Expert Systems with Applications 37 (2010) 1456–1464

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

An ontology-based Enterprise Architecture

Dongwoo Kang [a], Jeongsoo Lee [b], Sungchul Choi [b], Kwangsoo Kim [b,*]

[a] Integrated Operations & Management Research Department, Network R&D Laboratory, KT, 463-1, Jeonmin-dong, Yuseong-gu, Daejeon 305-811, Korea
[b] Department of Industrial and Management Engineering, Pohang University of Science and Technology, San 31, Hyoja-dong, Namgu, Pohang 790-784, South Korea

ARTICLE INFO

Keywords:
Enterprise Architecture
Ontology-based Enterprise Architecture
Enterprise ontology

ABSTRACT

The Enterprise Architecture refers to a comprehensive description of all of the key elements and relationships that make up an organization [Harmon, P. (2003). Developing an Enterprise Architecture, Business process trends: Whitepaper]. Through the Enterprise Architecture, enterprises can implement enterprise integration to cope with dynamically changing business environment. Existing Enterprise Architectures, however, lack of semantics for humans and systems to understand them exactly and commonly, which causes communication problems between humans or between systems or between human and system. These communication problems keep enterprises from implementing integration and collaborating with other enterprises. In order to solve this problem, the ontology-based Enterprise Architecture is suggested in this paper. The Enterprise Architecture ontology is composed of ontologies in three levels. Ontologies of business terms are in the first level, ontologies of Enterprise Architecture components are in the second level, and ontologies of relationships among Enterprise Architecture components are in the top level. The ontologies of business terms are defined in the approach of the WordNet, and the ontologies of Enterprise Architecture components and relationships of them are defined in the approach of the SBVR. Through these ontologies, it is expected that humans and systems can understand Enterprise Architectures exactly and commonly, which supports integrations in enterprises and collaborations between enterprises.

1. Introduction

Today, enterprises make efforts to cope with dynamically changing business environment. Among the various efforts, enterprises pay attention to Enterprise Architecture (EA). The Enterprise Architecture refers to a comprehensive description of all of the key elements and relationships that make up an organization (Harmon, 2003). In other words, it is thought of the enterprise blueprint which systematizes constituent units of an enterprise, such as business processes, organizations, data, and information technologies. The Enterprise Architecture enables enterprise members to understand detailed structure and components of the enterprise and how they work together. Brown asserts six common values of the Enterprise Architecture as follows (Brown, 2004):

(1) Readily available documentation of the enterprise.
(2) Ability to unify and integrate the business processes across the enterprise.
(3) Ability to unify and integrate data across the enterprise and to link with external partners.
(4) Increased agility by lowering the complexity barrier.
(5) Reduced solution delivery time and development costs by maximizing reuse of enterprise model.
(6) Ability to create and maintain a common vision of the future shared by both the business and IT communities, driving continuous business/IT alignment.

Current Enterprise Architectures, however, lack semantics so that humans and systems cannot understand the Enterprise Architectures exactly and commonly. The lack of semantics causes problems in the point of the first and sixth Enterprise Architecture values above. If the Enterprise Architecture, namely, documentation of the enterprise as stated in the first value, cannot be understood exactly and commonly by enterprise members, machines, suppliers, customers, and others, the Enterprise Architecture cannot but lose the value. This causes communication problems between humans or between systems or between human and system. For example, although a process of an enterprise is defined systematically, if a process manager, a process operator and systems understand details of the process incorrectly and differently, the process cannot be executed correctly and effectively. The communication problems keep enterprises from implementing integration and collaborating with others, which is against the second and third value of the Enterprise Architecture. In order to solve these problems from the lack of semantics of the Enterprise Architecture, an ontology-based Enterprise Architecture is suggested in this paper.

* Corresponding author. Tel.: +82 54 279 2195; fax: +82 54 279 5998.
E-mail addresses: dwkang@kt.com (D. Kang), jsrhyme@postech.ac.kr (J. Lee), blissray@postech.ac.kr (S. Choi), kskim@postech.ac.kr (K. Kim).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.06.073

An ontology is an explicit specification of a conceptualization (Gruber, 1993). The ontology includes definitions of concepts and an indication of how concepts are inter-related which collectively impose a structure on the domain and constrain the possible interpretations of terms (Uschold, 1998). The ontology is used to improve communication between either humans or computers (Uschold & Jasper, 1999). In the concrete, the ontology is used to assist in communication between human agents, to achieve interoperability among computer systems, or to improve the process and/or quality of engineering software systems (Uschold & Jasper, 1999). Therefore, if the Enterprise Architecture is defined in the form of the ontology, the Enterprise Architecture can get the abilities of the ontology so that it can solve the above communication problems. Concretely, stakeholders of an enterprise can share exact and common understandings about the Enterprise Architecture and systems under Enterprise Architecture can achieve interoperability. This leads to support internal enterprise integration and external cooperation with other enterprises.

In this paper, the Enterprise Architecture is modeled with ontologies in three levels. In the first level, business terms are defined with ontologies. Business terms are important in the point of the Enterprise Architecture. The Enterprise Architecture, as an enterprise blueprint, must manage business terms used by stakeholders and systems of an enterprise, and, reversely, the Enterprise Architecture is described with these business terms. These business terms must be understood by humans and systems exactly and commonly. But, enterprises have communication problems caused by usage of implicit and different business terms, which hinders integration of an enterprise and collaboration among enterprises. To solve the communication problems, business terms of the Enterprise Architecture is defined with ontologies, and the Enterprise Architecture is also described with these common business terms. In the second level, ontologies are used in order to describe components of the Enterprise Architecture exactly and commonly. Existing Enterprise Architectures have defined their components with natural languages so that they can be misunderstood by humans and systems. This misunderstanding problem hinders integration of an enterprise and collaboration among enterprises. In the third level, relationships among components of the Enterprise Architecture are described based on ontologies for common understanding. Existing Enterprise Architectures have modeled relationships among components of the Enterprise Architecture, but consideration for common understanding of them is insufficient. For example, relationships among strategies consist of super-sub relationship, same relationship, opposite relationship, and others, but it is vague what the super-sub relationship is, which character the relationship has, and so on. Naturally, people can understand the relationship in their own way after they interpret the model and search for related documents written in natural languages. But the model is almost system-friendly and does not have semantics and it is annoying to search other documents.

This paper is organized as follows: Section 2 provides related works. In Section 3, Zachman Enterprise Architecture is introduced as a base Enterprise Architecture in this paper. The ontology-based Enterprise Architecture is suggested in Section 4. Finally, conclusions and future works are discussed in Section 5.

2. Related works

Zachman (1987), FEA Program Management Office (2007), US Department of Defense (2007) are representative Enterprise Architectures. They, however, lack in consideration for common understanding of Enterprise Architectures. Although FEA-Reference Model Ontology (FEA-RMO) (Allemang, Hodgson, & Polikoff, 2005) is proposed in order to share meanings of FEA reference models, it is nothing but the model which describes FEA reference models with Web Ontology Language (OWL). It is only for FEA reference models and is short of concrete method to share common meanings of Enterprise Architecture components. Arseniev suggests ontology-based Enterprise Architecture to manage information systems of University of California, UC Irvine. But, it lacks in common understanding of business terms and does not deal with various relationships among Enterprise Architecture components. In addition, it has a weak point that it is specific to information systems of University of California, Irvine.

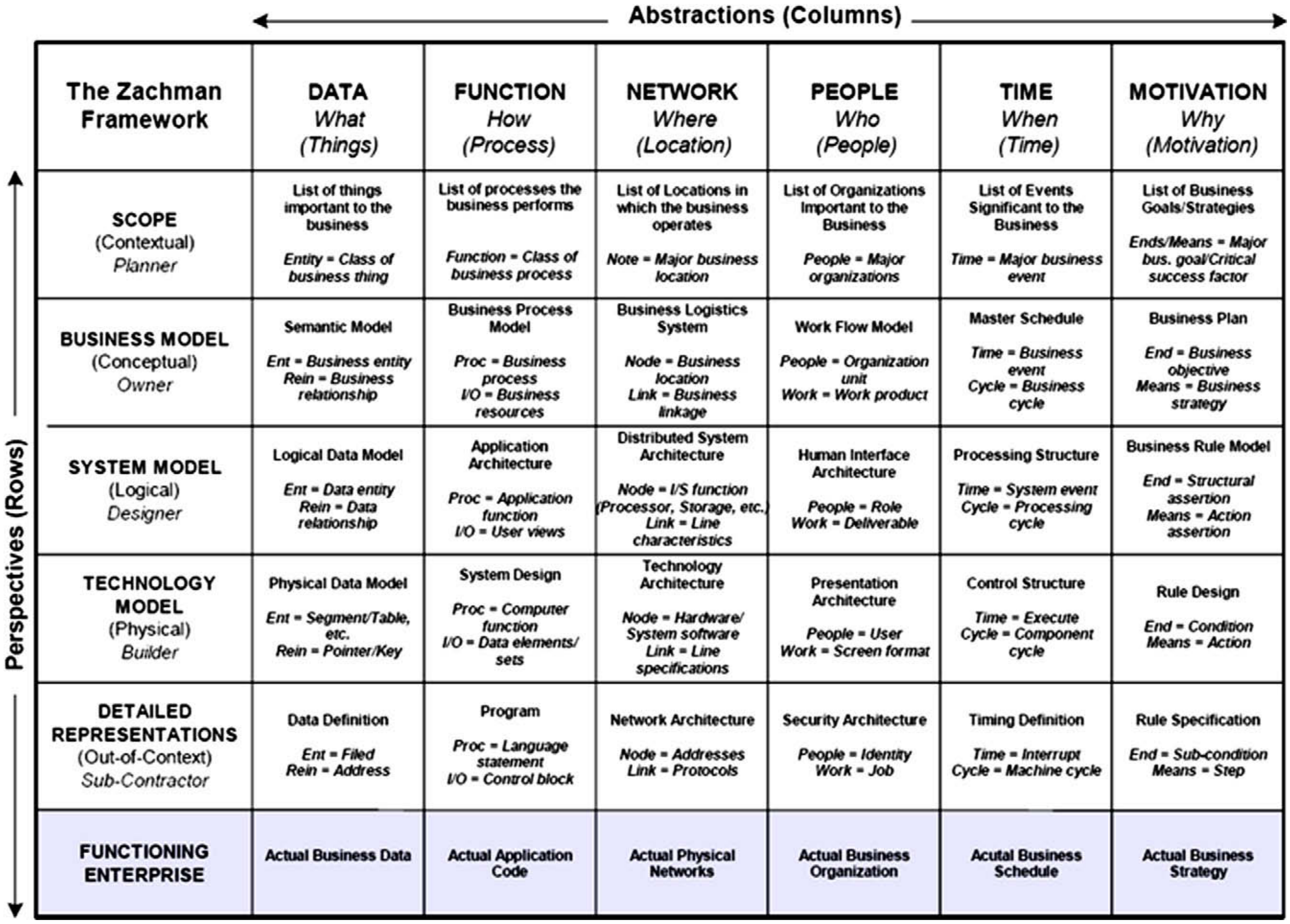
There are many researches on enterprise or business ontologies such as Enterprise Integration Laboratory (2002), Enterprise Ontology(EO) (Uschold, King, Moralee, & Zorgios, 1998), Core Enterprise Ontology(CEO) (Bertolazzi, Krusisch, & Missikoff, 2001), and so on. Leppänen suggests context-based enterprise ontology which aims to advance the understanding of the nature, purposes and meanings of things in enterprises (Leppänen, 2007). Missikoff and Taglino define business and enterprise ontology with SymOntoX which is a software prototype for the management of domain ontologies (Missikoff & Taglino, 2002). These researches, however, do not consider the Enterprise Architecture. Enterprise Architectures are the basic description for enterprises. Without Enterprise Architectures, it is difficult to grasp the whole features of enterprises. Therefore, enterprise ontologies without consideration for Enterprise Architectures are liable to be incomplete and fail to describe whole enterprises.

Researches in specific domains, such as National Institute of Standards and Technology (2005), Object Management Group (2008) and van Renssen (2005), provide methods to support clear communication among applications in each domain. They, however, do not deal with the whole enterprise model so that they cannot support share of meanings for various enterprise components totally.

3. Enterprise Architecture

In order to define Enterprise Architecture with ontologies, framework to define Enterprise Architecture is required. There are various Enterprise Architecture frameworks. Among them, the Zachman framework (Zachman, 1987; Zachman Institute for Framework Advancement) is selected as a base Enterprise Architecture framework to define the ontology-based Enterprise Architecture of this paper. The Zachman framework is created by John Zachman, and is valued as the beginning and standard Enterprise Architecture framework. The Zachman framework categorizes components of enterprises according to five perspectives and six abstractions as shown in Fig. 1 (Zachman Institute for Framework Advancement). Each cell which is intersection of perspective row and abstraction column is made up of Enterprise Architecture products such as documents, models, and so on. Although the Zachman framework lacks in modeling for detailed Enterprise Architecture components and relationships among them and does not provide concrete implementing method, it is valuable in the point that it presents general framework which every enterprise can use to build its Enterprise Architecture.

In this paper, it is assumed that Enterprise Architecture is modeled according to the Zachman Enterprise Architecture framework. Under this assumption, the Enterprise Architecture without common understanding is remodeled with ontologies so that humans and systems can understand the Enterprise Architecture commonly and exactly. First, Enterprise Architecture components, that is, Enterprise Architecture products in each cell as shown in Fig. 1 are defined with ontologies. Second, relationships of Enterprise Architecture components, that is, relationships of Enterprise Architecture products in the cell or between the cells are described with


Abstractions (Columns)
Perspectives (Rows)

The Zachman Framework	DATA What (Things)	FUNCTION How (Process)	NETWORK Where (Location)	PEOPLE Who (People)	TIME When (Time)	MOTIVATION Why (Motivation)
SCOPE (Contextual) Planner	List of things important to the business Entity = Class of business thing	List of processes the business performs Function = Class of business process	List of Locations in which the business operates Note = Major business location	List of Organizations Important to the Business People = Major organizations	List of Events Significant to the Business Time = Major business event	List of Business Goals/Strategies Ends/Means = Major bus. goal/Critical success factor
BUSINESS MODEL (Conceptual) Owner	Semantic Model Ent = Business entity Rein = Business relationship	Business Process Model Proc = Business process I/O = Business resources	Business Logistics System Node = Business location Link = Business linkage	Work Flow Model People = Organization unit Work = Work product	Master Schedule Time = Business event Cycle = Business cycle	Business Plan End = Business objective Means = Business strategy
SYSTEM MODEL (Logical) Designer	Logical Data Model Ent = Data entity Rein = Data relationship	Application Architecture Proc = Application function I/O = User views	Distributed System Architecture Node = I/S function (Processor, Storage, etc.) Link = Line characteristics	Human Interface Architecture People = Role Work = Deliverable	Processing Structure Time = System event Cycle = Processing cycle	Business Rule Model End = Structural assertion Means = Action assertion
TECHNOLOGY MODEL (Physical) Builder	Physical Data Model Ent = Segment/Table, etc. Rein = Pointer/Key	System Design Proc = Computer function I/O = Data elements/sets	Technology Architecture Node = Hardware/System software Link = Line specifications	Presentation Architecture People = User Work = Screen format	Control Structure Time = Execute Cycle = Component cycle	Rule Design End = Condition Means = Action
DETAILED REPRESENTATIONS (Out-of-Context) Sub-Contractor	Data Definition Ent = Filed Rein = Address	Program Proc = Language statement I/O = Control block	Network Architecture Node = Addresses Link = Protocols	Security Architecture People = Identity Work = Job	Timing Definition Time = Interrupt Cycle = Machine cycle	Rule Specification End = Sub-condition Means = Step
FUNCTIONING ENTERPRISE	Actual Business Data	Actual Application Code	Actual Physical Networks	Actual Business Organization	Acutal Business Schedule	Actual Business Strategy

Fig. 1. Zachman Enterprise Architecture framework.

ontologies. As shown in Fig. 1, it seems that the components are defined independently, but a component has complex relationships with other components in a same cell or different cells.

Fig. 2 shows concept definition of '*HR Strategy Process*', one of the components of Federal Enterprise Architecture which is based on the Zachman framework. Fig. 3 shows a part of concept definition for '*Establish HR Policy and Practices Process*' in '*HR Strategy Process*'. As shown in the figures, the concept definitions are described with natural language so that there is much room for misunderstanding or vague understanding. Fig. 4 illustrates a metamodel in order to model relationships of Enterprise Architecture components. As shown in the figure, it is difficult to grasp the meanings of '*MapsTo*', '*Implements*', '*Automates*', and others which describe the relationships.

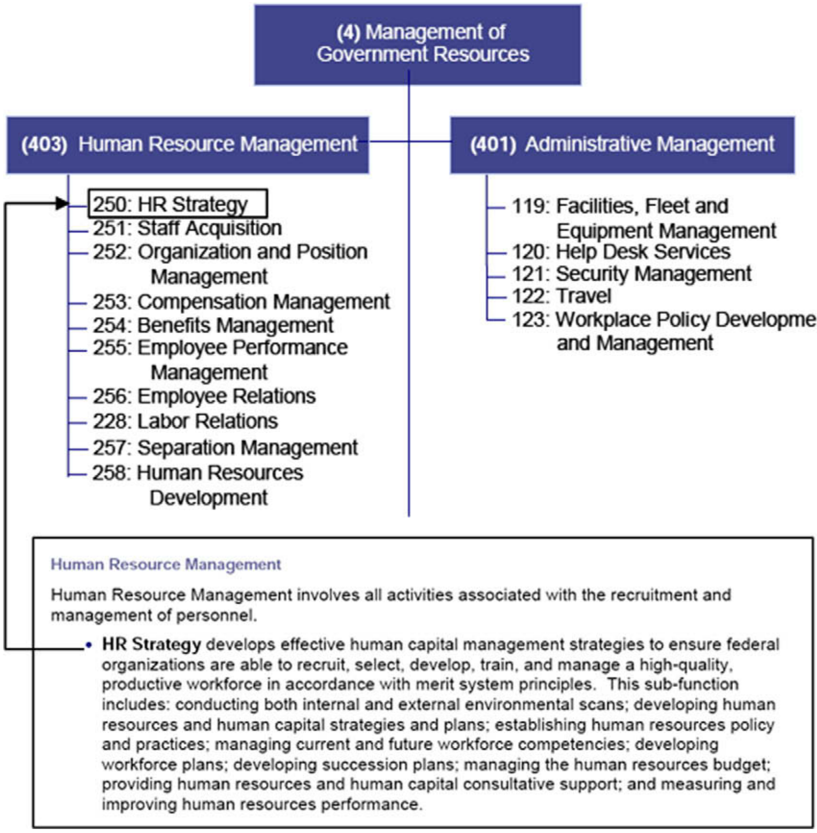
Fig. 5 shows a simple diagram which describes Enterprise Architecture components and their relationships with each other. The oval and sentences on its right side describe Enterprise Architecture components, and they correspond to concept definition of Enterprise Architectures in Figs. 2 and 3. The line which connects the ovals describes relationships of the components, and it corresponds to relationship definition in Fig. 4.

4. Ontology-based Enterprise Architecture

Enterprise Architectures are comprised of Enterprise Architecture components and relationships of them as shown in Fig. 5 simply. In this section, the ontology-based Enterprise Architecture for common understanding of Enterprise Architecture is suggested. The ontology-based Enterprise Architecture is modeled in three levels as shown in Fig. 6. The Enterprise Architecture ontology is composed of ontology for business terms, ontology for Enterprise Architecture components and ontology for relationships of the components. In Section 4.1, ontology for business terms is suggested. The ontology for business terms is an elementary ontology to define ontology-based Enterprise Architecture, which supports ontology for Enterprise Architecture components and ontology for relationships of them. Based on the ontology for business terms in Section 4.1, ontology for Enterprise Architecture components and ontology for relationships of the components are suggested in Sections 4.2 and 4.3, respectively.

4.1. Ontology for business terms

Business terms used in most enterprises are not managed systematically, and they are used implicitly without explicit definition so that humans and systems are in difficulties for communication. The Enterprise Architecture can support management of business terms because business terms can be treated as one of the Enterprise Architecture components. Reversely, business terms are used to define Enterprise Architectures. These business terms, however, are expressed with natural languages so that meanings of the terms are vague and humans and systems still cannot understand them exactly and commonly. In the same way, humans and systems cannot understand exactly and commonly Enterprise Architectures defined with the business terms. These misunderstandings cause problems in integration and collaboration of enterprises. Therefore, it is needed to share the meanings of business terms exactly and commonly, and there are already many

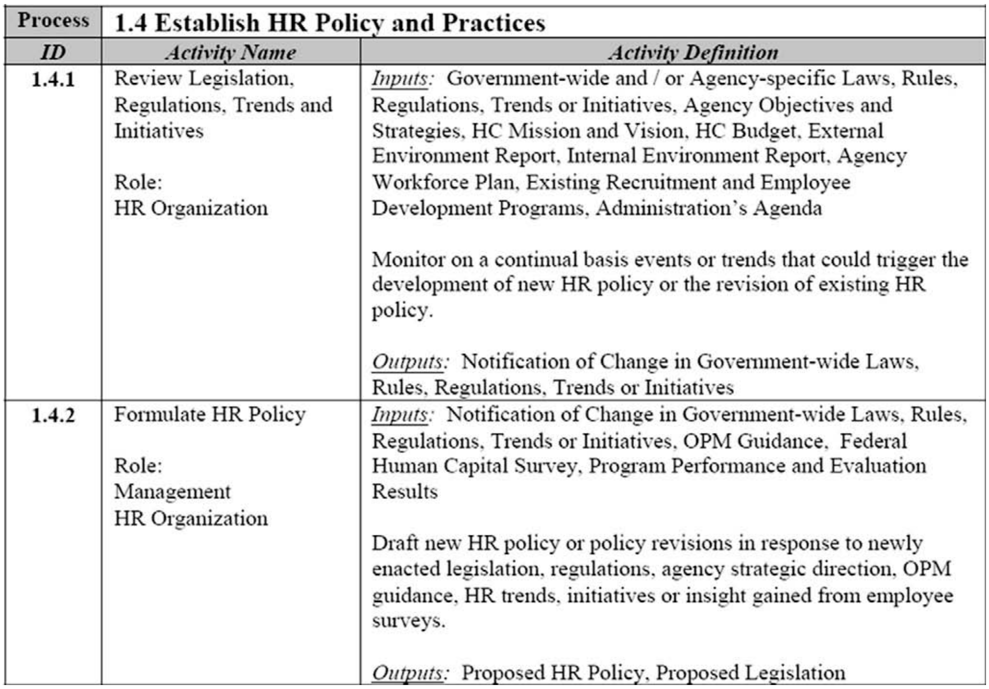

(4) Management of Government Resources
(403) Human Resource Management
(401) Administrative Management
250: HR Strategy
251: Staff Acquisition
252: Organization and Position Management
253: Compensation Management
254: Benefits Management
255: Employee Performance Management
256: Employee Relations
228: Labor Relations
257: Separation Management
258: Human Resources Development
119: Facilities, Fleet and Equipment Management
120: Help Desk Services
121: Security Management
122: Travel
123: Workplace Policy Development and Management

Human Resource Management

Human Resource Management involves all activities associated with the recruitment and management of personnel.

- **HR Strategy** develops effective human capital management strategies to ensure federal organizations are able to recruit, select, develop, train, and manage a high-quality, productive workforce in accordance with merit system principles. This sub-function includes: conducting both internal and external environmental scans; developing human resources and human capital strategies and plans; establishing human resources policy and practices; managing current and future workforce competencies; developing workforce plans; developing succession plans; managing the human resources budget; providing human resources and human capital consultative support; and measuring and improving human resources performance.

Fig. 2. Concept definition of '*HR Strategy Process*' of FEA (FEA Program Management Office, 2007).



Process	1.4 Establish HR Policy and Practices	
ID	*Activity Name*	*Activity Definition*
1.4.1	Review Legislation, Regulations, Trends and Initiatives Role: HR Organization	*Inputs:* Government-wide and / or Agency-specific Laws, Rules, Regulations, Trends or Initiatives, Agency Objectives and Strategies, HC Mission and Vision, HC Budget, External Environment Report, Internal Environment Report, Agency Workforce Plan, Existing Recruitment and Employee Development Programs, Administration's Agenda Monitor on a continual basis events or trends that could trigger the development of new HR policy or the revision of existing HR policy. *Outputs:* Notification of Change in Government-wide Laws, Rules, Regulations, Trends or Initiatives
1.4.2	Formulate HR Policy Role: Management HR Organization	*Inputs:* Notification of Change in Government-wide Laws, Rules, Regulations, Trends or Initiatives, OPM Guidance, Federal Human Capital Survey, Program Performance and Evaluation Results Draft new HR policy or policy revisions in response to newly enacted legislation, regulations, agency strategic direction, OPM guidance, HR trends, initiatives or insight gained from employee surveys. *Outputs:* Proposed HR Policy, Proposed Legislation

Fig. 3. Concept definition of 'Establish HR Policy and Practices Process' of FEA (US Office of Personnel Management, 2006).

ontology-based approaches to do. In this paper, the approach of WordNet is adopted to define business terms in the form of ontologies. The WordNet, which is suggested by Cognitive Science Laboratory of Princeton University, is an ontology database to define English terms. It defines English nouns, verbs, adjectives, and adverbs and links them through semantic relations that determine word definitions (Miller, 1990). Relational theories of lexical semantics hold that any word can be defined in terms of the other

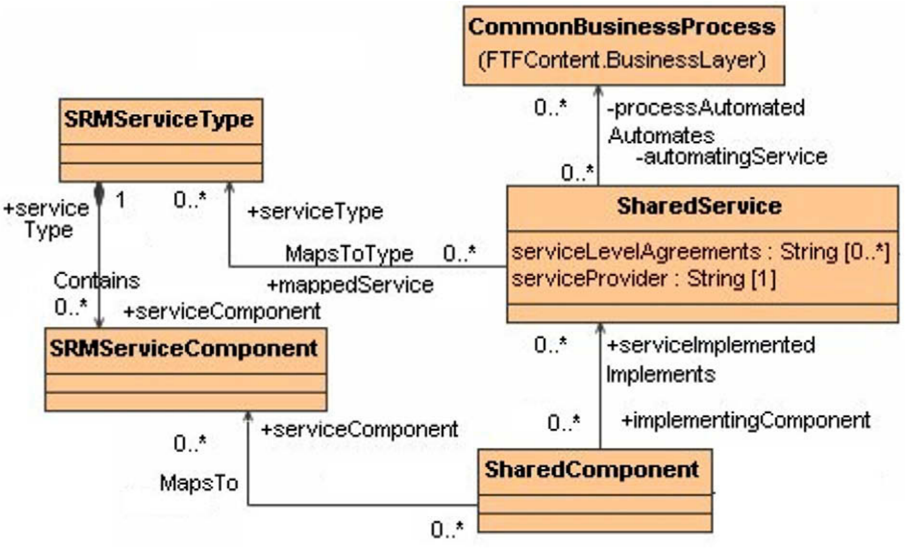


Fig. 4. Metamodel for relationships of Enterprise Architecture components of FEA (FEA-PMO, 2006).

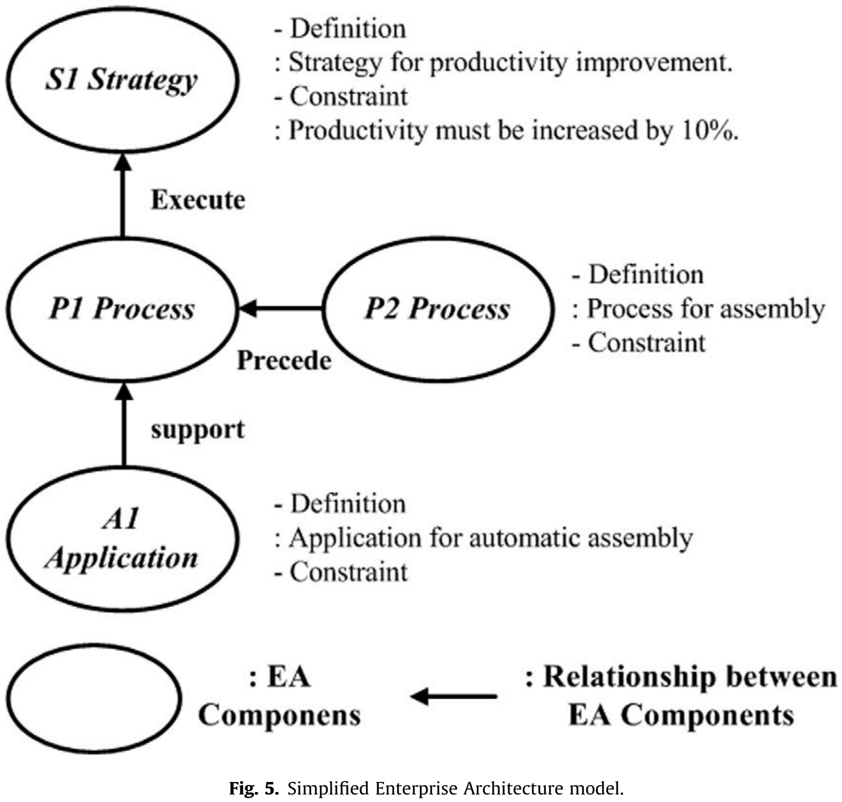


Fig. 5. Simplified Enterprise Architecture model.

words to which it is related (Miller, 1995). Because the approach of WordNet provides not only human readability but also machine readability, it is suitable to define the business terms of Enterprise Architectures which need to be interpreted by both humans and machines. Table 1 shows the semantic relations in WordNet, such as a similar relation, an opposite relation, a subordinate relation, and so on. Through these semantic relations, the WordNet forms semantic network of terms.

Business terms in Enterprise Architecture can be defined with ontologies in the same way that WordNet defines terms semantically. Because the Enterprise Architecture components are described through natural language mostly, enterprises can use basic terms of WordNet to interpret the description. In addition, business terms, which is created and used in specific enterprises or departments, are can be defined additionally in the way of WordNet by tagging ID. Fig. 7 shows the example of business terms defined in the way of WordNet. As shown in the figure, '*A enterprise*' defines a car as '*car*' and uses the term '*car*' in its documents and systems. To represent that the expression of '*car*' is specific to '*A enterprise*', '*A :*' is attached in front of the '*car*'. The terms used in the descriptive sentence '*a motor vehicle with four wheels*' can be interpreted through the predefined terms in WordNet. Because the '*A enterprise*' assembles cars, it focuses on description of cars through parts of cars. For example, '*A:car*' of '*A enterprise*' is defined

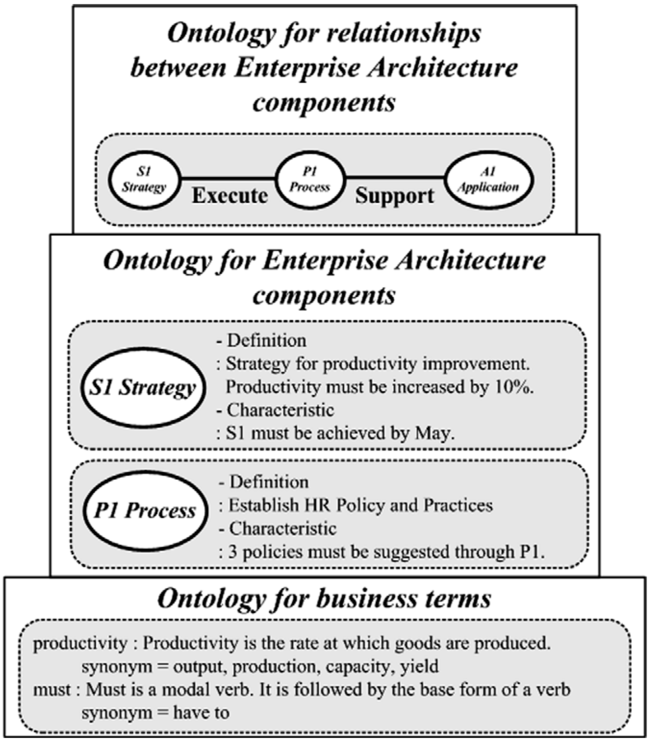

Ontology for relationships between Enterprise Architecture components
S1 Strategy
Execute
P1 Process
Support
A1 Application
Ontology for Enterprise Architecture components
S1 Strategy
- Definition
: Strategy for productivity improvement.
Productivity must be increased by 10%.
- Characteristic
: S1 must be achieved by May.
P1 Process
- Definition
: Establish HR Policy and Practices
- Characteristic
: 3 policies must be suggested through P1.
Ontology for business terms
productivity : Productivity is the rate at which goods are produced.
synonym = output, production, capacity, yield
must : Must is a modal verb. It is followed by the base form of a verb
synonym = have to

Fig. 6. Levels of Enterprise Architecture ontology.

Table 1
Semantic relations in WordNet (Miller, 1995).

Semantic relations	Syntactic category	Examples
Synonymy (similar)	N, V, Aj, Av	(pipe, tube), (rise, ascend), (sad, unhappy), (rapidly, speedily)
Antonymy (opposite)	Aj, Av, (N, V)	(wet, dry), (powerful, powerless), (friendly, unfriendly)
Hyponymy (subordinate)	N	(sugar maple, maple), (maple, tree), (tree, plant)
Meronymy (part)	N	(brim, hat), (gin, martini), (ship, fleet)
Troponomy (manner)	V	(march, walk), (whisper, speak)
Entailment	V	(drive, ride), (divorce, marry)

Note: N = Nouns, Aj = Adjectives, V = Verbs, Av = Adverbs.

as a thing composed of '*A:engine*' and '*A:wheel*' through the meronymy semantic relation. '*B enterprise*' defines a car as '*automobile*' or '*auto*' and uses the term '*automobile*' or '*auto*' in its documents and systems differently from '*A enterprise*'. As shown in Fig. 7, the fact that '*B:automobile*' has same meaning as '*B:auto*' is represented by synonymy semantic relation. Because '*B enterprise*' sells cars, it focuses on description of cars through kinds of cars. For example, '*B:automobile*' of '*B enterprise*' is defined as a thing which classified as '*B:sedan*' type or '*B:truck*' type. In order for '*A enterprise*' to collaborate with '*B enterprise*', '*A enterprise*' needs to know the fact that the term '*automobile*' of '*B enterprise*' is identical to the term '*car*' of it, and to represent the fact explicitly. '*A enterprise*' can judge that '*A:car*' is identical to '*B:automobile*' through the interpretation of '*B:automobile*' or agreement with '*B enterprise*'. And then, the relation of them can be defined as synonymy semantic relation. Through this definition, humans and systems of '*A enterprise*' can recognize that '*B:automobile*' is semantically same term as its '*A:car*' and collaborate with '*B enterprise*'.

4.2. Ontology for Enterprise Architecture components

After business terms are defined with ontologies, Enterprise Architecture components are described with ontologies. The description of Enterprise Architecture components includes definition, characteristic, and constraints of them. The ontology for Enterprise Architecture components can be modeled variously, but an approach of Semantics of Business Vocabulary and Business Rules (SBVR) is adopted in this paper. The SBVR is standard of Object Management Group (OMG) to support interchange of business vocabularies and rules among organizations (Object Management Group, 2008). The SBVR is based on fact-oriented approach. The fact-oriented approach is a conceptual modeling method that enables one to model and query business domains in terms of the underlying facts of interest, where all facts and rules may be verbalized in language readily understandable by non-technical users of those business domains (Halpin, 2007). Essential elements of the fact-oriented approach are facts. Facts build on fact types, and fact types build on concepts as expressed by terms. Fig. 8 shows an example of concepts, fact types and facts. The fact-oriented approach is less compact than other approaches, such as object-oriented approach, Entity-Relationship approach, and so on. The approach, however, has advantages of human-readability, semantic stability, expressiveness, extensibility, changeability and so on (Halpin, 2007; Object Management Group, 2008). Because Enterprise Architectures need to be understood by humans, be changed as business environment changes, be stable under frequent modification, it is suitable to define Enterprise Architecture components with ontologies in the way of the SBVR approach. In addition, because the SBVR is based on formal logic and provides formal representation through XML schema (Goedertier & Vanthienen, 2007), the ontology models of Enterprise Architecture components can get the formality.

The most convenient way to express a thing is using natural languages. The SBVR or any ontologies cannot fully support natural languages, but the SBVR provides Structured English which uses a small number of English structures and common words to provide a simple and straightforward mapping to its concepts, fact types

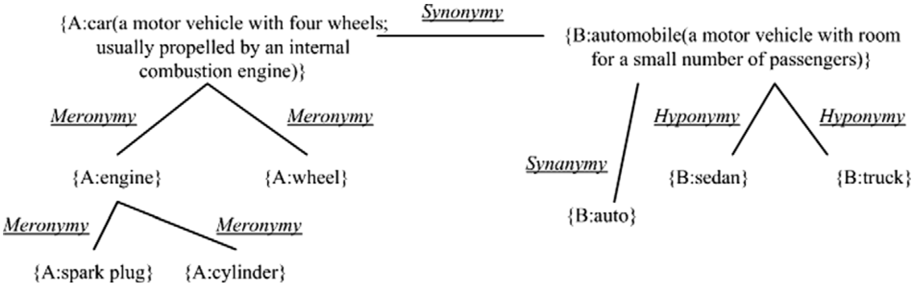
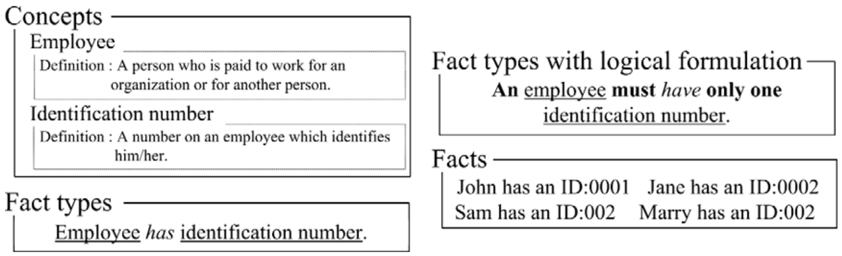

{A:car(a motor vehicle with four wheels; usually propelled by an internal combustion engine)}
Synonymy
{B:automobile(a motor vehicle with room for a small number of passengers)}
Meronymy
Meronymy
{A:engine}
{A:wheel}
Meronymy
Meronymy
{A:spark plug}
{A:cylinder}
Synanymy
{B:auto}
Hyponymy
Hyponymy
{B:sedan}
{B:truck}

Fig. 7. Example of business terms defined with ontologies.



Definition : A person who is paid to work for an organization or for another person.

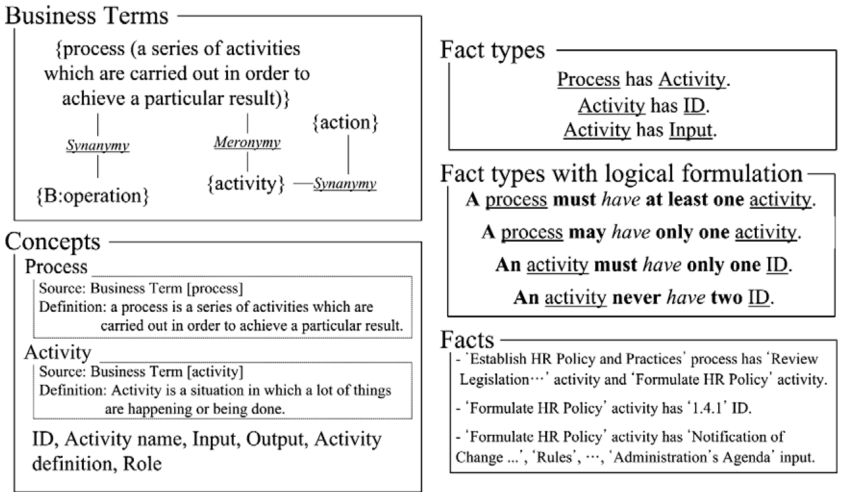
Definition : A number on an employee which identifies him/her.

Employee *has* identification number.

An employee **must** *have* **only one** identification number.

John has an ID:0001 Jane has an ID:0002
Sam has an ID:002 Marry has an ID:002

Fig. 8. Example of concepts, fact types and facts.



{process (a series of activities which are carried out in order to achieve a particular result)}

Source: Business Term [process]
Definition: a process is a series of activities which are carried out in order to achieve a particular result.

Source: Business Term [activity]
Definition: Activity is a situation in which a lot of things are happening or being done.

ID, Activity name, Input, Output, Activity definition, Role

Process has Activity.
Activity has ID.
Activity has Input.

A process **must** *have* **at least one** activity.
A process **may** *have* **only one** activity.
An activity **must** *have* **only one** ID.
An activity **never** *have* **two** ID.

- 'Establish HR Policy and Practices' process has 'Review Legislation···' activity and 'Formulate HR Policy' activity.
- 'Formulate HR Policy' activity has '1.4.1' ID.
- 'Formulate HR Policy' activity has 'Notification of Change ...', 'Rules', ···, 'Administration's Agenda' input.

Fig. 9. Ontology of Enterprise Architecture components with Structured English.

and facts (Object Management Group, 2008). Through the mapping, formality of the model expressed with the Structured English can be guaranteed. The example in Fig. 8 is defined with Structured English. Fig. 9 shows partial ontology of Enterprise Architecture components with Structured English. This ontology is for '*Establish HR Policy and Practices Process*' process, one of the FEA Enterprise Architecture components, as shown in Fig. 3. Through the concepts, fact types, fact types with logical formulation and facts, the details of '*Establish HR Policy and Practices Process*' process are described. First, concepts such as '*process*', '*activity*', '*ID*', and others are defined. The terms to describe concepts can be referred to business terms in Section 4.1. Second, based on the concepts, fact types are defined to describe the characteristic of the process. The '*Process has Activity*' fact type informs that a process includes activities. To describe constraints and rules of the fact type, logical formulation can be added to the fact type. The '*A process must have at least one activity.*' fact type informs that one process must have more

Table 2
Logical formulations of the SBVR.

Quantification	
Each	Some
At least one	At least n
at most one	At most n
Exactly one	Exactly n
At least n and at most m	More than one
Logical operations	
It is not the case that p	p and q
p or q	p or q but not both
If p then q	q if p
p if and only if q	not both p and q
Neither p nor q	p whether or not q
Modal operations	
It is obligatory that p	it is prohibited that p
It is necessary that p	it is impossible that p
It is possible that p	it is permitted that p
...may... only if p	it is permitted that q only if p
It is possible that q only if p	...must...
...must not...	...always...
...never...	...may...

Table 3
Relationships of Enterprise Architecture components.

Type of relationship	Relationships
Relationships of strategies	Identical strategy relationship Opposite strategy relationship Type strategy relationship Parts strategy relationship Exclusive strategy relationship
Relationships of processes	Identical process relationship Alternative process relationship Parts process relationship Pre–Post process relationship
Relationships of resources	Identical resource relationship Alternative resource relationship
Relationships between strategy and process	Core strategy–process relationship Supportive strategy–process relationship Adverse strategy–process relationship
Relationships between process and resources	Core process–resource relationship Supportive process–resource relationship Subordinate process–resource relationship

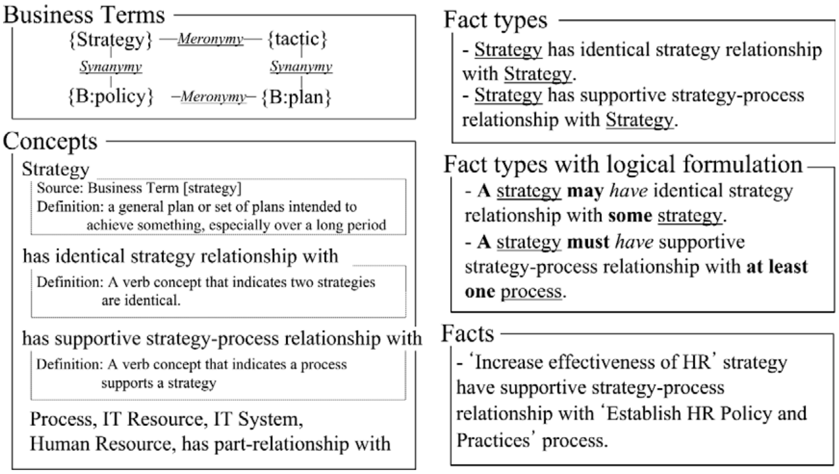

Business Terms
{Strategy} —Meronymy— {tactic}
Synanymy
Synanymy
{B:policy} —Meronymy— {B:plan}
Concepts
Strategy
Source: Business Term [strategy]
Definition: a general plan or set of plans intended to achieve something, especially over a long period
has identical strategy relationship with
Definition: A verb concept that indicates two strategies are identical.
has supportive strategy-process relationship with
Definition: A verb concept that indicates a process supports a strategy
Process, IT Resource, IT System, Human Resource, has part-relationship with
Fact types
- Strategy has identical strategy relationship with Strategy.
- Strategy has supportive strategy-process relationship with Strategy.
Fact types with logical formulation
- A strategy may have identical strategy relationship with some strategy.
- A strategy must have supportive strategy-process relationship with at least one process.
Facts
- 'Increase effectiveness of HR' strategy have supportive strategy-process relationship with 'Establish HR Policy and Practices' process.

Fig. 10. Ontology of relationships among Enterprise Architecture components with Structured English.

than one activity. Table 2 shows the logical formulations expressed with Structured English in the SBVR. Through these logical formulations, fact types for Enterprise Architecture components can be defined in detail. Finally, based on the fact types, the facts for Enterprise Architecture components are populated. Each fact as shown in Fig. 9 describes the details of '*Establish HR Policy and Practices Process*' process.

4.3. Ontology for relationships of Enterprise Architecture components

To share meanings of relationships among Enterprise Architecture components, ontologies for the relationships are required. The relationships include relationships among the components in the same cells, namely in the same domain, and relationships among the components in the different cells, namely in the different domain, as shown in Fig. 1. Table 3 shows the relationships among strategies, processes, resources, etc. which are some of Enterprise Architecture components. As Enterprise Architectures are updated, these relationships can be modified, new relationships can be added, useless relationships can be deleted, and so on. Because of the semantic stability, extensibility and changeability of the fact-oriented approach with other advantages, the SBVR approach helps to define these relationships in the form of ontologies. Fig. 10 shows partial ontology of relationships among Enterprise Architecture components with Structured English. In the same way of Section 4.2, the concepts, fact types, and fact types with logical formulations are defined with Structured English. The concepts include concepts of Enterprise Architecture components, concepts of relationships among them, fact types, and so on. Through the fact types, facts for the relationships are populated, and these facts describe details of the relationships.

5. Conclusions

The Enterprise Architecture is a descriptive document which explains all about enterprise components systematically. Through the Enterprise Architecture, Enterprises can implement enterprise integration to cope with dynamically changing business environment. Existing Enterprise Architectures, however, lack of semantics for humans and systems to understand them exactly and commonly, which causes communication problems between humans or between systems or between human and system. These communication problems keep enterprises from implementing integration and collaborating with other enterprises. In order to solve this problem, the ontology-based Enterprise Architecture is suggested in this paper. The Enterprise Architecture ontology is composed of ontologies in three levels. Ontologies of business terms are in the first level, ontologies of Enterprise Architecture components are in the second level, and ontologies of relationships among Enterprise Architecture components are in the top level. The ontologies of business terms are defined in the approach of the WordNet, and the ontologies of Enterprise Architecture components and relationships of them are defined in the approach of the SBVR. Through these ontologies, it is expected that humans and systems can understand Enterprise Architectures exactly and commonly, which supports integrations in enterprises and collaborations between enterprises.

Future researches still remain as follows: the Structured English is used to describe ontologies of Enterprise Architecture components and relationships among them. It helps humans to understand the ontologies, but it is still short of machine-readability on the other hand. The SBVR provides formal logics to guarantee machine-readability, but it is not sufficient to model complex Enterprise Architecture completely. Therefore, researches to extend the Structured English for Enterprise Architectures or new fact-oriented definition methods are needed.

Acknowledgement

This work is supported by Grant No. R01-2007-000-11040-0 from the Basic Research Program of the Korea Science and Engineering Foundation.

References

Allemang, D., Hodgson, R., & Polikoff, I. (2005). Federal Enterprise Architecture reference model ontologies: FEA-RMO version 1.1, online available at <http://www.topquadrant.com/documents/TQFEA-RMO.pdf>.

Bertolazzi, P., Krusisch, C., & Missikoff, M. (2001). An approach to the definition of a core enterprise ontology: CEO. In: *International workshop on open enterprise solutions: Systems, experiences, and organizations (OES-SEO 2001)* (pp. 14–15). Rome.

Brown, T. (2004). The value of Enterprise Architecture, Zachman Institute for Framework Advancement (ZIFA), online available at <http://www.zifa.com>.

Enterprise Integration Laboratory. (2002). TOronto Virtual Enterprise (TOVE) project, online available at <http://www.eil.utoronto.ca/enterprise-modelling/tove/index.html>.

FEA Program Management Office. (2007). FEA consolidated reference model version 2.3, online available at <http://www.whitehouse.gov/omb/egov/documents/FEA_CRM_v23_Final_Oct_2007.pdf>.

FEA-PMO. (2006). Federal transition framework metamodel reference version 1.0, online available at <http://www.whitehouse.gov/omb/egov/documents/FTF_Metamodel_Reference_Pilot_Final_December-_2006.pdf>.

Goedertier, S., & Vanthienen, J. (2007). Declarative process modeling with business vocabulary and business rules. *Lecture Notes in Computer Science, 4805*, 603–612.

Gruber, T. (1993). Toward principles for the design of ontologies used for knowledge sharing. Technical Report KSL93-04, Knowledge Systems Laboratory, Stanford University.

Halpin, T. (2007). Fact-oriented modeling: Past, present and future. In J. Krogstie, A. L. Opdahl, & S. Brinkkemper (Eds.), *Conceptual modelling in information systems engineering*. Berlin Heidelberg: Springer.

Harmon, P. (2003). Developing an Enterprise Architecture, Business process trends: Whitepaper.

Leppänen, M. (2007). A context-based enterprise ontology. *Lecture Notes in Computer Science, 4439*, 273–286.

Miller, G. A. (Ed.). (1990). WordNet: An on-line lexical database. *International Journal of Lexicography, 3*(4), 235–312.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM, 38*(11).

Missikoff, M., & Taglino, F. (2002). Business and Enterprise Ontology Management with SymOntoX. *Lecture Notes in Computer Science, 2342*, 442–447.

National Institute of Standards and Technology. (2005). The Process Specification Language (PSL) overview and version 1.0 specification, online available at <http://www.mel.nist.gov/msidlibrary/doc/nist-ir6459.pdf>.

Object Management Group. (2008). Semantics of Business Vocabulary and Business Rules (SBVR) version 1.0, online available at <http://www.omg.org/spec/SBVR/1.0/PDF>.

UC Irvine, UC Irvine Enterprise Architecture, online available at <http://apps.adcom.uci.edu/Enterpri-seArch/>.

Uschold, M. (Ed.). (1998). Knowledge level modelling: Concepts and terminology. *Knowledge Engineering Review, 13*(1), 5–29.

Uschold, M., & Jasper, R. (1999). A framework for understanding and classifying ontology applications. In *Proceedings of the IJCA-I99 workshops on ontologies and problem-solving methods*. Stockholm, Sweden.

Uschold, M., King, M., Moralee, S., & Zorgios, Y. (1998). The enterprise ontology. *The Knowledge Engineering Review*, 13.

US Department of Defense. (2007). Department of Defence Architecture Framework version 1.5, online available at <http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_I.pdf>.

US Office of Personnel Management .(2006). HR LOB Business Reference Model (BRM) version 2, online available at <http://www.opm.gov/egov/documents/architecture/BRM_Report_V2.pdf>.

van Renssen, A. S. H. P. (2005). Gellish: A generic extensible ontological language – Design and application of a universal data structure –. Doctoral Thesis, Delft University of Technology, the Netherlands, September 14.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal, 26*(3).

Zachman Institute for Framework Advancement, Zachman Framework, online available at http://www.zifa.com/.