Information Exchange Requirements (IER) and Information Exchange Models (IEM)

Robert Suzić[1,1] and Choong-ho Yi[1]

[1] Combitech AB, Command and Control Systems,
Torshamnsgatan 32 A, SE-164 84 Stockholm, Sweden
{robert.suzic, choong-ho.yi}@combitech.se

Abstract. Recent internationalisation effort of the Swedish Armed Forces has led to rapidly increased demand on interoperability in general and information exchange in particular. Flexibly composed units involve the building of never complete system. Nevertheless, the challenge of connecting systems turns out not to be only a matter of technical issues but also semantic (unique interpretation of information) and organisational (information exchanged is efficiently used) ones. In order to provide a domain specific model within a particular context we need an adaptive methodology that deals with how to build an information exchange model based upon information exchange requirements. By having a uniquely defined method to build the information exchange models we can provide a timely and resource efficient information exchange. This paper presents a methodology that is a first step towards meeting the challenges mentioned above.

Keywords: Interoperability, Information Exchange, Information Exchange Requirements, Information Exchange Models, Institutional Interoperability JC3IEDM.

1 Introduction

1.1. Background

Much work has been done within the field of interoperability from the agents' perspective, e.g., [1]. However, less work has been performed within interoperability from the *institutional* perspective. This paper addresses issues of institutional interoperability from how to adapt/create an information exchange model (IEM) to information exchange requirements (IER) and information exchange standards. In particular we address the following challenges:

- How to create a new IEM, that is based on information exchange standard (IES), from IER
- How to efficiently adapt IEM given changed IER
- How to efficiently adapt IEM given changed IES.

1.2. The context of the study

This paper presents part of *preliminary* results of a group of consultants from Combitech AB commissioned by the Swedish Defense Material Administration (FMV).

From the Swedish (Armed forces) perspective, cooperation within the European Union, with the United Nations and NATO are of great importance when dealing with prevention/peace keeping and managing disasters in the international arena. The Swedish Armed Forces (SwAF) participate in many types of missions throughout the world. The recent years of internationalisation of the SwAF have lead to rapidly increased demand on interoperability in general and information exchange in particular. Flexibly composed units involve the building of never complete system. In such missions systems emerge and are dynamical, mission-oriented, loosely coupled and federated; in one federation, it may occur the need to connect systems that weren't intended in the original specification to be connected. A system in our case includes both operational systems and technical ones, see **Fig.** 1.

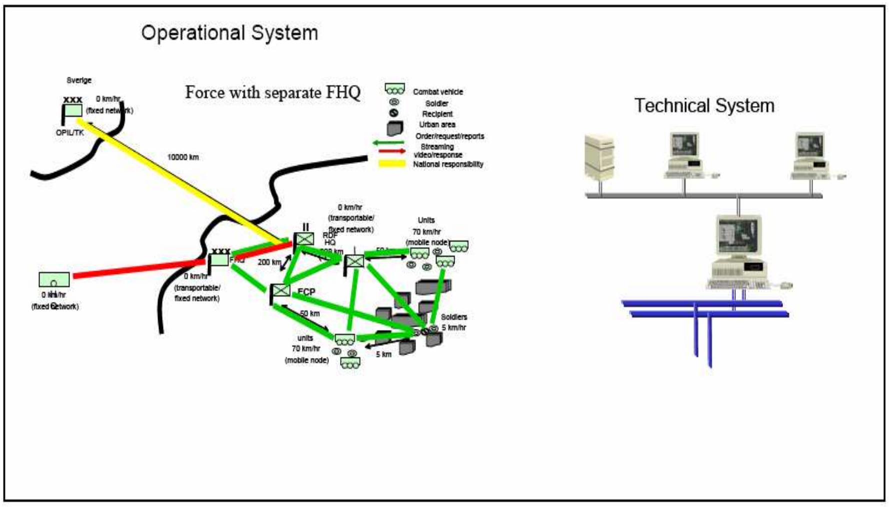

Fig. 1 Operative and Technical Systems [2]

Nevertheless, the challenge of connecting systems turns out not to be only a matter of technical issues but also semantic (in terms of unique interpretation of information) and organisational (in terms that information exchanged is efficiently used) ones.

In order to ensure efficient information exchange between different parts the Multilateral Interoperability Programme (MIP) organization [3] has proposed a model named Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM) [3] for information exchange. Its overall goal is to specify the minimum set of data that needs to be exchanged in coalition or multinational operations. The JC3IEDM is a standard in information exchange. However, the JC3IEDM is not enough. In some cases the JC3IEDM lacks a context specific description of information. In other cases it is not domain specific (one has to extend or adapt it). Moreover, requirements on information exchanged can change during time; in context of military and military operations other than war it may occur needs to adapt an IEM to new information exchange requirements.

In order to provide a domain specific model(s) within a particular context(s) we need an adaptive methodology that deals with on how to build IEM based upon information exchange requirements. By having a uniquely defined method to build the information exchange models we can provide a timely and resource efficient information exchange.

1.3 Aim

The aim of the paper is to present a methodology for how to deal with issues named above, i.e. we introduce a methodology of how to build an IEM from IERs. The method is constituted by a class diagram describing most important features and a process/activity diagram. We also discuss efficiency of different approaches.

2 Information Exchange Requirements (IER) and Information Exchange Models (IEM): Definitions

There are different definitions of what is an information exchange requirement. One of them is as follows: *An Information Exchange Requirement (IER) is the description, in terms of characteristics, of the requirement to transfer information between two or more end users. The characteristics described include source, recipients, contents, size, timeliness, security and trigger. IERs are defined to be independent of the communications medium. An IER can express both current and future requirements,* [4].

IER are divided into:
- operational
- system specific
- technical

The focus here is on operational and system specific IERs. From our point of view, the IEM is a type of model whose purpose is to enable semantic information exchange; with that we mean that both sender and receiver interprets information in same or "similar enough" manner.

3 The related work

In our work when developing IER we studied three approaches:

- IER from software development perspective
- National Information Exchange Model (NIEM)
- Architectural approach (NATO's Architecture Framework, NAF)

3.1 IER from software development perspective

In this section we consider that IERs are the subset/specialisation of requirements. This implies that all properties of requirements *are inherited* by IERs.

There are three types of requirements:

- functional
- non-functional
- business requirements

Business requirements relate to high-level business concern and are not mission specific (e.g., “keep the customer happy”, “improve product quality”). we describe Business requirements as a use case diagram that constitutes an example of business context. All the requirements presented exit on a high level conceptually.

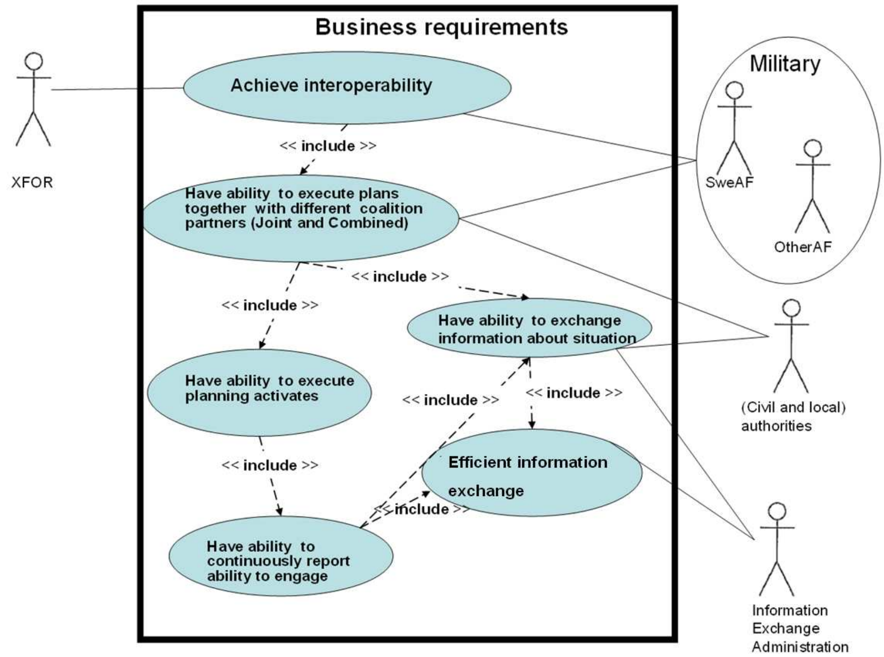

Fig. 2 Business Requirements

Fig. 2 represents stakeholders that have any *interest* in the business. Business requirements drive the development of functional requirements. All requirements [5] and consequently all IER have to be traceable back to the business requirements. If this is not the case for an IER it should be questioned why the requirement is necessary.

Functional requirements describe requirements that are typical for a system and define functionality of a system. In IERs it is hard to define from beginning which systems and in which situations are going to interact and exchange information.

Non-functional requirements *constrain* functional requirements. Example of Non-functional requirement is to use a particular standard, for information exchange like JC3IDEM.

Properties of requirements

Once requirements have been identified it is important that they are classified in some manner, for instance:

- source (where the requirement was originated)
- priority (e.g., ”essential”, ”desirable” , ”optional”)
- Verification and validation criteria (high-level information which gives brief idea on how to verify and validate the agreement between the requirement and product, in our case IEM)
- Ownership (who is responsible/owns the requirement; if there no owner of any stakeholder there might be some error in stake holder model)
- absolute reference (represents a unique key to allocate and identify the requirement; each part of the delivery in case the IEM should be traceable to the requirements)

3.2 National Information Exchange Model (NIEM)

The National Information Exchange Model (NIEM) is an XML-based information exchange framework from the United States (http://www.niem.gov/). NIEM is designed to facilitate the creation of automated enterprise-wide information exchanges which can be uniformly developed, centrally maintained, quickly identified and discovered, and efficiently reused [6]. NIEMs processes for building IEMs are focused on XML-level and are on data/object model [7] interoperability. NIEM with its processes and tools for development and integration of data components [6] could be important when developing IEM on semantic level.

3.3 Architectural approach

System/Enterprise/Software Architecture is commonly organized in views that are analogous to the different types of blueprints made in building architecture. An Architecture framework is a specification of how to organise and present a particular architecture. Nato Architecture Framework (NAF) is an example of an architectural framework. It defines a set of views and specifies a common meta-model that ensure consistency between views.

From our methodology point of view NAF gives following interesting views:

- a way to connect operational requirements and stakeholders to IER

- connection between IER, services and capabilities
- a standardised, but not extensive, connection of documented IER on operational level to IER on system specific level
- connection to higher level goals that information exchange enables
- it is easier to re-use and harmonise IEM, whose architecture is based on common architecture framework

The lack of architectural framework approaches is that they do not describe a process of traceability between changed IER and/or IEM.

4 Methodology for how to make an Information Exchange Model based upon IERs

This methodology explains how to build an IEM from IER in a uniform manner. The methodology is held on high level and should be understood by different stakeholders as for example system engineers, information modelers and operators. The benefit of the methodology is that it enables a way of dealing with new/modified IERs and modified IEMs. The methodology answers one of the key issues namely: How to document/deal with traceability between IER and IEM?

The methodology is explained in following sub-sections that explain relations between most significant classes and processes/activities.

4.1 Relations between classes

In this sub-section we visualize relations between classes that are relevant for information exchange. In Fig. 3 we present classes and relations between according to class-diagram of Unified Modeling Language (UML) [8] standard. The left part of the class-diagram shows class-IER relations in general and how IER can be documented and IER's different types. The right parts of the diagram, introduce those classes that are of essential importance for our methodology on how to create IEM based on IER.

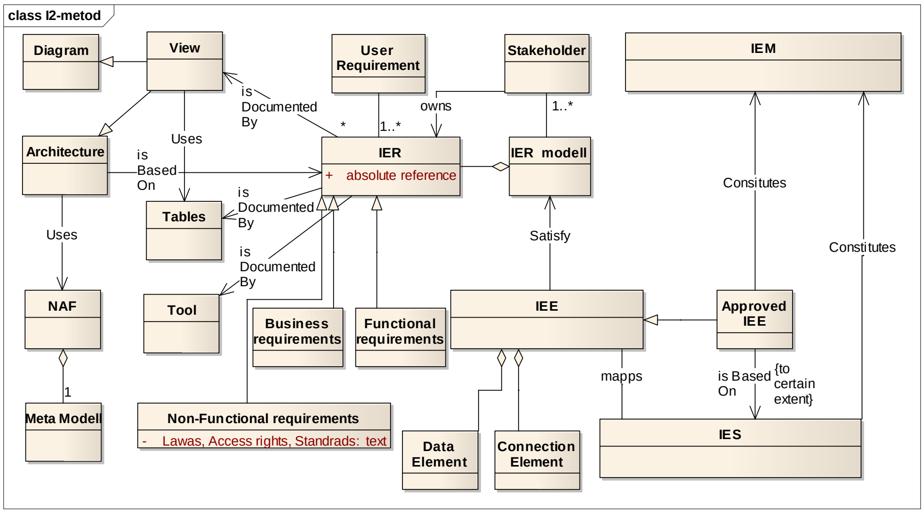

Fig. 3 Relations between classes

In following chapters we describe three parts that are important for the methodology:

- IER model
- Information Exchange Elements
- Relations to IEM

4.1.1 Information Exchange Requirements Model

The Information Exchange Requirements Model is constituted by a certain number of IERs. Each IER is *owned* by one or many stakeholders. In other words, there is no IER that has not at least one owner.

An IER can be functional, non-functional or a business requirement. All functional requirements should be related to a higher-level purpose that is expressed/encapsulated in business requirements. A documentation for traceability of requirements should contain whether information that each part of an IEM ought to be traceable back to business requirements.

IER can be documented in different manners. One way is to document IERs by text. Here we recommend that IERs are documented in an IER documentation tool. Other alternative/complement way is to document IERs by tables and architecture framework like NATO's Architecture Framework (NAF), [9], where consistency between views is ensured by its meta-model.

The absolute reference attribute play its role in identification of the IERs in different domains.

4.1.2 Information Exchange Elements

Here we introduce a new term and name it as Information Exchange Element (IEE).

Definition: An IEE *satisfies* a number of IERs (at least one) and is *represented* in such format (e.g., XML, OWL) that makes it possible to compare/map to information exchange models and standards.

In non-functional requirements there should be contained information on which information exchange standard should be applied. An IEE should have an attribute that we name here SatisfiedIERs. This attribute is a type of list and contains list of IERs that a particular IEE satisfies; in reality SatisfiedIERs is a list of IERs' absolute references. The content of IEE is description of entities and relations them between. In some cases relations are not contained in IEE due to the fact that they are not needed. This is the case when IEE is mapped to IEM or IES that does not contain relations. An IEE may consists of a number of meaningful parts called Data Elements and Connection Elements that describe connections between the Data Elements. One of the practical reasons is that an IEE can be easily adapted to changes by resuing Data Elements.

4.1.3 Relations to IEM

An IEM is created by identified entities and possible relations. Entities and relations can be contained in/originate from an IES and/or approved IEEs; by approved IEEs we mean a type of IEE that does not (completely) comply with IES but it has by some reason been approved to become a constitutive of an IEM.

4.2 Activities/Processes

In the previous section we introduced a class-diagram and we explained the meaning of the most important classes and relations among them. In this section we explain the methodology as a process that contains series of activities.

The main activities in the process are as follows:

- Formalization of IER
- Creation of IEEs
- Mapping IEE to IES
- Application for extension

- Creation of a new IEM/approval of an old IEM/ Harmonizing different IEEs with IEMs

The process includes a number of decision points where decision has to be made on how the future activities will proceed, see Fig. 4.

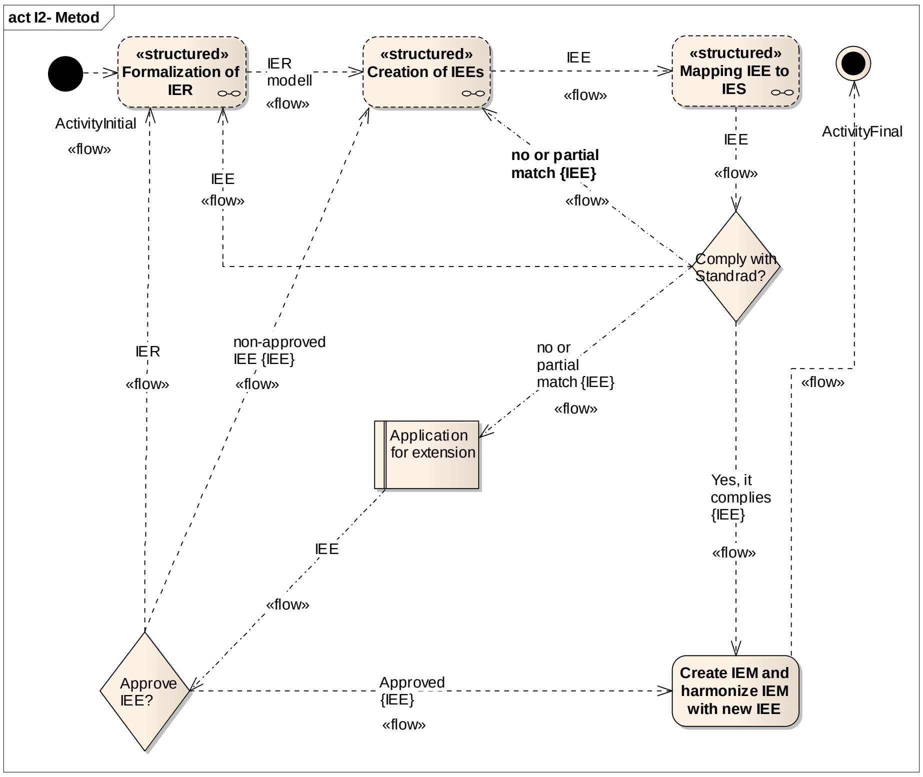

Fig. 4 Process/Activity Diagram

4.2.1 Formalisation of IER and Creation of IEE

The formalization of IER activity is to formalize and structure IER that eventually constitute an IER model. In creation of IEEs one seeks matches between “old” IEEs that comply with new IER. One way to create IEEs is to use tools that understand/can interpret IER language (the format on structured IERs) and can compare old IERs with new IER. The idea here is that each IEE know which attributes it satisfies. If search fails one has to create a new IEE.

4.2.2 Mapping IEE to IES

The activity "Mapping IEE to IES" aims to check and ensure compliance to IES. By IES we mean here both international standard but also approved organisational or cross-organisational standard. The first approach is to decide whether an IEE is a *subset* of an IES; e.g., in this activity we check whether a XML-schema of IEE is a subset of an IES. If this is the case then IEE become a part of the new IEM. To ensure traceability the new IEM has to be tagged with information on which parts are coming from which IEE.

4.2.3 Check complience

The next step is to decide if the IEE is a part of IES. If the IEE complies with IES then IEE will constitute the new IEM. If the IEE does not comply with IES there are two options. The first one is to suggest modification of IEE and the second one to suggest approval of IEE to eventually become a constitutive part the IEM. In the fist case IEE is sent to activity of creation of IEE. In the second case the request for approval will be issued. In the case request is approved then we obtain an approved IEE to constitute an IEM; in other case the request and IEE would be referred back to some of the activities.

4.2.4 Harmonising IEEs with IEMs

This section outlines a way of harmonising different IEEs with IEMs. In general, an IEE and an IEM can be harmonised on different levels and in different ways:

1. They are syntactically and semantically compatible, and can be connected without any major modification.
2. The one of them is adapted to the other one.
3. Build a "bridge" between them in terms of mapping rules (and convert the elements of them later during information exchange).
4. Create both of them from the beginning (if none of the alternatives above is applicable).

We focus on the alternatives 2 and 3, in fact, a combination of them, which is a typical situation in most cases. More precisely, IEE and IEM are harmonised through mapping rules, but also by adapting parts of the one the other. This leads to the following sub-phases:

1. Identify Data Elements (DEs) from the IEE.
2. Compare if these DEs exist already in the IEM and how are they interpreted

3. Identify which DEs are missing in the IEM. Complete the IEM with the missing DEs.
4. Identify the same concepts in the IEE and IEM.
5. Create mapping rules.

Among the phases above, only the phases 1, 4 and 5 will be addressed.

Firstly, identify DEs from the IEE. While an IEE represents a model external aspect (“black-box” perspective), a DE describes a model internal aspect (“white-box” perspective). For example, “requirement of exchange of situation awareness information within the ground domain” is an example of an IER and this requirement represented in an IEE. That is, a system functionality the can be observed from outside. This IEE can be decomposed to different DEs, e.g. terrain information, IER-link, sensor positions, etc, that are needed to realize the functionality. A DE can be decomposed further and the decomposition process continues until it is not only possible but also meaningful to establish a connection between Data elements in the models. See Fig. 5. Notice that IEEs are decompositions of the intended purpose of the IEMs.

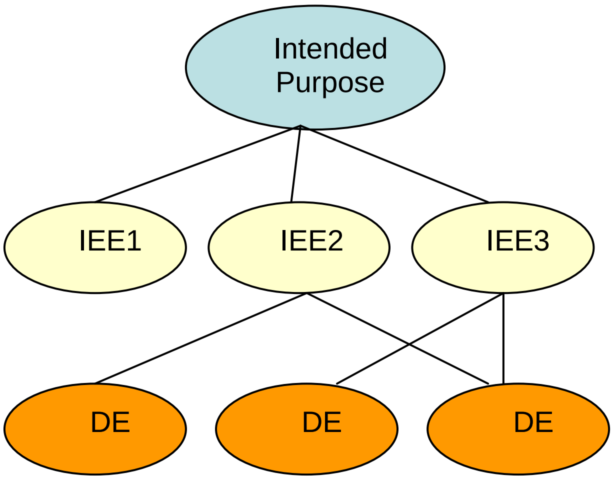

Fig. 5 Hierarchical relation between Intended Purpose, IEE and DE

Secondly, a natural way to harmonise and connect an IEE with an IEM would be through the concepts, i.e. DEs, that the models have in common. For assessing the degree of “being the same” of the DE, the following aspects are to be considered:

- Syntactic likeness, e.g. same names or same structure, may refer to the same phenomenon but not always (homonyms).
- Semantic descriptions such as the attributes, operations of the concepts and their interrelationship.

- Context; for example, SAAB and VOLVO are two different motorcar makes, i.e., they are syntactically and semantically different concepts. But in some circumstances, e.g., in the context of "transport of a person", they may be considered to have same semantic meaning.
- The IEE and IEM should be deeply analyzed as a whole.

Thirdly, create mapping rules. The mapping rules are recommended to be defined based on a common reference model, e.g. JC3IEDM. The advantage of using a common reference model to which both IEE and IEM have to adapt is:

- Semantic stability achieved
- Facilitates to maintain the harmonized constellation, see the Maintenance phase.

Once the IEE and IEM are harmonized, the harmonised constellation needs to be validated, i.e. it should be assessed whether the harmonisation performed satisfies its requirements and intended purpose. Validation of the mapping rules may be conducted differently depending on how the rules are structured. If possible, in addition to validation of the rules, "field test" may be done. That is, perform information exchange between the harmonised IEE and IEM and evaluate the exchange results.

Next step concerns maintenance of the harmonized constellation. A model is a representation of the Universe of Discourse (UoD). The UoD containing, e.g. organisation, processes, actors etc, may change over time. Accordingly the models are to be changed, and thereby even the established harmonisation. A very important issue to be raised here is an organisational one, since different stakeholders have their own models, view, interest, budget etc: Who decides how and when models should be changed, and how should the change be coordinated?

5 Use cases of the method

Here we present, in text, three use cases of the methodology:

1) case of new IER (this affects IEM)
2) some of IER are removed (this affects IEM)
3) new IEM and/or IES (how to deal with IER and old IEM)

5.1 Case 1: Case of new Information Exchange Requirements (IER)

The method proposed is incremental that means that is suited to take in new IERs without need of building new IEM every time new IER arrives.

5.2 Case 2: Some of IER are removed (this affects IEM)

Search of an IEE that satisfies that particular IER has to be performed. If the found IEE refers only to one (the particular) IER then the part of IEM corresponding to IEE can be removed; in the other case a possibility of fractioning IEE into Data elements that can be re-used when building a new IEE.

5.3 Case 3: New IES (how to deal with existing IER satisfied by an old IEM)

First step is to identify the relations and entities that does not exist anymore/are changed. This can be done automatically if both the new and the old IEM are described in a formal language. From the old IEMs documentation information of which IEEs would be obtained. Each IEE is supposed to refer to IERs it satisfies. This implies that the user can obtain the answer which IERs may not be satisfied new the new IES or IEM.

6 Conclusions

Here we address how to adapt/create an information exchange model (IEM) to/from information exchange requirements (IEM) and information exchange standards (IES) by incrementally developing an IEM from the ever-changing IER. The result of this paper is a methodology that is a first step in dealing with the named issue. The methodology can be used by different organizations and institutions as it is today. However, in order to be efficiently used, our methodology needs to be refined and tested on real-life scenarios. Moreover, a process/methodology for creating IEEs is required. The future challenge is to make context specific IEEs that can be used for a particular domain(s).

Acknowledgments

I would like to thank following persons for their invaluable support of this work: Truls Persson, Cecilia Unell, Linda Olofsson and Tord Pola.

References

1. Pinsdorf , U. and Volker, R., Mobile Agent Interoperability Patterns and Practice, In Proceeding of the 9th IEEE International Conference on Engineering of Computer-Based

Systems table of contents, Pages: 238 – 244, ISBN:0-7695-1549-5, Publisher IEEE Computer Society Washington, DC, USA (2002).
2. FMLS Architecture Design Framework, FMV, Sweden (2007).
3. Multilateral Interoperability Programme (MIP), URL: http://www.mip-site.org/
4. (British) Ministry of Defence, Information Exchange Requirements, URL: http://www.jsp600.ia.mod.uk/archive/ap/ap146.htm
5. Holt J., UML for System Engineering, UK, (2005).
6. NIEM, the National Information Exchange Model, http://www.niem.gov/
7. Tolk A., Beyond Technical Interoperability - Introducing a Reference Model for Measures of Merit for Coalition Interoperability, 8th CCRTS, National Defense University, Washington, USA (2003).
8. Eriksson H. E. and Penker M., UML Toolkit, ISBN 0-471-19161-2, John Wiley & Sons, USA, (1998).
9. NATO Architectural Framework (NAF), URL= http://194.7.80.153/website/book.asp?menuid=15&vs=0&page=volume2%2Fch02s06.html