DESIGN AND DEVELOPMENT OF A COMPACT ONTOLOGY FOR SEMANTIC MODELING OF A SMART HOSPITAL

K. Parveen[1], A. A. Malik[2], W. Azeem[3], M. Asad[4] and H.R. Kayani[5]

[1]Department of Computer Science, University of Engineering and Technology (UET), Lahore
[2,3,4,]Faculty of Computer Science, Lahore Garrison University (LGU), Lahore
[4]School of Electronics Information and Electrical Engineering,
Shanghai Jiao Tong University Shanghai China
[2]Email of Corresponding author: dr_afatb_malik@lgu.edu.pk

ABSTRACT: The IoT enables appliances globally to communicate with each other and produce a large quantity of information that can realize valuable information in different service domains. We will develop inter-connected semantic structures to promote semantic interoperability by combining the IoT device's data with semantic technologies. In research paper, we suggest a semantic engine platform (SEP) to promote ontological frameworks in different IoT-based service areas of the smart city. In specific, the main discussion is on three key concerns related to the delivery of advanced semantic infrastructure in IoT-related applications: semantic exploration, interactive semantic representation, and semantic data repository. The feasibility of this SEP is shown by developing a smart hospital prototype application, giving a preset hospital environment in which user's data is interpreted with the help of smart mobiles. The smart city scenario is explored in this paper with the help of SEP solutions based on services across different domains using IoT capabilities. In this way, our approach will ultimately lead to a smart city with more streamlined, robust facilities with help of semantic interoperability.

Keywords: Internet of things; ontology; knowledge representation; wireless communication.

(Received 16.03.2021 Accepted 23.05.2021)

INTRODUCTION

Today, IoT is a dominant technology field, spanning almost all electronic communication areas of life (Parveen *et al*. 2021). Like in smart cities (Zanella *et al*. 2014), (Mehmood *et al*. 2017), e-health(Islam *et al*. 2015), smart environment (Trasviña-Moreno, *et al.* 2017), smart home (Majumder *et al.* 2017),industry 4.0(Chen *et al.* 2017), etc. Similarly, worldwide items can communicate their knowledge against the alteration in any state in the environment, and in this way, they are sensitive to external stimuli and smart too. They can identify changes in service domains to produce effective usage (Gubbi *et al*. 2013).

In general, IoT may be a gateway to accessing a smart city's capacity for delivering integrated services throughout disciplines. In smart city current vertical frameworks of services, the invention and exploitation of resources along all service domains in a city has been difficult for services from every domain as shown in Figure 1. Therefore, the IoT smart city architectures of service grow from current vertical to horizontal service architectures. In the smart city, the main purpose of interoperability is to allow different service domains to share the IoT resources among them and to provide a smart service system to the end-users. Traffic accident information obtained from a transport area. For example, may be used to check the empty beds in the hospitals and sent an emergency call to the doctors. That is why implementing horizontal structures in a smart city would result in new services being developed and current facilities enhanced through interoperability across service domains.

IoT paradigm and semantics are *e*xplained for different services domain data interoperability via semantic data in (Ferretti et.al.2016).These different domain services can interpret as well as represent knowledge by performing semantic interoperability. The Semantic interoperability can be achieved in different domains with semantic technologies (Compton *et al*. 2012). However, a service domain must perceive and recognize the given service domain knowledge against some query. That is why an integrated solution information system is required to share store and manage the knowledge with the help of semantic technology among services domain.

IoT has given rise to a major interest in incorporating knowledge and skills generated by objective world artifacts into the global internet. As among the main components of the emerging Web, IoT has drawn significant attention across multiple research institutions and businesses. In the last several *y*ears, the range of design and technology has been greatly expanded, from the first emphasis on tracking and usability, utilizing IoT technology and software Rfid system, constraint system communication standards, (mobile) controllers and sensor systems, middleware, information protection, and plenty of others. Between these advances, semantic-oriented technology shows the potential to address the complex challenges of heterogeneity and interoperability because a significant number of devices have distinctive aspects.

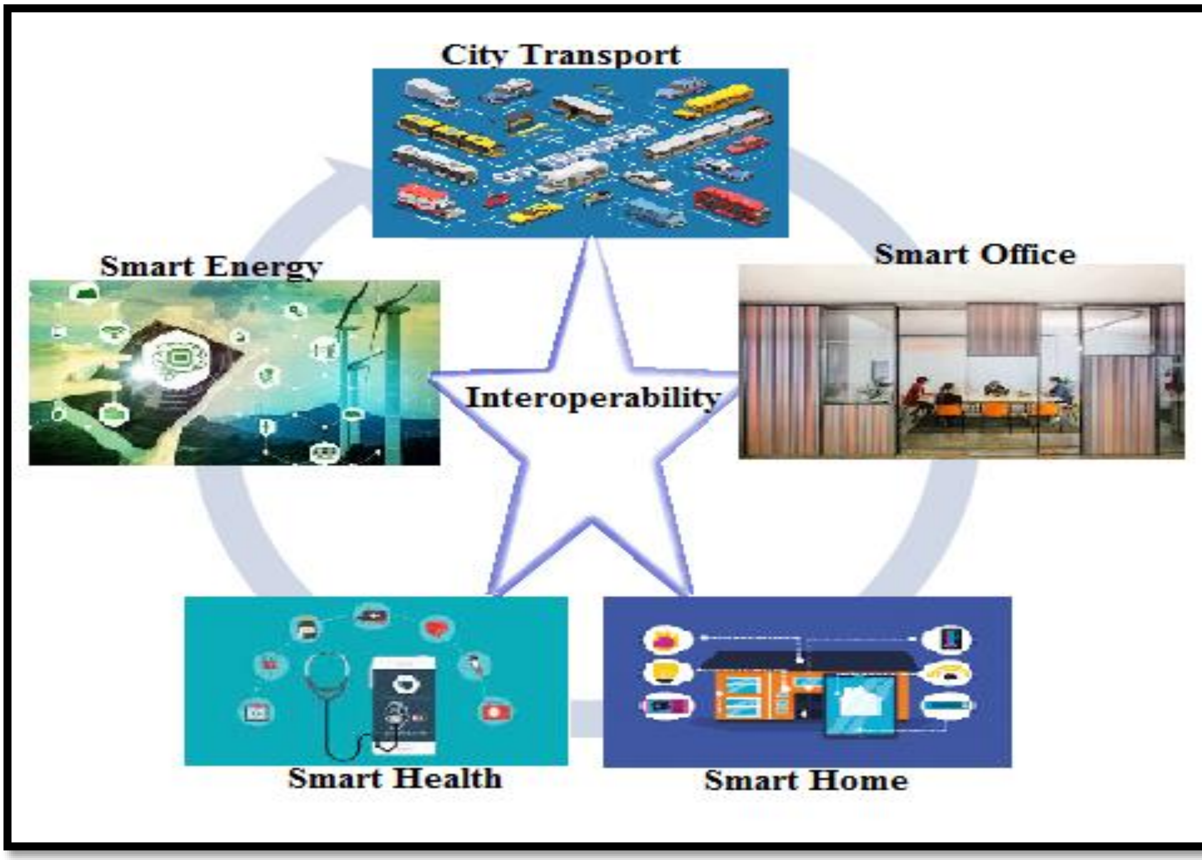

Figure1. Smart city-specific service domains Interoperability

We construct a SEP to address interoperability issues by managing and storing different service domain knowledge through the ontology of smart cities. It incorporates the various services with the help of ontology models. To create ontology IoT Amalgamation Ontology (IAO) is built to aid each service of the domain. It will serve as a highly advanced ontology for storing and managing different service domain ontologies, which are generated with the help of a tool known asa web constructor. We have also added an IoT system for observing and controlling capacities dependent on revelation by working together with external IoT administration frameworks.

SEP and IAO viability is checked by creating a prototype service for a hospital domain. With this project, we design an ontological workplace model, where users enter the text to create an application of web, which enables users to ask any query. We also establish monitoring and control mechanisms for IoT devices to conduct health services through the use of IoT healthcare tools. Mobius is used for monitoring and control functions of prototype service. It is an IoT-based application and used for the management as well as storage of data gathered from different IoT devices.

This paper is organized into three parts: initially, the development of a thorough and compact ontology for the modeling of fundamental principles in IoT systems. To understand the importance of interactive learning and querying information in IoT, a health-based ontology is also established. It is demonstrated that ontology's can also be utilized collectively to highlight important IoT-linked data with semantic captions for knowledge representation. Whereas the semantic realization of notions provides a common platform for automated system interpretation, the magnitude of the IoT and the variety of issues poses new opportunities for storage of metadata storage and retrieval of information from various resources. Our second contribution covers the distributed storage systems creation to expand the framework for sake of semantically annotated IoT skills and facilities. The third contribution adds an illustration of ontology effectiveness explanation using IoT service discovery method via the advancement of much IoT service discovery method. In general, semantic services can be linked for IoT and knowledge representation by ontology formulation.

The remainder of the article is organized into seven sections. Section 2 analyzes several conceptual semantic models applicable in service discovery methods and domain modeling of IoT. In Section 3 design and architecture for IoT ontology from design theory and knowledge discovery purpose are highlighted. In Section 4, we describe a hospital-based ontology and demonstrate usage with a description ontology to generate associated data of IoT. Section 5 highlights our solution for the storage of linked semantic data according to the ontology. Section 6 demonstrates that the online discovery method can be built, using ontology and IoT data. Section 7 sums up the research paper and future studies issues are highlighted.

LITERATURE REVIEW

This section covers state of art on the latest projects related to the IoT domain modeling and analysis of IoT exploration. This section discusses current work using semantic web technologies, including ontologies and linked data, to solve the problems of complex inter-relationships capture, data integration and reasoning, and interoperability of data in a sensor network.

Ontologies have played an essential part in overcoming a sensor network's existing issues. The research of Avancha *et al*. (Avancha *et al.* 2004), for example, suggested an ontology for sensor networks to gather essential sensor node characteristics, both for usability and present state. W3C has motivated the attempts to standardize sensor classification in allied areas and modeling of observational data in XML-based standards suite of Open Geospatial Consortium (OGC), Sensor Web Enablement (SWE), and Semantic Sensor Networks (SSN) (Compton *et al.* 2012). A basic level schema involves for SSN ontology to describe sensor tools, specifications, capabilities, as well as other relevant features for web services in wireless semantic communications and application. The SSN ontology doesn't include the modeling factors for interesting elements, measurements, and domain expertise which need to be affiliated with sensor research to enable independent information transmission, effective logic as well as aid in making decisions.

In (Henson *et al*. 2009) study offers monitoring by services activated semantically, named SemSOS, that makes it feasible for querying both high-level environmental information and raw sensor data at a low level. The 52North development aims to give sensors with data features for storing in a database in XML files. The research in (Kim *et al*. 2008) addressed ontological structure services, based on sensor data and networks. There are three essentials elements of ontology for the conceptual and physical characteristics of WSN architecture sensor nodes. However, the specification has not explained how sensor data in an application with a sensor network described and interpreted. In (Tao *et al*. 2017) the IoT-A project defined resources, IoT services, and entities as core concepts for IoT-domain. Here entity acts as the focal point of human and software agent interactions. In this project, IoT service reveals the functionality of resources that are hosted on devices that allow the organization to access them physically. The semantic models (De. *et al.* 2012) consist of entities *modeled* by associating attributes with the domain. The resource model defines the different types of resources location of the host system and the connection to resource display service model. It also exposes precondition, input, resource functionalities and output. The service form defines the specific technologies that also aid to raise the service like RESTful, OWL-S3, etc. The Table 1 shows the comparison among reviewed papers from interoperability, knowledge representation, and semantic modeling parameters. Here *shows partially, ✓ shows completely fulfilled, and ✗ not completely fulfilled. According to this table, there is a need for data interoperability based on domains.

Ontology for IoT-based application integration: Data interoperability for different domains can be achieved with the ontology (Avancha et.al.2004) .The IoT-based Health Amalgamation Ontology (IAO) is defined in this section. The IAO is a high-level ontology for storing and handling the ontology from each field of service to communicate the knowledge. To understand the ontology, the entire knowledge is divided into three parts of IoT-based services, services, testimonials, and consumer. As shown in Fig 2, the hierarchy graph support ontology, built from different IoT-based application domains. In IAO terms service in a smart city means that there are several services. It has further sub-classes distinguished with the help of class Topic. This Topic class is developed by either by developer or administrator in the service domains with the help of the web constructor tool, which is explained in section 4.1.1. “Thing” have class “services” have a relation with other classes which have object properties “has referenced”, “has repositories” and “has the method”. The “consumer” class contains information about the end-users like user name, user ID in the ontology. Web Constructor tool is used to describe classes.

Table 1. Comparison table of various technologies.

Reference #	Application	Coverage: sensors	Coverage: observation	Coverage: Domain	Data Interoperability	Knowledge Representation	Semantic Modeling
Avancha et.al.2004	Adaptive Sensor Networks	*	✓	*	×	×	×
Compton et.al.2012	Sensors and Observations	✓	✓	*	✓	✓	✓
Henson et.al.2009	Semantic Sensor Observation	✓	✓	*	×	✓	×
Kim et.al.2008	Service-Oriented Ontology for Wireless Sensor Networks	✓	✓		×	✓	×
De.et.al.2012	Real World and digital objects	✓	✓	*	✓	✓	✓

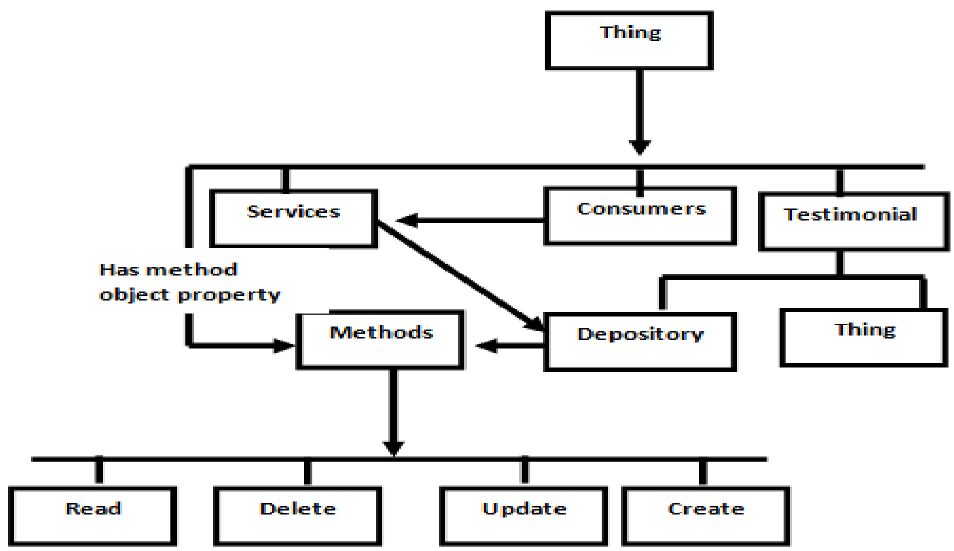

Figure 2. IoT-based Health Integration Ontology (IAO)

Framework for SEP: The SEP gives an integrated semantic application by storing and operating the IAO, with external IoT application frameworks. The SEP gives web service with the help of different smart gadgets, including android mobiles and computers. The SEP description is shown in Fig 3. In Fig. 3, the SEP components are web constructor and semantic engine. First, developers and administrators use services domain knowledge ontology with the help of a web constructor.

1. This tool generates an ontology that fits the values input and afterward transmits that ontology to the SE for inclusion in IAO.
2. The SEP applies semantic methods to explain ontology and supplements these by IAO. This method helps in the coordination of given data with other IoT services that are not on the platforms and individuals of class URLs and of the system. Here SE will provide interoperability between hospital facilities according to end-user requests through these procedures. Finally, as shown in Figure 3, end-users here ask for services with their applications, for instance, observing and management of IoT; Two key software packages consist of the SEP: the web constructor and SE as illustrated in Figure 4.

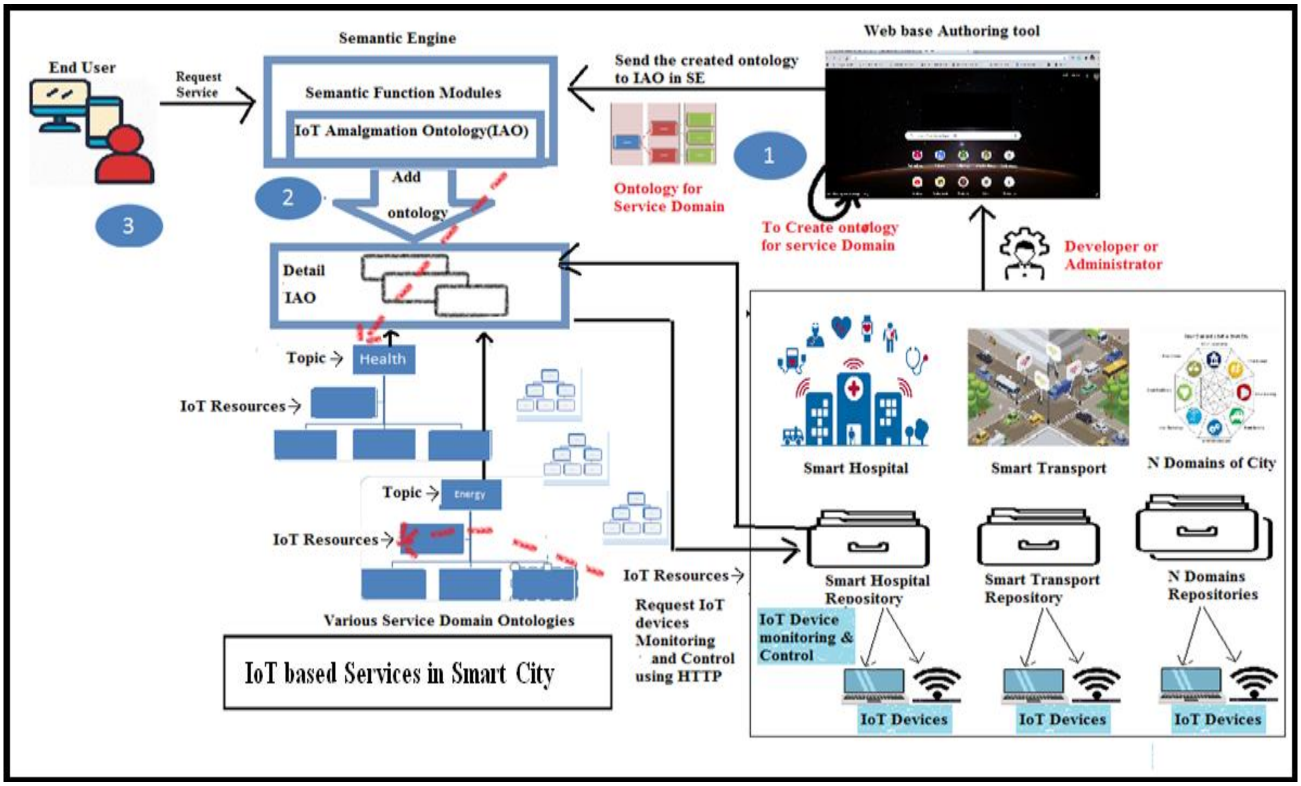

Figure 3. The overview of the Semantic Engine Platform

Web-based platform: The Web-browser is used by web constructor to build ontology for adding services in the hospital without any ontological creation tool. The web constructor provides four key input fields: (1) Title of domain service(2) ontological scheme and relation; (3) Related URL resources; and(4)Rule language of the semantic web known as semantic web rule language (SWRL).Title of domain service is helping developers or administrators to construct the subject class in each service domain. Further, it is to be converted into a high class of ontology that represents special service domain information generated with the values of the other input field. This area can be used to call the subject class. The ontology schema is used to enter the class name, object and data properties, context, limitation, and range, therefore, an ontology is created that represents health domain awareness.

Semantic Engine (SE) Architecture: The SE contains all the semantic roles in the SEP and its core components are five. Figure 4 shows the SE architecture having components the semantic translator, ontology adder, semantic Explorer, Service collaborator, and IAO Manager. Here semantic translator is used to translating data semantically to represent values obtained as semantic data from both the web constructor tool as well as from the external IoT service network.

The values are entered as Web ontology language (OWL) from web authoring tool with the help of semantic translation based on the W3C supported RDF schema and OWL syntax. An IoT service has its specific resource framework and application specifications for each service domain based on standards like ETSI M2M or oneM2M. The semantic translator renders RDFs significance in domain-specific forms. In the end, the textual translator sends the RDFs to the adder of the ontology. The translator adder and ontology adder applies the OWLs and RDFs from the semantic translator to the IAO. Here ontology adder performs OWL registration and RDF registration.

The OWL registration introduces to the IAO the structure of ontology to connect people or IoT resources with the IAO. The IAO updates files using registration OWL to its specification, using the IAO manager. The RDF registration requires people (i.e. RDFs) introduced into the IAO by the service applied to the IAO as an ontology scheme. To differentiate service domain ontology from ontology in the IAO, the RDF registration interprets the explicit meaning of the Ontology Scheme in the IAO. Until all RDFs have been added to the IAO, the RDF registration will then be added to the IAO in the same manner as the OWL and repeated. The Pellet reasoner is used by the ontology Adder to define a particular ontology scheme directly and the IAO so that the ontology adder can apply OWL and RDFs to the IAO.

The semanticized explorer makes semantic discoveries with SPARQL. The SPARQL is used to test semanticized discoveries in the IAO. The semanticized explorer makes semanticized discoveries in Ontology for a particular service domain and a few ontology's in the IAO by SPARQL. The semanticized explorer will consider people in connection with the requested facilities as well as their open APIs. The semantic explorer uses the Pellet reasoner as the ontology adder to discover the individuals.

Now, the Pellet reasoner is used for the analysis of the relationships among individuals as well as SWR's identified by the Web authoring method. The semantic explorer sends to the connector facility the appropriate individuals and their open APIs. This style of semantic exploration enables SE to access the IoT resources needed for end-user applications, which are registered on external IoT service platforms with their IDs, and URLs. The semantic explorer will thus maintain a kind of metadata about the IoT tools, the actual data, and control values of which reside in remote IoT databases, i.e., we execute exploration over their centralized database metadata. The software adapter creates commands for communicating with other IoT systems using individuals received from the textual explorer. The open APIs provided the service collaborator uses HTTP verbs address to the portal for external IoT service to a network entity of the URL type. The individuals received and the open APIs are used to execute the services requested from end-user applications as the query string arguments in HTTP. The SE will provide semantic, discovery-based IoT monitoring and control through this process. According to OWL and RDFs introduced from ontology, the IAO manager is responsible for managing and upgrading the IAO as shown in Figure 5.

Implementation: Within this section, we explain how the SE is implemented. Figure 5 demonstrates the authoring method that is focused on the Internet. Smart gadgets such as, cellphones, laptops as well as personal computers support this application. In this application hypertext markup language (HTML) is used with the Query. In this application different ontologies of different domains can be added by using the add button.

We use JavaScript and extended markup language (XML) asynchronous to send input values and links to the SE.Fig. 6 describes the IAO class hierarchy web protégé 4.3(Protégé,2021). It enables to specifically define the ontology created from the Web constructor tool.

In each service domain, the IAO supports various basic data types of IoT services using explicit data properties. The SE is structured and designed with the help of java, the Jena library, and the Web Application Server (WAS) Pellet-2.3.1 reasoner, i.e. Jetty9.0. We constructed a java servlet for communication to the web constructor tool inside the SE. We have used Java's Jena library and the Pellet-2.3.1 reasoner Jetty9.0 to establish the semantic translator, ontology adder, and semantic explorer. We have developed query procedures for semantic exploration using the IAO SPARQL. Finally, the service collaborator is built using the Eclipse HTTP client library [18](eclipse,2021).

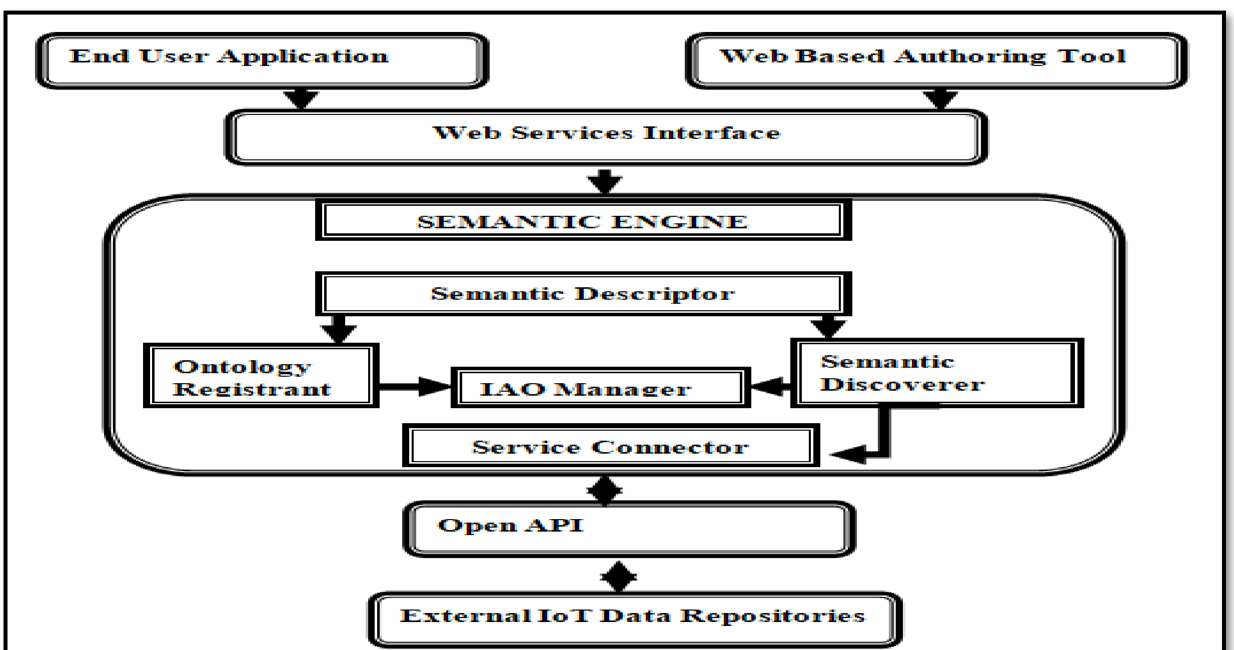

Figure 4. The architecture of the Semantic Engine (SE)

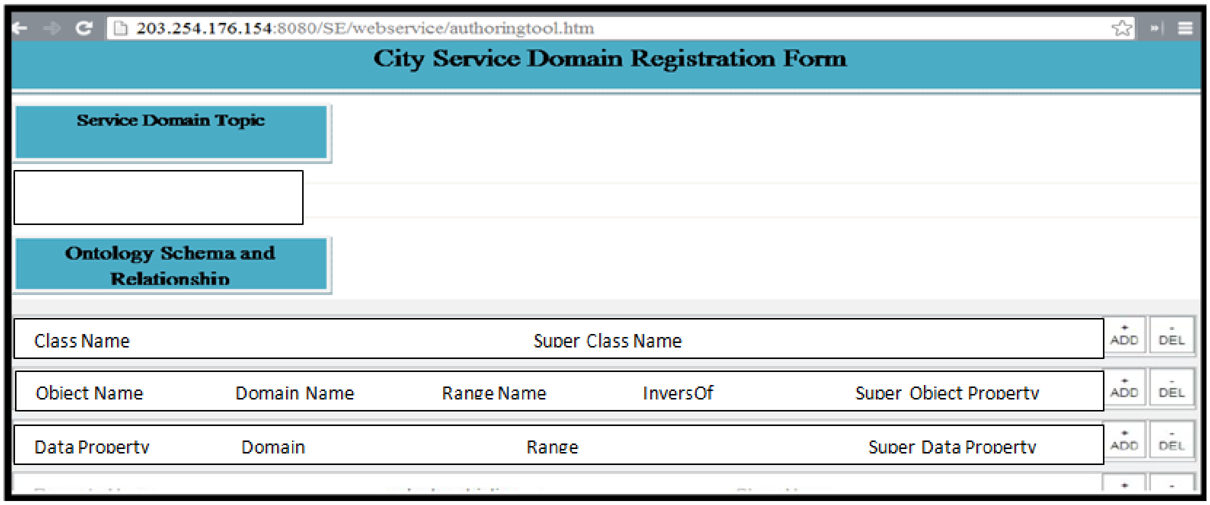

Figure 5. A preview of the authoring method available on the site.

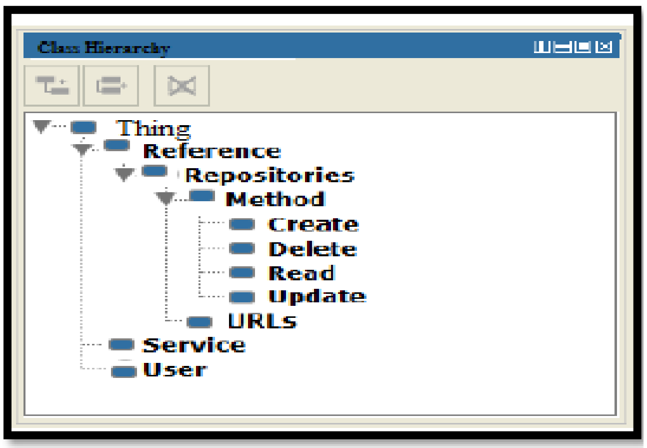

Figure 6. IAO class hierarchy

Evaluation of Performance: In this segment, we evaluate the SE's output by creating a framework for a smart hospital. The framework steps are as follows:

(1) First step: Adding ontology which is built with the help of web constructor tool to SE for the ontological hospital model.

(2) Second Step: The Dynamic semantic translation of IoT tool data using the semantic translator on the SE's main semantic server.

3) Third step: To monitor as well as control with the help of IoT devices and Mobius.

In the first and second step, we use the ontological hospital model with the help of protégé for testing and focused on schema and the annotation. We demonstrate research results of system monitoring and control achieved through a mix of Mobius 'HTTP verbs with its IoT resource characterized as semantic info to test this phase.

Case Study: A prototype of smart hospital service is generated for the SE evaluation. The service offers a customized hospital environment through Web App semantic interpretations of input users' data. It functions in two modes to deliver the service.: (1) user profiles mode for registration; and (2) Service operation request mode.

User profile mode usually collects data of user like ID, password, and user preset, for its customized hospital environments like as hospital activities and their relevant devices and orders. The role of the service activity request mode is to request a smart hospital service by looking at the user profile and its registered data preset.

The Figure 7 shows the prototype service ontological model of hospital data flow. There are seven steps :

1. Mobius adds a repository to the desktop utility. The service consumer then registers their profiles and uses the main server of the smart hospital for the web app to monitor the hospital utilities, as shown in Figure 7.
2. Individuals are created according to the ontological hospital model specification, using the registered user profile. The main smart hospital server adds to the ontological hospital model the registered user profile, which is stored as an OWL (i.e. ontology) script. Using the registered user profile, individuals are generated on the ontological hospital specification.
3. In its main server of the smart hospital a registered user profile in the ontology of the hospital., which is further stored as an OWL in form of an ontology, as shown in Figure 7.
4. In the service request mode, as shown in Figure 7, the service consumer can ask for service with the help of a web app.

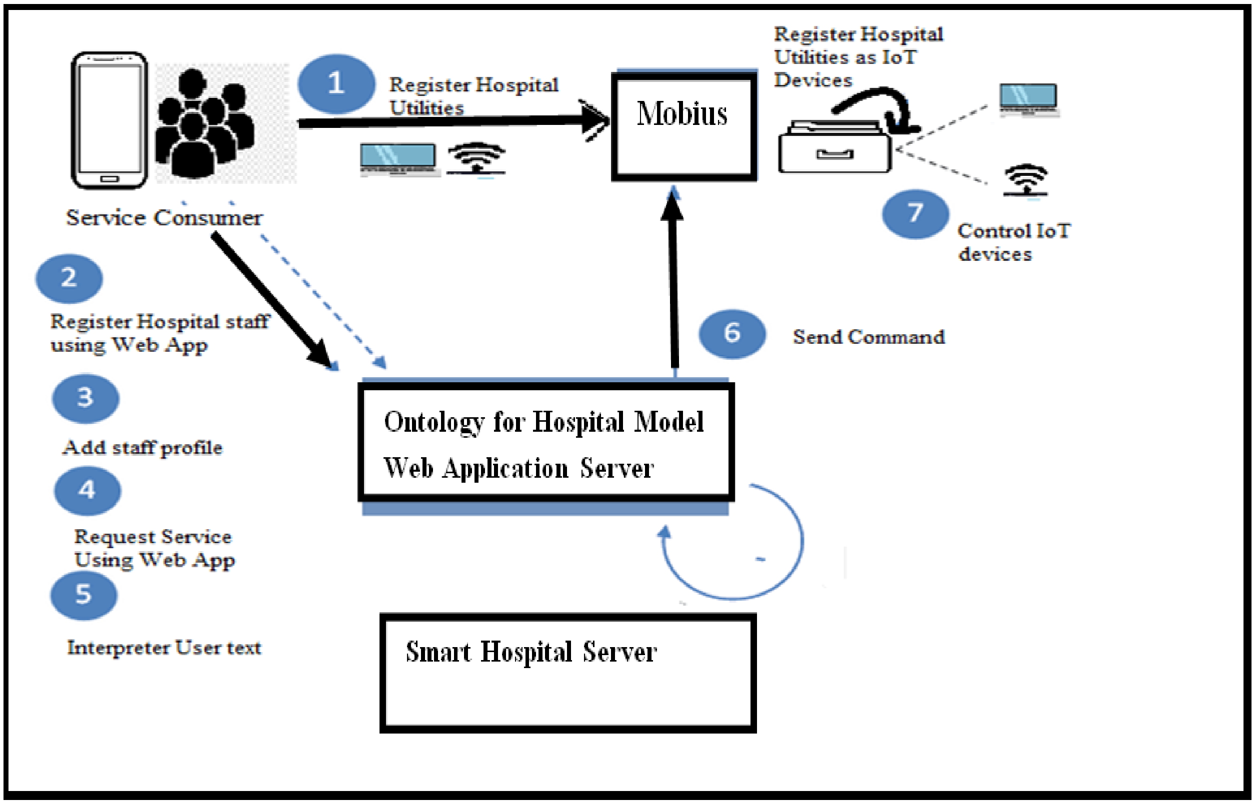

Figure 7. The prototype service ontological model of hospital data flow

5 and 6: In the process, the web application translates the user's data from in the smart hospital main server according to the ontological hospital model and then sends the interpretation results to Mobius using free Mobius APIs as shown in Figure 7-.

7: Finally, Mobius executes an IoT device to monitor the order issued accordingly.

Hospital Models Ontology: The hospital's model ontology is applied, for interpreting the user's data entered and for managing the service's hospital utilities. The hospital model ontology, hospital model has been designed in the service based on a hospital. The hospital is composed of Physician clinics, wards, rooms, and Emergency as shown in Fig. 8. Instead, through an interview with hospital staff, we selected six user hospital acts, such as going to the hospital, working in hospital, staff meeting, and relaxing leaving the hospital, break and lunchtime. We also assessed the relationships in each field between user hospital behavior and the corresponding devices in the interview. Finally, ontological hospital model has been established with the help of protégé, as illustrated in Figure 9.Sub-classes of the first main class device are ID, Mode, and name. The sentence, the second main and abstract classes of user texts which are entered from the web app. It is further subdivided into subclasses like an emergency, doctor's availability, working, working, device availability, and conference.

The User profile configuration mode adds individuals of the subclasses through the web app. The subclasses currently have the corresponding individuals' data entered from the web app with the reference of the user's action type. The system users related to at least one or more than one User Action via the Action object property The fourth main class of User shows an abstract collection of the user's knowledge.

Similarly, User Name, User PW of these subclasses is applied like class Sentence's subclasses. Users have relationships with one or more Computer, User Action, and Sentence class individuals through object properties expressed respectively by has Device Info, has Action, has space, and has Sentence. The ontological hospital model developed is used to establish and incorporate ontology in the IAO evaluation, i.e., Evaluations (1) and (2).The protégé system in Figure 9 is based on four main classes 1) Device, 2) Sentence, 3) space, and 4) User.

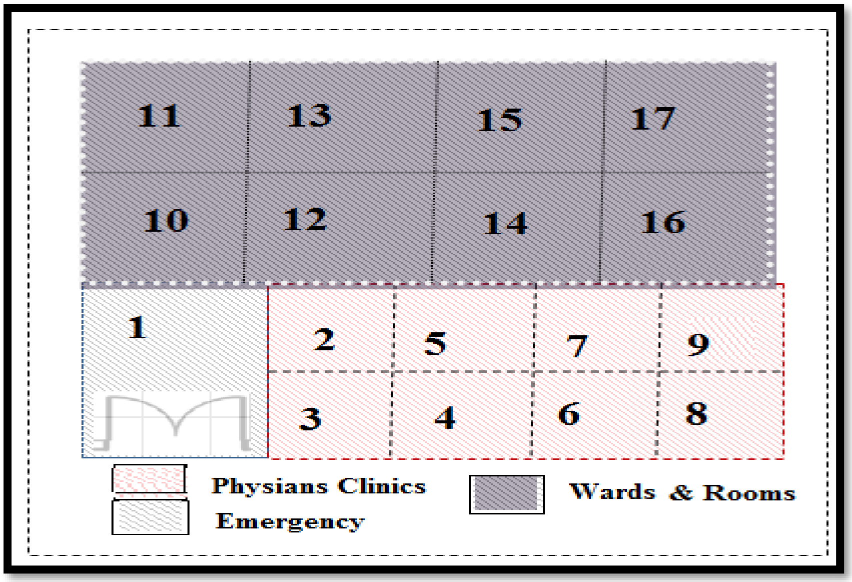

Figure. 8. Classification of the hospital according to room division.

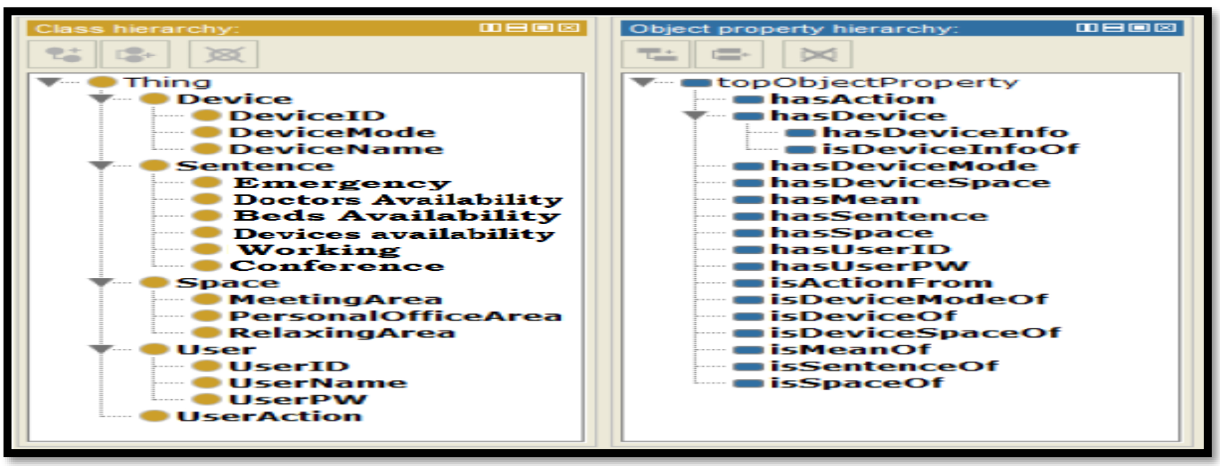

Figure 9. Ontological hospital model was established with the help of protegé

Mobius and IoT Application: IoT application platform is Mobius. It can handle and store the different IoT app's data. It also provides IoT system storage, monitoring, and control capabilities. Mobius monitors and controls IoT devices via RESTful interfaces with its open APIs.

Evaluation of SE: This section explains the scenario of prototype services for the evaluation of the SE. The scenario is based on six steps as shown in Fig. 10.

1. Register the hospital utilities on Mobius
2. Register Web Application services in SE.
3. Create hospital ontology with the help of web constructor tool and add ontology to the IAO
4. Users profiles registry must be done on a web application in SE.
5. In operation mode services must be requested from the web applications.
6. Last step is the verification of the results produced by the IoT control.

Figure 11 shows the smart hospital class the domain name is generated .It is also subclass in the IAO the subclass under Service class. It has a further four subclasses in the Hospital layout, except for user classes, as described in Section 3. These classes refer to knowledge about users have information of User PW, Account, and UserID names, the User in the IAO subclasses are introduced as seen in Figure 11. The implementation of these entity properties shown in Figure 11 by applying it Hospital ontological model in IAO. here it is assumed that the ontology is properly applied to the IAO according to the clear definition of the IAO.

We test the second component of semantic translation with the help of protégé. The research is done on two groups of people. The first kind of type of individuals is represented by the user profile and the preset feedback from the web interface, while the second type of individuals is automatically transformed by interacting with Mobius.

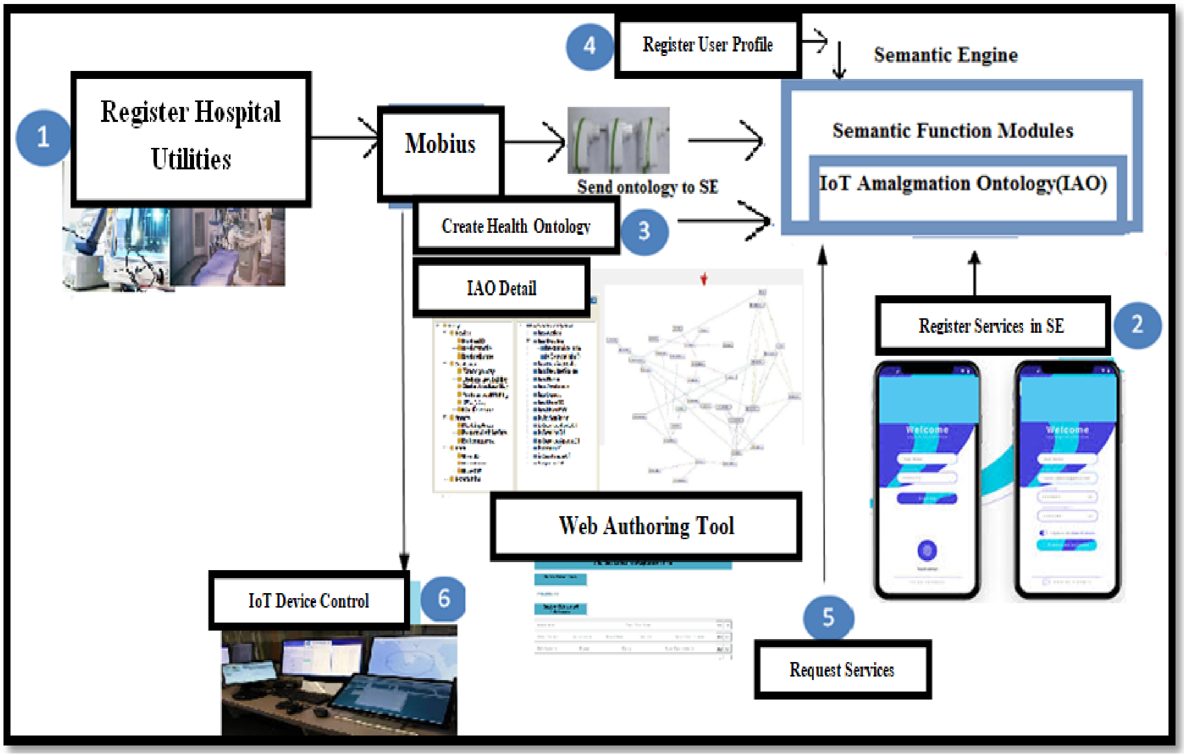

Figure 10. The evaluation process of the SE

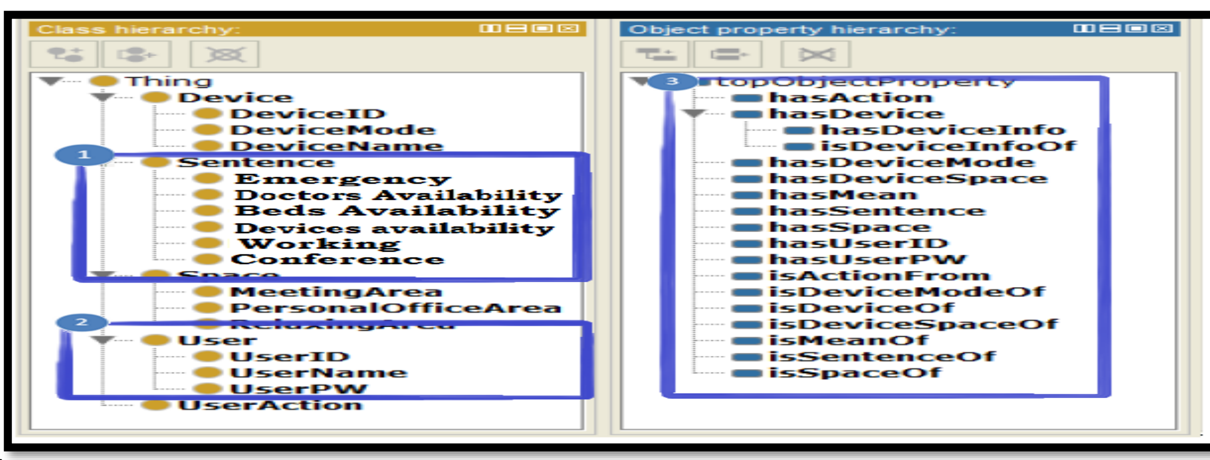

Figure 11:Theresult of the ontology added in the IAO

Smart City Semantic Interoperability Scenario based on SE: We describe the semantic interoperability situation based on SE in this section. We focus in particular on how semantics can be used in a smart city to achieve interoperability between specific business domains. The scenario is as follows: after finishing his or her hospital shift, a service customer needs to turn on the HVAC (heating, ventilation, and air conditioning) device in his or her home. In other words, when a machine in the workplace turned off, the person must turn on the HVAC.1 them.

In both the home and the workplace, the user creates the same user ID. Developers or managers will then use the device, such as in Fig.12, to apply their understanding of the business domain to the IAO. The SE can work with service domains and discover relevant entities in the IAO after the user signs his or her profile and preferences using a service program (e.g. mobile web), as shown in Figure 12.

The SE defines the relationships needed to provide interoperability through the object properties specified in the IAO in this method. For example in order to determine the relationship between a service consumer and a service domain, the SE first finds the connection between the input user ID (i.e. k2014). The related service topics (i.e. home and hospital) in the User class and its subclasses and then uses the entity property to determine the relationship between the matched user ID and the related service topics (i.e. home and hospital) (i.e. service).

Furthermore, the SE will take URLs into account (i.e. iotmobius.com and smarthome.com). Fig. 15 shows that by using semantic interoperability, the SE can now provide a user command-based service to the user. The customer gives their ID and order to the SE, who then uses the Hospital's URL and request that My PC be turned off. The SE verifies the use of My PC after this process. When My PC use mode is disabled, the SE instructs the home repository to turn on the HVAC in the same manner as the My PC. As a result of semantic interoperability, the SE will be able to deliver services across multiple service domains in a smart city. The same type of device would exist in several service domains at the same time, according to IoT.

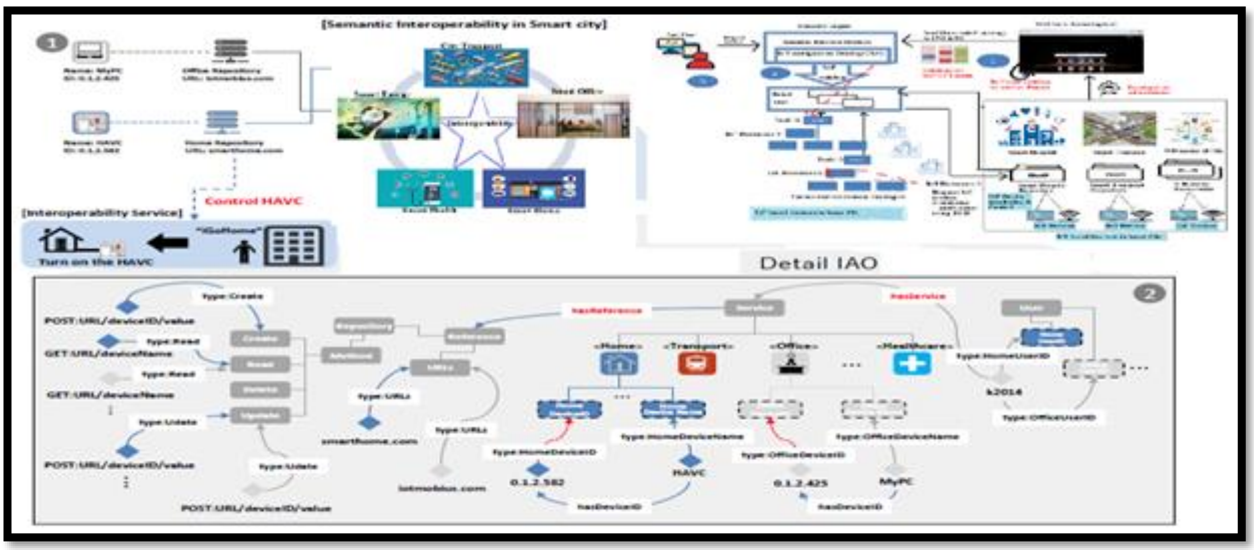

Figure 12. Semantic interoperability in applications for smart city infrastructure.

The service criteria may identify and store each as a separate device on the corresponding IoT service platform (e.g. IoT data repository). At this point, the system's corresponding functions may be required in a specific service domain. For instance, we believe there are two monitors with the same model number that can be found in both the Workplace and Home domains. Depending on the service needs of each service domain, the monitors given a different name, type, function, and role. It is, of course, natural for every service domain.

However, because of the duplicate existence of the device, such a domain-dependent definition of the same type of device results in increased semantic data (i.e. individuals) and the complexity of the relationship between them when contemplating the integrated service system infrastructure for semantic interoperability.

Individuals who use the same facility. The best solution is to identify and redefine the profiles (such as name and function) of the same category of IoT devices in the interconnected service network as IoT devices are implemented from each business domain, but this is a tricky task.

Deal with a wide range of IoT devices that may or may not be similar.

DISCUSSIONS

We need to provide an integrated service system infrastructure that can accommodate the full-service expertise of IoT-based service domains to share IoT resources across service domains, as demonstrated by the importance of interoperability across the entire service domain (e.g. smart cities) in this paper. We designed a SE to provide such Integrated Service System Infrastructures..; nonetheless, there are certain extra practical considerations to be made. Another question is how to accommodate the growing number of IoT applications. The use of mobile apps or web applications to deliver IoT system tracking and control services is the way to go across IoT-based software domains. Even though the SE and the web application are combined, Ajax was unable to support the connection of countless IoT device tracking and control applications by manually adding these to the prototype service. As a result, we consider to efficiently accommodate, an ever-increasing number of applications. Although we have proposed an integrated service platform for applying semantics technologies to IoT-based service domains, semantics technology's primary goal is to share represented knowledge within each service domain using the current site. As a result, we may recommend that data be made freely available for semantic interoperability between IoT-based service domains. This procedure, on the other hand, may not be adequate for providing semantic interoperability for IoT-based services. It must be dynamically generated and then released. Various IoT apps have been introduced and their data captured in real-time.

Table 2 shows Current work achieves the interoperability in different domains.

Application	Coverage			Data Interoperability	Knowledge Representation	Semantic Modeling
	sensors	observation	Domain			
Service-Oriented Ontology for Wireless Sensor Networks Real World and digital objects	✓	✓	✓	✓	✓	✓

Table.2. Domains and interoperability

Conclusions: We present a semantic engine platform (SEP) to support ontological models in a variety of IoT-based service domains in this paper. We designed a web-based authoring tool to build ontology for a service domain for the ontological models to be a formal, overt representation of knowledge within a service domain. We have also built an ” IoT Amalgamation Ontology (IAO)” as a top-level ontology to keep track of and manage the ontology is generated by the web-based authoring tool. Besides, for semantic translation of the IoT resources used in each service domain, we developed a semantic descriptor.

IoT-based systems that are massively distributed, heterogeneous and resource-constrained: integrated semantic discovery in highly distributed IoT domains, the real-time dynamic semantic expression between a large number of IoT resources, and a semantic data repository to archive a massive amount of data extracted from IoT devices. To demonstrate the practical viability of our proposed approach, we created a prototype service for a smart hospital that can provide a preset, customized office environment, such as automatically turning on or off office utilities or lighting based on user text input sent from his or her smart phone.

We demonstrated that the SEP would allow the prototype service to perform semantic discovery, dynamic translation, and finally IoT device control in real-time while working with our IoT service platform through the assessment. We have also spoken about a situation for SEP-based semantic interoperability in a smart city, where people can get integrated and complete services across service domains. As a result, we can confidently infer that the SEP-based approach would help to realize semantic interoperability across the entire service domain using IoT-based systems and semantic technologies.

REFERENCES

Avancha, S., C. Patel and A. Joshi, (2004). Ontology-driven adaptive sensor networks. UMBC Student Collection.

Chen, B., J. Wan, L. Shu, P. Li, M. Mukherjee and B. Yin (2017). Smart factory of industry 4.0: Key technologies, application case, and challenges. Ieee Access, 6, 6505-6519.

Compton, M. M., P. Barnaghi, L. Bermudez, R. G. Castro and O. Corcho, *et al*. (2012). The ssnontology of the w3csemantic sensor network incubator group. Web Semantics: Science, Services, and Agents on the World Wide Web, vol.17, pp. 25-32, 2012.

Compton, M., P. Barnaghi, L. Bermudez, R. Garcia-Castro, O. Corcho, S. Cox and K. Taylor (2012). The SSN ontology of the W3C semantic sensor network incubator group. Journal of Web Semantics, 17, 25-32.

De, S., T. Elsaleh, P. Barnaghi and S. Meissner (2012). An internet of things platform for real-world and digital objects. Scalable Computing: Practice and Experience, 13(1), 45-58.
Eclipse Client Library.

Ferretti, M., and F. Schiavone, (2016). Internet of Things and business processes redesign in seaports. The case of Hamburg. Business Process Management Journal.

Gubbi, J., R. Buyya, S. Marusic and M. Palaniswami, (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. Future generation computer systems, 29(7), 1645-1660.

Henson, C. A., J. K. Pschorr, A. P. Sheth and K. Thirunarayan, (2009, May). SemSOS: Semantic sensor observation service. In 2009 International Symposium on Collaborative Technologies and Systems (pp. 44-53). IEEE.

http://protege.stanford.edu, 2021.https://www.eclipse.org/jetty/javadoc/current/org/eclipse/jetty/client/HttpClient.html ,2021.

Islam, S. R., Kwak, D., Kabir, M. H., Hossain, M., andKwak, K. S. (2015). The internet of things for health care: a comprehensive survey. IEEE access, 3, 678-708.

Kim, J. H., Kwon, H., Kim, D. H., Kwak, H. Y., and Lee, S. J. (2008, May). Building a service-oriented ontology for wireless sensor networks. In Seventh IEEE/ACIS International Conference on Computer and Information Science (icis 2008) (pp. 649-654). IEEE.

Majumder, S., E. Aghayi, M. Noferesti, H. Memarzadeh-Tehran, T. Mondal, Z. Pang, and M. J. Deen (2017). Smart homes for elderly healthcare—Recent advances and research challenges. Sensors, 17(11), 2496.

Mehmood, Y., F. Ahmad, I. Yaqoob, A. Adnane, M. Imran, and S. Guizani (2017). Internet-of-things-based smart cities: Recent advances and challenges. IEEE Communications Magazine, 55(9), 16-24.

Parveen, K., G. A. Shah, M. Aslam and A. Farooq (2021). Monitoring of Unaccounted for Gas in Energy Domain Using Semantic Web Technologies. COMPUTER SYSTEMS SCIENCE AND ENGINEERING, 36(1), 41-56.

Protégé for Ontology Editor.

Tao, M., K. Ota and M. Dong (2017). Ontology-based data semantic management and application in IoT-and cloud-enabled smart homes. Future generation computer systems, 76, 528-539.

Trasviña-Moreno, C. A., R. Blasco, Á. Marco, R. Casas, and A. Trasviña-Castro (2017). Unmanned aerial vehicle based wireless sensor network for marine-coastal environment monitoring. Sensors, 17(3), 460.

Zanella, A., N. Bui, A. Castellani, L. Vangelista, and M. Zorzi, (2014). Internet of things for smart cities. IEEE Internet of Things journal, 1(1), 22-32.