ResearchGate

See discussions, stats, and author profiles for this publication at: https://www.researchgate.net/publication/265084358

Data-Centric Enterprise Architecture

Article · August 2012

DOI: 10.5815/ijieeb.2012.04.08

CITATIONS
5

READS
550

2 authors, including:

Zeinab Rajabi
Malek Ashtar University of Technology
4 PUBLICATIONS **13** CITATIONS

SEE PROFILE

Some of the authors of this publication are also working on these related projects:

Project Applying ontology to Enterprise Architecture View project

All content following this page was uploaded by Zeinab Rajabi on 05 February 2019.

The user has requested enhancement of the downloaded file.

I.J. Information Engineering and Electronic Business, **2012, 4, 53-60**
Published Online August 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijieeb.2012.04.08

Data-Centric Enterprise Architecture

Zeinab Rajabi
Lecturer in University of applied science and technology Tehran, Iran
Rajabi.Ze@gmail.com
Maryam Nooraei Abade
Department of computer engineering, Science and Research Branch, Islamic Azad University Tehran, Iran
S.Nooraei@gmail.com

***Abstract* —** Enterprises choose Enterprise Architecture (EA) solution, in order to overcome dynamic business challenges and in coordinate various enterprise elements. In this article, a solution is suggested for the Enterprise Architecture development. The solution focuses on architecture data in the Enterprise Architecture development process. Data-centric architecture approach is preferred product-centric architecture approach. We suggest using Enterprise Ontology (EO) as context for collecting architecture data; Enterprise Ontology enhances quality of architecture data and lead to effective architecture results for decision-making. First, Enterprise is modeled using the ontology. Then how collecting Enterprise Architecture data based on the Enterprise Ontology is explained. Finally, the results and advantages of the solution are demonstrated.

Index Terms — Enterprise Architecture (EA), Enterprise Ontology (EO), Data-Centric Enterprise Architecture, Repository, Zachman Framework

I. INTRODUCTION

The Enterprise Architecture refers to a comprehensive description of all of the key elements and relationships that constitute an organization [1]. Enterprises pay remarkable attention to Enterprise Architecture to increase their flexibility and to adapt to business environment changes. By architecture aid, enterprises can achieve organizational integration and overcome business dynamics [2]. Enterprise Architecture refers to a discipline that attempts to integrate, govern and analyze enterprise elements. Alignment of elements creates synergy in achieving enterprise objectives.

Gruber [3] defined ontology as follows: "Ontology is an explicit specification of a conceptualization". In other research, Campbell and Schapiro [4] defined ontology as "Ontology consists of a representational vocabulary with precise definitions of the meanings of the terms of this vocabulary plus a set of formal axioms that constrain the interpretation and well-formed use of these terms". Ontology defined prevalent terms and concepts in a domain to exchange information and provided the way to share and to reuse knowledge among people and asymmetrical application systems. In other words, ontology provides a common understanding of a domain that facilitates communications between people and systems [5].

Existing Enterprise Architecture methodology has some major weak point. Firstly, there has not been common and exactly semantic understanding between human and system yet and it causes communication problems among humans or among systems or between human and system [6]. In addition, data collected in developing Enterprise Architecture are not based on a common definition of concepts and data communication; for example, planner has one definition for action and the developer has another definition; in some cases a specific data is called with different names. Such problems cause incoordination and lack of integrity of the architecture data. There is not a semantic foundation for collecting architecture data.

Another weak point is none-effective architecture results in decision-making. Architects have to accompany models and explain results to the managers in order to they make decisions according to results. Most of Enterprise Architecture frameworks and methodologies rely on traditional and routine products; Architects attempt to produce a product and present it for managers; for example, they produce business process model or system model. Since creating Enterprise Architecture models is expensive and lacks intrinsic value, it is desirable to only create Enterprise Architecture models that fit for purpose and support decision making well [7]. Often technical models, which are suitable technically are produced, while they are not suitable for decision-making. Enterprise Architecture products should be defined according to stakeholders' purposes and needs; therefore, to have qualified data as the architecture process backing is very helpful.

In this article, a solution is suggested for the Enterprise Architecture based on a conceptual model of Enterprise Ontology. First, conceptual model of Enterprise Ontology based on the Zachman Framework is proposed. Second, Data-Centric Enterprise Architecture based on Enterprise Ontology is proposed. Furthermore, the features of the Enterprise Architecture repository are examined in the new solution. This paper is organized as follows: Section 2 provides related works. Data-centric Enterprise Architecture development is suggested in Section 3. Finally,

Copyright © 2012 MECS

conclusions and future works are discussed in Section 4.

II. RELATED WORKS

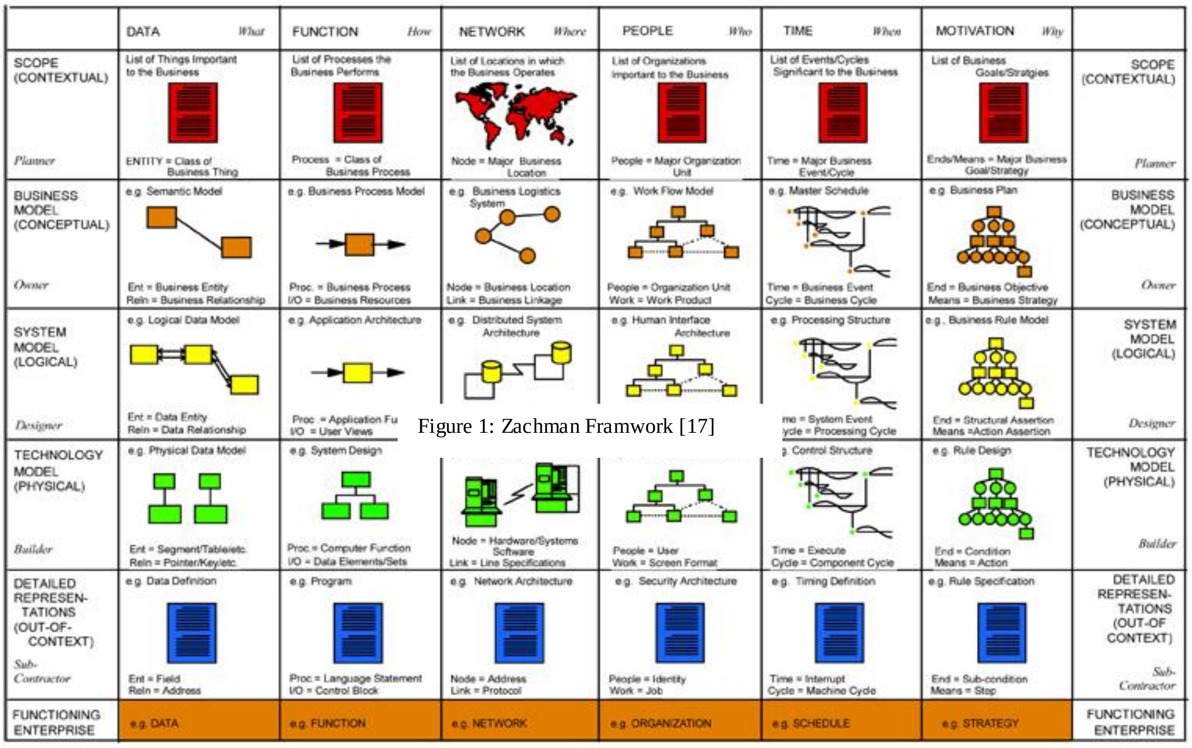
Allemang et al. [8] proposed FEA Reference Model Ontology (RMO) to have the shared meanings of Federal Enterprise Architecture (FEA) reference models, but it is nothing except FEA description in Web Ontology Language (OWL) and it only applies to FEA. In other research, Fuchs-Kittowski and Faust [7] presented semantic architecture tool (SemAT) for collaboration in Enterprise Architecture development and management. The Enterprise Architecture ontology model is used to design wiki-like collaborative environment. It presented only an ontology model for Enterprise Architecture and it did not point to how to develop Enterprise Architecture. Ghani et al. [9] offered a user-centric semantics-oriented Enterprise Architecture management and paid specific attention to users as architecture audience. He tries to provide meaningful information for the enterprise users, along with their needs and functional scope. Kang et al. [6] presented an ontology-based three-level Enterprise Architecture in order to solve the lack of semantic understanding in common among different systems and between human and system and among stakeholders in the enterprise. It emphasizes to use Semantics of Business Vocabulary and Business Rules (SBVR) and Fact-oriented approach. This approach focuses on formal level of ontology and it does not present a systematic method for Enterprise Architecture development. All researches attempt to use ontology in Enterprise Architecture, but none of them has presented a fundamental method for Enterprise Architecture development and has not paid attention to Enterprise Ontology as a baseline for Enterprise Architecture development. They also have not pointed to a general process for Enterprise Architecture development based on ontology.

2.1 Zachman Framework

There are several Enterprise Architecture frameworks such as Zachman [10], [11], FEA [3], TOGAF[12] and DODAF.[13],[14] Among them, all researchers agree on the Zachman as base for Enterprise Architecture research. Zachman Framework is a two-dimensional schema, used to organize the detailed representations of the enterprise. Figure 1 shows the Zachman Framework. The Zachman Framework summarizes a collection of perspectives involved in Enterprise Architecture. These perspectives are represented in a two-dimensional matrix that defines along the rows the type of stakeholders and with the columns the aspects of the architecture. While there is no order of priority for the columns of the Framework, the top-down order of the rows is significant to the alignment of business concepts and the actual physical enterprise. The rows of Zachman are described as follows: Planner's View (Scope), Owner's View (Enterprise or Business Model), Designer's View (Information Systems Model), Builder's View (Technology Model), Subcontractor View (Detailed Specifications). The columns of Zachman are described as follows: The data description (What), the function description (How), The Network description (Where), the people description (Who), the time description (When), and the motivation description (Why). The kinds of models or architectural descriptive representations are made explicit at the intersections of the rows and columns. Each cell which is the intersection of row (perspective) and column (abstraction) is made up of Enterprise Architecture products such as documents, models, graphics, and etc. [10], [11], [15]. Zachman is better than the other framework in the point of completeness of taxonomy. This is almost the entire focus of Zachman. None of the other framework focuses as much on this area [16]. The framework is a simple and logical structure for classifying and organizing the descriptive representations of an enterprise [15]. All requirements of Enterprise Architecture are collected in the Zachman grid at once. Zachman is the best starting point for determining required concepts in ontology development. The Zachman Framework is considered as a base for Enterprise Ontology in this article.

2.2 The Enterprise Ontology

Te important issue to achieve integrity and effective business planning is that all agents and stakeholders acquire a common understanding of different abstractions of an enterprise. Enterprise Ontology is provided for this purpose and it includes a set of well-formed terms that are used to describe enterprise widely while it covers the concepts of enterprise domain carefully. The set provides a common understanding of enterprise and can be stable a basis to specify requirements for end user applications. Therefore, perceptional errors are reduced in cases when terms are used with different interpretations, and it causes interaction improvement and facilitation between factors that is an important pace to promote efficiency. Accordingly, Enterprise Ontology works as a communicating medium between different people like users and developers in various enterprises, between people and systems, and between different systems [18], [19]. Toronto Virtual Enterprise (TOVE) [20], The Enterprise Ontology [21] and Context-Based Enterprise Ontology [18] are important researches in the field of Enterprise Ontology which presented models for Enterprise Ontology.



	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*	
SCOPE (CONTEXTUAL) *Planner*	List of Things Important to the Business ENTITY = Class of Business Thing	List of Processes the Business Performs Process = Class of Business Process	List of Locations in which the Business Operates Node = Major Business Location	List of Organizations Important to the Business People = Major Organization Unit	List of Events/Cycles Significant to the Business Time = Major Business Event/Cycle	List of Business Goals/Stratgies Ends/Means = Major Business Goal/Strategy	SCOPE (CONTEXTUAL) *Planner*
BUSINESS MODEL (CONCEPTUAL) *Owner*	e.g. Semantic Model Ent = Business Entity Reln = Business Relationship	e.g. Business Process Model Proc. = Business Process I/O = Business Resources	e.g. Business Logistics System Node = Business Location Link = Business Linkage	e.g. Work Flow Model People = Organization Unit Work = Work Product	e.g. Master Schedule Time = Business Event Cycle = Business Cycle	e.g. Business Plan End = Business Objective Means = Business Strategy	BUSINESS MODEL (CONCEPTUAL) *Owner*
SYSTEM MODEL (LOGICAL) *Designer*	e.g. Logical Data Model Ent = Data Entity Reln = Data Relationship	e.g. Application Architecture Proc. = Application Fu I/O = User Views	e.g. Distributed System Architecture	e.g. Human Interface Architecture	e.g. Processing Structure me = System Event ycle = Processing Cycle	e.g., Business Rule Model End = Structural Assertion Means =Action Assertion	SYSTEM MODEL (LOGICAL) *Designer*
TECHNOLOGY MODEL (PHYSICAL) *Builder*	e.g. Physical Data Model Ent = Segment/Table/etc. Reln = Pointer/Key/etc.	e.g. System Design Proc.= Computer Function I/O = Data Elements/Sets	Node = Hardware/Systems Software Link = Line Specifications	People = User Work = Screen Format	. Control Structure Time = Execute Cycle = Component Cycle	e.g. Rule Design End = Condition Means = Action	TECHNOLOGY MODEL (PHYSICAL) *Builder*
DETAILED REPRESEN-TATIONS (OUT-OF-CONTEXT) *Sub-Contractor*	e.g. Data Definition Ent = Field Reln = Address	e.g. Program Proc.= Language Statement I/O = Control Block	e.g. Network Architecture Node = Address Link = Protocol	e.g. Security Architecture People = Identity Work = Job	e.g. Timing Definition Time = Interrupt Cycle = Machine Cycle	e.g. Rule Specification End = Sub-condition Means = Step	DETAILED REPRESEN-TATIONS (OUT-OF CONTEXT) *Sub-Contractor*
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

Figure 1: Zachman Framwork [17]

TOVE is one of the pioneer projects in the field of Enterprise Ontology development. TOVE project has produced some subsets including; Organization [22], Resource [23], Activity [24] and etc. We compare TOVE concepts with Zachman columns in order to examine Enterprise Ontology adaptation with Zachman Framework. Considering Organization subset ontology, we place its core concepts in the Zachman columns. Table 1 shows that concepts are distributed in some of the Zachman columns. Activity and Resource subsets are placed in "How" and "Who" columns, respectively. As you see, some columns are left empty in table 1; Therefore, TOVE concepts do not cover all abstractions of Zachman Framework.

The second project is The Enterprise Ontology. In this project, a collection of business Enterprise terms and definitions was presented. Project core concepts are divided into five main parts, including: Activity, Organization, Strategy, Marketing, and Time [5]. We compare The Enterprise Ontology concepts with Zachman columns in order to examine Enterprise Ontology adaptation with Zachman Framework. In table 2, The Enterprise Ontology core concepts are selected; they are placed in the Zachman columns as representatives. Related concepts with Activity section are distributed among "How" and "Who" columns. None of the concepts can be found in the "Where" column; Therefore, its concepts do not cover all abstractions of Zachman.

The third research is a Context-Based Enterprise Ontology [18]; it aims to promote the understanding of the nature, purposes and meanings of the things in the enterprise, with providing concepts for representing things within contexts, and/or as contexts. It provides a unified view of the enterprise as an aggregate of contexts, as well as generic concepts for each of the contextual domains. The ontology advantage is its focus on defining context for enterprise activities. The project divided its concepts into seven domains: Purpose, Actor, Object, Facility, Location, and Time. We compare its concepts with Zachman columns in order to examine Enterprise Ontology adaptation with Zachman. As you see in table 3, the domains are adaptation of Zachman columns; therefore, Context-Based Enterprise Ontology concepts cover all Zachman abstractions, but uniformity has not been noticed. Some concept concepts are high abstraction level and some are in the low abstraction level. As a whole, none of the three Enterprise Ontology models are based on Zachman Framework.

Table 1 Examining TOVE adaptation with columns of Zachman Framework

DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
What	How	Where	Who	When	Why
Resource(sub ontology)	Activity Constraint Resource Communication_link Activity(sub ontology)		Organization Division Subdivision Team Organization_agent Role Skill Authority	Time(sub ontology)	Organization_goal Sub_goal Organization ontology

Table 2 Examining The Enterprise Ontology adaptation with columns of Zachman Framework

DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
What	How	Where	Who	When	Why
Entity Role Relationship Attribute Resource	Activity Activity Specification Execute Plan Process Specification		Person Organization Unit Actor Machine Skill Capability Authority	Time point Time Interval T-Begin T-End Time Line Calendar Date Duration	Purpose Critical Success Factor Objective Vision Mission Goal Decision Strategy

Table 3 Examining Context-Based Enterprise Ontology adaptation with columns of Zachman Framework

DATA	FUNCTION	NETWORK	PEOPLE	TIME	MOTIVATION
What	How	Where	Who	When	Why
Object domain Facility domain	Action domain	Location domain	Actor domain	Time domain	Purpose domain

III. DATA-CENTRIC ENTERPRISE ARCHITECTURE

3.1 Enterprise Ontology based on Zachman Framework

Enterprise Ontology does not aim to provide the ontology with so many concepts in enterprise domain but it detect core concepts and express how to development architecture by applying this ontology. We focus on how to use the Enterprise Ontology in Enterprise Architecture development.

Enterprise Ontology based on the whole Zachman abstractions (columns) are suggested. Moreover, in conceptual level, the Enterprise Ontology focuses on first and second rows (views) of Zachman but other rows will detail in formal level. These concepts provided planner's view and owner's view.

Ontology is defined in English and presented in meta models in a UML-based ontology representation language. UML class diagram is used for representing concepts and association, and aggregation relations are used for representing relations between concepts. Figure 2 shows ontology concepts and their relationships in the conceptual level. The core concepts of enterprise include:

- **Goal:** Desired state or condition that enterprises should achieve it.
- **Action:** A deed and action or event that is done, actions is divided into sub-actions, these further into sub-sub-actions etc. Parts of actions are functions, activities, tasks or operations. Decomposition aims at reaching the level of elementary actions, where it is not possible or necessary to further decompose.
- **Person:** Persons are the organization Staffs.
- **Role:** A collection of responsibilities that is relevant to a person.
- **Data:** Actions use data for execution and produce data when execution. Data is a type of resource.
- **Organizational unit:** An organization consists of organizational units. An organizational unit is composed of roles with the established supervision relationships.
- **Resource**: Actions can be produced or consumed resources in the enterprise. The resource has various types such as system and data.
- **System:** A system is a thing that is designed, built and installed to serve in a specific action affording a convenience, efficiency or effectiveness. Systems can be manual, computer aided, or computerized. The system is a type of resource.
- **Time:** Time Refers to when actions are executed.
- **Location:** Refer to place where actions are executed. A point or extent in space that may be referred to physically or logically.

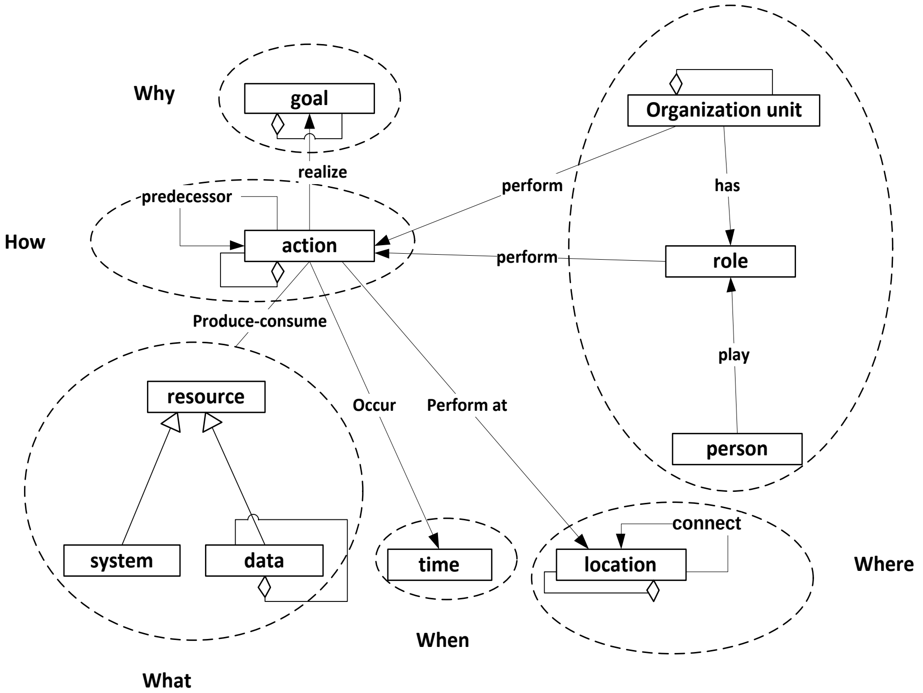


Figure 2: Conceptual model of Enterprise Ontology and how it adapted to the columns of the Zachman

Core concepts are considered, because concepts are provided conceptual level. Enterprise Ontology is not static, and defining concepts can be evolved. Enterprise Ontology can be improved and extended within the Enterprise Architecture project progress and based on organization type. Enterprise Ontology is an infrastructure to collect architecture data.

Furthermore, figure 2 shows the comparison between and with abstractions of Zachman model by closed lines, and indicates that core concepts are considered in a way that Zachman columns have been covered. The columns of the framework represent different abstractions or different ways to describe the real world. It indicates that the model covers the whole enterprise abstractions and suites for Enterprise Architecture.

3.2 Data-Centric Enterprise Architecture development

In this Section, how to use the ontology for Enterprise Architecture development is explained. This study introduces two approaches for Enterprise Architecture development process: *product centric* and *data centric*. Product centric approach means Enterprise Architecture development process which looks for creating the model as Enterprise Architecture product, and all efforts are towards producing traditional product. Furthermore, it is important to produce products such as process model, system model, data model and to present them to stakeholders. There is not a systematic method for collecting architecture data and data might be collected irregularly in different way. Data-centric approach means all efforts towards collecting data for Enterprise Architecture development process in accurate way. Architecture data are considered as input of Enterprise Architecture product. Accurate data influences on product quality remarkably. Furthermore, according to accurate data, architecture trends to “fit for purpose” product effectively. The architecture data support architecture process and “fit for purpose” product. Presence of qualified data as the architecture effort backing is very helpful to "fit for purpose" modeling. At first, the process focuses on collecting accurate architecture data and, then producing "fit for purpose" results.

Data-centric approach is proposed in this study. Enterprise Ontology provides common bases to conceptualize share and reuse information (achieved from enterprise) as architecture data. Architect requires common understanding of collecting data, and architecture participants should agree on the concepts. Common understanding eliminates semantic conflicts. Enterprise Ontology supports data-centric Enterprise Architecture. Enterprise Ontology is considered as base for collecting architecture data. First, required concepts and relations of the Enterprise Ontology are determined based on clear purpose of architecture.

Second, architecture data are collected based on concepts and relations.

After collecting data from enterprise, architects try to create models fit for stakeholder's need. In this solution, we can provide *views* and *analyses* as architecture results. The views are preliminary product in the Enterprise Architecture development process. Views are created based on collecting data and their relation. Sometimes, it is not essential to provide views as result to meet decision makers’ demands. Since architecture data were collected accurately, exactly and consistently, we can analyze them in a variety of ways;

for example statistic analysis, strategic planning analysis, that is another advantage of this approach.

Analysis can be performed by human experts and/or BI tools. Any type of models and tools can be used accompanying ontology and ontology is applied as supporting. Enterprise Architecture Decision makers' questions are answered. Customized analysis and "fit for purpose" results can be provided by using derived data. Of course, often we can analyze potentially based on collected data, but all of them are not required. If ontology implements in formal level, we can use ontology reasoning in this step. Ontology reasoning engine performs reasoning, regarded to defined concepts, their relationships, and ontology rules and constraints. Collected data based on ontology can be used as a basis for inference and reasoning.

3.3 Repository

The Enterprise Architecture repository is an automated model storage facility to keep track of architecture [25]. Treasury Enterprise Architecture Framework (TEAF) [26] defined repository as an information asset used to organize, store, access and share all Enterprise Architecture information, relationships between the information elements, and products. A repository is simply a database built specifically to store and relate the various kinds of documents and diagrams described in the Zachman Framework.

One of the weak points of Enterprise Architecture modeling is dispersal of products i.e. they are not integrated with the repository. Individual models are produced as products, and their consistency is too complexity. Furthermore, lack of integrity leads to change management for Enterprise Architecture products and descriptions, so tracking and updating is not possible. Integrated repository also increases reusability of architecture data. As a whole, Enterprise Architecture management is very difficult without an integrated repository.

In this approach, we have views and analyses as architecture results. The views are preliminary product in the Enterprise Architecture development process. Since data are collected consistency, integrity and precision, the views that based on data also have consistency, integrity and precision features. The views relate tightly to architecture data. Therefore, Enterprise Ontology provides context for Enterprise Architecture repository. In general, since this solution introduced Enterprise Ontology as a foundation for Enterprise Architecture, it provides appropriate Enterprise Architecture repository; the features include:

- Consistency, integrity, precision, concordance and interoperability between architecture data
- Consistency and integrity between architecture views as artifacts and descriptions
- Predicting enterprise changes effects on artifacts
- Facilitation of updating architecture artifacts
- Increasing reusability of architecture data
- Facilitation of Keeping track of plan transition from "As-Is" to "To-Be" architecture.

Ontologist updates Enterprise Ontology based on architecture requirements and/or enterprise conditions, and changes base of repository. Architecture data also is entered into repository by architects or sometimes by personnel or stakeholders; and updates data of repository. After collecting data, architects provide *views* and *analysis* based on architecture data in the repository and present to stakeholders. If Enterprise Ontology is implemented by ontology language in formal level, Ontology reasoning engine can reason on architecture data. Stakeholders use repository to get views. Figure 3 shows repository, based on Enterprise Ontology and their use cases.

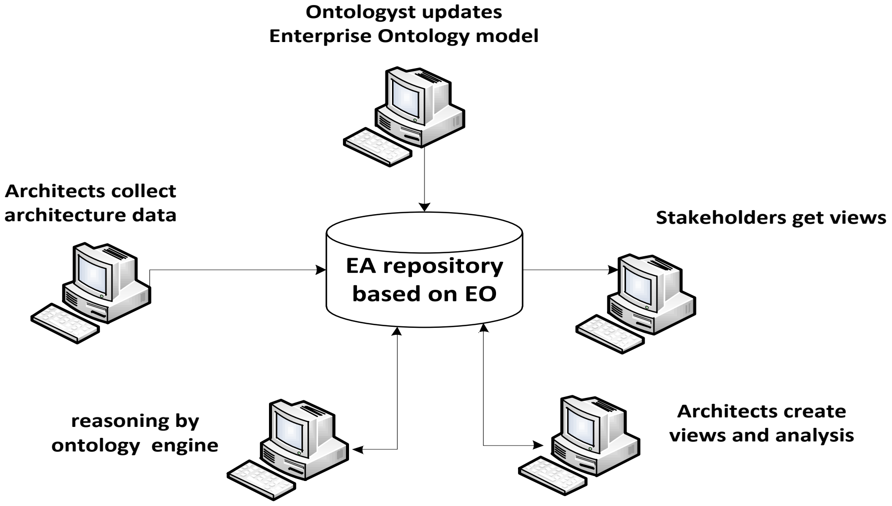


Figure 3: Enterprise Architecture repository based on Enterprise Ontology and use cases

V CONCLUSIONS

Enterprise Ontology eliminates semantic inconsistency in the enterprise, create communication channel for all participants of Enterprise Architecture, and also provide context and foundation for collecting architecture data from enterprise. Collecting data based on Enterprise Ontology causes to achieve accurate architecture data from enterprise. The solution considers data-centric architecture instead of product-centric architecture. Enterprise Architecture development process emphasizes collecting accurate data with qualified structure and recognizing decision makers' and stakeholders' needs and purpose of architecture that helps to create "fit for purpose" views and analyses as architecture results. Accurate data support "fit for purpose" modeling as product and an architect will move toward fit for purpose results well. we can use advantages of ontology in the Enterprise Architecture, and get accurate architecture data and "fit for purpose" modeling. Future researches remain as follows: Applying solution in a case study, selecting appropriate language for Enterprise Ontology according architecture requirements, extending ontology concepts.

REFERENCES

[1] Harmon P. Developing an Enterprise Architecture, Business process trends:Whitepaper. 2003, [online]. Available: http://www.bptrends.com/resources_publications.cfm?publicationtypeID=DFC61D66-1031-D522-3EBDAB1F65A451AA.

[2] Chief Information Officer Council. A Practical to Guide Federal Enterprise Architecture Version 1.0. 2001, [online]. Available: http://www.gao.gov/bestpractices/bpeaguide.pdf.

[3] Gruber T. Towards Principles for the Design of Ontologies Used for Knowledge Sharing. Knowledge Systems Laboratory, Stanford University, Technical Report KSL93-04, 1993.

[4] Campbell E, Schapiro S C. Ontologic Mediation: An Overview. in Proceedings of the IJCAI Workshop on Basic Ontological Issues in Knoweldge Sharing, Menlo Park CA, USA: AAAI Press, 1995.

[5] Uschold M, Gruninger M. Ontologies Principles Methods and Applications. The University of Edinburgh, 1996.

[6] Kang D, Lee J, Choi S. An ontology-based Enterprise Architecture. Expert Systems with Applications, 2010, 37:1456–1464.

[7] Fuchs-Kittowski F, Faust D. The Semantic Architecture Tool (SemAT) for Collaborative Enterprise Architecture Development. in Proceedings of CRIWG, Springer-Verlag Berlin Heidelberg, 2008, 151-163.

[8] Allemang D, Hodgson R, Polikoff I. Federal Enterprise Architecture reference model ontologies: FEA-RMO version 1.1. ,2005, [online]. Avalable: http://www.topquadrant.com/documents/TQFEA-RMO.pdf.

[9] Ghani I, Lee C Y, Juhn S H. Semantics-oriented approach for information interoperability and governance: towards user-centric enterprise architecture management. Zhejiang University-SCIENCE C (Computers & Electronics), 2010, 11(4): 227-240.

[10] Sowa F , Zachman J A. Extending and formalizing the framework for information systems architecture. IBM Systems Journal, 1992, 31(3):590-616.

[11] Zachman John P. A Framework for Information Systems Architecture. IBM Systems Journal, 1987, 26(3): 276-292.

[12] TOGAF version9 Enterprise Edition http://pubs.opengroup.org/architecture/togaf9-doc/arch/chap05.html

[13] Department of Defense. The Essential DoDAF: A User's Guide to Architectural Description Development Version 0.5. 2009, [Online]. Available: http://cio-nii.defense.gov/sites/dodaf20/products/Essential_DoDAF_V2-0_2009-05-20.pdf.

[14] Department of Defense. The DoDAF Architecture Framework Version 2.02. 2010, [online]. Available: http://cio-nii.defense.gov/sites/dodaf20/products/DoDAF_v2-02_web.pdf

[15] Wikipedia, http://en.wikipedia.org/wiki/Enterprise_architecture

[16] Sowa F , Zachman J A. Extending and formalizing the framework for information systems architecture. IBM Systems Journal, 1992, 31(3):590-616.

[17] Zachman John P. The Zachman Framework Evolution. Zachman International website, 2009-2011.

[18] Leppänen M. A Context-Based Enterprise Ontology. in Proceedings of 10th International Conference on Business Information Systems BIS 2007, Springer Berlin/Heidelberg, 2007, 273-286.

[19] Uschold M, King M, Moralee S, Zorgios Y. The Enterprise Ontology. Artificial Intelligence Application Institute (AIAI),The Univercity of Edinburgh,1996.

[20] Enterprise Modelling http://www.eil.utoronto.ca/enterprise-modelling/index.html

[21] The Enterprise Ontology http://www.aiai.ed.ac.uk/project/enterprise/enterprise/ontology.html

[22] Fox M S, Barbuceanu M, Gruninger M, Lin J. An Organization Ontology for Enterprise Modelling. in Proceedings of Simulating Organizations: Computational Models of Institutions and Groups, 1998, 131-152.

[23] Fadel F G, Fox M S. Gruninger M. A Generic Enterprise Resource Ontology. in Proceedings of the 3rd Workshop on Enabling Technologies: Infrastructure forCollaborative Enterprise, USA, 1994, 117-128.

[24] Gruninger M, Fox M S. An Activity Ontology for Enterprise Modelling. Submitted to Workshop on Enabling Technologies - Infrastructures for Collaborative Enterprises, West Virginia University, 1994.

[25] Spewak S H, Hill S C. Enterprise Architecture Planning, Developing a Blueprint for Data, applications, and technology. John Wiley & Sons, 1992.

[26] Chief Information Officers Council. Treasury Enterprise Architecture Framework Version 1. Department of the Treasury, 2000, [online]. Avalable: http://www.ea.or.kr/upload_files/TEAF1.0.pdf.

Z.Rajabi received her BS at Isfahan University of Technology in 2005 and MSc in Information Technology Engineering (eCommerce) at Nooretuba University in 2012. Her research interests include Information System, Enterprise Modeling and Enterprise Integration.

M.Nooraei is a Ph.D. Candidate in Computer Engineering Department, at Islamic Azad University Science and Research Branch, Tehran, Iran. She is a faculty member in the department of Computer Engineering of Islamic Azad University.

View publication stats