Journal of Theoretical and Applied Information Technology
31st March 2017. Vol.95. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CONTEXTUAL VARIABILITY MANAGEMENT OF IOT APPLICATION WITH XML-BASED FEATURE MODELLING

[1]ASAD ABBAS, [2]ISMA FARAH SIDDIQUI, [3]SCOTT UK-JIN LEE*

[1,2,3]Department of Computer Science and Engineering, Hanyang University ERICA, Ansan, South Korea
*Corresponding Author: [3]SCOTT UK-JIN LEE
[1]asadabbas@hanyang.ac.kr, [2]isma2012@hanyang.ac.kr, [3]scottlee@hanyang.ac.kr

ABSTRACT

Internet of Things (IoT) can be used anywhere, anytime and with any device. IoT devices are used broadly in recent industry such as health care, transport, education and communication channels. IoT applications make enable the use of IoT systems in different environment such as heat sensor functionality varies for indoor and outdoor, water sensors in automobile for screen viper movement (auto working during rain and stop if there is no raining), heartbeat sensors for old and young patients. Moreover, IoT devices are same but the operating functionality is different according to environment and end user's perspective. Therefore, this kind of variability management is very important for reusability of resources in multiple environments. However, variability management is difficult task due to the dependency of resources, relationships between them and compatibility issues. Feature Modeling of Software Product Line (SPL) is extensively used to manage the variability management for family of software that share common resources and varies in different points. In this paper we have proposed simple and effective approach for variability management of IoT-based applications by using feature modeling. We have proposed XML-based feature modeling for IoT-based applications for variability management and increase the reusability of existing resources. Our approach defines the complete package from domain modeling to application development by using XML-schema and XSLT files to select and deselect the features for final application development. Our final architecture of IoT-based feature model shows the complete relationships and include exclude associations between them without any constraint violations.

Keywords: *Internet of Things (IoT), IoT-based applications, Software Product Line, Feature Modeling, XML-based modeling.*

1. INTRODUCTION

Internet of Things (IoT) enable the connectivity of multiple internet devices to enhance the usability, accessibility and better efficiency to access the data around the world such has health, transportation, mobile phone. IoT can be used anywhere, anytime with any other devices to access and operate the data according to end user perspective [1,2]. IoT system are composed with lightweight devices that operate with different interconnected applications to exchange the functionalities [3]. Environmental selection of IoT systems are perform different functionalities with same devices due to end user perspective applications. Heat sensor IoT device perform different functionalities according to environment such as indoor heat sensing is different and outdoor sensing is different with same device but different operating application [4]. Moreover, multiple IoT devices are connected with each for common goal such as GPS and wifi in mobile and transport systems. IoT systems are composed of physical and virtual components to globally connect with interactive and highly dynamic network that continuously changing and react in own context [5, 6]. Contextual variability (environmental) modeling is most important before development of IoT systems for easy selection of features and functionalities. Contextual variabilities are handled by IoT applications as devices are same in multiple systems but operating applications are different with different environmental features.

Software Product Line (SPL) is an approach to handle the commonalities and variability of families of products. SPL consists on domain engineering and application engineering. Domain engineering defines the scope of SPL where all possible features are exist that can be used in all products. However, application engineering is the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

development process of actual products by using different features from domain of SPL [7, 8]. SPL enhance the reusability of existing resources by using features in multiple products according to end user requirements. Common feature are necessary to be part of each product of SPL. However, variable features are used in products according to required functionalities. Feature model is tree structure that is used to manage the common and variable features of SPL. Variable features are categories as alternative group, optional and OR group. Selection or not selection of variable features are based on end user requirements [9, 10].

IoT applications are developed based on environmental features selection and end user requirements. Environmental variability can be managed by using effective modeling approach that can hold all constraints and enables the application development. In same IoT devices with different applications have contextual variabilities that needs to be manage for effective development and increase reusability. XML-based feature modeling has been proposed to handle the variability management of SPL. In this paper we have proposed an effective and novel framework based on XML-based feature model for variability management of IoT application. Proposed approach enables the effective modeling of IoT application with variability management and cross tree constraints (dependency between features include or exclude). We have defined all possible constraints and relationships between IoT features at XML-schema level, convert to XML and final automatic IoT application development model is converted to XSLT that shows complete structure of application development with variability management. Results of proposed approach is IoT application development without constraint violations.

Further paper is organized as, section 2 explaining the Background of feature model, section 3 is related work, section 4 is XML-based feature modeling, section 5 is xml-based iot applications feature modeling and section 6 is conclusion.

2. BACKGROUND

SPL is used to quick development, time to market and low cost due to reusability of features in multiple products [11]. Feature-Oriented Domain Analysis (FODA) [12] has been proposed by Kyu C. Kang in 1990 to represent the modeling of multiple products that have some similar functionalities and variations. Development of applications are easy according to end user requirements due to clear identification of features and relationships. Domain engineering develop all common and variable features as SPL assets and use them in multiple products. Feature Oriented Reuse Method (FORM) [13] has been proposed by Kyu C. Kang in 1998 to increase the reusability of domain features. Once the development of all common and variable features is completed in domain of SPL, reusability of these features in multiple products with some variation points is easy, time to market and less cost effective. Independent development of all features enables to increase the reusability without any interference of other modules.

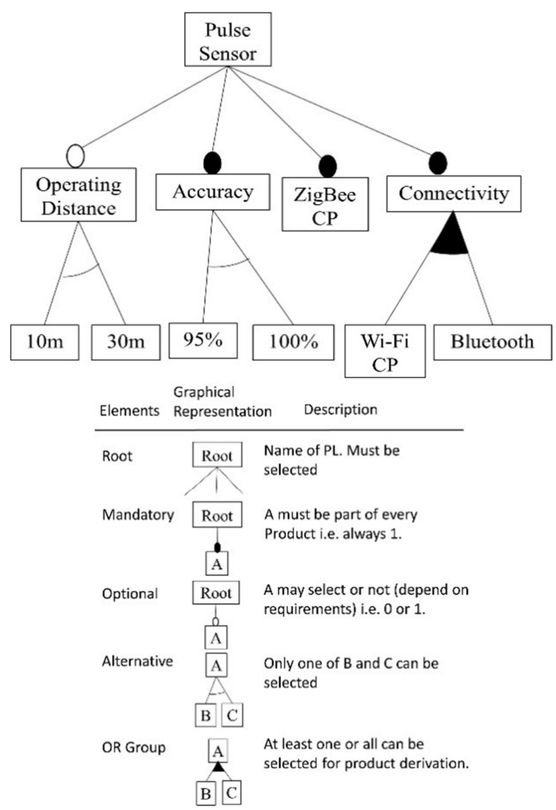

Figure 1. Pulse Sensor IoT Feature Model

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Fig 1 shows the pulse sensor feature model with predefined relationships and constraints for healthcare IoT system. Four main process of feature oriented development approach has discussed below [14].

a) Domain Analysis and Engineering: Primary process defines the complete scope of product line where organization can predict number of products and in each product which features are needs to be select with different variation points by end user perspective.
b) Domain Implementation: This is development process of every feature independently in domain for products derivations.
c) Requirement Analysis: This process defines the requirements analysis from end user. End user defines the complete requirements of system how much efficient and what features should be in product. Moreover, requirements must meet the scope of domain engineering.
d) Product Derivation: Final development process of application according to end user requirements. Developers select the features from domain implementation and generate the product. Reuse ability of features is cost effective and reduce development time for next generation of product line.

3. RELATED WORK

Cechticky et. al. has been proposed XML-based modeling approach of SPL feature model. XML-based modeling approach enables to identify all primary constraints of feature model. Feature model is mapped by using XML-schema, convert it XML file and translate to XSD to select the features for application development by specifying the end user requirements and constraints. To map all the constraints and relationships of feature model in XML-schema, primary information of features relationship such as alternative, mandatory, optional and OR group are need to predefine at the stage of domain engineering. Moreover, at application engineering level, end user requirements, the constraint relationship between features are needs to specify at XSD level for proper feature selections [15].

Hartman et. al. proposed contextual variability of feature model that needs to handle at different environments of application development. Contextual Variability differentiate the all products of SPL in different context according to end user requirement. In multiple product line contextual variability increase due to number of systems communicate each other. This study proposed contextual variability modeling of multiple product lines feature model. Merging the multiple feature models that have some similarity between features in domain engineering such as mobile phone and tab product lines. By merging feature models reusability of features from multiple product lines increase and require less time to market and development cost. [16]

Contextual variability modeling at requirement engineering stage of product lines is required relationships of features at specific environment. Multiple features influence the other features during application development due to complex relationships (include or exclude) between features. To handle the contextual variability of SPL, Ali et. al. proposed unified framework that starts from goals of application development and then maps the goals in feature model at early stage of modeling. Feature model categories each goal as terminal feature and define the relationship between them. As goals are increased during requirement elicitation, the complexity of feature model increases, problem frame of this framework handles the crosstree constraints between features to reduce the complexity [17].

Multiple devices are interconnected in IoT system, integration and communication between these devices are needed to be managed under the variability constraints. Gligoric et. al. has proposed the XML based modeling for IoT applications to resolve the constraint problems between applications of multiple communication devices. XML web service approach is more effective due to efficiently handling the constraints by using XML-schema than the rest of web service applications such as Representational State Transfer (REST) [18].

Ayala et. al. has proposed the modeling approach of IoT applications and agents by using SPL technique to increase the reusability the of

IoT application features. Agents are the medium of communication in cloud computing where large number of different devices are interconnected with different application. However, these applications have some functional similarities and some variations. To increase the reusability of similar features author has modeled the IoT applications in feature model and multiple products can be generated from domain of IoT system [19].

Variability management of Body Area Network (BAN) applications where large number of similar IoT devices has used with different environmental functionality such as healthcare and transport. Functional requirements such as data collection from heart beat and non-functional requirements such as security and performance time are needed to model at early stage of development. Trade-off between functional and non-functional requirements are also important to handle such as security is more important than execution time e.g. email services. However, in some cases execution time is more valuable than security e.g. data collection of heart beat. To increase the reusability of BAN application features and quick development, feature model is effective approach to manage the variability and increase the reusability of BAN applications [20].

Variability and capability management of IoT applications in ecosystems is open challenge for modeling. Contextual variability of IoT applications where large number of different features are exist to use as alternate and some optional and OR groups of features are the part of system. Author has proposed semantic modeling approach for variability management of relationships and crosstree constraints of features of IoT applications. Web semantic approach is effective to manage the contextual constraints of large and small IoT applications [21].

The given research is limited to modeling with only family of software but not IoT application where large number of constraints exist due to different environmental selections of features.

4. XML-BASED FEATURE MODELING:

Meta-modeling and modeling level are two-layer structure for feature modeling. Types of features are defined in meta-level layer where all properties such cardinalities and relationships are defined. Modeling level layer defines the features construction according to the interest of end user specifications and perspectives. This approach can be exist where family of products exist that share common resources and assets. Feature modeling to application development requires multiple layer structure as given below [15]:

a) Family meta-model
b) Family Modeling Level
c) Application Modeling level

XML language is used to define the feature model and XML-schema is used to define the meta-model where all relationships and constraints can be defined. XML-based model has to express the correctness of defined relationships in XML-schema. Both domain and application models are used to express the XML-based feature models that initiated from predefined notations in XML-schema [XML]. XML-schema represents the collection of all reusable components of SPL. SPL consists on root node to leaf nodes where multiple relationships and constraints exist that are covered and defined in XML-schema for feature modeling [22].

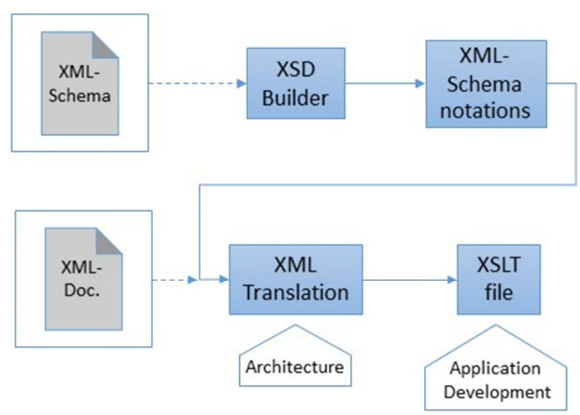

Figure 2. XML-based Family Meta Modeling

Fig. 2 shows the complete process of XML modeling from XML-schema to application development by translating the XML document to XSLT file. XSD builder map the cardinalities and constraints that are exist between functional and non-functional requirements. Application development is based on the final architecture that construct from end user requirements for specific functional.

Journal of Theoretical and Applied Information Technology
31st March 2017. Vol.95. No 6

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

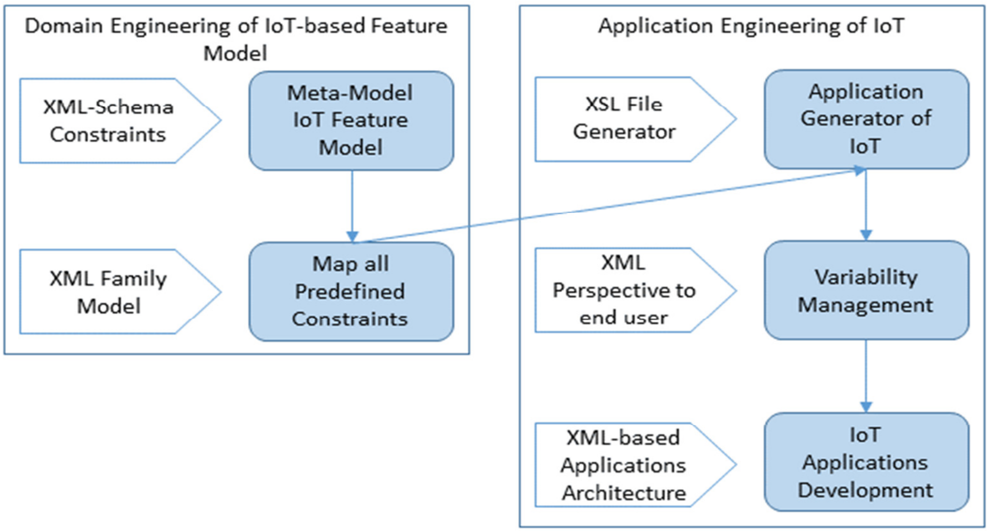

Figure 3. XML-based Feature Modeling of IoT-based family of applications

4.1. Feature Configuration Rules

Family of products consist on complete number of features and varied in different products. This process increase the reusability of existing resources in multiple applications [23]. The rules for composition of features in different applications are defined in previous studies such as FODA [12] and FORM [13]. Application features are 1) the subset of features that are defined by domain engineering i.e. family meta-model and 2) configuration of features in every application are according to constraints and relationships that are mentioned at family meta-modeling level. Therefore, feature modeling approach gives the complete process of features combinations and configuration rules for application development. Current feature modeling techniques for IoT applications do not support the reusability of resources where some common features exist. Feature modeling approaches for IoT-based applications where nested constraints and relationships exist are difficult to manage. FODA approach cover the cross-tree constraints (features include of exclude, incompatibility, require or from other part of feature model) between features.

XML language is efficient for modeling the cross-tree constraints as XML-schema defines the relationships and dependencies between family meta-model and application model. By using XML document four models are constructed for application development:

Constraint Meta Model: All mandatory and variable features with constraints and relationships of feature model are defined on early stage of meta-model. Therefore, during domain development all features need to develop independently and then configure them by end user perspective during application development [24].

Constraint Model: In this process, variable features need to extract from meta-model to manage the variability of feature model [25].

Constraint Model Compiler: In this process, variable model is compiled to check if there is any constraint violations or dependency issues. If any constraint violation occurs, then redefine the constraints in meta-model.

```
<xs:element name="IoT-PL">
  <xs:complexType>
    <xs:sequence maxOccurs="1" minOccurs="1">
      <!-- Child_A is mandatory feature -->
      <xs:element name="child_A" minOccurs="1" type="P" maxOccurs="1"/>
      <!-- Child_B is mandatory feature -->
      <xs:element name="Child_B" type="APP" minOccurs="1" maxOccurs="1"/>
      <!-- Child_C is optional feature -->
      <xs:element name="Child_C" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

Algorithm 1. XML Vocabulary Root Node To Child Node

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Application Constraint Checker: Final process of application development is to find the best features for specific requirements and no constraint violation occur.

5. XML-BASED IOT APPLICATIONS FEATURE MODELING:

To increase the reusability of existing resources of IoT-based family of products, proper and efficient modeling is required at early stage of features development. Avoidance of constraint violation is necessary and important from domain engineering to application engineering of IoT systems. Fig. 3 shows the complete process of XML-based feature modeling of IoT systems. Proposed process handles all features from domain modeling to application modeling and define the constraints and relationships. Development of domain model of IoT system by using XML language consists of two processes.

Domain model needs to identify all the features that are under the scope of IoT systems. In IoT-based feature models, contextual variability mentioned by this process such as heat sensor indoor and outdoor applications are different but same device. Meta-model contains the cardinality and dependency constraints to map the contextual variability in multiple applications under the user's perspectives. Other predefined variabilities such optional, OR group and alternative features, XML language supports and handle the constraints by using XML-schema vocabulary.

For development of IoT application, suitable features need to select according to end user requirements and also configuration rules that are defined in domain model. To select the suitable features XSLT file is generated from XML file and find the possible features for application development.

Mandatory and optional relationship exist from root node to next level of feature model. Algorithm 1 maps three features and relationships between root and children of root node. Mandatory

```
<xs:simpleType name="P">
    <xs:restriction base="xs:ID">
        <!-- Terminal nodes, only one
        feature is selected for final application -->
        <xs:enumeration value="Child_A_1"/>
        <xs:enumeration value="Child_A_2"/>
    </xs:restriction>
</xs:simpleType>
```

Algorithm 2. XML Vocabulary Parent Node To Leaf Nodes

```
<xs:complexType name="APP">
    <!-- Mandatory Alternative -->
    <xs:choice maxOccurs="1" minOccurs="1">
        <!-- Non-terminal nodes, only one
        feature is selected for final application -->
        <xs:element name="Child_B_1"/>
        <xs:element name="Child_B_2"/>
    </xs:choice>
</xs:complexType>
```

Algorithm 3. XML Vocabulary Parent To Leaf Nodes

features must be part of every application so the relationship with root node is minimum occurrence is 1 and maximum occurrence is also 1. However, optional features are not necessary to be part of every application so minimum occurrence is 0 and maximum occurrence is 1.

Further relationships such as alternative, OR group and optional are exist at second level of feature model. In Algorithm 2 we map alternative terminal features (leaf nodes) and apply restrictions that only one feature can be selected in one application development at same time.

Algorithm 3 maps the non-terminal features where alternative and OR group relationships exist. Minimum occurrence of APP child is 1 due to mandatory feature and maximum occurrence is 1 due to alternative relationship.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

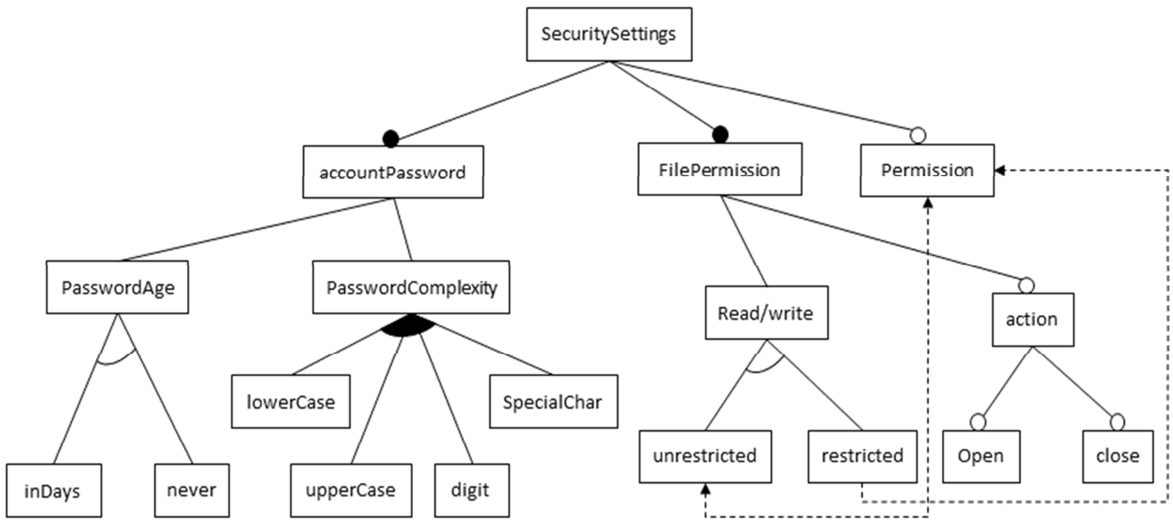

Figure 4. Security Setting Feature Model

Fig. 4. shows the real-time case study of security settings. We map in feature model for environmental variability management from domain engineering to application engineering.

XML-based feature modeling on real time case study "SecuritySettings" where multiple relationships (alternative, optional and OR group) exist. "PasswordAge" has two terminal features "inDays" and "never" where alternative relation exist i.e. only one feature is required in each application development, "PasswordComplexity" has four features where OR relationship exist i.e. at least one feature must be selected and maximum four. Cross Tree Constraint between unrestricted and permission shows that both require each other.

The suitable selection of features for each application development, XML-based feature modeling has applied due to protect the constraint violations.

Fig. 5. Shows the meta-model of given feature model in fig. 4. The cardinality relationships clearly indicate the possible features selection in every application development such as 0ne-to-one, one-to-many and many-to-many. In Fig. 5. APP, F11, F12 and FP show the constraints that we have applied in each group that have similar relationship. Doted line of "Permission" shows the optional feature that may or may not be part of application according to end user's requirement. Our proposed approach clearly indicates that there is no any constraint violation occurred and easy selection of features for final development of applications.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

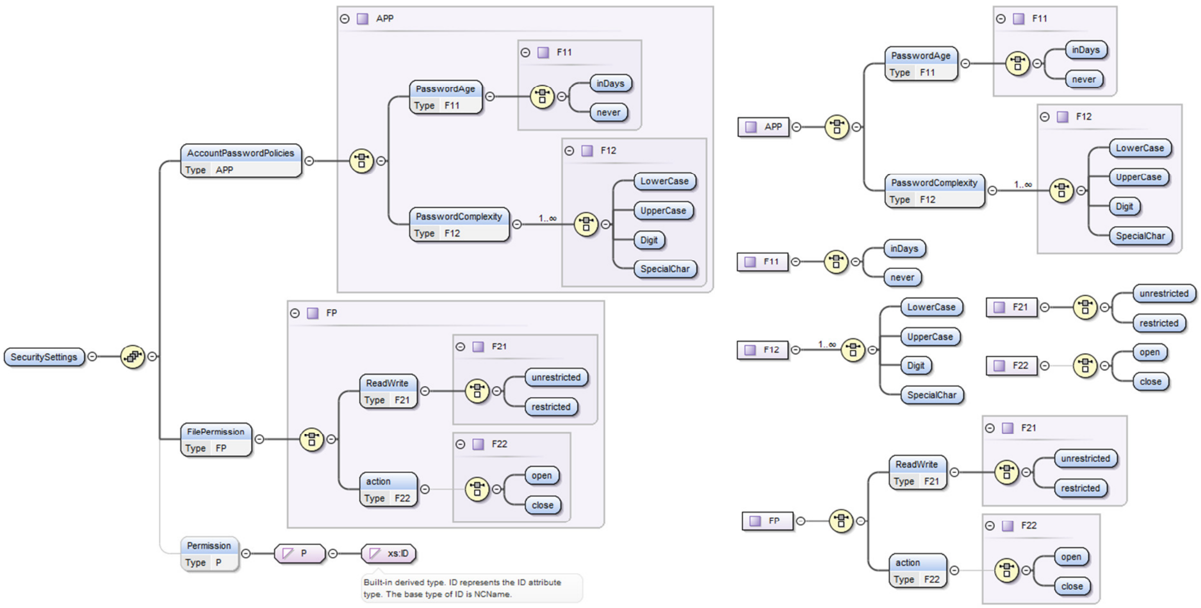

Figure 5. Meta-model of IoT-based application

6. CONCLUSION:

Internet of Things are used in every field for data management and communication. Functionality of IoT applications vary in different environments where family of application share common and variable resources. Therefore, variability management is very important and primary task for IoT applications development. In this paper, we have proposed simple and effective approach for IoT-based feature modeling by using XML documents i.e. XML-schema and XSLT. Our model shows zero constraint violation and applicable for small and large IoT-based feature models. XML-schema meta-model map all type of relationships (alternative, optional and OR group) and constraints (include and exclude) and finally generate model for application development.

ACKNOWLEDGMENT

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIP) (No. NRF-2016R1C1B2008624).

REFERENCES

[1] M. T. Lazarescu, "Design of a WSN platform for long-term environmental monitoring for IoT applications," *IEEE Journal on Emerging and Selected Topics in Circuits and Systems,* vol. 3, pp. 45-54, 2013.

[2] M. Endler, J.-P. Briot, V. P. Almeida, F. S. E. Silva, and E. Haeusler, "Towards Stream-based Reasoning and Machine Learning for IoT Applications," 2017.

[3] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises," *Business Horizons,* vol. 58, pp. 431-440, 2015.

[4] M.-S. Pan and S.-W. Yang, "A lightweight and distributed geographic multicast routing protocol for IoT applications," Computer Networks, vol. 112, pp. 95-107, 2017.

[5] I. F. Siddiqui, A. Abbas, and S. U.-J. Lee, "A HIDDEN MARKOV MODEL TO PREDICT HOT SOCKET ISSUE IN SMART GRID," *Journal of Theoretical and Applied Information Technology,* vol. 94, p. 408, 2016.

[6] C.-S. Kim, S.-K. Yoo, Y.-S. Jeong, Y.-W. Kim, and H.-K. Jung, "A Study on Cooperative System between Devices to Construct Internet of Things," *International Journal of u-and e-Service, Science and Technology,* vol. 8, pp. 343-350, 2015.

[7] A. Abbas, Z. Wu, I. F. Siddiqui, and S. U.-J. Lee, "An Approach for Optimized Feature Selection in Software Product Lines using Union-Find and Genetic Algorithms," *Indian Journal of Science and Technology,* vol. 9, 2016.

[8] A. Heuer and K. Pohl, "Structuring variability in the context of embedded systems during software engineering," in *Proceedings of the Eighth International Workshop on Variability Modelling of Software-Intensive Systems*, 2014, p. 21.

[9] H. Holdschick, "Challenges in the evolution of model-based software product lines in the automotive domain," in *Proceedings of the 4th International Workshop on Feature-Oriented Software Development*, 2012, pp. 70-73.

[10] A. Abbas, I. F. Siddiqui, and S. U.-J. Lee, "Multi-Objective Optimization of Feature Model in Software Product Line: Perspectives and Challenges," *Indian Journal of Science and Technology,* vol. 9, 2016.

[11] A. Abbas, I. F. Siddiqui, and S. U.-J. Lee, "GOAL-BASED MODELING FOR REQUIREMENT TRACEABILITY OF SOFTWARE PRODUCT LINE," *Journal of Theoretical and Applied Information Technology,* vol. 94, p. 327, 2016.

[12] K. C. Kang, S. G. Cohen, J. A. Hess, W. E. Novak, and A. S. Peterson, "Feature-oriented domain analysis (FODA) feasibility study," DTIC Document 1990.

[13] K. C. Kang, S. Kim, J. Lee, K. Kim, E. Shin, and M. Huh, "FORM: A feature-; oriented reuse method with domain-; specific reference architectures," *Annals of Software Engineering,* vol. 5, pp. 143-168, 1998.

[14] F. Loesch and E. Ploedereder, "Optimization of variability in software product lines," in Software Product Line Conference, 2007. SPLC 2007. 11th International, 2007, pp. 151-162.

[15] V. Cechticky, A. Pasetti, O. Rohlik, and W. Schaufelberger, "XML-based feature modelling," in *International Conference on Software Reuse*, 2004, pp. 101-114.

[16] H. Hartmann and T. Trew, "Using feature diagrams with context variability to model multiple product lines for software supply chains," in *12th Int. Software Product Line Conference, SPLC*, 2008, pp. 12-21.

[17] R. Ali, Y. Yu, R. Chitchyan, A. Nhlabatsi, and P. Giorgini, "Towards a unified framework for contextual variability in requirements," in *Software Product Management (IWSPM), 2009 Third Int. Workshop on*, 2009, pp. 31-34.

[18] N. Gligorić, I. Dejanović, and S. Krčo, "Performance evaluation of compact binary XML representation for constrained devices," in *Distributed Computing in Sensor Systems and Workshops (DCOSS), 2011 International Conference on*, 2011, pp. 1-5

[19] Ayala, M. Amor, L. Fuentes, and J. M. Troya, "A software product line process to develop agents for the iot," *Sensors,* vol. 15, pp. 15640-15660, 2015.

[20] A. Venčkauskas, V. Štuikys, N. Jusas, and R. Burbaitė, "Model-Driven Approach for Body Area Network Application Development," *Sensors,* vol. 16, p. 670, 2016.

[21] M. Tomlein and K. Grønbæk, "Semantic model of variability and capabilities of iot applications for embedded software ecosystems," in *Software Architecture (WICSA), 2016 13th Working IEEE/IFIP Conference on*, 2016, pp. 247-252.

[22] S. U.-J. Lee, "An Effective Methodology with Automated Product Configuration for Software Product Line Development," *Mathematical Problems in Engineering,* vol. 2015, 2015.

[23] A. Murguzur, R. Capilla, S. Trujillo, Ó. Ortiz, and R. E. Lopez-Herrejon, "Context variability modeling for runtime configuration of service-based dynamic software product lines," in *Procedings of the 18th International Software Product Line Conference: Companion Volume for Workshops, Demonstrations and Tools-Volume 2*, 2014, pp. 2-9.

[24] K. Czarnecki and S. Helsen, "Feature-based survey of model transformation approaches," IBM Systems Journal, vol. 45, pp. 621-645, 2006.

[25] F. Heidenreich, J. Kopcsek and C. Wende, “FeatureMapper: mapping features to models”. In Companion of the 30th international conference on Software engineering, ACM, May 2008, pp. 943-944.