

Contents lists available at GrowingScience

## Management Science Letters

homepage: [www.GrowingScience.com/msl](http://www.GrowingScience.com/msl)

### A new model for designing uncertain enterprise architecture

Jamshid Afshani<sup>a\*</sup>, Ali Harounabadi<sup>b</sup> and Mashallah Abbasi Dezfouli<sup>a</sup><sup>a</sup>Department of computer engineering, Science and Research Branch, Islamic Azad University, khuzestan-Iran<sup>b</sup>Department of computer engineering, Islamic Azad University, Central Tehran Branch, Iran**ARTICLE INFO***Article history:*

Received October 5, 2011

Received in Revised form

January, 4, 2012

Accepted 5 January 2012

Available online

6 January 2012

*Keywords:**Uncertain Enterprise Architecture**C4ISR Framework Products**Fuzzy-UML***ABSTRACT**

C4ISR framework describes architecture in three views of information architecture and defines a set of products, which are main outputs of enterprise architecture design. Formats and templates, which are presented for C4ISR products, cannot describe uncertainty in process or data. Meanwhile, uncertainty in many information systems is unavoidable and using the concept of fuzzy numbers in architecture design helps architects handle uncertainty in process and data of organization. In this paper, we present a new template based on fuzzy-UML concept for some of C4ISR products such as Logical Data Model (OV-7), Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c). To explain further, fictional Fast Pass system used at OilCo gas stations is used to demonstrate details of our proposed model.

© 2012 Growing Science Ltd. All rights reserved.

### 1. Introduction

The Command, Control, Communications, Intelligence, Surveillance, and Reconnaissance (C4ISR) Architecture Framework, Version 2.0, developed by the U.S. Department of Defense (DoD) C4ISR Architecture Working Group, provides guidance for describing architectures (AWG, 1997). Actually, C4ISR framework describes enterprise architecture with its products and for each of the C4ISR product special template has been presented. In addition, these products are shown and described by UML diagrams, and OSAN and other templates (Bienvenu et al., 2000). There are many capabilities associated with these methods and templates, but they cannot describe uncertainty in enterprise architecture, which is the case for many real-world applications. During the past few years, there have been tremendous efforts on developing methods to handle uncertainty associated with different applications.

In this article, we present a new template to describe and to show some of the C4ISR products by the Fuzzy-UML concept. In addition to the ability of UML diagrams, Fuzzy-UML diagrams has the ability to describe uncertain concepts.

\* Corresponding author. Tel: +

E-mail addresses: [j.afshani83@gmail.com](mailto:j.afshani83@gmail.com) (J. Afshani)

With this method, the uncertainty in the data and process of enterprise architecture, which are inevitable requirements of the users will be considered and described. The C4ISR products describe both dynamic and static behavior of Enterprise Architecture. Here we focus on Logical Data Model (OV-7) used to show information data of the architecture in terms of static behavior and Operational Event/Trace Description (OV-6c), Systems Event/Trace Description (SV-10c) to extend the capture of business requirements and concept of operations information introduced by the Logical Data Model (AWG, 1997). In this paper, we first explain C4ISR Framework and Fuzzy-UML diagram in this section that need for our idea, and then we will present our idea that is new templates for C4ISR products. Finally, to explain further, fictional Fast Pass system used at OilCo gas stations has been studied.

### *1.1. C4ISR Framework*

The purpose of defining a C4ISR architecture specification is to assure that system-wide characteristics such as mission effectiveness, interoperability, etc. are achieved across the evolution or deployment phases of the systems, which are built in conformance with the architecture. These systems may be associated with time evolution, differing deployment options, or both. Time evolution of systems is common in the U.S. DoD, where individual systems, which are interconnected in a large, distributed systems are replaced by upgraded versions. Systems may also differ over deployment schemes, as in the case of large building security systems, which consist of a large number of component types, with specific deployments having varying mixes and numbers of components. The U.S. DoD has found numerous situations where such problems have arisen, and, in response, has developed the C4ISR Architecture Framework (AF) (AWG, 1997).

The framework has four main parts:

- Definitions of three standard views of any given architecture: the operational view, the systems view, and the technical view
- Common products (descriptive models) and data: To consistent and integrated linkages among its various views. These products are the main outputs of enterprise architecture design.
- Common building block references
- High-level guidance: how to use the Framework to describe architecture.

### *1.2 Fuzzy UML*

Because of natural characteristics of requirements, uncertainty in information systems is inevitable. For system development, object orientation methodologies, express their concepts with UML diagrams. The UML as a standard object-oriented language is able to cope with behavioral and structural aspects of software engineering.

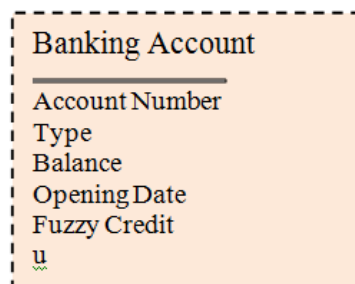
With inject uncertainty in UML, extended version has been developed called Fuzzy-UML (Haronabadi & Teshnehlab, 2007; Lu, 2005; Ma, 2004; Ma, 2005). Fuzzy-UML is propounded in two sections: fuzzy structure and fuzzy behavior. To support system fuzzy structure, a fuzzy data model, and to support system fuzzy behavior, the models are needed to support system functionality using the concept of fuzzy. The first part of this section is about data modeling where the uncertainty in data structure is entered by presenting a class diagram in fuzzy form. The fuzzy behavior modeling is inspected in the second part. This subject is realized by the use case, sequence and state diagrams in fuzzy form. Here, we present fuzzy sequence diagram and other diagrams have been investigated in (Haronabadi & Teshnehlab, 2007; Lu, 2005; Ma, 2004; Ma, 2005).

### 1.3. F-UML data model

The class diagrams in UML are the logical models, which describe the system main structure. The classes and the relationships among them consist of the elements in class diagram. By entering the uncertainty into these elements, the Fuzzy-UML data model is produced. According to Ma (2004, 2005) in the context of classes, there are three levels of fuzziness defined as follows:

- Fuzziness in the extent to which the class belongs in the data model as well as fuzziness on the content (in term of attributes) of the class.
- Fuzziness related to whether some instances are instances of a class; even though the structure of a class is crisp, it is possible that an instance of the class belongs to the class with degree of membership.
- The third level of fuzziness is on attribute values of the instances of the class; an attribute in a class defines a value domain, and when this domain is a fuzzy subset or a set of fuzzy subset, the fuzziness of an attribute value appears.

The attribute or the class name in the first level should be described by this phrase WITH mem DEGREE where,  $0 \leq \text{mem} \leq 1$ . This value shows the degree of membership of the attribute to the class or the class to the data model. The second level of fuzziness, the membership degree in an instance of the class, which belongs to the Class should be specified. So an additional attribute in the class is defined for representation of the instance membership degree to the class where its domain is  $[0, 1]$ . The classes with the second level of fuzziness are specified by a rectangle fuzzy number where its lines are dash. In the third level, a fuzzy keyword is appeared in front of the attribute. Fig. 1 shows the banking account fuzzy class (Merseguer & Campos, 2004). In the mentioned class, the credit attribute could have the fuzzy value (the third level of fuzziness). On the other hand, the credit attribute is a linguistic variable, and it has a domain like fuzzy sets (for example: little / much). The account type specifies the membership degree of credit attribute to the class (the first level of fuzziness): "Credit With 0.8 membership Degree"



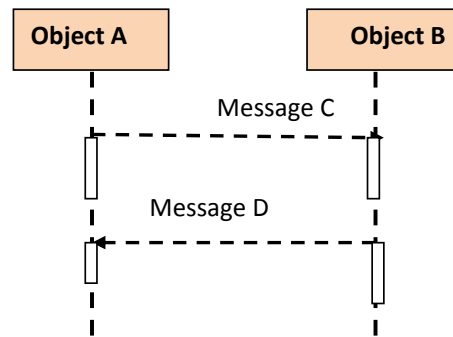
**Fig. 1.** A fuzzy class of banking account

The relationships among the classes are divided into four categories and they are propounded in fuzzy form (Ma, 2005): fuzzy generalization, fuzzy association, fuzzy aggregation and fuzzy dependency.

### 1.4 Fuzzy UML behavior model

A Use Case is a sequence of actions where a system prepares an observable result for the user. In other words, for each Use Case, the scenarios could be set, which explain the steps of the Use Case. In UML, sequence diagram is used to realize the Use Cases. If the selected Use Case is uncertain, the

sequence diagram will be uncertain too. The dash lines for uncertain messages representation in mentioned diagram are used. Fig. 2 represents this subject.



**Fig. 2.** Fuzzy-Sequence diagram in Fuzzy-UML

The messages in sequence diagram explain the methods. The uncertainty in method has two levels of fuzziness: The first one is that the method belongs to the object with membership degree and the second one is that the method is fuzzy in nature. In Fig. 2, the method C, which is produced from message C belongs to the object B with the membership degree between 0 and 1 (first level fuzziness). In other part, the essence of message D is also uncertain (second level of fuzziness). State diagrams, models different states of an object. This diagram is mostly used to show the dynamic behaviors of a system.

## 2. Fuzzy model for C4ISR

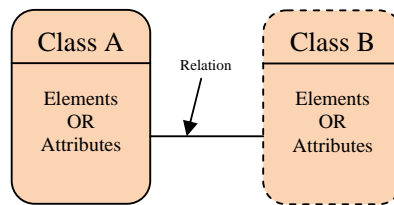
In this section, we present new template to show and describe Logical Data Model (OV-7), Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c), with Fuzzy-UML. Both static and dynamic behavior of enterprise architecture can be described with use of Fuzzy-UML concept. In this section, we present new template for describing products used to show dynamic behavior of enterprise architecture (Operational Event/Trace Description (OV-6c), Systems Event/Trace Description (SV-10c),) and a product that used to describe static behavior of enterprise architecture (Logical Data Model(OV-7)) using concept of Fuzzy Class Diagram and Fuzzy-Sequence and Fuzzy-Sequence Diagram respectively.

### 2.1. Logical data model (OV-7)

The Logical Data Model (LDM) is used to document the data requirements and structural business process rules of the architecture's operational view. It describes the data and information, which is associated with the information exchanges of the architecture, within the scope and to the level of detail required for the purposes of the architecture. AWG (1997) presents information items and/or data elements, attributes or characteristics, and interrelationships. Logical Data Model can be described by class diagram in UML (Bienvenu et al., 2000). The purpose of a given architecture helps determine the level of detail needed in this product. A formal "data" model (IDEF1X) that is detailed down to the level of data, their attributes, and their relationships are required for some purposes, such as validation of completeness and consistency requirements. However, for other purposes, a higher-level information focused data model of the domain of interest such as an entity-relation model without entity attributes will be sufficient. The proposed model of this paper assumes that detail of data such as attributes of entities is required. Templates and models presented since they cannot describe uncertainty in data model of enterprise architecture. So this idea can be used by architect for more realistic design architecture. Uncertainty in a Logical Data Model (OV-7) has four levels of fuzziness are defined as follows:

1. *A class belongs to the architecture with membership degree in domain [0, 1].* We know that Logical Data Model (OV-7) describes data and information, which is associated with the information exchanges of the architecture. So some of the classes in this product are produce in fuzzy form. We assign the class to architecture *WITH mem DEGREE* where,  $0 \leq mem \leq 1$ .
2. *Fuzziness related to whether some elements or attribute belong to the class.* It is possible that an element or attribute of the class belongs to the class with degree of membership. We assign element or attribute to the class *WITH mem DEGREE* where,  $0 \leq mem \leq 1$ .
3. *Fuzziness in attribute values.* An attribute in a class defines a value domain, and when this domain is a fuzzy subset or a set of fuzzy subset, the fuzziness of an attribute value appears.
4. *Fuzziness related to an instance of the class that is objects of a fuzzy class.* If a class uses instance of a fuzzy class, inherits its fuzzy nature.

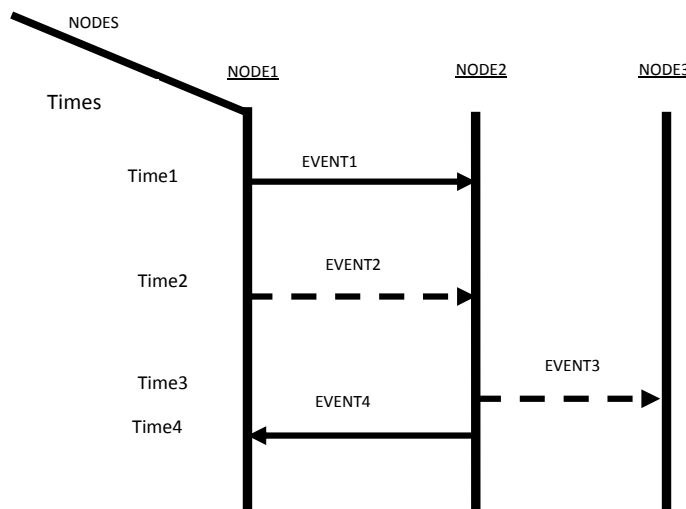
A class may include one or more fuzzy levels. Fig. 3 shows a new template for describe Fuzzy Logical Data Model. In this template, fuzzy classes are drawn by dash line.



**Fig. 3.** Template for Logical Data Model (OV-7) with considering uncertainty in data, Class B is a fuzzy class

2.2.Operational Event/Trace Description (OV-6c), Systems Event/Trace Description (SV-10c)

Operational Event/Trace Descriptions, sometimes called sequence diagrams, event scenarios, and timing diagrams, allow tracing of actions in a scenario or critical sequence of events (AWG, 1997).



**Fig. 4.** Template for Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c) products

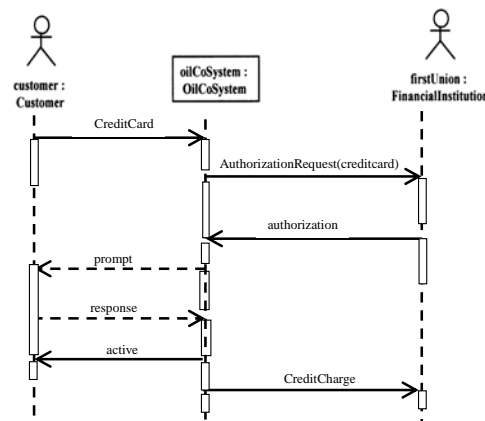
This product can be described with sequence diagram in UML (Bienvenu et al., 2000). For each process and services in High-level Operational Concept Graphic (OV-1) we have one Operational Event/Trace Description (OV-6c). If a process or service is uncertain, related Operational Event/Trace Description will be uncertain too. In addition, Operational Event/Trace Description shows sequence of events, so if an event is uncertain we must show its uncertainty. There are a lot of templates and formats used for design Operational Event/Trace Description, but none of them can show uncertainty in events. Fig. 4 provides a template for an Operational Event/Trace Description with uncertain event. The items across the top of the diagram are nodes, usually roles or organizations, which must take action based on certain or uncertain types of events. Each node has a timeline associated with it, which runs, vertically. Specific points in time can be labeled running down the left side of the diagram. Directed lines between the node time lines represent events, if the event is uncertain, related line will be dash line. The uncertainty in events has two levels in fuzzy forms: The first one belongs to the node with membership degree and the second one specifies that the events essence is fuzzily (Ma, 2005). This template also can be used for design Systems Event/Trace Description (SV-10c) product easily.

### 3. Case study

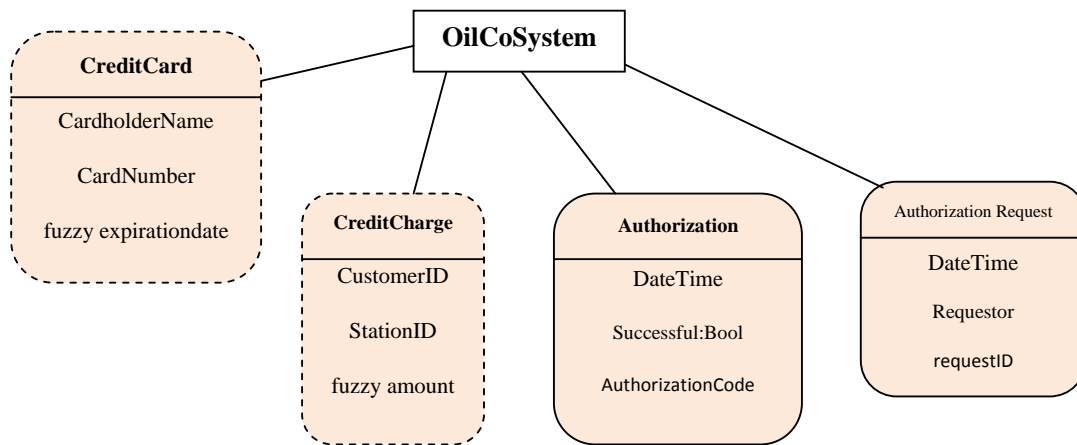
In this section, we use an illustrative example, which is associated with fictional Fast Pass system used at OilCo gas stations. In this system driver enters credit card, then information transmit to OilCoSystem database using LAN network and OilCoSystem sends information to financial institution for authorization. The OilCoSystem station determines, according to the account given to the driver, how much and what type of fuel, type A or B, can be refueled and the driver also has the option to choose an appropriate one. Pump station with the driver selections, prepares to refuel and the amount and type of fuel used sent to the OilCoSystem station and the amount of fuel is calculated. For updating, the information is sent to financial institution issued credit card. The following events contain fuzzy data shown with dash line:

- Calculation amount and type of fuel that driver can use (*from OilCoSystem to drivers pump*),
- The amount and type of fuel that the driver uses (*from driver pump to OilCoSystem*).

Fig. 5 shows top-level Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c) of fictional Fast Pass system and Fig. 6 shows top-level Logical Data Model (OV-7) of fictional Fast Pass system.



**Fig. 5.** Top-level Operational Event/Trace Description (OV-6c) and Systems Event/Trace Description (SV-10c) of fictional Fast Pass system



**Fig. 6.** Top-level Logical Data Model (OV-7) of fictional Fast Pass system (Not complete) CreditCard class and CreditChargh are fuzzy classes

#### 4. Conclusion

In this paper, we have presented new templates, which consider uncertainty for C4ISR products. The proposed model of this paper uses new templates for describing three numbers of C4ISR products used for dynamic and static behavior of architecture using Fuzzy-UML concept. C4ISR products have been already been shown and described by UML diagrams but a template for describe Logical Data Model (OV-7), Operational Event/Trace Description (OV-6c), Systems Event/Trace Description (SV-10c) was not considered any uncertainty. So the idea presented in this paper, can be used by architects as a new model to design these products. The implementation of this hybrid model has been demonstrated using a simple well known example. These templates can be converted to Fuzzy Colored Petri Nets for evaluating architecture. It is also possible to use fuzzy concept in enterprise architecture and we can use linguistic variables and use specific question to evaluate architecture.

#### Acknowledgment

The authors would like to thank the anonymous referees for their constructive comments on earlier version of this work.

#### References

- Bienvenu, M.P., Shin, I., & Levis, A.H. (2000). C4ISR Architectures III: An Object-Oriented Approach for Architecture Design. *Systems Engineering*, 3(4), 288-312.
- Bai, X.H. (2008). An application with UML-based Petri nets for C4ISR Architecture simulation validation. *Proceedings of the Seventh International Conference on Machine Learning and Cybernetics*, Kunming, 2257-2263.
- Haronabadi, A., & Teshnehlal, M. (2008). A Novel Method for Behavior Modeling in Uncertain Information Systems. *World Academy of Science, Engineering and Technology*, 41, 959-966.
- Levis, H., & Wagenhals, L.W. (2000). C4ISR Architectures I: Developing a Process for C4ISR Architecture Design. *Systems Engineering*, 3(4). 225-247.

- Lu.W. (2005). Fuzzy UML. Seminararbeit, Summer semester, 2005.
- Ma, Z. (2005). Fuzzy Information Modeling With the UML. Idea Group Publishing, 153-176.
- Ma, Z. (2004). Extending UML for Fuzzy Information Modeling in Object Oriented Database. Theories and Practices, Idea Group Publishing.
- Motameni, H., Daneefar, I., Bakhshi, J., & Nematzadeh, H. (2009). Transforming fuzzy state diagram to fuzzy Petri net. Journal of Computer Engineering, 1(1), 29-44.
- Merseguer, J., & Campos, J. (2004). Software Performance Modeling using UML and Petri nets. Lecture Notes in Computer Science Journal, 2965, 265 – 289.
- Shin, M., Levis, A., & Wagenhals, L. (2002). Transforming of UML-Based system Model into CPN Model for validating system behavior. Workshop of UML Conference, California USA.
- US DoD C4ISR Architecture Working Group (AWG) (1997). C4ISR Architecture Framework Version 2.0. Washington, D.C.