of standard semantic data models with Web protocols. In this paper, we offer a comprehensive overview of existing SWoT components and propose a SWoT reference architecture, designed to address vital considerations, such as scalability, extensibility, modularity, and interoperability for a diverse range of devices and services. Finally, this paper includes a gap analysis, examining main components of SWoT.

Semantic technologies enhance WoT's ability to effectively utilize and understand data. In the context of SWoT, semantic technologies play a crucial role in facilitating the extraction of actionable knowledge from various heterogeneous information sources, ultimately promoting interoperability across a diverse range of applications. Interoperability is a major challenge within SWoT, particularly due to the complexity of connecting and managing a multitude of devices from different vendors. The utilization of ontologies can foster homogeneity among systems and introduce self-management capabilities, thereby increase system efficiency.

As future work, we will explore methods to bridge the identified gaps. While there are plausible solutions to each of the gaps, which require in-depth exploration and practical considerations. This involves, for example, a comprehensive examination of existing ontologies as well as the development of new ones when necessary. We will also investigate surrogate solutions for aspects of SWoT where ontologies may face limitations. Furthermore, various approaches to the knowledge base will be thoroughly examined.

ACKNOWLEDGEMENTS

This research is partially supported by Nordforsk Nordic University Cooperation on Edge Intelligence (168043) and Academy of Finland (323630).

REFERENCES

[1] P. Barnaghi, W. Wang, C. Henson, and K. Taylor, "Semantics for the internet of things: early progress and back to the future," *International Journal on Semantic Web and Information Systems (IJSWIS)*, vol. 8, no. 1, pp. 1–21, 2012.

[2] T. BERNERS-LEE, J. HENDLER, and O. LASSILA, "The semantic web," *Scientific American*, vol. 284, no. 5, pp. 34–43, 2001. [Online]. Available: http://www.jstor.org/stable/26059207

[3] X. Su, J. Riekki, J. K. Nurminen, J. Nieminen, and M. Koskimies, "Adding semantics to internet of things," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 8, pp. 1844–1860, 2015.

[4] D. Reynolds, C. Thompson, J. Mukerji, and D. Coleman, "An assessment of rdf/owl modelling," *Digital Media Systems Laboratory, HP Laboratories Bristol*, vol. 28, 2005.

[5] G. Muller, "A reference architecture primer," *Eindhoven Univ. of Techn., Eindhoven, White paper*, 2008.

[6] M. Noura, A. Gyrard, S. Heil, and M. Gaedke, "Automatic knowledge extraction to build semantic web of things applications," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8447–8454, 2019.

[7] H. Abdelhalim and K. Halimi, "An integration of health monitoring system in public transport using the semantic web of things," *JUCS - Journal of Universal Computer Science*, vol. 27, pp. 1325–1346, 12 2021.

[8] M. Mrissa, L. Médini, J.-P. Jamont, N. Le Sommer, and J. Laplace, "An avatar architecture for the web of things," *IEEE Internet Computing*, vol. 19, no. 2, pp. 30–38, 2015.

[9] A. J. Jara, A. C. Olivieri, Y. Bocchi, M. Jung, W. Kastner, and A. F. Gómez-Skarmeta, "Semantic web of things: an analysis of the application semantics for the iot moving towards the iot convergence," *Int. J. Web Grid Serv.*, vol. 10, pp. 244–272, 2014.

[10] D. D. Guinard, "Building the web of things : with examples in node.js and raspberry pi," Shelter Island, 2016.

[11] I. Nadim, Y. Ghayam, and A. Sadiq, "Towards a semantic web of things framework," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, p. 443, 12 2019.

[12] E. Gilman, *Exploring the use of rule-based reasoning in ubiquitous computing applications*. Doctoral dissertation, University of Oulu, 2015.

[13] E. J. Y. Wei and A. T. S. Chan, "Towards context-awareness in ubiquitous computing," in *Embedded and Ubiquitous Computing*, T.-W. Kuo, E. Sha, M. Guo, L. T. Yang, and Z. Shao, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 706–717.

[14] T. Strang and C. Linnhoff-Popien, "A context modeling survey," in *First International Workshop on Advanced Context Modelling, Reasoning And Management at UbiComp 2004, Nottingham, England, September 7, 2004*, September 2004, lIDO-Berichtsjahr=2004, Potential plagiarism at http://dl.comsoc.org/cocoon/comsoc/servlets/GetPublication?id=9014634. [Online]. Available: https://elib.dlr.de/7444/

[15] S. Poslad, *Ubiquitous computing: smart devices, environments and interactions*. John Wiley Sons Ltd, 2009.

[16] J. Viterbo, *Decentralized Reasoning in Ambient Intelligence*. Doctoral dissertation, Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), 2009.

[17] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni, "A survey of context modelling and reasoning techniques," *Pervasive and Mobile Computing*, vol. 6, no. 2, pp. 161–180, 2010, context Modelling, Reasoning and Management. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1574119209000510

[18] D. L. M. D. N. P. F. P.-S. Franz Baader, Diego Calvanese, Ed., *The Description Logic Handbook. Theory, Implementation and Applications*. Cambridge University Press, 2003.

[19] A. I. Maarala, X. Su, and J. Riekki, "Semantic reasoning for context-aware internet of things applications," *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 461–473, 2017.

[20] A. Malki, M. Barhamgi, S. M. Benslimane, D. Benslimane, and M. Malki, "Composing data services with uncertain semantics," *IEEE Transactions on Knowledge and Data Engineering*, vol. pp, p. 1, 09 2014.

[21] D. Brickley, R. V. Guha, and A. Layman, "Resource description framework (rdf) schema specification," 1999.

[22] K. Sengupta and P. Hitzler, "Web ontology language (owl)," *Encyclopedia of Social Network Analysis and Mining*, 2014.

[23] N. Dalwadi, B. Nagar, and A. Makwana, "Performance evaluation of semantic reasoners," in *Proceedings of the 19th International Conference on Management of Data*, 2013, pp. 109–112.

[24] A. Khamparia and B. Pandey, "Comprehensive analysis of semantic web reasoners and tools: a survey," *Education and Information Technologies*, vol. 22, no. 6, pp. 3121–3145, 2017.

[25] Apache jena. [Online]. Available: https://jena.apache.org/

[26] M. O'connor, H. Knublauch, S. Tu, and M. Musen, "Writing rules for the semantic web using swrl and jess," *Protégé With Rules WS, Madrid*, 2005.

[27] X. Su, J. Riekki, J. K. Nurminen, J. Nieminen, and M. Koskimies, "Adding semantics to internet of things," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 8, pp. 1844–1860, 2015. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/cpe.3203

[28] L. Roffia, F. Morandi, J. Kiljander, A. D'Elia, F. Vergari, F. Viola, L. Bononi, and T. Salmon Cinotti, "A semantic publish-subscribe architecture for the internet of things," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1274–1296, 2016.

[29] A. Gyrard, K. Tabeau, L. Fiorini, A. Kung, E. Senges, M. De Mul, F. Giuliani, D. Lefebvre, H. Hoshino, I. Fabbricotti *et al.*, "Knowledge engineering framework for iot robotics applied to smart healthcare and emotional well-being," *International Journal of Social Robotics*, pp. 1–28, 2021.

[30] S. Dustdar and W. Schreiner, "A survey on web services composition," *International Journal of Web and Grid Services*, vol. 1, no. 1, pp. 1–30, 2005. [Online]. Available: https://www.inderscienceonline.com/doi/abs/10.1504/IJWGS.2005.007545

[31] O. Davidyuk, I. S. Milara, E. Gilman, and J. Riekki, "An overview of interactive application composition approaches," *Open Computer Science*, vol. 5, no. 1, pp. 79–95, 2015. [Online]. Available: https://doi.org/10.1515/comp-2015-0007

[32] J. Rao and X. Su, "A survey of automated web service composition methods," in *Semantic Web Services and Web Process Composition*, J. Cardoso and A. Sheth, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 43–54.

[33] M. Kovatsch, K. Kajimoto, K. Toumura, T. Kawaguchi, M. Lagally, and R. Matsukura, "Web of things (wot) architecture," Apr 2020. [Online]. Available: https://www.w3.org/TR/wot-architecture/

[34] I. Mashal, O. Alsaryrah, T.-Y. Chung, C.-Z. Yang, W.-H. Kuo, and D. P. Agrawal, "Choices for interaction with things on internet and underlying issues," *Ad Hoc Networks*, vol. 28, pp. 68–90, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1570870514003138

[35] L. Heuser, J. Scheer, P. Den Hamer, and B. De Lathouwer, "Reference architecture & design principles," *EIP-SCC Work Stream*, vol. 2, 2017.

[36] S. N. A. U. Nambi, C. Sarkar, R. V. Prasad, and A. Rahim, "A unified semantic knowledge base for iot," in *2014 IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 575–580.

[37] A. Mämmelä, J. Riekki, and M. Kiviranta, "Loose coupling: An invisible thread in the history of technology," *IEEE Access*, vol. 11, pp. 59 456–59 482, 2023.

[38] M. Özpınar, "A flexible semantic service composition framework for pervasive computing environments," 2014.

[39] R. Ramanathan, "Service-driven computing: Challenges and trends," in *Information Retrieval and Management: Concepts, Methodologies, Tools, and Applications*. IGI Global, 2018, pp. 2287–2311.

[40] G. Zhang, L. Kou, L. Zhang, C. Liu, Q. Da, and J. Sun, "A new digital watermarking method for data integrity protection in the perception layer of iot," *Security and Communication Networks*, vol. 2017, 2017.

[41] S. Al-Sarawi, M. Anbar, K. Alieyan, and M. Alzubaidi, "Internet of things (iot) communication protocols: Review," in *2017 8th International Conference on Information Technology (ICIT)*, 2017, pp. 685–690.

[42] P. Patierno, "Strengths and weaknesses of iot communication patterns - dzone iot," May 2017. [Online]. Available: