Teaching Enterprise and Service-Oriented Architecture in Practice

Conference Paper · June 2007

Source: OAI

Alain Wegmann
École Polytechnique Fédérale de Lausanne

Gil Regev
École Polytechnique Fédérale de Lausanne

Lam-Son Lê
Ho Chi Minh City University of Technology (HCMUT)

Irina Rychkova
Université de Paris 1 Panthéon-Sorbonne

TEACHING ENTERPRISE ARCHITECTURE AND SERVICE-ORIENTED ARCHITECTURE IN PRACTICE

Wegmann, Alain; Regev, Gil; de la Cruz, José Diego; Lê, Lam-Son; Rychkova, Irina
Ecole Polytechnique Fédérale de Lausanne, CH-1015 Lausanne, Switzerland
{alain.wegmann, gil.regev, jose.delacruz, lamson.le, irina.rychkova}@epfl.ch}

Abstract

Many companies expect their IT developers to understand their business strategy and to specify IT systems that will impact favorably the execution of their business strategy. Enterprise Architecture (EA) and Service-Oriented Architecture (SOA) address these issues. In this paper, we present a course that introduces EA and SOA to undergraduate CS students. The course is based on an immersive problem-based pedagogy coupled with role playing. The goal is to have the students conceptualize the theory out of the practical experience they gain in the course. Their experience is developed through a game in which the student teams manage competing companies, specify and then develop an IT system (using workflow and web-services). The course places an emphasis on the enterprise-wide impact of the IT systems. Through their practice, the students discover some of the important good-practices used in the industry. They also learn a systemic and systematic approach to address enterprise-wide problems.

Keywords: Systemic Modeling, Enterprise Modeling, Competitive Environment Analysis, Enterprise Architecture, BPMN, Service-Oriented Architecture, Problem-Based, Active Learning. .

1 INTRODUCTION

Business and IT alignment has been steadily gaining in importance. The available technologies (e.g. enterprise service buses, web services) as well as the growing importance of outsourcing change the nature of IT projects. Frequently, less local development is required and more effort is expected for specifying adequate IT systems, creating new IT services by combining existing ones and managing the changes introduced by the IT systems within the corporation. More emphasis is placed on the business impact of the IT solutions and on their applicability rather than on the implementation of the solutions themselves. The growing importance of compliance (e.g. SOX, Basel II) is also strengthening this new trend.

Computer science and information systems education needs to address these new dimensions of IT system development. Our goal with this course is to show the importance of business-IT alignment and to illustrate how Enterprise Architecture (EA) and Service-Oriented Architecture (SOA) can contribute to reach this alignment. The course targets master-level CS students. One of the challenges is to teach EA and SOA to students who do not have practical experience yet. This is addressed through a problem-based learning pedagogy. In most weeks, the students are first immersed in a business game and then conceptualize what they have discovered in the game. Thus, their understanding emerges from their practice. A theoretical session confirms the value of their findings. In the second part of the course, the students need to design the changes the company needs to implement (at the business level as well as at the IT level). To support their design activity, they develop an enterprise model. This is done using SEAM (*Systemic Enterprise Architecture Method*), an EA method that we have developed. This paper presents the course structure and what is taught. It only sketches the specific problem the students need to address; to avoid disclosing too much information that the students need to discover on their own.

2 EA COURSE OUTLINE AND SEAM METHOD OVERVIEW

Our EA course is based on a real-life situation, the development, promotion, sale and support of diesel engines in the light aircraft market. The goal is the development of an IT infrastructure necessary to provide a better customer service to aero clubs who own airplanes equipped with these modern engines. The course lasts 14 weeks with 6 hours per week: 3 hours for practice + 3 hours for debriefing and theory. In most weeks, the students are first immersed in a practical problem. They gain experience and frustrations. They then go through a technical debriefing and an emotional debriefing in which they formalize what they have learned. After the debriefings, a short lecture presents the related theory, strengthening the key learning points.

The students first play the role of a management team of a company – the company of interest - (first 4 weeks) and then the IT development team (remaining 10 weeks). In the role of the IT development team the students specify the IT system requirements (7 weeks) and implement the IT system (3 weeks). In this section, we present the course structure week by week. The figures throughout this paper illustrate the kind of representations the students are required to produce. We intentionally drew hand-made diagrams to highlight that the course places an emphasis on early requirements in which the goal is to discover what needs to be done rather than being complete and formal. This is similar in style and aim to Soft Systems Methodology (SSM) (Checkland and Scholes 1990). In the course, the students eventually create models using tools that we have developed (Lê 2006) and eventually implement the business process with a tool available as open-source (Intalio 2007).

2.1 Business Introduction (Week 1 – 4)

The first four weeks introduce the students to general business issues, and to the main business processes found in companies developing hard goods. They have to manage a company that markets diesel engines for light aircrafts. The students experience the manufacturing process (week 1), engineering development process (week 2), the financial process (week 3), and the production planning as well as the quality process in manufacturing (week 4). This part ends with a presentation, by a business expert, of the role of the Enterprise Resource Planning (ERP) applications.

We have developed a business game for teaching these processes. The students, working in groups, form the management teams of competing companies. The companies design, manufacture and service diesel engines. Each company competes with each other. They encounter difficulties similar to those experienced in real businesses. The game is supported by a custom-developed IT system that simulates the minimal functions of an ERP. In the debriefing sessions, the students analyze how they could manage their business more effectively. The theory sessions confirm that the findings made by the students correspond to industry good practices (e.g. ISO 9001).

2.2 Requirement Elicitation (Week 5 – 6)

The fifth and sixth weeks introduce the main purpose of the course. Each group of students now plays the role of the IT team of their company (the company they learned to manage in the first 4 weeks). They are requested to write the requirements for an IT system that is supposed to improve the sales situation of their company. The IT system improves the service offered to the customers who operate the company's diesel engine powered aircrafts. As the student's company is short on cash, it is closely monitored by its venture capitalist. The goal of the students is to write a document that both convince the venture capitalist to fund the project and that describes the system to be built to its future developers. The format of the document is defined in the IEEE standard 830 called *Software Requirements Specifications (SRS).* Our goal is to give students the experience of interviewing, writing, presenting IT requirements and obtaining funding for projects. In the corresponding debriefing session, we analyze how efficient were the interviews, how useful was the IEEE 830 standard and the reactions of the venture capitalist to the presentation of the SRS document. This highlights the need for contextual interviews (Beyer and Holtzblatt, 1997) to discover the situation of the customers on their work site. This also helps to the students in understanding the impact of the IT system on the company's value proposition, on the company's business strategy as well as on the company's operations.

2.3 SEAM Enterprise Modeling (Week 7)

In the seventh week, the students receive lectures about SEAM, our enterprise architecture method. SEAM (Systemic Enterprise Architecture Method) focuses on enterprise-wide projects and on change management (Wegmann and al. 2005). Using SEAM, the students analyze their company and the problems their company experiences (***as-is*** situation). From this analysis, the students evaluate different solutions, select the most adequate one and implement it. This implies redesigning their company's partnerships, its internal organization and its IT systems (***to-be*** situation).

The main originality of SEAM is its systemic paradigm (Wegmann 2003) that combines a generic modeling technique together with discipline specific heuristics. For example, the same kinds of model elements are used to represent the company in its competitive environment or the business processes within the company. Conversely, when the competitive environment is analyzed, specific marketing heuristics are applied; when the business process is analyzed, other heuristics are applied. The SEAM systemic paradigm defines a set of principles necessary to analyze and design systems. The key principle is that we perceive reality as hierarchical and that we can use the concept of system to structure this hierarchical perception. We subscribe to the interpretive view of Checkland and Holwell (1998), and Weinberg (1975) who argue that systems do not exist as such in reality. In this view, a system is an intellectual construction that we use, as modelers, to represent the fact that we perceive sets of interrelated entities. The same sets of interrelated entities exhibit emergent properties when seen as a whole. In other words, a system can be analyzed as a whole (abstracting the components that belong to the system and making explicit the emergent properties) and as a composite (modeling the components that construct the system). When using SEAM, the modelers develop one enterprise model that represents all the relevant systems across all organizational levels. The modelers work on their enterprise model using views that represent parts of the model, following the heuristics that will be explained in the following sections. All the figures presented in this paper are views of the enterprise model.

After having been exposed to the concept of interpretive view, the students have to agree on how they perceive their reality. This perception was built implicitly in the first 6 weeks of the course. The kinds of systems that are typically perceived are: the market segments, the value networks (supplier or adopter), the companies, the employees & IT systems and the web services. Figure 1 illustrates this system hierarchy.

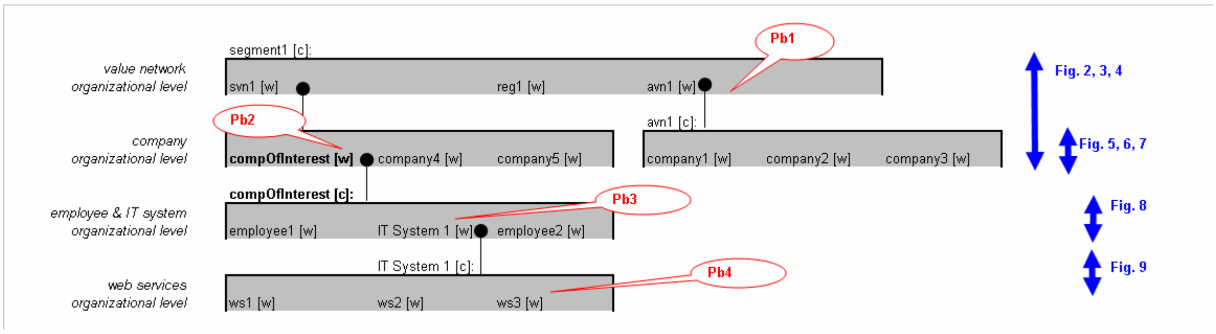


Figure 1: Organizational levels in the enterprise model, the problems to address are highlighted in each level

As a deliverable in the second class of the seventh week, the students identify the hierarchy of systems that will be described in their enterprise model.

2.4 Analysis and Design of the Enterprise Model (Week 8 – 10)

In this part of the course, the students develop the enterprise model. Using the enterprise model, they analyze which solution to select and then implement the selected solution. The enterprise model describes all systems relevant to the identified problem. The students work in parallel on the different views and they have to synchronize their different representations within each group. This trains the students on the challenges of working collaboratively on the same model. This is also necessary to achieve the design (and its representation in the enterprise model) in the three weeks scheduled for this task. The modeling is complete when all diagrams represented in this Section are complete, consistent and useful. We consider the diagrams as useful if they describe the problem and whole or part of the solution. As time-boxing is recommended, the students can choose to only specify and implement a subset of the solution.

2.4.1 Analysis and Design of Service to Customer

First the students analyze the service offered to the customer. This is done by making a view of the segment as illustrated in Figure 2. The segment includes the supplier/adopter relationship that represents the service offered, the main supplier value network *SVN1* that provide the service, the adopter value network *AVN1* that benefits from the service and the regulators (*regulator1, regulator2*) that control this segment. A value network is a group of companies that collaborate for a common goal (e.g. providing a service). When the students model the main supplier value network, it is represented as a whole: only the externally visible behavior is shown. It is not possible to see how the behavior is implemented and the role of each company in the value network. But, it is possible to see the net effect (the emergent property) of their collaboration – including the quality of the service provided (e.g. response time). This is a very interesting feature as it expresses the customer experience in interacting with the supplier value network. On the other hand, the main adopter value network is represented as a composite. This is necessary to understand the relations between the features offered and the value created for each company in the adopter value network.

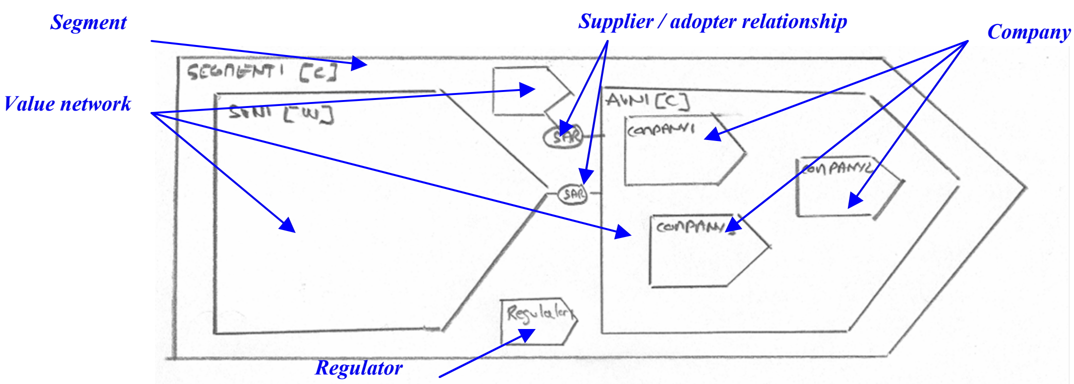


Figure 2: View of the function of SVN1 as a whole and AVN as a composite, useful for identifying the companies in the adopter value network (part of the project scoping).

Once the relevant systems are identified, the students need to develop a second view that represents the beliefs and the goals of the companies. A goal describes a state to be reached or to be maintained during a given time by a system (like a person or a company). A belief describes the knowledge of a system about itself or about its environment. Beliefs drive the definition of the goals. Belief-goal modeling is one of the specific theories we use to analyze the enterprise model (Regev and Wegmann 2004). The belief-goal model is useful to identify the value that needs to be created. Figure 3 illustrates a simplified belief-goal view of the actors identified in a market segment. For example, one of the beliefs for the company that operates diesel powered airplanes is that the airplanes need to be serviceable[1] most of the time to be profitable. The resulting company goal is that, in the case of failure, the engine needs to be repaired quickly. This goal becomes a belief of the repair shop; this belief then becomes the goal to obtain the necessary spare parts as soon as possible. This increases the serviceability of the plane. This analysis is useful to understand what services the company manufacturing the engine and its partners need to provide. In the belief-goal model, some beliefs and goals can be annotated as being "to-be". This means that they are expected as a result of the project.

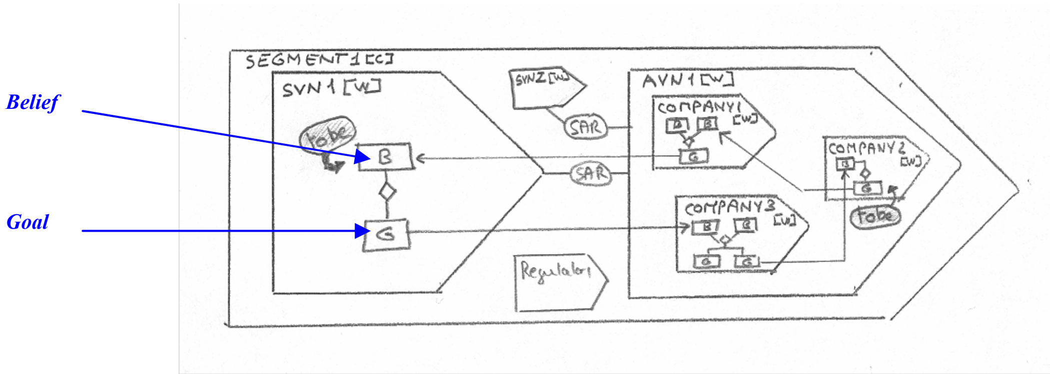


Figure 3: View of the beliefs and goals for SVN1 (as a whole) and AVN1 (as a composite), useful to understand what is important for the companies in the adopter value network

The next view (Figure 4) captures the specifics of the supplier / adopter relationship (SAR). It makes explicit the features that are provided by the main supplier value network and which are the relevant values that the companies on the adopter side perceive. The main goal for this view is to define the value (*value1* to *value4* in Figure 4) that the adopters perceive in the features (*feature1* to *feature4* in Figure 4) of the product or service delivered by the supplier value network. For example, a feature *"fast delivery of spare parts"* contributes to the value *"airplane highly serviceable"*. The students also analyze which information flows back from the adopter side to the supplier. An example of information is *"number of serviceability hours"*. The students use the SAR to analyze the value created by this information for the adopter side and for the regulators, as well as the exchange of information / value between the supplier and the adopter sides. The feature/value part of the SAR is related to the concept of house of quality as used in the industry.

svn1 [w]				avn1 [c]				
↔				↔				
				company1 [w]	company2 [w]	company3 [w]	regulator1 [w]	regulator2 [w]
X	feature 1	→	value1	+			X	
X	feature 2	→	value2		++			X
X	feature 3	→	value3	+	-			
X	feature 4	→	value4		+	+		
X	value5	←	feature5	X				
X	value6	←	feature6		X			
X	value7	←	feature7			X		
X	value8	←	feature8			X		
			to-be					

Figure 4: View of the supplier / adopter relationship, useful to describe the value created for the companies in the adopter value network

[1] A serviceable airplane is an airplane that is fit for use.

2.4.2 *Analysis and Design of Company Responsibility and Partnership*

Once the service to the customer has been specified, the students analyze who does what in the main supplier value network. For this they consider the main supplier value network as a composite. The view in Figure 5 illustrates how they represent the companies in the main supplier value network (which was analyzed as a whole before). The lines between the companies represent linkages. A linkage exists when a same company plays multiple roles in one or more segments or when two companies are closely related. In our course an example of linkage could be that a repair shop may repair traditional engines as well as diesel engine (i.e. the repair shop works with the company of interest and with its competitors). Linkages are often at the centre of business strategies. For example, contracts can be signed to anticipate potential conflict of interests due to a linkage between two companies. Linkages are represented as annotations of the model to make sure that students do not forget important information.

To-be

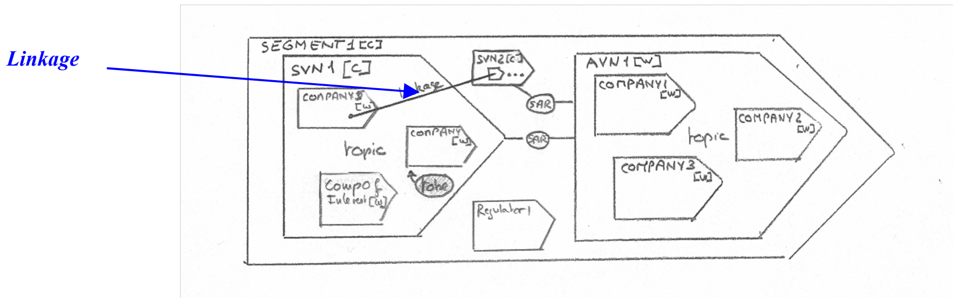


Figure 5: View of the segment composed of value networks, composed of companies, useful for identifying the companies in the main supplier value network (part of the project scoping).

At this point the students need to identify which company, in the main supplier network, contributes (in which role) in providing the features. The left part of Figure 6 is called the responsibility matrix. Filling the responsibility matrix makes the students think on possible outsourcing strategies as well as on the specific role of each company (*r1* to *r13* in Figure 6).

	SVN1 [c]						avn1 [c]					
	←		→				←		→			
	compOfInterest [w]	company4 [w]	company5 [w]				company1 [w]	company2 [w]	company3 [w]		regulator1 [w]	regulator2 [w]
	R1	R2		feature 1	→	value1	+				X	
	R3			feature 2	→	value2		++				X
	R4	R5	R6	feature 3	→	value3	+	-				
	R7		R8	feature 4	→	value4		+	+			
	R9			value5	←	feature5	X					
	R10			value6	←	feature6		X				
	R11		R12	value7	←	feature7			X			
			R13	value8	←	feature8			X			
						to-be						

Figure 6: View of the supplier / adopter relationship, useful to define the roles of the companies in the supplier value network

The students now need to design the inter-company business process necessary to create the features. Figure 7 describes the view they develop. With this view it is possible to define the responsibilities of the company of interest. This is a very important step as it defines the result, at the company level, of the business process the students will implement within the company.

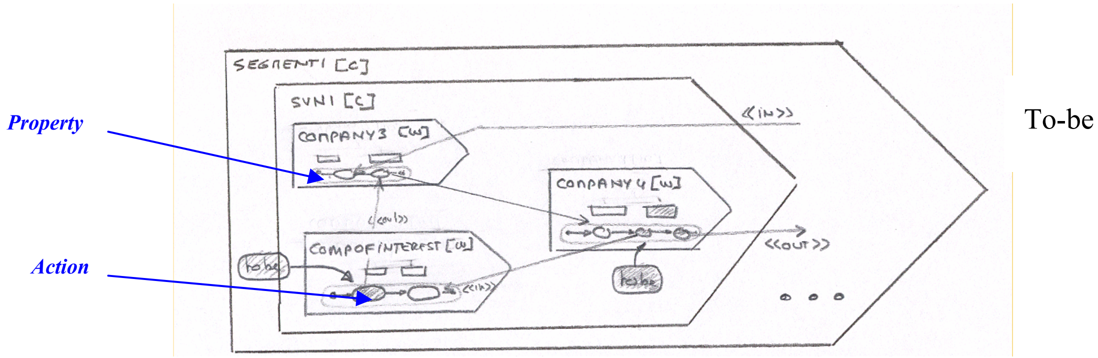


Figure 7: View of the function of SVN1 as a composite, useful to design the inter-company business process

2.4.3 Analysis and Design of Company Organization

The students then focus on the company of interest and they model the functionality provided by its employees and its IT systems implementing the features described in the SAR. The corresponding view is shown in Figure 8. This model is close to what will be later described in the Business Process Modeling Notation (BPMN) (OMG 2007) but less details are represented (e.g. exception cases are no shown). In order to create these views, the students have the option to interview the relevant people (adopters, employees of the company, etc.)

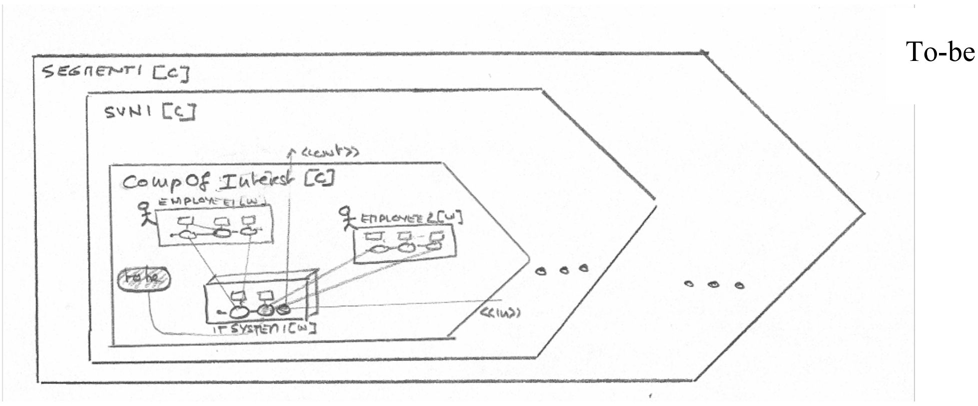


Figure 8: View of the function of Company3 as a composite (and of Employee1, Employee2 and IT System1 as wholes)

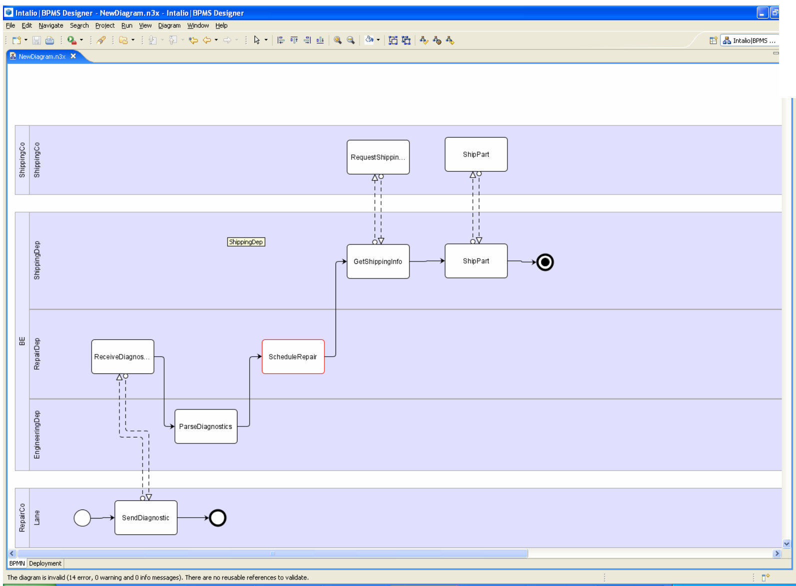
At this point the impact of the project on the whole enterprise has been analyzed. The different views represent some of the important aspects of the project – mainly the ones related to the functionality. The traceability between the levels exists because there is one hierarchy of systems that spans the whole model. This traceability is maximized because of the common modeling technique used to model all systems. Note that the resulting enterprise model is not complete. The model is useful as it defines the functionality of the IT system, of the company and of the value network. This represents a shared agreement that can help the students playing different specialists to federate their work. However, additional dimensions might be necessary (such as security or finance). Note that in this part of the course, we present to the students the other existing EA methods, such as Togaf (The Open Group 2007) or Zachman (1987). We also invite a guest speaker that presents the importance of reasoning at the enterprise level when developing IT systems.

2.5 Project Plan & Project Funding (Week 11)

Once the enterprise model is developed, the students need to evaluate the impact of the project (the goal is still to convince the venture capitalist to fund the project). The first views (Figure 2 to 4) are used to evaluate the value created by the project. The students make a quantitative estimation of the revenue increase for the company of interest by assessing the business impact of their project. The other views (Figure 5 to 8) represent the changes to be implemented across the company. The project and operating costs of these changes are also evaluated. To make this evaluation, the students need also to decide of the implementation strategy of the IT system. Note that we do not ask the students to justify the transition to an SOA approach. The students need to take into consideration that the company has already decided to take a service-oriented architecture (SOA) approach. At that point we ask the student to complete a simple break-even time analysis. The students also need to choose between a *buy-* and a *make-strategy* for developing the IT system. They then need to summarize the results of their study and present them to the venture capitalist. The project usually gets approved at this point. At this stage of the course, we revisit the debriefing done in week 4 (when the SRS was made). We discuss with the students the importance of aligning business and IT. We also invite a guest speaker to illustrate that what the students have learned in the course is actually happening in real industrial projects.

2.6 IT Design & Business Activity Monitoring (Week 12 – 14)

Once the project is funded, a detailed representation of the interaction between the users and the IT systems is developed. This is done with an open source Business Process Management System (BPMS) (Intalio 2007). The BPMN notation is used (OMG 2007) (see Figure 9). The resulting business process is deployed, as a BPEL file, in a business process management server. The business process then interacts with the web services provided by the software simulating the ERP in week 1-3. This simulated ERP is developed with WebLang (Buchwalder and Petitpierre 2007) and runs on a JBoss (JBoss 2007) platform. These three weeks are essentially an introduction on business process management (BPM), workflow and web services. In addition to learning BPM and web service concepts, the students learn the importance of data modeling. In the process they develop interfaces with the simulated-ERP that they used during weeks 1 to 4. They need to reverse-engineer the data model from their experience at the beginning of the course to be able to document the web services.



To-be

Figure 9: BPMN model of a business process

In week 14, once the business process is deployed, the students monitor the performance of the developed system. This is done by simulating its execution. We can then validate the impact of the project and decide on correcting actions, if necessary. This closes the project.

3 RELATED WORK

Even if publications highlighting the need to teach business and IT are not rare in the literature; to our knowledge not many academic publications have been written on courses relating business and IT. One of them is (Zack 1998); the approach presented is less participatory than in our course.

Simulations are widely used to teach business. The most well known is the Beer Game (MIT Forum for Supply Chain Innovation 2007). The originality of our course is to address both business and IT development in relative depth. So, it is no surprise that we use a software simulator. The software we use was developed specifically for the course. Its functionality is similar to a very small ERP (capability to place orders, to manage bills of material, to plan production, etc). The Beer game software puts more emphasis on quantitative simulation.

Finally, our course is based on SEAM. In one of the session, we explain "traditional" EA methods such as Zachman (Zachman 1987) and TOGAF (The Open Group 2007). We explain the differences with these methods (no visual notation, different ways to represent business, business process and IT issues). We also present more modern EA methods that do provide a visual notation, such as ArchiMate (Lankhorst 2005) and we explain the main differences with SEAM. These differences are usually related to the existence of a systemic paradigm in SEAM.

4 CONCLUSIONS

This course is an improved version of a course given some years ago and documented in (Wegmann 2004). The new features are the method and the technologies. In the previous course, we used traditional non-integrated methods (e.g. fishbone analysis to model problems, value system to model business, UML-like diagram to model business processes). This new course uses SEAM that provides a holistic approach to enterprise modeling. The same business and IT aspects are taught as in the previous course, but now the modeling technique has the same foundational principles regardless of the fact that we model business or IT. This eases considerably the teaching and enables the development of more consistent views of the enterprise model.

In the previous version of the course we taught ActiveX, DCOM, JDBC and MSMQ as integration technologies. The new course uses Business Process Management (BPM) and web-services.

All these changes explain that the course has evolved from 3 weeks devoted to the business game, 4 weeks devotes to requirements and 7 weeks to implementation into 3 weeks devoted to the business game, 7 weeks to business modeling and 3 weeks to implementation. This change of ratio reflects the change in the industry and the growing maturity of integration technologies.

Among other improvements for future versions of the course, we might ask the students to play the roles of different specialists (e.g. we can ask the students to develop an ISO 17799 model to cover the security aspects).

Many design choices for this course were the same as the ones in the previous version, the choice of the hard-goods industry, for example. The main reason is that this industry is a very good metaphor for understanding the service industry. Understanding the quality systems and the key processes in the hard-goods industry (where processes are concrete) significantly helps in understanding what is a good process in the services industry. Another important choice was to fully immerse the students in the business context, through the business game in module 1 and role playing in module 2. This is the way we found to give them some sense of business experience. This stems from our belief that practicing EA requires business experience whereas companies require ever younger engineers to perform it.

References

Beyer H. and Holtzblatt K. (1997). Contextual Design: A Customer-Centered Approach to System Designs, Morgan Kaufmann

Buchwalder O. and Petitpierre C. (2006), WebLang: A Language for Modeling and Implementing Web Applications, 18th International Conference on Software Engineering and Knowledge Engineering (SEKE), San Francisco, http://ltiwww.epfl.ch/WebLang

Checkland, P. and Scholes, J. (1990). Soft System Methodology in action. Wiley, Chichester, UK.

Checkland, P.and Holwell, S. (1998). Information, Systems and Information Systems – making sense of the field. Wiley, Chichester, UK.

Intalio (2007). http://www.intalio.com, accessed March 2007.

JBoss (2007). http://labs.jboss.com/portal/, accessed March 2007.

Lankhorst, M et al. (2005). Enterprise Architecture at Work, Modelling, Communication and Analysis. Springer, Berlin, Germany.

L. S. Lê and A. Wegmann, SeamCAD: Object-Oriented Modeling Tool for Hierarchical Systems in Enterprise Architecture, Proceedings of 39th Hawaii International Conference on System Sciences, p. 179c, Track 8, IEEE Computer Society, Hawaii, USA, January 2006.

MIT Forum for Supply Chain Innovation (2007). http://beergame.mit.edu/, accessed March 2007

OMG (2007). http:// http://www.bpmn.org/, accessed March 2007.

Regev, G. and Wegmann, A., (2004). Defining Early IT System Requirements with Regulation Principles: The Lightswitch Approach, Proc. 12th IEEE International Requirements Engineering Conference (RE04) p.144 – 153.

Stabell, C. B. and Fjeldstad, Ø. D. (1998). Configuring value for competitive advantage: on chains, shops, and network, Strategic Management Journal 19(5): p. 413 – 437

The Open Group (2007), The Open Group Architecture Framework (TOGAF), http://www.opengroup.org/togaf/

Wegmann, A. (2003). On the Systemic Enterprise Architecture Methodology (SEAM). Proceedings of 5th International Conference on Enterprise Information Systems Angers, France, 483-493.

Wegmann, A. (2004). Theory and practice behind the course designing enterprise-wide IT systems IEEE Transactions on Education, 47 (4), 490 - 496

Wegmann, A., Regev, G. and Loison, B. (2005). Business and IT Alignment with SEAM. Proceedings of 1st International Workshop on Requirements Engineering for Business Need, and IT Alignment, Paris.

Weinberg, G.M. (1975). An Introduction to General System Thinking, Wiley & Sons. New York.

Zachman, J. A. (1987). A Framework for Information System Architecture, IBM System Journal 26(3): 276-292.

Zack, M. (1998). An MIS course integrating information technology and organizational, ACM SIGMIS Database, Volume 29 Issue 2.