The Design and Implementation of Ontology for Architecture Framework (ONT-DAF) in Military Domain

Young Min Kwon[1], Mye Sohn[1]* and Hyun Jung Lee[2]

[1]*Department of Industrial Engineering, Sungkyunkwan University,
300 Cheoncheon-dong, Jangan-Gu, Suwon, Korea*
[2]*Yonsei Institute of Convergence Technology, School of Integrated Technology
Yonsei University, 162-1, Songdo-dong, Yeonsu-gu, Incheon, Korea
{mini8716@;myesohn}@skku.edu, hjlee5249@yahoo.com
(*corresponding author)*

Abstract

Architecture framework (AF) is a guideline to define components needed to develop and operate enterprise architecture (EA), and to identify relationships among the components. There are many AFs to operate EA for of governments or businesses such as Zachman framework, DoDAF, TOGAF, FEAF, and TEAF. DoDAF is the most representative AF to support the development of the EA in the military domain. It is composed of eight viewpoints and 40 views that are affiliated with the viewpoints. However, views in DoDAF are structurally inter-related explicitly and/or implicitly. Similarly, data are twisted and complicated, too. So, developing an AF for a specific goal is going to be a project to be carried out over a long haul. To reduce the burden of its development, in this paper, we develop ONT-DAF (Ontology for DoDAF) that can infer inter-relationships like referential and transitive relationships and the sequences among the views. Furthermore, to promote reusability and consistency of the views and the data within an AF, we adopt the view-data separation strategy. To prove the effectiveness of ONT-DAF, we perform a case study.

Keywords: Enterprise Architecture, Architecture Framework, DoDAF 2.0, Ontology

1. Introduction

Architecture framework (AF) is a guideline to define components needed to develop and operate enterprise architectures (EAs), and to identify relationships among the components. There are many AFs to support the development of EAs for governments or businesses such as the Zachman framework, DoDAF, TOGAF, FEAF, and TEAF [3, 4, 5, 6]. In the military domain, the development and documentation of EAs must be required to assure interoperability and integratedness between legacy military systems (including weapon and information systems) and new ones when new defense acquisition projects are launched. In order to support the development of the EAs for the military domain, the guidelines such as DoDAF (U.S.), MODAF (U.K.), NAF (NATO), and MND-AF (ROK) are proposed by the governments of several countries [1, 6, 7, 8, 9]. Among them, DoDAF is the most representative AF that serves as the foundation of other AFs. So, we perform our research based on DoDAF 2.0 as the newest version of DoDAF. DoDAF is composed of a series of viewpoints, views, and models..

To develop an EA using DoDAF 2.0, system architects who are in charge of the development of the AF have to set a goal and decide a set of views that fit in with the goal of the AF to be developed. However, DoDAF 2.0 is composed of eight viewpoints and 40 views that are affiliated with the viewpoints. Furthermore, views in DoDAF 2.0 are explicitly and/or

implicitly entangled with inter-referential and/or transitive relationships. It is very difficult for the system architects to develop the AF by taking all relationships into consideration. Even though, to support the development of the AF, CASE tools are developed like System Architect, FrameWork, and Architect Framework [14, 15, 16], these CASE tools have limitations such as their inability to understand the implicated relationships among the views and/or data in the AF and/or to perform inference of inter-relationships between views and data for a specific AF. In this paper, we develop ONT-DAF (Ontology for DoDAF) to support the development of AFS in the military domain. It can infer inter-relationships and the sequences among the views and data in DoDAF 2.0.

The contributions of the developed ONT-DAF are as follows. ONT-DAF contributes to the reduction of the burdens of developing the AF by decreasing trials and errors. It improves the completeness of the AF. Furthermore, ONT-DAF can be expected to contribute to the evaluation of legacy AFs.

The outline of this paper is as follows. The next section describes related works. Section III presents the overall schema of ONT-DAF. A case study is performed in Section IV. Finally, Section V brings the conclusions and suggestions for further research.

2. Related Works

To develop complete AFs based on improved understandings on document-based AFs, ontology is widely being applied. AF-related ontology is classified into two types; data model ontology and ontology for AF components. The former ontology is to boost interoperability among data that are used to develop AFs, and the latter is used to share the understanding of the structures and components of AFs. The representative examples of the data model ontology are the DoDAF Meta model (DM2) and the MoDAF Meta model (M3) [6, 7]. They are provided to maintain consistency of data that are used for architecture descriptions and reference models, and composed of conceptual, logical and physical models. Each model describes data and constraints to satisfy the objectives of each model. The Zachman framework is constructed using applying ontology developed by Kang, et al. [2] and Zhuozhi Chen and Rob Pooley [10]. Kang, et al. adopts ontology to develop semantics-applied AFs to enhance understanding and communication between humans and systems. And Chen and Pooley build a meta-model using the BWW (Bunge-Wand-Weber) ontology and the Enterprise ontology (implemented on TOVE), which is applicable to the machine-processible Zachman framework. Turo Kilpel änen and Miika Nurminen [11] used the constructed Genre-Based Ontologies as a conceptual base for the EA description. Allemang et al. [12] developed the FEA-Reference Model Ontology (FEA-RMO) to share the meaning of FEA reference models. FEA-RMO supports useful queries to use information in FEA documents and provides a guideline for direction or the scope of modeling when the EA is constructed by seven models of FEA. Arseniev [13] provided ontology-based EA for the management of the information systems of a university. Oddrun Pauline Ohren [16] applied ontology to implement specified information in model-based AFs (FEAF, DoDAF etc.) for the comparative evaluation of AFs.

To sum it up, the ontology related to AFs is classified into data-level ontologies and ontologies supporting the construction of AFs that is applying semantics on AF. In this research, the proposed ONT-DAF also supports the systematic construction of AFs. To do this, data are separated from views to improve the data-sharing and to clarify the relationships among views and among data.

3. Overall Schema of ONT-DAF

3.1. Analysis of DoDAF

Before the development of ONT-DAF, we perform in-depth analysis of DoDAF 2.0, which is the root of almost every AF in the military domain. DoDAF 2.0 is comprised of eight viewpoints and 40 views which include directions, data, and rules that should be obeyed by the views. As depicted in Figure 1, the views in DoDAF 2.0 are intertwined with inter-referential and transitive relationships. To develop the AF precisely, first of all, system architects have to trace the relationships that exist among views, find the relevant view set, and decide the sequence of the development of the views in the set. However, finding the relevant views is not easy because the linkages among the views are very intricate. The analysis confirmed that inference tools are needed to interpret the intricate relationships among views.
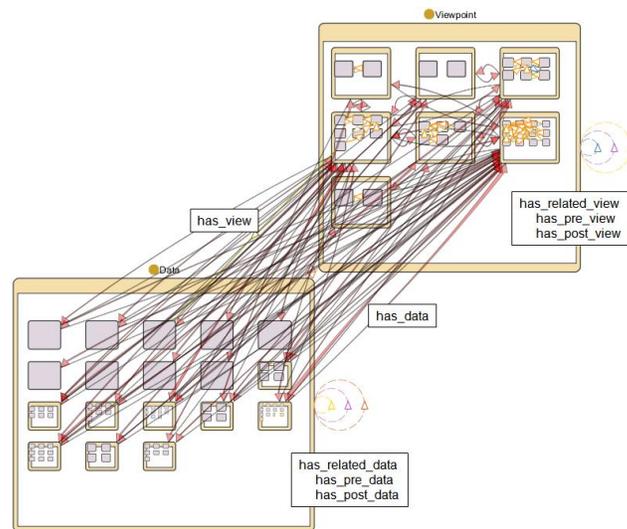

Figure 1. Referential and Transitive Relationship among Views

In addition, to assure the data consistency and reusability in an AF, we try to identify the referential relationships among data that are needed to develop the AF. An explicit definition of the relationships among views and/or data in DoDAF 2.0 can contribute to the precise development of the AF and the reduction of the burden of constructing the AF by decreasing trials and errors.

3.2. Design Principles of ONT-DAF

As we designed ONT-DAF, we pursued four main principles:

- *Expressional completeness*: ONT-DAF has to completely include the relationships and development sequences of views and/or data in DoDAF 2.0.

- *Relevance linkability*: Links of relevance among views and/or data in DoDAF 2.0 should be identified completely.

- *Data consistency*: Consistency among data should be maintained within an AF.

- *View and data reusability*: The data and views should be separated and reused within an AF.

To achieve the above principles, we establish the development strategy for ONT-DAF to separate data from views. To develop and document the views, relevant data with the views should be identified and applied. Unfortunately, the views and data are closely coupled in the document-based AFs. Each view only specifies data needed to develop and document it regardless of which views use the data. It causes data inconsistency in the developed AF and can hinder the reuse of the data. In this light, we separate data from views when we develop ONT-DAF. So, we accomplish the development principles named 'data consistency' and 'data reusability.'

3.3. Overall Structure of ONT-DAF

The overall structure of ONT-DAF is depicted in Figure 2. ONT-DAF is composed of a set of classes and properties that represent relationships among classes. There are five types of classes: '*TargetEA*,' '*Viewpoint*,' '*Data*,' '*Expression*,' and '*Reference*.' Each class has a lot of subclasses. As shown in Figure 2, class '*TargetEA*' relates to class '*Viewpoint*,' '*Data*' and '*Viewpoint*,' and '*Data*' have referential relationships via properties '*has_view*' and '*has_data*.'
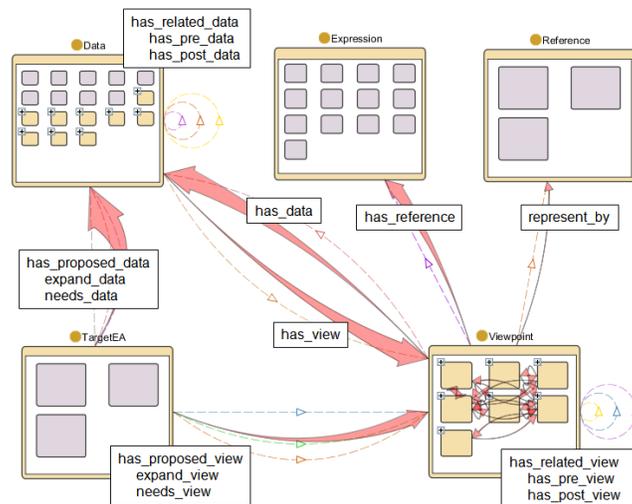
Figure 2. Overall Structure of ONT-DAF

Class '*Viewpoint*' is a set of classes that are mapped to eight viewpoints in DoDAF 2.0. Eight viewpoints have a set of instances that are mapped to views. Relationships among eight viewpoints are established by property '*has_related_view*.' '*has_related_view*' has domains (*rdfs:domain*) and ranges (*rdfs:range*) as viewpoints. Furthermore, '*has_related_view*' has two sub-properties named '*hasPreView*' and 'h*asPostView*.' These are used to set the development sequences between the views. IThe overall relationship among 'Viewpoint,' 'View,' and their properties are summarized in Figure 3.
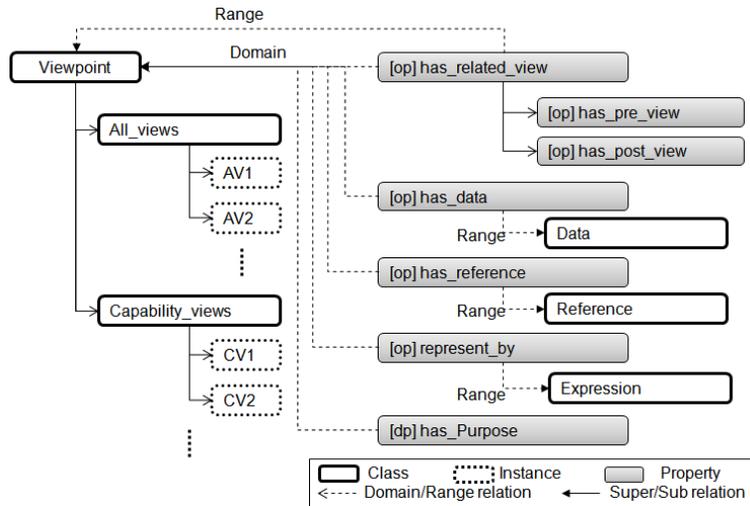
Figure 3. Overall Relationships among 'Viewpoint,' 'View,' there Properties, and Instances

Class '*Data*' represents the data that are needed in the modeling of views. In DoDAF DM, the data are classified into nine types. In addition, '*Resource Data*' is further classified into three types of sub-data. In ONT-DAF, class 'Data' is mapped to 12 data types. To treat the data flow among views, we add the extra-data named '*Data exchange*' and '*Data flow*.' Class '*Data*' is related to property '*has_related_data*.'

Class '*TargetEA*' is comprised of a set of '*views*' and '*data*' that are needed to develop the new AF. To implement the class '*TargetEA*,' we design three types of properties named '*has_proposed_data*,' '*has_proposed_view*,' and '*has_Proposed*.' Class '*Expression*' and '*Reference*' provide additional information about the development methods of views and reference models.

3.4. Reasoning Rules

To infer meaningful knowledge from Onto-DAF, we develop the SWRL (Semantic Web Rule Language)-based rule set and perform some inferences using JESS. In ONT-DAF, inferences are classified into four types. First, ONT-DAF recommends all the relevant views that are linked to views selected by system architects to develop a new AF (*TargetEA*). SWRL rule is depicted in Rule 1 and Rule 2. Additionally, Rule 1 and 2 can infer the development sequences among views.

(Views → Views)

Target EA(?x) ∧ needs_view(?x, ?y) ∧ Viewpoint(?y) ∧ has_pre_views(?y, ?z) →

TargetEA(?x) ∧ expand_view(?x, ?z) (Rule 1)

Target EA(?x) ∧ expand_view(?x, ?y) ∧ Viewpoint(?y) ∧ has_pre_views(?y, ?z) →

TargetEA(?x) ∧ expand_view(?x, ?z) (Rule 2)

We are developed additional rule set to infer the relationship from '*Views*' to '*Data*,' from '*Data*' to '*Data*,' and from '*Data*' to '*Views*.' However, we do not depict further due to the limit of the space.

4. Case Study

To show the effectiveness of ONT-DAF, we perform a case study. The target problem of the case study is selected from examples in the MODAF handbook (Version 1.2.003) [7]. As depicted in Figure 10, to provide guidance to bidders as part of the ITT for the SMS(s) project, the development of an AF is initiated. To do so, system architects identify views developed as follows: AV-1, AV-2, StV6, OV1s, and OV2. In our case study, we substituted StV6 for CV6 because the strategic view of MODAF is compatible with the capability view of DODAF 2.0.

The execution results of ONT-DAF based on the above example are summarized as follows. First, ONT-DAF recommends additional views developed aside from five views identified. To do so, ONT-DAF applies Rule. As a result, it additionally recommends OV3, OV5, CV1, and CV2. As the next step, ONT-DAF performs inference to decide the development sequences among views abiding by rule2. The execution results of ONT-DAF are summarized in Figure 4. To show the execution results in Figure 4, we adopt Onto-Viz as a plug-in application of Protégé

① Among nine views, the first developed by system architects is AV-1 and AV-2 because AV-1 does not have property 'hasPreView' and AV-2 has nothing to do with any views. From Figure 11, we know that data named 'Enterprise_Phase,' 'Enterprise_vision,' 'Architectural_Description,' and 'Terms_and _Abbreviations' are needed to develop AV-1 and AV-2.

② To find the following views of AV-1 and AV-2, ONT-DAF try to select views that have the property 'hasPostView.' In this example, ONT-DAF recommends OV-1 and CV-1, which are related to AV-1 with the property 'hasPostView.' To develop OV-1, the value of data 'Enterprise_Vision,' 'Information_Flow,' 'Enduring_Task,' 'System,' 'Node,' and 'Organization' should be required. Among them, the value of data 'Enterprise_Vision' is taken from AV-1. Furthermore, the value of data 'Enterprise_Goal,' 'Capability,' and 'Enduring_Task' is defined, too.

③ CV-2 is a post view of CV-1 and needs data 'Capability_Composition.' At this moment, data 'Capability_Composition' are derived from 'Capability' in CV-1.

④ OV-2 is developed using OV-1 and CV-2. It needs the value of data 'Needline,' 'Service,' 'Information_Security,' 'Logical_Flow,' and 'Operational_Activity.'

⑤ OV-3 is developed by the previous view OV-2 and needs data 'Information_Exchnage.' At this moment, 'Information_Exchnage' is derived from data 'Operational_Activity' and Node' in OV-2.

⑥ Data 'Operational_Activity_Flow' in OV-5 is developed using data 'Operational_Activity' in OV-3.

⑦ Finally, CV-6 is developed by OV-5 and CV-2. Furthermore, overall data in CV-6 are defined using 'Operational_Activity' and 'Enduring_Task' in OV-1 and OV-5, and 'Capability' in CV-2.

5. Conclusion and Further Works

We implemented the ONT-DAF that infers inter-relationships like inter-referential and transitive relationships and the development sequences among the views in DoDAF 2.0. To

develop the overarching ontology for DoDAF 2.0, we define four development principles: expressional completeness, relevance linkability, data consistency, and view and data reusability. Furthermore, we establish the development strategy for ONT-DAF, which is to separate data from views to achieve the above principles.

Contributions of this study are summarized as follows. First, ONT-DAF contributes to the reduction of the burden of developing AFs by decreasing of trials and errors. Second, it improves the completeness of the AFs. Finally, ONT-DAF can be expected to evaluate legacy AFs.
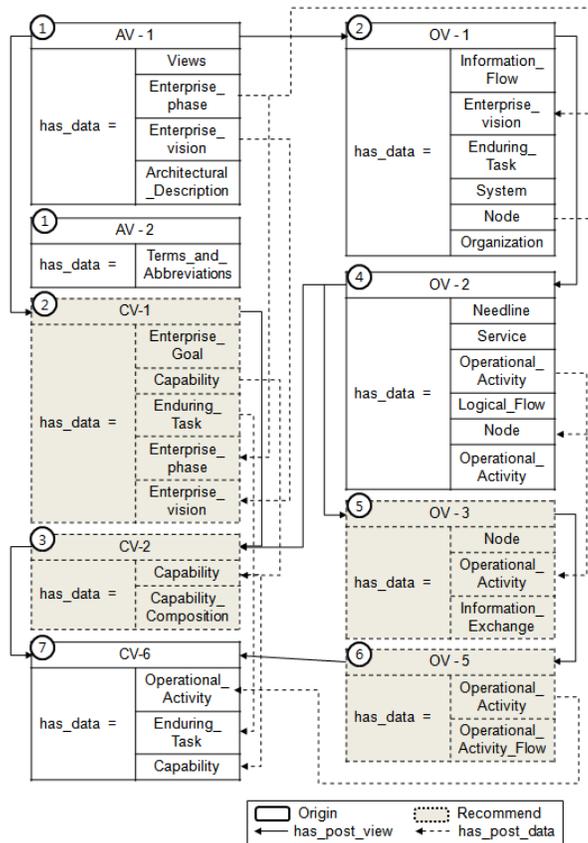
Figure 4. Inference Result of ONT-DAF

This research can be extended in several directions. We need to implement user interfaces that are applied to the development of the AFs. Currently, we use the plug-in programs of protégé Second, we will perform the evaluation on ONT-DAF to prove its adequacy and efficiency.

Acknowledgements

References

[1] L. Namkyu, S. Hyunsik, P. Sanggun and L. Taegong, "Trend of the Defense Architecture Framework and Developmental Direction of MND-AF", Journal of ITA, vol.6, no.2, **(2009)**, pp.171-182.

[2] K. Dongwoo, L. Jeongsoo, C. Sungchul and K. Kwangsoo, "An ontology-based Enterprise Architecture", Expert Systems with Applications, vol. 37, **(2010)**, pp. 1456–1464.

[3] TOGAF handbook v9.1, Open Group, **(2011)**.

[4] Federal Enterprise Architecture guidance, FEA Program Management Office, **(2007)**.

[5] Treasury Enterprise Architecture Framework, US Department of the Treasury, **(2000)**.

[6] DoD Architecture Framework v2.02, DoD Chief Information Officer of US, **(2010)**.

[7] MoD Architecture Framework v1.2.004, MoD of UK, **(2010)**.

[8] NATO Architecture Framework v3, NATO, **(2010)**.

[9] MND-AF v1.2, MND of Korea, **(2008)**.

[10] C. Zhuozhi and R. Pooley, "Rediscovering Zachman Framework using Ontology from a Requirement Engineering Perspective", Proceedings of 33rd Annual IEEE International Computer Software and Applications Conference, **(2009)** July 20 – 24; Washington, USA.

[11] T. Kilpel änen and M. Nurminen, "Applying Genre-Based Ontologies to Enterprise Architecture", Proceedings of 18th Australasian Conference on Information Systems, **(2007)** December 5-7; Toowoomba, Australia.

[12] D. Allemang, R. Hodgson and I. Polikoff, "Federal Enterprise Architecture reference model ontologies: FEA-RMO version 1.1", http://www.topquadrant.com/documents/TQFEA-RMO.pdf, **(2005)**.

[13] A. Marina, "Enterprise Architecture Implementation: Practical Steps Using Open Source Tools", http://net.educause.edu/ir/library/CMR0409.pps.

[14] M. Y. Hong, Introduction about Rational Software Architect, IBM, **(2010)**.

[15] Critical Success Factors for Selection of Enterprise Architecture Tools, 2e Consulting, Korea, **(2002)**.

[16] R. David, "Selecting an Enterprise Architecture tool", Antevorte consulting, **(2003)**.

[17] O. P. Oddrun, "Ontology for Characterising Architecture Frameworks", Workshop of Enterprise modeling and ontologies for Architecture Frameworks, **(2004)**.

Authors

Young Min Kwon

2010 Bachelor's degree in Sungkyunkwan University

2010 ~ Master course in Sungkyunkwan University

Research   interests Enterprise   Architecture,   Business   Process Reengineering, Project Management

Mye Sohn

1985 Bachelor's degree in Sungkyunkwan University

1988 Master degree in KAIST

2002 Ph. D in KAIST

1988 ~ 2004' Researcher of Korea Institute for Defense Analyses

2004 ~ Current Professor in Sungkyunkwan University

Research interests Semantic Web, Ontology, Semantic Web Service, Case-Based Reasoning, Ubiquitous Computing, Enterprise Architecture

Hyun Jung Lee

1991 Bachelor's degree in Ewha Womans University

1994 Master degree in Ewha Womans University

2004 Ph. D in KAIST

2006 ~ 2011 Research professor at Korea University Business School

2012 ~ Current Researcher at Yonsei Institute of Convergence Technology in Yonsei University

Research interests Decision Support System, Semantic Web, Ontology, Recommend System, Knowledge Management Systems