Available online at www.sciencedirect.com
SciVerse ScienceDirect
Procedia Computer Science 16 (2013) 363 – 372
Procedia Computer Science

Conference on Systems Engineering Research (CSER'13)
Eds.: C.J.J. Paredis, C. Bishop, D. Bodner, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013.

On using multiple interoperating models to address complex problems*

Ahmed Abu Jbara[a], Alexander H. Levis[a], Abbas K. Zaidi[a]

[a] *System Architectures Laboratory, George Mason University, Fairfax, VA 22030, USA*

Abstract

Models, created using different modeling techniques, usually serve different purposes and provide unique insights. While each modeling technique might be capable of answering specific questions, complex problems require multiple models interoperating to complement/supplement each other; we call this Multi-Modeling. This Multi-Modeling approach for solving complex problems is full of syntactic and semantic challenges. In this paper, a systematic methodology for addressing Multi-Modeling problems is presented. The approach is domain specific: Identification of the domain and the supporting modeling techniques is the first step. Then a new Domain Specific Multi-Modeling Workflow Language supported with a Domain Ontology is used to construct the workflow that defines the interoperation of the selected models. The Domain Ontology provides semantic guidance to effect valid model interoperation. The approach is illustrated using a case study from the Drug Interdiction and Intelligence domain. The Joint Inter-Agency Task Force (JIATF) -South, an agency well known for interagency cooperation and intelligence fusion, receives huge amounts of disparate data regarding drug smuggling efforts. Analysis of such data using various modeling techniques is essential in identifying best Courses of Action (COAs). The proposed methodology is illustrated by creating workflows of model interoperations involving Social Networks, Timed Influence Nets, Organization Structures and Geospatial models.

Keywords: Multi-Modeling; Meta-Modeling; Model Interoperation; Domain Specific Workflow Language

1. Introduction

Traditionally, Modeling and Simulation (M&S) environments were designed with the assumption that a single type of models would be developed and analyzed. A model in such an environment is developed using some known

* This research was supported in part by the Office of Naval Research under Contract No. N00014-11-1-0129.

doi:10.1016/j.procs.2013.01.038

modeling techniques to address a certain class of problems. While single modeling techniques might be capable of answering specific questions, solving complex problems usually requires multiple models interoperating together (Multi-Modeling). The move towards supporting Multi-Modeling in various Modeling and Simulation platforms is already taking place. The Command and Control Wind Tunnel (C2WT) [1] developed by Vanderbilt University and the Service Oriented Architecture for Socio-Cultural Systems (SORASCS) [2] developed by Carnegie Mellon University are examples of Multi-Modeling capable platforms. While the first provides a federated approach, utilizing the High Level Architecture (HLA) [3] standard and the meta-programmable Generic Modeling Environment (GME) [4]; the second employs Service Oriented Architecture techniques in providing model interoperation capabilities.

In achieving Multi-Modeling, and to provide powerful supporting platforms, many challenges have to be faced. Beside the technical issues that usually arise in allowing interoperations between models through their modeling tools, there is also a major challenge of improving the human interface to the Multi-Modeling process itself [5]. This includes addressing both syntactic and semantic aspects of interoperation.

In this paper, a systematic methodology for addressing both syntactic and semantic issues in developing a Multi-Modeling approach to solve complex problems is presented. The focus of our approach is on helping users of Multi-Modeling platforms in designing workflows of Multi-Modeling activities that guarantee both syntactic and semantic correctness of the interoperations across models. Our approach is domain specific; the rationale behind this is twofold: first, problems to be solved by employing Multi-Modeling techniques are usually domain specific themselves; second, it narrows down the scope of meaningful interoperations among several modeling techniques where each technique offers unique insights and makes specific assumptions about the domain being modeled. We begin with identification and characterization of a domain of interest and its supporting modeling techniques. A Domain Analysis (DA) follows aiming to provide formal representations of syntactic and semantic aspects of the domain. A new Domain Specific Multi-Modeling Workflow Language is then developed to construct workflows that capture Multi-Modeling activities in the selected domain. A domain Ontology resulting from the Domain Analysis step is utilized to provide semantic guidance that effects valid model interoperation.

Our approach is illustrated in an application from the Drug Interdiction and Intelligence domain. The Joint Interagency Task Force - South (JIATF-South), an agency well known for interagency cooperation and intelligence fusion [6], receives huge amounts of disparate data regarding drug smuggling efforts. Analysis of such data using various modeling techniques is essential in identifying best Courses of Action (COAs). We apply our methodology to solve a class of problems in this domain by creating workflows of model interoperations involving Social Networks, Timed Influence Nets, Organization Structures, and Geospatial models.

In Section 2 we present a discussion of the basic concepts and approaches. The proposed methodology is presented in Section 3. Section 4 illustrates the application of the methodology. Conclusions and discussion of future work are in Section 5.

2. Multi-Modeling, Meta-Modeling and Workflows

Model interoperation has been addressed repeatedly, and from different perspectives, in the M&S research community. In this section we discuss some preliminary concepts and related approaches.

Modeling and Multi-Modeling: Modeling is the process of producing a model; a model is a representation of the construction and working of some situation of interest. [7] Fig. 1.a represents the modeling hierarchy where a Model is obtained using a Modeling Tool that applies a Modeling Language to represent a specific situation. The model itself should always conform to the Modeling Language used to create it. We call this combination of Model, Modeling Language and Modeling Tool a Modeling Technique.

We use multiple models because each Modeling Technique provides certain capabilities and makes specific assumptions about the domain being modeled. For example, Timed Influence Nets [8] describe cause and effect relationships among groups at high level but have no capability of capturing social aspects among the groups of interest. Social Networks [9], on the other hand, can describe the interactions among groups and members of the

groups. In this context, a Multi-Modeling approach addresses a complex problem through the use of a number of interconnected domain-specific models where each model contributes insights to the overall problem. The interoperations between the interconnected models could serve different purposes and can happen in various forms.

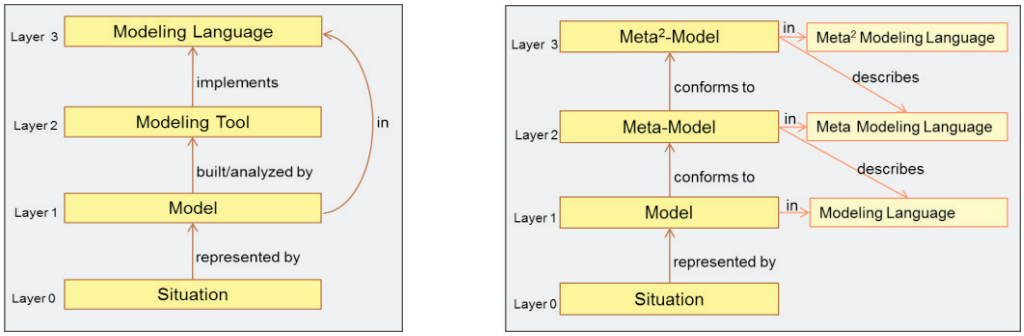

Fig. 1. (a) Modeling Hierarchy; (b) Meta-Models Hierarchy

Meta-Models and Meta-Modeling: A Meta-Model is an abstraction layer above the actual model and describes the Modeling Language used to create the model; the model has to conform to its Meta-Model. A Meta-Model conforms itself to a higher Meta-Model (Meta^2-Model) which describes the Meta-Modeling Language as shown in Fig. 1.b.

The typical role of a Meta-Model is to define how model elements are instantiated. Meta-Modeling is defined to be the process of constructing a Meta-Model in order to model a specific problem within a certain domain. In the context of this Multi-Modeling research effort, the Meta-Modeling concept is extended to include the analysis of the conceptual foundations of a model ensemble. These models interoperate as part of a workflow developed to address a specific problem. Meta-Modeling then becomes a process of constructing a Meta-Model of a Multi-Modeling Workflow Language that captures interoperations between models.

Multi-Modeling Workflows: Four layers to be addressed in order to achieve Multi-Modeling. [10] The first layer, the Physical layer, i.e., Hardware and Software, is a platform that enables the concurrent execution of multiple models expressed in different modeling languages and provides the ability to exchange data and also to schedule the events across the different models. The second layer, the Syntactic layer, ascertains that the right data are exchanged among the models. The C2WT [1] and SORASCS [2] achieve that. In the third layer, the Semantic layer, interoperation across models should be examined to ensure that the exchange of data is semantically correct with respect to the problem domain. The fourth layer, the Workflow layer, is where workflows of interoperating models are captured.

A Multi-Modeling workflow is itself a model of an analysis process. A formal approach to capture a Multi-Modeling workflow requires a formal Modeling Language with its own rules. Developing workflows using such approach allows for translating visual views of model interoperation into an executable implementation. There already exist generic techniques for designing and implementing workflows such as Business Process Model and Notation (BPMN) [11] and Business Process Execution Language (BPEL) [12]. The domain specific nature of our approach requires us to develop a Domain Specific Multi-Modeling Workflow Language for the selected domain of interest. Such a language would be tailored to a problem domain and would offer a high level of expressiveness and ease of use compared with a General Purpose Language (GPL) [13] and can be a specific profile of an existing GPL, i.e., BPMN. Fig. 2 shows the mapping between our proposed Domain Specific Multi-Modeling Workflow Language (and its Meta-Model) to the Meta-Models Hierarchy.

Defining the Meta-Model of the workflow language in Layer 2 in Fig. 2 is a Meta-Modeling process itself. To capture those constructs of the Meta-Model that define the new language, a Meta-Modeling Language that conforms to a higher Meta-Model, Meta^2-Model, is also required. The research community in this area has addressed such hierarchy from different perspectives and many approaches were developed. One of these approaches is the Generic Modeling Environment (GME) [4], a configurable toolkit for creating domain-specific modeling languages and program synthesis environments, developed by Vanderbilt University. The configuration is accomplished through

Meta-Models specifying the modeling paradigm (Modeling Language) of the application domain. The modeling paradigm contains all the syntactic and presentation information regarding the domain including the concepts used to construct models, relationships between concepts, different views and organizations of the concepts, and rules governing the modeling process. Defining the modeling paradigm is a modeling activity itself. GME has a Meta-Modeling paradigm that configures the environment for creating Meta-Models of the domain of interest. These models of the Meta-Models are then automatically translated into GME configuration through model interpretation. The Meta-Modeling paradigm is based on the Unified Modeling Language (UML).

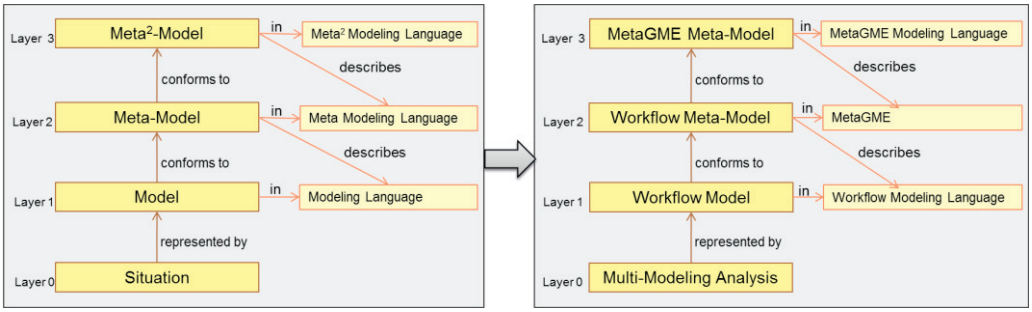

Fig. 2. Mapping Our Domain Specific Workflow Language to the Meta-Models Hierarchy

Ontologies: Ontology is the term used to refer to the shared understanding of a domain of interest. It entails some sort of a global view of a specific domain. This view is conceived as a set of concepts, their definitions and their inter-relationships; this view is referred to as conceptualization. In computer systems ontologies can be thought of as a means to structure a knowledge base. [14]

A Multi-Modeling Workflow Language needs a mechanism that guarantees semantic correctness of model interoperation. We propose the use of ontologies to guide interoperations between models. A Domain Specific Ontology that represents possible mappings between different concepts in the domain serves this purpose. The use of ontologies as a mean for representing the semantic aspects of model interoperability has been explored in [10] and [15]. The first approach is based on comparing ontologies (for each Modeling Technique) to help identify the similarities, overlaps, and/or mappings across the models under consideration and then constructing a higher level "Meta" ontology that determines which sets of models can interoperate. The second maintains a clear distinction between Meta-Models and Ontologies; they are different but complementary concepts, and both are needed to allow for model interoperation. We employ concepts from these two approaches in our methodology.

Domain Identification: Since the proposed Multi-Modeling approach for solving complex problems is domain specific, domain characterization becomes an essential task to be conducted prior to any other activity. The output of the domain characterization should provide enough information to perform domain analysis, construct domain ontology, and develop a Meta-Model of a Domain Specific Multi-Modeling Workflow.

In the context of software and systems engineering, a domain is most often understood as an applications area, a field for which systems are developed [16]. It is also defined to be a class of problems, where the types of problems to be solved and the context in which the system elements can be used are clearly identified [17]. In our approach, we consider a domain to be a specific class of problems to be solved using a set of Modeling Techniques and the appropriate required data.

The domain identification process itself has been approached in many research efforts, specially the research on software reusability in late 80's and early 90's. In [17] a comparison of Domain Analysis (DA) approaches for software reuse purposes was presented. Domain identification was pointed out as a first and essential step prior to any DA activities. Domain identification methods in those approaches include informal description in the form of statements, use of object oriented techniques, employing classification schemes, determining domain boundaries and collecting examples of similar problems.

3. The Multi-Modeling Approach

The focus of our approach is to provide a systematic methodology for creating and implementing Multi-Modeling Workflows that are both syntactically and semantically correct. It consists of five major steps as shown in Fig. 3. In this section we will discuss each step and its sub-steps in detail.

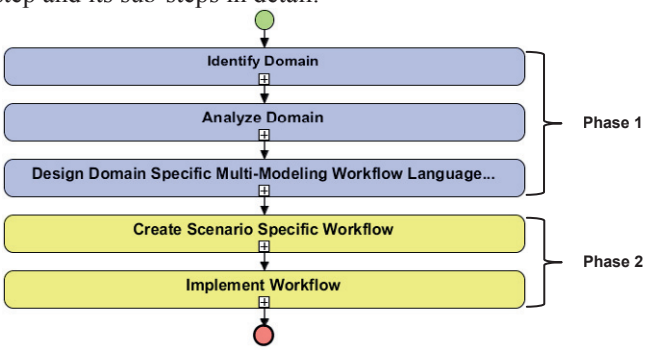

Fig. 3. Overview of the proposed methodology

Domain Identification: This is the first step which deals with characterizing a specific domain of interest, in which, interoperating models are used to solve certain problems. We address the domain identification challenge by employing different techniques. As shown in Fig. 4.a, we begin with an informal description of the domain in the form of statements that try to identify the problems to be solved, Modeling Techniques usually used in solving these problems, data sources and types, and main actors involved including domain experts, modelers and analysts.

After that we proceed with deciding the domain boundary in order to scope the domain and to exclude any unrelated elements. Then, a classification of concepts applicable to the domain takes place. These concepts serve as a repository for the final step. In the final step, Concept Maps [18] are used to represent those concepts. Concept mapping is a representation technique to organize knowledge about a specific domain. In our approach, we consider Concept Maps as a semi-formal representation of the domain. Generating Concept Maps is an iterative process until a satisfactory domain representation is reached.

Domain Analysis: Once satisfactory Concept Maps that represent the domain of interest and its supporting Modeling Techniques are ready, the Domain Analysis (DA) process starts. The process, as shown in Fig. 4.b, goes into two parallel, but complementary, paths. On the outer path, UML class diagrams derived from the concept maps are produced to capture the structural aspects of the domain and its supporting Modeling Techniques. A mapping between these class diagrams follows to produce consolidated diagrams that include interoperations between the Modeling Techniques. On the inner path, ontologies based on the concept maps of the Domain and the Modeling Techniques are constructed to capture the semantic aspects. These ontologies are represented using the formal Web Ontology Language (OWL) standard. Mapping of these ontologies follows by employing Upper Ontology [19] and Ontology Matching [20] techniques.

Domain Specific Multi-Modeling Workflow Language: A Meta-Model of the new language has to be created and it should include the set of fundamental language constructs that represent the essential concepts of the domain, the set of valid relationships that exists between the domain concepts, and a set of constraints that govern how the language constructs can be combined to produce valid models. Accordingly, in the third step of our methodology, the UML class diagrams obtained from the DA step are used as the basis for the Meta-Model that defines the Domain Specific Multi-Modeling Workflow Language. The GME is used to create the Meta-Model of the Multi-Modeling Workflow Language. This Meta-Model is then automatically translated into a GME configuration that allows the use of GME itself to create workflows of specific Multi-Modeling scenarios. In general, we propose the use of a profile of BPMN as the basis of any Domain Specific Multi-Modeling Workflow Language.

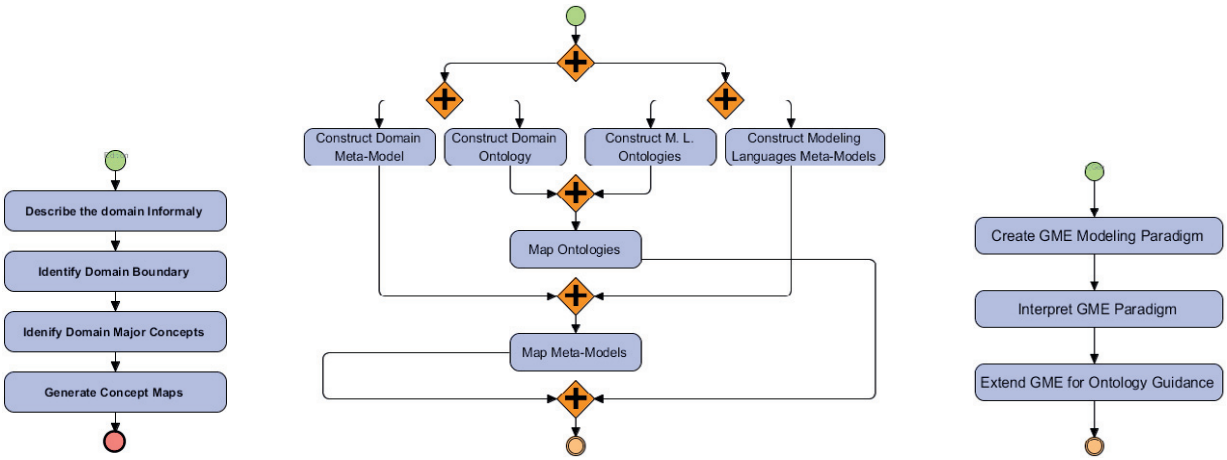

Fig. 4. (a) Domain Identification (b) Domain Analysis (c) Multi-Modeling Workflow Language Definition

Semantic Guidance of Multi-Modeling Workflows: The semantic concepts identified in the domain identification process and then captured in the Ontology in the domain analysis step should be enforced while using the new Domain Specific Multi-Modeling Workflow Language. Since our ontologies are represented in OWL [21] and we are using GME to create Multi-Modeling workflows, there should be a way to allow OWL ontologies to guide the creation of workflows; that is to guarantee their semantic correctness. GME allows for different types of extensions to the environment; basically using Plug-ins or Add-ons [4].

Utilizing these GME extensibility features and in order to address the semantic guidance issue, we implemented a GME Add-on extension. This extension reacts to GME events, and in case of any interoperation connection while using our Multi-Modeling Workflow Language, the OWL ontology is checked on the semantic validly of this connection. We use SPARQL [22] queries that are passed to a SPARQL Query Server to query the ontology. Based on the query result, our GME extension could allow or disallow the interoperation connection.

Multi-Modeling Workflow Creation: After defining the domain specific Multi-Modeling Workflow Language and having its GME modeling paradigm interpreted, GME can be used to create workflows for specific situations of interest. This is the fourth step of our methodology. Workflows constructed with our domain specific workflow language to capture interoperations across models are guaranteed to be both syntactically and semantically correct.

Workflow Implementation: The final step is to implement our workflows in an appropriate platform. In order to achieve this, an interpretation of the workflow to an executable form is required. For this purpose, a GME interpreter can be coded. One example platform that can be used to execute our workflows is the SORASCS [2]. SORASCS utilizes BPEL to execute workflows of analysis activities. Since we proposed basing our Domain Specific Multi-Modeling Workflow Language on BPMN, which can be mapped to BPEL [23], workflows created using our Multi-Modeling Workflow Language can be converted to executable workflows that SORASCS can execute.

Two Phase View of the Approach: The overall process of the methodology can be viewed as a two phase approach. Phase 1 is where the first three steps, domain identification, domain analysis, and workflow language definition take place. For a specific domain, this phase goes into multiple iterations until a Multi-Modeling Workflow Language that addresses a domain of interest and is capable of capturing model interoperations is reached. Phase 2 takes place when the Workflow Language is used to create workflows for specific scenarios. It is always possible to go back to Phase 1 to refine and enhance the Multi-Modeling Workflow Language; this might be the case when a new Modeling Technique is introduced in the domain of interest.

4. Application: JIATF-South

In this section we present an application of our approach to a decision making problem in the Drug Interdiction domain. The Joint Interagency Task Force - South (JIATF-South) is a Drug Interdiction agency well known for interagency cooperation and intelligence fusion [6]. The agency usually receives disparate data regarding drug trafficking and drug cartels from different sources. Quick and effective analysis of data is very essential in addressing drug trafficking threats effectively. A typical case begins with JIATF-South receiving information form the Drug Enforcement Administration (DEA). This prompts the deployment of Unmanned Airborne Vehicles (UAVs) that subsequently detect and monitor a suspect vessel until JIATF-South can sortie a Coast Guard cutter to intercept. If drugs are found, jurisdiction and disposition over the vessel, drugs and crew are coordinated with other agencies. Courses of Actions (COAs) identified by the agency are dependent on efficient analysis of received data. In order to proceed in applying our approach, we first present a fictitious scenario of a possible drug trafficking activity reported to and monitored by JIATF-South. The scenario is presented briefly in Fig. 5.

- A US based drug cartel is involved in drug trafficking activity from Country R in the Caribbean to the USA.
- A cargo ship with R flag is being loaded with drugs. A drug cartel operating in country R is responsible.
- JIATF-South receives information about the drug smuggling activity from its intelligence sources in country R.
- The cargo ship disembarks the port of country R on Day x and goes under JIATF-South surveillance in international waters starting Day x+1. The ship is scheduled to arrive to the USA on day x+5.

Fig. 5. Scenario Brief

Analysts at JIATF-South are trained to use various Modeling Techniques to analyze data and then to identify possible COAs. In a traditional manner, analysts would be using each modeling technique individually. By applying our methodology, an efficient Multi-Modeling based analysis would make such analysis process more accurate and faster. The rationale is that while each Modeling Technique might be capable of capturing certain aspects of the available data, interoperation between models will definitely improve the results of the overall process. Also, the ability for analysts to create visual workflows of the Multi-Modeling activity provides a mean of reusability of the constructed workflows in addressing similar scenarios. In this section we show a practical example of applying our 5-steps methodology to the JIATF-South drug interdiction scenario.

Domain Identification: We first begin with an informal description of the domain. Looking back at the JIATF-South operations description and the brief scenario, the following partial list of statements shown in Fig. 6 describes the main concepts of the domain.

- Drug Interdiction involves information sharing, fusion of intelligence data and monitoring of drug trafficking activities.
- Given (incomplete and uncertain) information, decisions to be made on best COAs.
- Drug Interdiction involves dealing with Drug Cartels and Smugglers (RED groups) and Law Enforcement and Intelligence (Blue groups).
- Drug Smuggling takes different routes and originates from different sources.
- Analysts use Social Networks, GIS, Influence Nets, Asset Allocation and Scheduling and Organization Models techniques.

Fig. 6. Informal Description of Domain

These informal statements are then revised to scope the domain and exclude any concepts that are outside its boundary. In this example, the Asset Allocation and Scheduling problem is not addressed. A repository of related concepts is then identified. Table 1 shows examples of some related concepts. The concepts are classified into two major categories, Domain Concepts, and Modeling Techniques Concepts.

Table 1. Domain and Modeling Techniques concepts

General Domain Concepts	Specific Domain Concepts	Modeling Techniques Concepts	Specific M. Techniques Concepts
Drug Interdiction	Drug Smuggling Drug Cartels	Geospatial Analyses	Incidents, Time Location, Route WebTas
Data	Geospatial Time Individuals	Influence Nets	Node Link Proposition Probability
Interagency Collaboration	Intelligence Agencies Law Enforcement Agencies		

After identifying related concepts, we construct Concept Maps to capture the relations between these concepts. Concept Maps are generally constructed to answer specific questions in the domain of interest. Fig. 7 shows a concept map that addresses the question: How does JIATF-South perform Drug Interdiction? The same applies to other aspects of the domain and the Modeling Techniques. As part of the process, we refer to similar experiences and make use of existing assets. In [10], concept maps for Influence Nets and Social Networks were constructed.

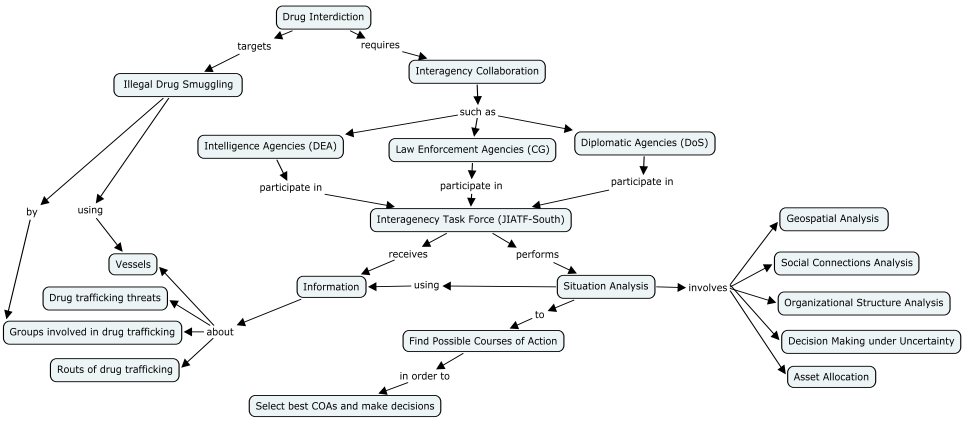

Fig. 7 Concept Map: How does JIATF-South perform Drug Interdiction?

Domain Analysis: In this step, we use the generated Concept Map to perform domain analysis. We construct UML class diagrams to represent the constructs of the domain and the Modeling Techniques. Parallel to that, we identify semantic concepts and relations and capture them in our OWL Ontology. In Fig. 8, we show a partial UML class diagram that represents the constructs of the drug interdiction domain.

Domain Specific Multi-Modeling Workflow Language: Using GME, a Meta-Model for our domain's Workflow Multi-Modeling Language is defined. This Meta-Model defines the constructs of this new language. In addition to basic constructs borrowed from BPMN, we have introduced some new constructs and imposed some constraints. A workflow in our domain has two types of activities, operations and interoperations. Operations are those activates performed on a specific model using the modeling tool that supports its modeling language. Interoperations are those activities that involve interoperations across models through their modeling tools. Operations in our language can be in one of two flavors, Thick or Thin Operations. This is due to the fact that Multi-Modeling platforms can support the integration of Modeling Tools in one of two forms. Thin Operations represent the case when service based integration takes place, given that the modeling tool of interest exposes its functionalities as services. Thick Operations represent the case in which the whole Modeling tool is integrated as a package in the Multi-Modeling platform.

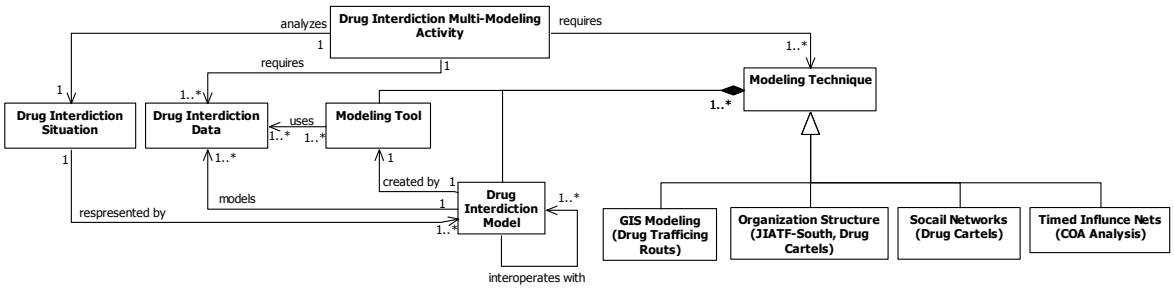

Fig. 8 UML Class diagram representing main constructs of the drug interdiction domain

Multi-Modeling Workflow Creation: Once the GME Meta-Model of our Domain Specific Multi-Modeling Workflow Language is interpreted and registered as a new Modeling Paradigm in GME, we begin using the GME environment to create workflows that capture specific domain scenarios. In Fig. 9 we show a workflow that involves the use of GIS, Timed Influence Nets, Social Networks and Organization Models to analyze data and then generate and select best COAs.

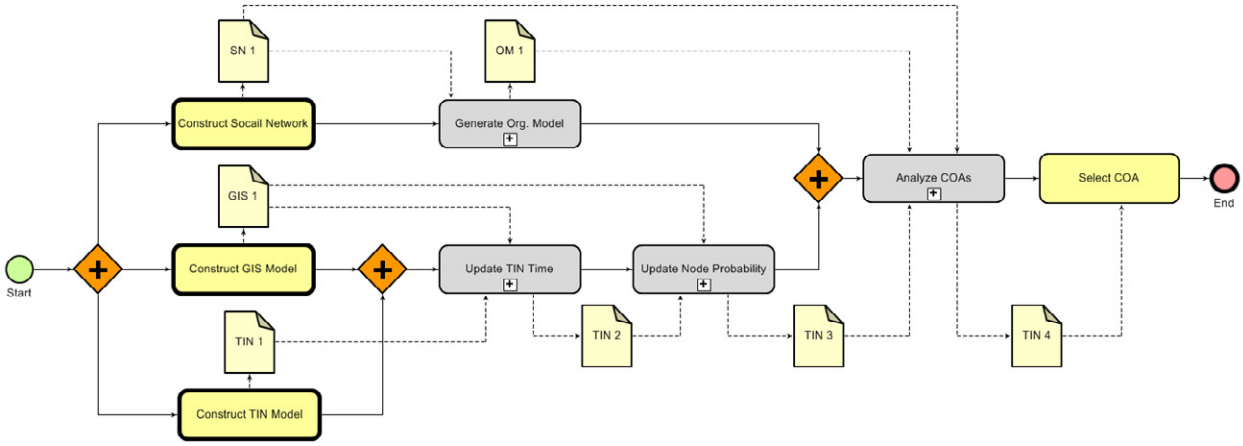

Fig. 9. Workflow of a Drug Interdiction Multi-Modeling Activity

5. Conclusion And Future Work

In this paper we presented a systematic methodology for addressing Multi-Modeling problems by employing a Domain Specific Multi-Modeling Workflow Language and a supporting Domain Ontology. Our approach is domain specific and requires the characterization of a specific domain, modeling techniques used in solving problems in that domain, and data sources available for creating models of specific scenarios in that domain. Domain characterization is a complex problem by itself. It has been addressed in our proposed methodology by building on top of previous research in this area. We believe that this is an area that deserves more future attention as it is essential for making our approach capable of capturing different possible combinations of Multi-Modeling activities for a specific domain.

The utilization of a Domain Ontology to guarantee the creation of semantically correct Multi-Modeling workflows is another area of our research that requires more attention. Creating a Domain Specific Ontology is an iterative process that requires domain experts' knowledge and advanced techniques to match and combine low-level Ontologies into higher-level "Upper" Domain Ontology. This would be a major focus in our future research.

References

1. G. Hemingway, H. Neema, H. Nine, J. Sztipanovits, and G. Karsai, Rapid Synthesis of High-level Architecture-Based Heterogeneous Simulation: a Model-Based Integration Approach, Simulation, (2011).
2. D. Garlan, K. M. Carley, B. Schmerl, M. Bigrigg, and O. Celiku, Using Service-Oriented Archi-tectures for Socio-Cultural Analysis, Software Engineering and Knowledge Engineering, (2009).
3. HLA Evolved Working Group. (2010) IEEE Standards Association. [Online]. http://standards.ieee.org/findstds/standard/1516-2010.html
4. James Davis, GME: The Generic Modeling Environment, Conference on Object-oriented programming, systems, languages, and applications , (2003).
5. Paul A. Fishwick, Toward an Integrative Multimodeling Interface: A Human-Computer Interface Approach to Interrelating Model Structures, Simulation, (2004).
6. Evan Munsing and Christopher Lamb, Joint Interagency Task Force - South: The Best Known, Least Understood Interagency Success, 2011.
7. A. Maria, Introduction to Modeling and Simulation, Proceedings of the 29th Winter Simulation Conference, (1997).
8. S. Haider and A. H. Levis, Effective Course-of-Action Determination to Achieve Desired Effects, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, vol. 37, no. 6, pp. 1140 - 1150 , (2007).
9. K. Carley, On the Evolution of Social and Organizational Networks, Steven B. Andrews and David Knoke (Eds.) special issue of Research in the Sociology of Organizations, vol. 16, pp. 3-30, (1999).
10. Alexander H. Levis, Abbas K. Zaidi, and Mohammad F. Rafi, Multi-modeling and Meta-modeling of Human Organizations, 4th International Conference on Applied Human Factors and Ergonomics, (2012).
11. OMG. (2011) Business Process Model and Notation. [Online]. http://www.omg.org/spec/BPMN/
12. OASIS. (2007) Business Process Execution Language. [Online]. https://www.oasis-open.org/committees/wsbpel/
13. M. Mernik, J. Heering, and A. M. Sloane, When and how to develop domain-specific languages, ACM Computing Surveys (CSUR), vol. 37, pp. 316–344, (2005).
14. M. Uschold, M. Gruninger, M. Uschold, and and M. Gruninger, Ontologies: Principles, Methods and Applications, Knowledge Engineering Review, vol. 11, pp. 93-136, (1996).
15. P. Höfferer, Achieving business process model interoperability using metamodels and ontologies, 15th European Conference on Information Systems, (2007).
16. Ruben Prieto-Diaz, Domain Analysis: An Introduction, Software Engineering Notes, vol. 15, no. 2, p. 47, (1990).
17. X. Ferré and S. Vegas, An Evaluation of Domain Analysis Methods, 4th International Workshop in Evaluation of Modeling Methods in Systems Analysis and Design, (1999).
18. Joseph D. Novak and Alberto J. Cañas, The Theory Underlying Concept Maps and How to Construct and Use Them, 2006,.
19. I. Niles and A. Pease, Towards a standard upper ontology, International conference on Formal Ontology in Information Systems, (2001).
20. K. Kotis and M. Lanzenberger, Ontology Matching: Current Status, Dilemmas and Future Challenges, International Conference on Complex, Intelligent and Software Intensive Systems, (2008).
21. Sean Bechhofer et al. OWL Web Ontology Language. [Online]. http://www.w3.org/TR/owl-ref/
22. W3C. (2008) SPARQL Query Language for RDF. [Online]. http://www.w3.org/TR/rdf-sparql-query/
23. C. Ouvans, From BPMN Process Models to BPEL Web Services, International Conference on Web Services, (2006).