Abstract

It is important for a systems architect to be able to "zoom in" and "zoom out" different levels of details, i.e., to see the enterprise and business level views as well as to go into details of systems and different components of systems. The enterprise knowledge and information systems are two fundamentally different disciplines. An intermediary approach that can connect the two bidirectionally can be very useful in such scenarios. This short paper proposes an intermediary modeling approach that can provide means to computer assisted creation of higher-level business architectural artifacts, that can hold business meaning for the upper management, as well as information system meta-model descriptions that can be used for MBSE (Model Based System Engineering) approaches to develop concrete software systems. This would allow enterprise planning at intermediary level to provide upstream and downstream independence. This preliminary research uses a computer aided planning environment that has repository of multi-disciplinary patterns at different levels of detail for rapid development of these models, capturing various views and viewpoints in a step by step manner.

Keywords: Enterprise Modeling; Decision Support System; Enterprise Architecture Frameworks; Model Based System Engineering

1. Introduction

Models are conceptual representations of reality. Enterprise models are models that try to capture the conceptual representation of an organization. A great deal of research in enterprise modeling has been reported in the
As enterprises are complex multi-domain multi-level entities, so are the models that need to represent diverse enterprises. Collectively, different models of an enterprise can create a complete architectural description of an enterprise. This architectural description can hold enterprise elements, information elements and system elements of an organization. The two important aspects of these architectural descriptions are:

1) the enterprise or business architecture that holds the information about the organizational goals, its environment, people, processes, business functions and its governance concerns, and
2) the technical or information system architecture, that has the structure of information and its interactions, the partitioning and allocation of functionality, the infrastructure in terms of communications network, hardware/software platform and the applications and software solutions.

Researchers have categorized different aspects of the enterprise architectural description in numerous ways, but, afore mentioned partition, illustrated in Figure 1, is based on interdisciplinary nature of artifacts that are used for the purpose in practice.

Computer aided development of enterprise models using multi-level models and patterns is an interesting area of work. Model Driven approaches based on these patterns are suggested in this short paper to develop and maintain information system models. First different enterprise models compliant to many different enterprise architectural frameworks are generated, and then model transformations are discussed, as shown in Figure 1.

2. Background

Information systems alone cannot provide long-term sustainable strategic advantage to the enterprise. They have to perform under the guidance of higher level environment to achieve business goals. This suggests that the enterprise or business architecture serves as the source of the requirements that the information systems need to fulfill. The enterprise architecture provides many viewpoints to different stakeholders.

There are many different and diverse approaches and frameworks for describing Enterprise Architecture. A number of Frameworks are developed for a specific industry domain in mind. For example, manufacturing has IDEF23 (Integration Definition method), TOVE15 (Toronto Virtual Enterprise), CIMOSA13 (Open System Architecture for CIM), PERA32 (Purdue Enterprise Reference Architecture) and GRAI-GIM7 (GRAI Integration Methodology).

Government, on the other hand, has FEAF8 (Federal Enterprise Architecture Framework), TEAF12 (Treasury Enterprise Architecture Framework), xGEA6 (cross-Government Enterprise Architecture), NORA18 (Nederlandse Overheid Referentie Architectuur) and GEA-NZ11 (Government Enterprise Architecture Framework for New Zealand) developed and maintained by different governments or departments. For defense we have DoDAF10 (Department of Defense Architecture Framework), MoDAF24 (Ministry of Defence Architecture Framework), AGATE9 (Atelier de Gestion de l’Architecteur) and NATO Architecture Framework25. In addition, several frameworks for different industry domains are evolving. Several EAFs (Enterprise Architecture Frameworks) are available that are not specific to any domain such as Zachman33, TOGAF31 (The Open Group Architecture Framework) by open group, GERAM14 (Generic Enterprise Reference Architecture and Methodology) and GEAM2 (Gartner Enterprise Architecture Method) by Gartner.
The diversity of these frameworks suggest the heterogeneous nature of enterprise knowledge that needs to be described for business or strategic value, and therefore can be complex to fully grasp in one framework. A study evaluated different architectural frameworks and concluded that no single framework can address all needs of an enterprise. In practice, organizations tend to develop an EA program to determine which architectural framework can help their organizational goals and how. This leads to an industry trend of developing hybrid enterprise architecture approaches. A survey found that most enterprise architecture programs chose hybrid approach for architectural description to gain individual framework advantages like best-fit model, compliance to industrial architecture frameworks, flexibility/interoperability and better IT alignment.

Enterprise Modeling approaches tend to focus on different aspects of the enterprise for different reasons. We here focus on two reasons for modeling the enterprise: 1) managing the enterprise and 2) developing and maintaining the IS for the enterprise as emphasized by many researchers. To combine the enterprise architecture concerns and information system development concerns, the enterprise modeling seems to be at the right level to address both.

3. Creating intermediary models using a decision support system

The Authors of this paper are developing a decision support environment that heavily uses business and technology patterns and creates multiple views of an enterprise based on user input and information captured in related views. In this section, generation of these models is elaborated.

3.1. Patterns driven decision support system.

The planning environment is not one expert system, but a collection of intelligent decision support systems that sequentially collaborate with each other. This sequence, as well as the configuration of advisors depend upon the type of the problem user needs to solve. Main components of the system, depicted in Figure 2, are elaborated in this section.

User Interface (Interviews): acts as the controller, which helps user go through with the sequence and configuration of the advisors in the system. It provides user with the ability to enter the User Requirements (UR) or User Input for the advisor, which are also intelligently guided by the advisor. In return, it gets a Presentation Model (PM), which itself holds strategic value to the user.

Integration Bus: Integration bus is the connector and can be thought of as the blackboard in blackboard architecture style, it is based on simple shared data access and connects all the advisors with each other as well as with the user. The data elements as shown in the Figure 2 are UR, PM and OM (Object Model). PM can be viewed as the user friendly presentation or any industrial standard presentation of the OM. The OM is a specialized representational
model that the system works with. It will always follow a schema to facilitate integration with other applications or advisors. One $OM$ may contribute in production of multiple models.

Advisor: An advisor ($ADV_i$) based on the input knowledge provided by users ($UR_i$) and by previous advisors ($OM_{i-1}$), produces $PM_i$ that is valuable to the user, and $OM_i$ that could be valuable to later advisors in the stream. So an $OM$ is enriched with input from each advisor, and this can be described as the following relationship:

$$OM_i = f(UR_i, OM_{i-1}, PR_i)$$

The $OM_{i-1}$ is heavily based on the user input $UR_i$. Each Advisor makes many inferences but the user has the ability to override the default values, hence $OM_i$ is influenced by both $UR_i$ and $OM_{i-1}$.

Patterns Repository: Patterns repository is systematically organized repository of business and technology patterns. A Pattern $P_i$ is identified from the repository $PR_i$ for relevance to the requirement ($UR_i$ and $OM_{i-1}$) by the $ADV_i$, customize and generate a model $OM_i$ out to the integration bus.

This model elicitation interview progresses on step by step basis, and in some scenarios in multiple iterations, until all of the information for that scenario is captured in these models. The schema ensures that the information stored in these models is valid, while linking to the knowledge in the previously generated models allows making inferences in an incremental manner, so that the new model is coherent with respect to the previous one.

Using the approach mentioned above we can capture a vast knowledgebase. In order to be effectively used with model driven approach, these models need to have certain properties. These properties can be assured by using meta-models that provide the grounds for creating semantically correct and reusable models. Moreover, the design of meta-models should be such as to support the Query, View and Transformation (QVT) capabilities.

4. Example: Modeling of a Healthcare Enterprise

Let us take a scenario to develop enterprise plans for a healthcare organization. We go from top level to detailed level model generation. At the top level we develop the enterprise context by eliciting the high level organizational view. This will hold the information about the business processes grouped into high level business functions, the people and high level objectives of the enterprise. Figure 3 shows the EM (Enterprise Model) diagram that captures the healthcare specific processes such as healthcare administration and healthcare services (e.g., patient care). It also captures the general administrative processes such as customer services, sales, marketing, finance and accounting, and corporate management. In addition, the operational processes of warehousing and supply chains for medicines, medical equipment, and patient comfort (e.g., beds) are included in this EM.

![Fig.3. Enterprise Model (EM) Diagram for a Healthcare Enterprise](image)

The EM can now be partitioned where we allocate different processes into different organizational units and the topology of the enterprise. For example, three different buildings could house the healthcare processes, the general administrative processes, and the operational processes.
After these steps, the application and platform models are developed that take into account the groupings of enterprise units and Information system needs. A communication network model can also be developed based on the location of the organization units. For example, the three buildings located in three different cities may have their own local area networks that are interconnected through a wide area network that serves as an enterprise integration bus, as discussed below. These technical models are developed for each and every organizational unit and are based on the configuration of business processes as captured in the EM.

![Integration model for Healthcare information Network](image)

After these individual models are created then the integration model is developed. This will take into account integration at each enterprise level, from unit to enterprise wide to external resources. This integration bus, shown in Figure 4, enriches the network model developed previously to emphasize the integration of applications across the enterprise by using the principles of Service Oriented Architectures (SOA).

At the end of these interviews, the system will have information about the enterprise, its business processes model, stakeholders, the information models, the interactions, the information systems, the integration choices, the network technology architecture, the governance and deployment choices.

This repository of models specific to one scenario can be linked with other scenarios to make composite structures for larger enterprises, or as service bundles for very large initiatives like health information exchanges. These compositions can be between multiple industrial domains and among multiple enterprises as well.

The multi-view knowledge represented models generated can now be further transformed into more formal structure for model to model and model to code transformations as illustrated in Figure 5. This requires meta-models to be defined, discussed in next section, for different upstream/downstream modeling.

![Model driven Upstream and downstream artifacts](image)

5. Future Directions and Conclusions

Developing downstream artifacts using upstream architectural descriptions is a challenging task. Some of the challenges are the informal structure of enterprise architectural data, narrow scope of enterprise modeling approaches, inadequate tools support, out of context system models, industrial domain specific meta models and MDA (Model driven Architecture) support. This paper has proposed an approach that can harness the potential of enterprise models developed using a decision support tool based on business and technology patterns. The approach presented can be
extended to hold diverse knowledge about a diverse range of enterprises in a number of scenarios in forms of Advisors that use Patterns. The models generated are content rich and broadly scoped. The Model generated by an Advisor uses the Model(s) generated by previous Advisor(s), with some user input. So the model to model transformation is automatically accomplished by the Advisors.

More domains and knowledge concerns can be elicited using the same approach i.e., defining the rules, designing the patterns and developing the advisors. Our current research is focusing on making the content of the generated models semantically rich to enable extensive transformations. Our goal is to augment the intermediate models with the downstream system models and upstream business models. The business content models such as Archimate business process modeling, DoDAF views and FEAF Reference architectures need better transformations. This transformation also can be facilitated by the computer aided interviews based on patterns as suggested earlier. The system level models for downstream transformations need refinements to support the MDA approach. Healthcare HL7 and Insurance Application Architectures are examples of such domain models, but MDA solutions development in this context needs a repository of diverse domain models – a future area of work.

References