Ontology Development based on Generic Object Oriented Smart Home Model

Yulistiyan Wardhana, Bob Hardian and Gladhi Guarddin
Faculty of Computer Science, Universitas Indonesia
Email: yulistiyan.wardhana@ui.ac.id, hardian@cs.ui.ac.id and adin@.ui.ac.id

***Abstract*—Smart home development has many methods that cause the rise of various smart home systems which has its own uniqueness on every system. Nevertheless, those systems could not accommodate their communication and information sharing. It causes the difficulties of collaboration on a system to another, and it makes hard to implement an existing models on a new system. One of solution to answer this problem is to implement ontology. Ontology allows the preparation of structured system that corresponds to a specific character in accordance with the needs of the system. Ontology implementations in Smart Home are expected to ease in managing information and using information. In this research, an ontology is built. The ontology would be mapped from the smart-home object based generic design. We developed a simple system by utilizing the ontology that has been built. Based on the experiment has been done, the ontology has had a consistent structure and appropriate with the smart home system design requirement. Based on tests that we have done, the ontology is easy to be implemented into a system. Besides, with a limited instance, the ontology has been successfully delivered 81.027 % appropriate response with 0.0146 second average response time to the residents activity in the smart home environment.**

Keywords*—*Ontology; Smart Home; Object Oriented

I. Introduction

The smart home system is usually processed data that has been retrieved from the system environment with various methods [1]. The method, it could be rule-based system, neural network, reinforcement learning, hidden markov model, fuzzy logic [2] and ontology [3][4], is used to produce an appropriate responses from the smart home system. Every kind of method has its own advantages and disadventages that produces uniqueness on every system. The uniqueness of every smart home system raises a lot of variation of smart home system and cause so many diversity in the development of smart home systems. Unfortunately, there is no available collaboration feature from a smart home system to other systems [5].

In this paper, we propose a method that is combined with a general model of smart home system concept. The general model of smart home system concept is combined with ontology modeling to produce more general and applicable system. Ontology is chosen as the method we used because it gives simplicity in sharing information and structure to the smart home systems [6]. Ontology also allows knowledge reuse in the research or other smart home systems [3].

II. Generic Object Oriented Smarthome Model

Although various model has been developed for smart home, most of them have a lot of similiarities on their components and functions [7]. Based on that fact, we use more generic model that combine functionalities, concepts and technologies of smart home models. The generics model is developed based on object oriented concept in embedded system that allows it to be reused later. The model that we use can be seen in Fig. 1.

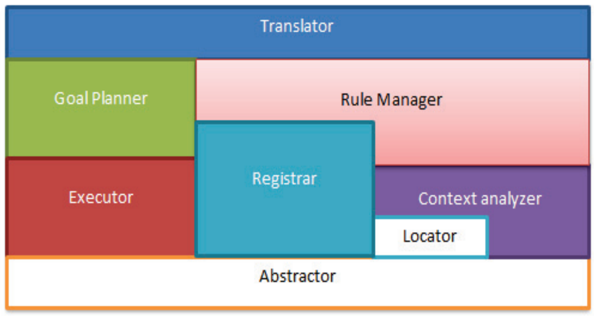

Fig. 1: Generic Object Oriented Smarthome Model

The model is divided into eight part. The role of each part is described as:

- Registrar
  The model is designed like an operating system. It has its own drivers and registers that we call Registrar. The Registrar can be accessed by other components that run above the model. The Registrar has role as an exchange point that provide the model with mechanism service system to other components.
- Abstractor
  Inter-part interaction in smart home is managed in this component. There is a specific mechanism that enables the smart home system to make direct interaction with its parts. Besides, this component also processes the abstraction of definition,

location, command execution and information retrieval from the smart home's parts. The results of this component processes will be distributed to other components in the smart home system.

- Translator
  This component is responsible to translate user's response and manage interaction between the system and users. High level language approach [8] is used to establish a well connection and interaction between them.
- Goal Planner
  In making services, the system is ran based on Goal which is made based on occupant's needs. Goal represents an aim of one service of the system. To achieve that, there are some steps should be taken.
  The steps that the system would take is represented by a Plan. Plan is built with burdens that are represented by Intelligence Rules. Intelligence Rules are created based on a learning process that is performed by the system.
  The execution of Plan in the system to achieve an exact Goal based on user's needs is decided by this component. The component can use a predefined relationship model between Goal, Plan and Intelligence Rule.
- Executor
  This component has a role to connect the decisions that the system takes to Abstractor that pass them to smart home devices. Plan, a result of information processing in the system, is translated and connected to a Task.
  Task contains some commands on what the system should do next based on its environment information. Although Executor produces Task that has role to control smart home devices, it does not have direct connection to have interaction with users.
- Locator
  Locator has a role as a component that records all of object location in the smart home environment. The location information that has been captured would be processed as a parameter in context processing in Context Analyzer.
- Context Analyzer
  Information in the smart home environment, such as devices status and object location, is processed by this component. The result from this component is a relevant context that is used by Rule Manager to produce suitable responses from the system.
- Rule Manager
  This component has a role to create Goals based on users Rules and captured information, such as system state, situation and context. The result of this component is passed to be parameters of Goal Planner.

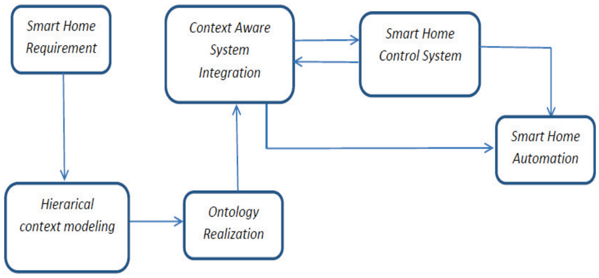

Fig. 2: Smart home's ontology development process

III. Ontology Development

In smart home context, ontology is a specification of a concept of smart home modeling by connecting a smart home structure to the others [6]. The usage of ontology could be interpreted as the contextual overview of smart home with semantic definition that represents real world knowledge. The ontology could be used in activity context analyzing and the distribution of information based on structural context of users, devices and services [9].

The implementation of smart home ontology is briefly introduced in Fig. 2 . The ontology modeling itself is started from Hierarical Context Modeling based on requirement that has been gathered until the context aware system integration. After ontology is integrated with context aware system, the control system will be able to use ontology to manage the flow of information in the smart home system.

A. Hierarical Context Modeling

This process has four sub-processes that start from determining ontology domains from system requirement, checking existing ontology that could be used, enumerating key words based on system requirements and specifying class and its hierarchy.

1) Determining The Ontology Domain: Ontology model in this research is based on object oriented smart home model by Guarddin in 2011 [7]. The model has consisted requirements list of smart home model and its system domain. From available requirements in existing research, we gathered assessment elements that will become ontology's domain. The elements that has been gathered can be seen in Table I.

2) Checking Existing Ontology: After the ontology domain has been determined, the next step is to choose existing available ontology that could be used in the model. The candidate that could be used is described as follows:

- FOAF[1]

FOAF is a model that has connectivity between person information. The model has three section, they are social network and people collaboration, friendship, and association; person point of view on actual concept; and person information network that use web network. In this smart home model, FOAF is used for completing the occupant data model.

- Location aware ontology[2]
  Location aware ontology is an ontology for context aware based system in ubiquitous, pervasive, interdomain and intradomain organization that is focused on the location context. This ontology also could be combined with occupants and devices model part to complete its location context information. This model is implemented with DARPA Agent Markup Language (DAML) 2.

TABLE I. ONTOLOGY ASSESSMENT ELEMENTS

No	Element
1	The availability of system information to smart home user/occupant based on system's service and responses in certain conditions.
2	The availability to gather location context of smar home devices in particular time.
3	The availability of information to multimedia content that lies in the system.
4	The system can control and execute a planned goal based on contextual information that is gathered from smart home's environment and occupant's preference.
5	The system can report an event while there is a device's status alteration.
6	There is notice of the system's current conditions to occupant.
7	The availability of heterogeneous device information in the system.
8	The availability of a eventual report while device's status is changed.
9	The availability of location processing, context processing and occupant's preference alteration as a function that is implemented in the system.
10	The availability of information about location processing's function that is used in device's presence detection.
11	The availability of information about system interaction with smart home sensors and devices.

- Devices ontology[3]
  Device ontology is an ontology of devices model that is developed by OpenCyc. OpenCyc is an open source general knowledge base system and commonsense reasoning engine which is claimed to be the most complete and generics in the world knowledge base system. This model could be used for completing device model in the ontology. The ontology is implemented with DARPA Agent Markup Language (DAML).
- Time basic ontology[4]
  Time basic ontology is an ontology that is a part of COBRA-ONT ontology implementation. This ontology is represented time and its temporal relations. The implementation of the ontology is based on DAML-TIME ontology and Time-Entry ontology.

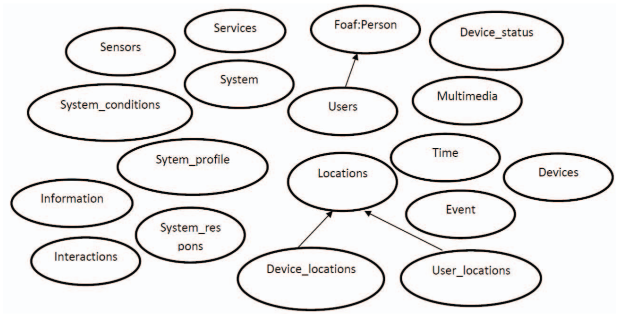

Fig. 3: Initial ontology class

From those models that are considered, FOAF is the only model that is used in the development of the smart home ontology. FOAF is chosen because its properties have suitability with the smart home requirements and has suitability with the current model ontology development language. Furthermore, FOAF provides URIs for its classes and properties that one can easily use.

3) Class and Class Hierarchy Modeling: The class of ontology is determined by enumerating lexicons from ontology assessment components. The lexicon is chosen by its importance and suitability with the requirements. The lexicon list building is described in Table II.

TABLE II. LEXICON LIST

Assessment Element No	Lexicon(s)
1	information, information type, user, service, system response
2	device location, user
3	multimedia
4	users, information
5	device status, event
6	system condition, system, user
7	device, system
8	device status
9	location, user, system
10	location, device
11	sensor, device, interaction

After the lexicon list is available, ontology's class and its hierarchical structure is built with adopting bottom-up and top-down method combination. Initial hierarchical can be seen at Fig. 3.

B. Ontology Realization

The next step after hierarchical structure is built was class and property implementation. Property will be

[1]http://xmlns.com/foaf/spec/
[2]http://www.daml.org/ontologies/336
[3]http://www.daml.org/ontologies/317
[4]http://www.daml.org/ontologies/427

determined based on the relationship between classes which is developed based on the requirement. After the properties were determined, the ontology was built with Protege 4.30[5].

We also use Protege 4.30 to check the ontology logical consistency and evaluate the ontology whether it has been fulfilled the requirements. The evaluation was done by applying SPARQL's query based on the ontology's competency question.

1) Property Definition: There should be two types of property that had to be determined, data property and object property.

(a) Data property: Data property is determined from classes necessity on Literal values. There should be seven data properties that are used by four classes. The data property list can be seen at Table III.

TABLE III. DATA PROPERTY

Class	Data Property	Type
Information	information type	Literal
Locations	pointX	Integer
Locations	pointY	Integer
Time	time section	Literal
Time	date	date (DD:MM:YYYY)
Time	timestamp	dateTimeStamp (HH:MM:SS)
System	activate	Boolean

(b) Object property: Object property is defined by based on relationships of classes that are lied on the smart home requirement. After that, the relationships are mapped into object property that is connected over classes. The initial object properties after relationship mapping can be seen at Table IV.

TABLE IV. INITIAL OBJECT PROPERTY

Domain	Object Property	Range
System respon	asks	Device
User	at	User location
Device	do	Interaction
Sensor	gathers	Information
System	gathersInfo	Sensor
System	hasCondition	System condition
System	hasProfile	System profile
Device	hasStatus	Device status
Device	liesOn	Device location
Sensor	locatedIn	Device location
User	makes	Interaction
Device	plays	Multimedia
System	produces	System respon
Device	produces	Service
Interaction	respondsTo	User
System	runsAt	Time
Device	sends	Event
Interaction	interactWith	Sensor

[5]http://protege.stanford.edu/

Next, the properties is used to build an initial ontology that would be iterated. The iteration is to be done using SPARQL[6]until the model matched system requirements and its assasement element. After the iteration, ontology's hierarchical and properties has been changed. The final hierarchical can be seen at Fig. 4 and the final model can be seen at Appendix A.

IV. Testing and Evaluation

The evaluation process of the final ontology is to be described in this section. Three test will be conducted, the ontology consistency test, the ontology-system suitability test and the ontology-occupant activity suitability test.

A. Ontology Model Consistency Test

The model can be described as the consistent one if it is logically consistent while it is tested by reasoner [10]. The consistency test will be held by testing the model with ontology reasoner. The query will be tested with three ontology reasoner; FaCT++, Hermit, and Pellet[11]. The test that has been held using those three reasoners didn't show any logical inconsistency on the ontology. Thus, it is concluded that the ontology has had an consistent model.

B. Ontology and System Suitability Test

This test will be held by implementing SPARQL query based on ontology assessment elements. First, the ontology assessment elements are converted into competency questions. Then, the competency questions are converted into SPARQL queries. Last, the result of SPARQL queries would be checked whether it matches with ontology assessment elements description.

The test was done iteratively with iterative improvement into the ontology. The final result of the test is an ontology that is matched with the system requirements. Thus, it could be said that the ontology has represented all requirement of the system.

C. Ontology and Occupant Activity Suitability Test

This test is held by implementing the ontology into the smart home control system and use occupant's activity data as its input. The system responses based on the occupant's activity data were being compared with

[6]http://sparql.org/

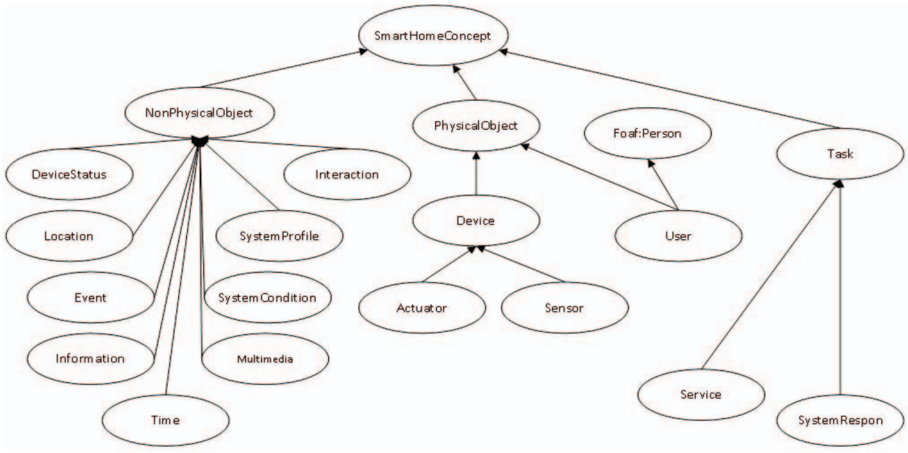

Fig. 4: Final ontology class

the actual responses that occupant expected. There were three scenarios to be done based on the ontology's instances; manual configuration instances, instances with PART learning in many sensors, instances with PART learning in three sensors. PART learning algorithm was being used because it could provide rule set that is converted into ontology's instances [12].

1) Testing with Manual Configuration: This scenario was using knowledge base that was set manually by creating instances that represents custom habit without considering occupant's usual activity. The result of this scenario is just about 41.370 % with 0.015 seconds response time. The result is indicated that there must be a learning mechanism to make the instances correspond with occupant's habit. Yet, the response time shows that the system could give fast responses to the occupants.

2) Testing with Learning Instances in many Sensors: The scenario used 40 sensors from the occupant's activity. The data was processed first with PART learning algorithm before it was converted into instances. The learning process was ran based on the occupant's monthly activity. Then, the result of the learning process was converted to be ontology instances to make the system able to give responses.

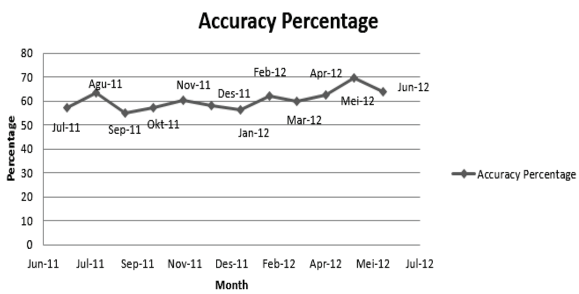

Fig. 5: Testing result with learning algorithm in many sensors

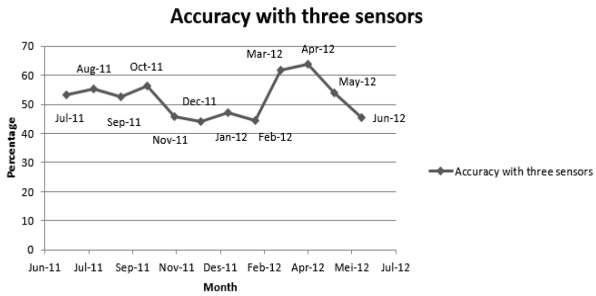

Fig. 6: Testing result with learning algorithm in many sensors

The test produced 59.731 % total matched result with 0.019 seconds response time. Detailed test result can be seen at Fig. 5.The test shows that learning algorithm could improve the response's accuracy. Yet, the learning features should be filtered to produce better accuracy. Per-month result also shows that per-time section learning is needed to segmented the learning to produce better result.

3) Testing with Learning Instances in Three Sensors: This scenario has similarity with previous scenario. The difference is just the sensor's usage. There were just three sensors data that were used into learning algorithm. Yet, the scenario has worse accuracy, with just 53.211 % accuracy with 0.0176 seconds response time. Detailed test result can be seen at Fig. 6. It means that more sensor data could provide better accuracy to response occupant activity. The difference result per-time section also shows that per-time section learning disseverance is needed to produce better result.

D. Evaluation and Improvement

The previous tests shows that there must be diversification in learning the data. In this test, the learning

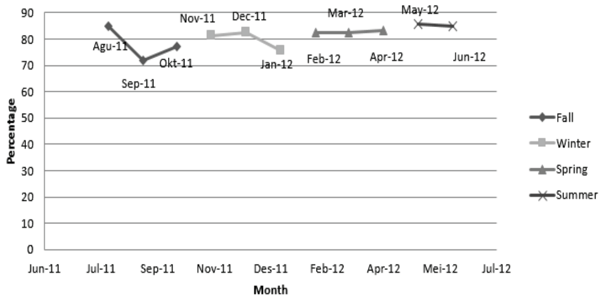

Fig. 7: Testing result with learning algorithm in many sensors

phrase was done in four parts. Each part represents per-season data which is affect occupants habit. The result of the test shows that there was 81.027 % match responses with 0.0146 response time. Detailed test result can be seen at Fig. 7. It shows that the model has been suitable to be implemented in the system and can provide appropriate response in good accuracy. Yet, more improvement and research in the usage of machine learning to build the knowledge base is needed to produce more accurate responses.

V. Conclusion

The ontology has been implemented and tested in three scenarios. Based on the tests, the ontology has fulfilled smart home system requirements needs. The tests also gave an example of an simple implementation of the ontology into a system. With the implementation, the ontology has been successfully gave 81.027 % appropriate response with 0.0146 second average response time to the residents activity in the smart home environment. Besides, more complex implementation and more complete instances are needed to be researched in the future.

References

[1] D. Valtchev and I. Frankov, "Service gateway architecture for a smart home," IEEE CommunicationsMagazine, pp. 126–131, 2002.

[2] D. J. Cook and S. K. Das, "How smart are our environments? an updated look at the state of the art," Pervasive and Mobile Computing, p. 5373, 2007.

[3] X. H. Wang, D. Q. Zhang, T. Gu, and H. K. Pung, "Ontology based context modeling and reasoning using owl," IEEE Computer Society, 2004.

[4] A. Fensel, S. Tomic, V. Kumar, M. Stefanovic, S. V. Aleshin, and D. O. Novikov, "Sesame-s: Semantic smart home system for energy efficiency," Informatik-Spektrum, vol. 36, no. 1, pp. 46–57, 2013.

[5] D. Bregman and A. Korman, "A universal implementation model for the smart home," International Journal of Smart Home, pp. 15–30, 2009.

[6] B. Chandrasekaran, R. J. Josephson, and V. R. Benjamins, "What are ontologies, and why do we need them?," IEEE Intelegent System, pp. 20–26, 1999.

[7] G. Guarddin, "Rancang bangun rumah cerdas berorientasi obyek di atas framework osgi," Master's thesis, July 2011.

[8] S. Yemini, G. Goldszmidt, A. Stoyenko, Y.-H. Wei, and L. Beeck, "Concert: a high-level-language approach to heterogeneous distributed systems," in Distributed Computing Systems, 1989., 9th International Conference on, pp. 162–171, Jun 1989.

[9] T. V. Nguyen, W. Lim, H. A. Nguyen, D. Choi, and C. Lee, "Context ontology implementation for smart home," 2007.

[10] N. F. Noy and D. L. McGuinness, "Ontology development 101: Aguide to," tech. rep., 2001.

[11] K. Dentler, R. Cornet, A. ten Teije, and N. de Keizer, "Comparison of reasoners for large ontologies in the owl 2 el profile.," Semantic Web, vol. 2, no. 2, pp. 71–87, 2011.

[12] E. Frank and I. H. Witten, "Generating accurate rule sets without global optimization," in Proceedings of the Fifteenth International Conference on Machine Learning, ICML '98, (San Francisco, CA, USA), pp. 144–151, Morgan Kaufmann Publishers Inc., 1998.

APPENDIX

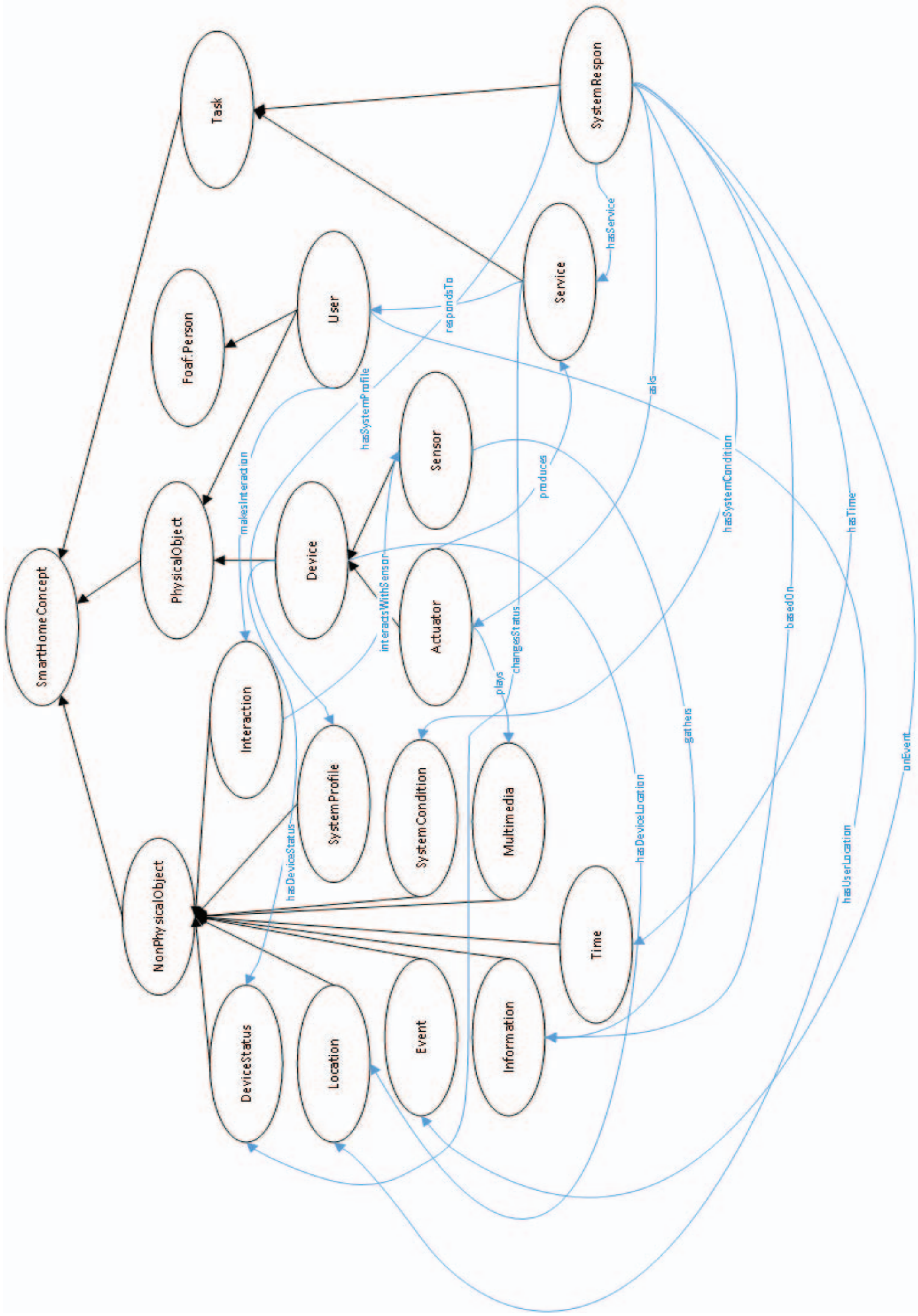

Fig. 8: Generic Object Oriented Smart Home Model's Ontology

 978-1-5090-0363-1/15/$31.00