Model-Based Architecture for Learning in Complex Organization

Francesco Basciani(✉) and Gianni Rosa(✉)

Dipartimento di Ingegneria e Scienze dell'Informazione e Matematica,
Università degli Studi dell'Aquila, L'Aquila, Italy
{francesco.basciani,gianni.rosa}@univaq.it

Abstract. To improve their service quality modern organization employees have to understand and put in action latest procedures and rules while coping with quickly changing contexts and decreasing resources. To this end a model-based architecture with interrelated enriched models is required in order to fosters an informative learning approach in the learning-by-doing paradigm. Such architecture enables organization employees to learn by accessing and studying enriched business process models and related material in a process-driven learning approach. Zachman Framework is used to organize all the models through the definition of the relations among them.

Keywords: Model driven engineering · Enterprise architecture · Zachman framework · Learning · Organizations

1 Introduction

In the complex organization domain it is increasingly demanded greater effort in terms of quality and efficiency in the services provided by employees in doing their jobs. To ensure this efficiency and this quality is necessary that employees with expertise in a given task (process) can share their experience. To facilitate both the knowledge elicitation and the learning process, a wide variety of models, tools and techniques have to be provided and integrated. In this respect several technical spaces are identifiable. This represents a major challenge because while the informative content of the various models is comparable, the way they are represented is based on different formats and standards. Furthermore, all these artifacts at the same time may confuse organizations, because it is not very obvious which one to choose or which purpose is served and bridging the different notation presents intrinsic difficulties whenever the artifacts are not belonging to the same technical space regardless of their content. Moreover, all the process and its sub-processes have to be developed and managed independently from other domains processes. Integrated models are needed, which put the various approaches into perspective. Such integration is meant to improve the speed of working, improve quality of documentation, products and processes, reduce costs, enhance responsiveness to customer needs and handle the overall system inherent complexity.

S. Hammoudi et al. (Eds): MODELSWARD 2016, CCIS 692, pp. 293–311, 2017.
DOI: 10.1007/978-3-319-66302-9_15

In this paper we propose a model-based architecture conceived to provide a learning experience in which learner acquires knowledge while serving real requests, supporting an informative learning approach besides the learning-by-doing paradigm. For being effective, the architecture must provide the requirements for a modeling notation which describe the learners level, the acquired competencies and knowledge to perform a procedure described by means of a business process. This approach permits the learner to access and study these enriched models and operate within a simulated environment reproducing real requests through the promulgation of a process and monitoring activities in order to provide feedbacks for the evaluation of learners, business processes, and associated learning contents. To fulfill the need of share knowledge, manage and improve the processes in enterprise, the Learning Architecture *LA* provide a machine-processable model that exploit the correlation among the activities and/or concerns in order to provide enriched informations to the organization. The Zachman Framework [29] is used to describe all the interrelations, that provides a logic structure for classifying and organizing the knowledge about business activity of an organization in different dimensions, and each dimension can be perceived in different perspectives with respect to the Enterprise Architecture.

Structure of the Paper. The paper is organized as follows. Next section illustrates a motivating example related to a complex organization. In Sect. 3 we present an analysis of the required informations in order to design a LA; in Sect. 4, we outline how are integrate all the artifacts involved in learning in complex organizations using the Zachman Framework. Related work is discussed in Sect. 5 and finally, in Sect. 6 we draw some conclusions and future work.

2 Motivating Example

In this section, we present an example where an organization submits a project to the European Union (EU). To do that, the organization have to be aware of the environment complexity in which it is working because the ability to deal with this complexity is critical for the success of the project proposal. They must be able to handle in different ways a process as well as use different tools, models, reporting documentation and so on. Moreover, to successfully participate in a project proposal and to support administrative reporting activities, for a complex organization is required to involve a unit of administrative personal. For this reason, and also due to the typical employees high mobility, the availability of an electronic learning platform is therefore highly desired.

In order to better understand the complexity of managing public administrative procedures, a real world scenario is presented. Such scenario reference the administrative offices of an Italian public research body and is related to his participation to an European Project Budget Reporting (EPBR) [7]. We will start from the University organization structure description to the detail of the Business Process under analysis.

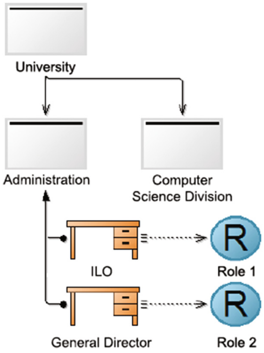

Fig. 1. Organization model: university organization (partial).

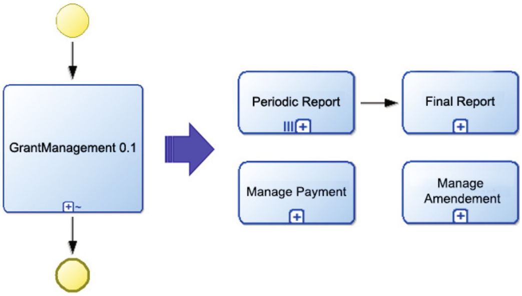

Fig. 2. Grant management different level of detail.

Figure 1 denotes a fragment of the University organization in which there is an administration and different schools (e.g. the Computer Science Division). In turn, an administration may have several employees, each one with its own role.

This scenario engages different partners in the definition of models and documentation for a Business Process and will permit to assess applicability of the proposed solution within real working contexts. For the sake of clarity, we are going to explain only a portion of the entire process and, after a first analysis of the domain, the Grant Management BP has been selected as reference point (see Fig. 2). It includes some sub-process, such as: Periodic Report, Final Report, Manage Payment and eventually Manage Amendment.

Without going into the details of each of the sub-processes involved in the scenario (this is not the purpose of our work) we consider the *Periodic Report* as motivating example. It is the data object representing the periodic report written by each partner participating to the project. In this process are involved different participants such as the officer, the coordinator (one pool), the grant beneficiary (multiple in parallel) and optionally the third part. Figure 3 describes how the coordinator organizes the process of periodic reports with respect to all the involved stakeholders.

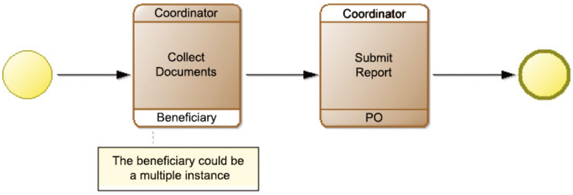

Fig. 3. Periodic report - choreography diagram.

Moreover, each Public Officer (PO) according to her experience, might have her own view of the process for the production of her private periodic report (Fig. 4 shows an example of a private process done by an EU public officer). In this way, different versions of the same process could be created, so we may have different diagrams for the same process. All these diagrams should have documentation so we need other models for this purpose.

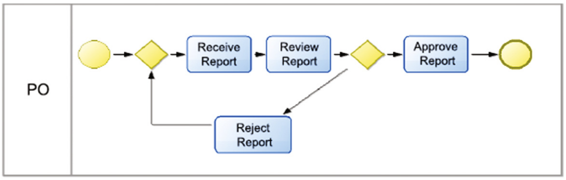

Fig. 4. Periodic report - private process of EU Officer.

In its turn, a documentation have to describe, textually and graphically, the state of the data-object. In particular, Fig. 5 shows as the Periodic Report is composed by a set of data-objects.

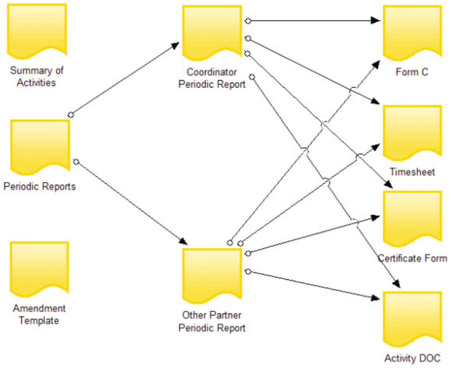

Fig. 5. Document model: periodic report.

Focusing on the role of Coordinator, they can be determined specific data-object: Amendment Template, Summary of Activities and Periodic Reports. Finally, Fig. 6 shows the private process of the Coordinator in relation to these documents.

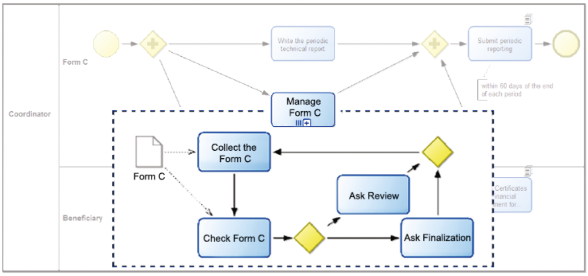
Coordinator
Form C
Beneficiary
Write the periodic technical report
Submit periodic reporting
within 60 days of the end of each period
Manage Form C
Form C
Collect the Form C
Ask Review
Check Form C
Ask Finalization
Certificates inancial nent for...

Fig. 6. Periodic report - private process of coordinator relationship with documents.

This scenario shows the use of a wide variety of models and diagrams (i.e. organization model, choreography diagram, collaboration diagram, Document Model, etc.) at different levels of detail both in term of modelling and learning (i.e. according to the learner skill you should focus on different abstraction level regarding how to deal with reporting).

The disadvantage in using all these models is represented by the increase of the whole process complexity and the problem of proper integration of all these artifacts. In the next section, we will show how the complexity emerged in this scenario is handled and the integration is done.

3 Learning Architecture

As illustrated in [17], in complex organisations there are many information resources that represents the complete set of activities consumed to perform missions, goals, and objectives. The knowledge must be systematically formalized, organized and consistently categorized in order to support effectiveness in learning. The architecture proposed in this paper supports:

- *informative learning* by which the learner can access and study the enriched BP model and related material with additional descriptive contents and,
- *procedural learning*, by which the learner operate within a simulated environment reproducing real requests through the enactment of a process and monitoring activities (*learn by doing* approach). Such environment allows us to capture useful feedback for the evaluation of: *(i)* learners, *(ii)* business processes, and *(iii)* associated learning contents. To this end, open-source communities principles and cooperation spirit will be fostered: contents are produced by the community, and meritocracy is naturally promoted, with leaders emerging because of their skill and expertise.

The above strategies, are *off-line* because the learner acquire such knowledge before serving real requests. However the typical complexity of processes defeats the human capacity to acquire a full knowledge on any aspect just through informative and procedural approaches. It is necessary that learner can retrieve and process useful and context-dependent information while she is working on real cases. The architecture therefore, must provide learning experience with *on-line* strategies in which learner acquires knowledge while serving real requests, supporting *"training on the job"* or *"learn while doing"* approach. To this end, it is of crucial relevance to be able to provide the user with contextually selected task and user-specific background knowledge [6]. In particular, the learner should be able to access the required knowledge in an optimal manner. This can be achieved by coupling the process (formal or informal) description with the descriptive units about the kind of data and document type being considered by the process. Ideally, the notion of context provided by the process permits users to know:

- what to do,
- who does what, and
- what to do after the task.

In this context arise the necessity to analyze the business process from a knowledge-management perspective and this is largely recognized (for instance, see [16]). In such a way the users engaged in their daily work routines have not to spend much time and effort in knowledge, information retrieval and management activities additional to their operative ones. Starting from the aforementioned premises, the LA exploits models in order to have informative specifications *(i)* for the learners and *(ii)* at the same time informations able to simulate and monitor the processes in organizations. The architecture proposed in this paper is composed by two main components as illustrated in Fig. 7:

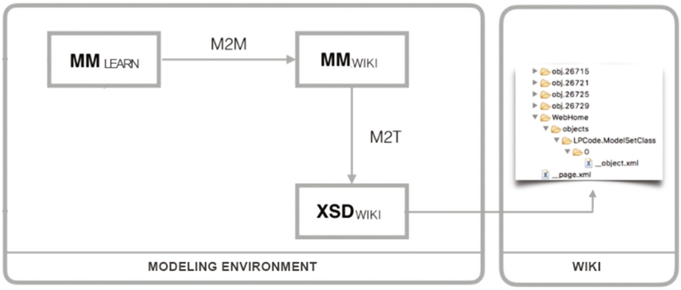

Fig. 7. The learning architecture.

- the Modeling Environment *ME* adopts state-of-the-art techniques and tools provided by the Eclipse Modeling Framework (EMF)[1]. The component provide metamodels, transformation and tools able to create a WIKI structure

[1] https://eclipse.org/modeling/emf/.

representing the processes starting from the diagrammatic modeling stage. The generation of e-learning artifacts out of specified business processes will be performed by LA means of horizontal (Model-to-model) and vertical (Model-to-code) model transformations as discussed in Sect. 4. Each of them represents an overall process phase that, starting from a representation of the modeled business process, create the XWiki structure from which will be created the wiki pages. The availability of complex meta-models for representing the business process structure, its data, and its business rules, permits to exploit its use also to assess the quality of the provided documentation with natural language processing techniques;

- the integration of a WIKI platform as a space for collaborative learning. The wiki-based content can be edited directly by the *experts* in order to enrich the learning material and to provide support to colleagues. Sharing and cooperation will be strongly fostered by the platform, introducing mechanisms inspired by the open source and open model communities. The WIKI is able to automatically reflect the structure of the specified BP.

The ME must be able to represent and transform by means of metamodels and model knowledge used in complex organization including factual, conceptual, procedural and meta-cognitive artifacts. Following we briefly illustrate the metamodeling architecture MM_{LEARN} involved in organizations that represent knowledge needed for learning as discussed in [24].

The models, as shown in Fig. 8, are obtained with an in-depth analysis of *(a)* three business processes in the domain of the Italian *Public Administration*

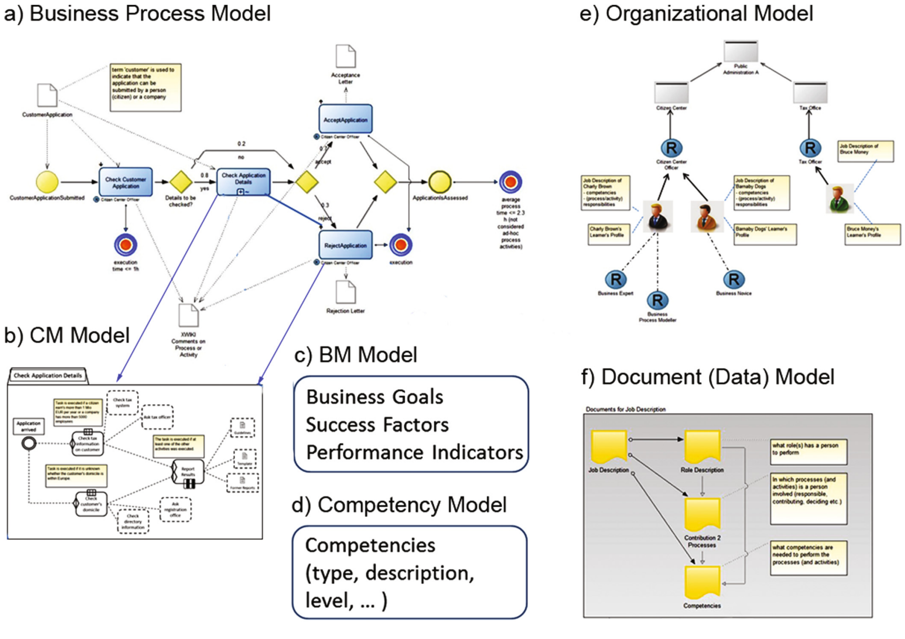
a) Business Process Model
term 'customer' is used to indicate that the application can be submitted by a person (citizen) or a company
CustomerApplication
CustomerApplicationSubmitted
Check Customer Application
Citizen Center Officer
Details to be checked?
0.8
yes
0.2
no
Check Application Details
0.7
accept
0.3
reject
AcceptApplication
Acceptance Letter
RejectApplication
Rejection Letter
ApplicationIsAssessed
average process time <= 2.3 h (not considered ad-hoc process activities)
execution
execution time <= 1h
XWIKI Comments on Process or Activity
b) CM Model
Check Application Details
Application arrived
Task is executed if a citizen earn's more than 1 Mio EUR per year or a company has more than 5000 employees
Check tax system
Ask tax officer
Check tax information on customer
The task is executed if at least one of the other activities was executed
Report Results
Guidelines
Template
Former Reports
Task is executed if it is unknown whether the customer's domicile is within Europe
Check customer's domicile
Ask registration office
Check directory information
c) BM Model
Business Goals
Success Factors
Performance Indicators
d) Competency Model
Competencies
(type, description,
level, ...)
e) Organizational Model
Public Administration A
Citizen Center
Tax Office
Citizen Center Officer
Tax Officer
Job Description of Charly Brown
- competencies
- (process/activity) responsibilities
Charly Brown's Learner's Profile
Job Description of Barnaby Dogs
- competencies
- (process/activity) responsibilities
Barnaby Dogs' Learner's Profile
Job Description of Bruce Money
Bruce Money's Learner's Profile
Business Expert
Business Process Modeller
Business Novice
f) Document (Data) Model
Documents for Job Description
Job Description
Role Description
Contribution 2 Processes
Competencies
what role(s) has a person to perform
In which processes (and activities) is a person involved (responsible, contributing, deciding etc.)
what competencies are needed to perform the processes (and activities)

Fig. 8. Models involved in learning.

(the family reunion, the grant citizenship, and the bouncer registration); and *(b)* a number of relevant modeling notations [6]. The business modelling language, defined to provide a process notation that could be easily understood by all business stakeholders is BPMN 2.0 [1] as represented in Fig. 8(a). BPMN is a standard for modeling processes described as a predefined sequences of activities with decisions (gateways) to direct the sequence along alternative paths or for iterations, flow of activities. Unfortunately, its semantic, as discussed in [6], is limited, and it is not useful for some organizational aspects as for instance when the activities in a process

- can occur in any order and/or in any frequency,
- are not predefined, repeatable and knowledge intensive,
- depend on evolving circumstances and ad-hoc decisions by knowledge workers regarding a particular situation.

The standard notation CMMN [2] as depicted in Fig. 8(b), allows us to deal with the aforementioned limits. As discussed in [27], the importance to introduce intentional modeling in enterprise architecture entails potential benefits and pitfalls. In learning context, it is of crucial relevance to model intentionality providing a scheme for developing, communicating and managing business plans in an organized manner. The BMM [3] focuses on that. It has been proposed as a standard under the Object Management Group (OMG) and provides elements and relationships of intentional modeling as depicted in Fig. 8(c). Central elements include *Means*, *Ends*, *Influencer*, *Potential Impact* and *Assessments* that are specialized into more detailed elements as discussed in [24]. The modelling notation in learning must be able to describe the learners level, acquired competency and learning progress respect to a business process or procedure in organizations. In Fig. 8(d), the Competency model unlike the other models is not defined in specific standard leaving to the modeller the responsibility to define such aspects. The implementation we take into account is defined in [5] and it is partly based on the framework the European Committee for standardisation, CEN WS-LT LTSO (Learning Technology Standards Observatory)[2]. To achieve their means and ends, organizations are structured (often hierarchically) in units where each one has a set of job functions or tasks assigned to a group of people belonging to the organization. Therefore, an organization structure is composed of units, each encompassing the relevant people who work to achieve the mission of the organization [22]. The need to keep track of "who does what, how and when" is demanded to in Organizational model as depicted in Fig. 8(e) whose implementations is provided in [5]. About the management of knowledge and documentation, instead of using the BPMN 2.0 data object element for modeling information/documents used in a process, e.g. as input or output for an activity, we use a separate model, as shown in Fig. 8(f). This allows to define a data object (and its meta data), and adding more details, e.g. providing references to operative templates or guidelines, knowledge products or resources, which are utilized in the processes (input, output to activities etc.).

[2] 2EN WS-LT Learning Technology Standards Observatory. URL: http://www.cen-ltso.net/Main.aspx. Main contact: University of Vigo 36213 SPAIN.

4 Learning Using the Zachman Framework

The huge amount of informations and resources gathered from models in Sect. 3 is not independent because several technical spaces are identifiable. This represents a challenge because while the informative content of the various models is comparable, the way they are represented is based on different formats and standards. Bridging the different notation presents intrinsic difficulties whenever the artifacts are not belonging to the same technical space regardless of their content.

To fulfill the need of learning in enterprise, the *Learning Architecture* provides a machine-processable model that exploits the correlation among the activities and/or concerns in order to provide enriched informations to the organization. In the following we use the Zachman [29] framework to describe:

- the interrelations of above mentioned models,
- the logic structure for classifying and organizing the knowledge about business activity of an organization in different dimensions and perspectives with respect to the Enterprise Architecture.

Specifically, the Zachman Framework is a framework for enterprise architecture, which provides a formal and highly structured way of defining an enterprise. In essence, the framework is a two dimensional matrix consisting of 6 rows and 6 columns which defines 6 levels relevant to any enterprise, as well as 6 aspects. The structuring provided by the Zachman Framework provides that attention is placed on all the relevant scales, as well as on all relevant aspects, of any situation under consideration. Any Zachman Framework should be calibrated so that all relevant scales occur within its boundaries. Each row represents a total view of the enterprise from a particular perspective. These rows starting from the top include: Planner's View (Scope), Owner's View (Enterprise or Business Model), Designer's View (Information Systems Model), Builder's View (Technology Model), Subcontractor's View (Detail Representation), and Actual System View (The Functioning Enterprise). The columns describe various abstractions that define each perspective. These abstractions are based on six questions that one usually asks when s/he wants to understand an entity. The columns include: The Data Description (What?), The Function Description (How?), The Network Description (Where?), The People Description (Who?), The Time Description (When?), The Motivation Description (Why?). Further information and cell definitions of Zachman Framework can be found in [28]. The Zachman Framework can form the backdrop for a decision making process, ensuring that no mistaken collapse of attention occurs.

In this respect in Fig. 9 we outline how the models can be structured by Zachman's matrix [29]. The vertical dimension (the rows) in Fig. 9, describes the perspectives in terms of the participants involved in the organization's Information Systems [18] that use the models or descriptions contained in the cells. The top row represents the most generic perspective of an organization, while lower rows are successively more concrete, i.e.:

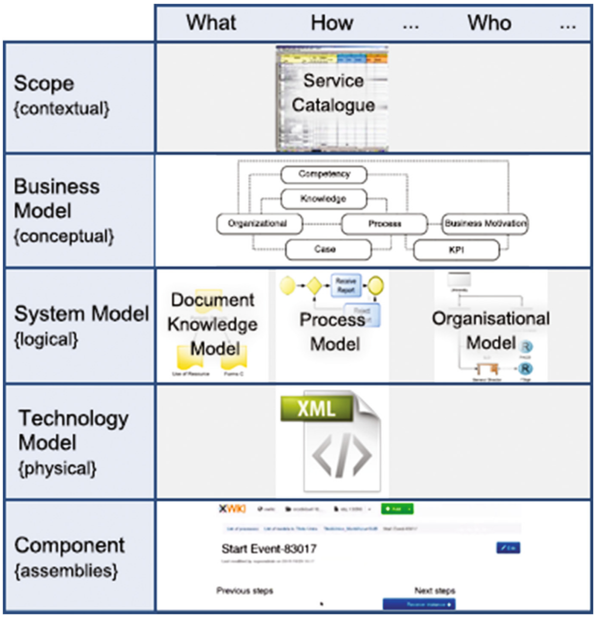

Fig. 9. The learning architecture structured by Zachman's matrix.

- *Scope* (Planner's Perspective), the planner defines the catalogue of services and the boundary of an organization which describe concrete information about a specific organisation, the context of learning, and business scope. The specification is written in natural languages and structured by means of a table that gather the aforementioned information;
- *Business Model* (Owner's Perspective), the owner is interested in modelling, at high abstraction level, the services defined in the *Scope*. The relevant data involved in a learning architecture, consists of a number of component metamodels illustrated in Fig. 10. The following have been defined by adapting current industrial standards:
  - business motivation (BMM) [3];
  - business process management and notation (BPMN) [1];
  - case management and notation (CMMN) [2].

  The remaining have been defined from scratch and are described in [24]:
  - competency metamodel (CM);
  - document and knowledge metamodel (DKM);
  - key performance indicator metamodel (KPI);
  - organization metamodel (OM).

  The relations are implicit and, hence, a process defined in a service catalogue (Scope Concepts level), may occur in the process description on the Business Concepts level but that relation is not formalized and therefore hard to trace;
- *System Model* (Designer's Perspective) the designer works with the specifications defined above, instantiating all elements involved in business organization to ensure that it will, in fact, fulfill the owner's expectations.

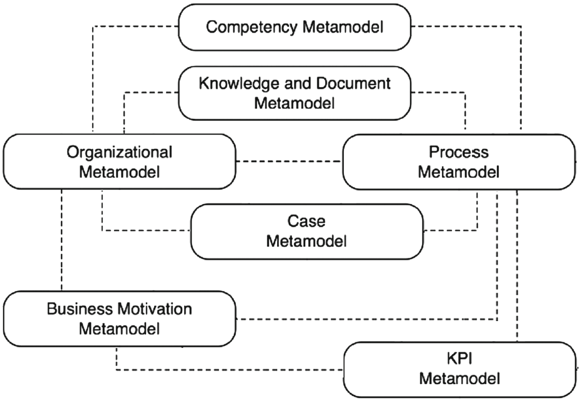

Fig. 10. The conceptual model.

The problem about tracing the relation between a process model on the System Logic Layer and the process description (at conceptual level), holds true;
- *Technology Model* (Builder's Perspective) the builder manages the process of define the language and functionalities able to satisfies the requirement of the learning platform. To this respect, the model set defined in *System Model* must be transformed in a standard exchange format, eg. XMI (see Sect. 4.2), in order to be machine readable;
- *Component* (Learning platform's Perspective) the learning architecture takes the instance models provided by the *Technology Model* and enables process-driven learning, fostering the cooperation and knowledge sharing among the learners.

While the horizontal dimension in the Zachman Framework describe the participants involved in the learning architecture, the columns provide a focus on each dimension [15]: What, How, Where, Who, When, Why and each of them is a descriptive of a single model. The architecture exploit a subset of them as following:

- Data (What?): in this column, "Document and Knowledge" concepts are defined. In particular, about the perspectives *Business Model*, *System Model*, and *Component*, the enterprise's informations about knowledge and resources used for business activity;
- Function (How?): the process of the organization are defined in several abstraction level. Starting from a service catalogue, the models are refined and enriched with structured information. In such way, learner can retrieve and process useful and context-dependent information while she is working on real cases;
- People (Who?): describes who is involved in activities, assigning them to business or IT perspective and classifying them w.r.t. to several aspects.

The matrix structure of the LA, allow us to perform an in depth analysis on some intrinsic characteristics:

- *horizontal relations*: bridging the various modeling notations (and their representation formats) between considered Business Objects;
- *vertical relations*: factorizing part of the transformation chaining in order to produce artifact needed for learning;
- enhance relevant quality factors, e.g. maintenance, extendibility, etc.

4.1 Horizontal Relations

As already discussed in Sect. 3 there are many information resources in an enterprise that serve several purposes and that usually reside in different information systems. The separation of concerns in software system modeling avoids the construction of large and monolithic models which could be difficult to handle, maintain and reuse. At the same time, having different models (each one describing a certain concern) requires their integration into a final model representing the entire domain [25]. The integration in LA is made through horizontal relations in Zachman Framework and, for the sake of clarity, only relations between models on the *System Model* layer will be discussed in this paper (see the related row in Fig. 9). To make these relations explicit and machine processable we provided the specification in terms of weaving models for defining correspondences between modeling elements belonging to different metamodels[3].

The concept of weaving is not new. Typical applications of model weaving are database metadata integration and evolution as in [21] which proposes Rondo, a generic metamodel management approach which uses algebraic operators such as Match and Merge to manage mappings and models. In [14] a UML extension is introduced to express mappings between models using diagrams, and illustrates how the extension can be used in metamodeling. The extension is inspired by mathematical relations and is based upon ideas presented in [4] which proposes an approach for defining transformation relationships between different components of a language definition rendered as a metamodel. The definition of model weaving that will be considered in this paper is that provided by Didonet Del Fabro et al. in [12]. They leverage the need of a generic way to establish model element correspondences by proposing a solution aimed at reaching a trade-off between genericity, expressiveness and efficiency of mappings which are considered models that conform to a weaving metamodel. The weaving metamodel is not fixed since it might be extended by means of a proposed composition operation to reach dedicated weaving metamodels. A weaving model *WM* represent the mapping between the *LeftMM* and *RightMM* metamodels. Like other models, this should conform to a specific weaving metamodel *WMM*.

In the context of *horizontal relations* we use the weaving models for specifying some form of semantics of given modeling elements. For instance, in BPMN the semantics of *Lane* is not precisely given, therefore we provide a weaving model which can associate a *Lane* to an *OrganizationalUnit* deferring the semantics of

[3] Implemented metamodel resources can be found in the repository: https://github.com/LearnPAd/learnpad/tree/master/lp-model-transformer/src/main/resources/metamodels.

the former to the latter (see diagram in Fig. 13). This technique is a simplification of the semantic anchoring [10] which adopts model transformations for anchoring the meaning of a concept from a metamodel into a concept to another metamodel (for which typically the semantics is already given). In other cases, the weaving is more relational and serves the scope to link different entities, like a competence profile which points to a document describing a job description.

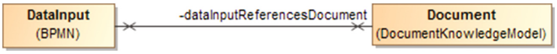

Fig. 11. The dataInput weaving.

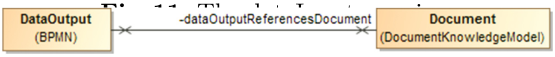

Fig. 12. The dataOutput weaving.

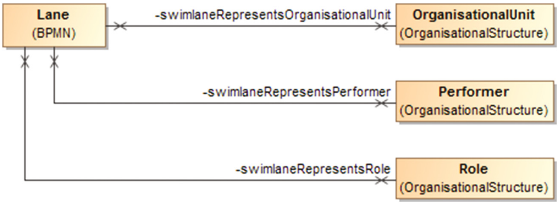
Lane
(BPMN)
-swimlaneRepresentsOrganisationalUnit
OrganisationalUnit
(OrganisationalStructure)
-swimlaneRepresentsPerformer
Performer
(OrganisationalStructure)
-swimlaneRepresentsRole
Role
(OrganisationalStructure)

Fig. 13. The swimlane-lane weaving.

In the following, each weaving is given by means of a weaving metaclass denoting the correspondences between two or more metaclasses in different metamodels. The weaving models are given according to the component metamodels defined in [24], and the definition of each model can encompass one or more association:

- Business Process Modelling Notation (BPMN 2.0)[4]: several kinds of weaving are defined; the link with Document Knowledge Model permit to have the resources used as input (Fig. 11) and/or output (Fig. 12) in a process or activity.

  The lack of a specific semantic in the BPMN specification for the *Lane* concept required the definition of the Lane-weaving (Fig. 13). Such interconnection links a *Lane* in BPMN, with respectively *(i) OrganizationalUnit*, *(ii)* the *Perfomer*, and *(iii)* the *Role* in the Organisational Model. Finally, the Activity-weaving interconnects information linked to a given activity in accordance with the Fig. 14. In particular, given an *Activity*, it denotes: *(i)* the competencies needed for realizing it; *(ii)* the criteria used for evaluating its performance; *(iii)* the organizational unit, which has been assigned the responsibility; *(iv)* who is performing it; *(v)* the performer position and *(vi)* her role.

[4] http://www.omg.org/spec/BPMN/2.0/.

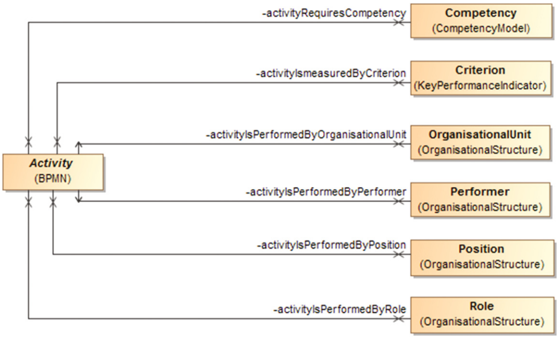
-activityRequiresCompetency
Competency
(CompetencyModel)
-activityIsmeasuredByCriterion
Criterion
(KeyPerformanceIndicator)
-activityIsPerformedByOrganisationalUnit
OrganisationalUnit
(OrganisationalStructure)
Activity
(BPMN)
-activityIsPerformedByPerformer
Performer
(OrganisationalStructure)
-activityIsPerformedByPosition
Position
(OrganisationalStructure)
-activityIsPerformedByRole
Role
(OrganisationalStructure)

Fig. 14. The activity weaving.

- Case Management and Notation (CMMN)[5]: the ProcessTask-weaving denotes the reference to an *Activity* (regular task) to be invoked by the process task (Fig. 15).
- Organization Model: the Position-weaving links the *Position* described or reported in a resource in a Document and Knowledge Model, e.g., a job description (Fig. 16).

The above relations are just only a subset of all possible ones, according to motivating scenario in Sect. 2. A more in depth analysis, and other kind of relationship tailored for learning in complex organizations, like the Public Administrations, are discussed in [5].

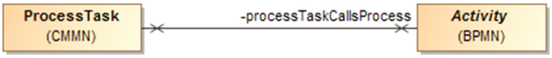

Fig. 15. The process task weaving.

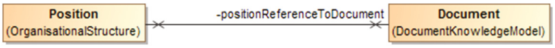

Fig. 16. The process task weaving

4.2 Vertical Relations

The LA exploits models in order to have informative specifications for the learners and, at the same time, informations able to simulate and monitor the processes in organizations. As said, models in the Zachman matrix are organized using different abstraction levels, therefore, the learning contents that describe multiple aspects of processes in organizations, should rely on adequate means that automatically relate and trace over the multiple views. The generation of learning artifacts out of specified business processes will be performed

[5] http://www.omg.org/spec/CMMN/.

by means of vertical model transformations chain as depicted in Fig. 17[6]. In order to enhance the automation in finding model transformation chains, we use the proposed process of deriving model transformation chaining depicted in [8]. Moreover, there is the need of techniques introducing automation in the management of artifacts that have to be kept consistent to each other.

In this respect the modeling facilities offered by the Eclipse Modeling Environment (EMF) can be used in order to support the management of the artifacts involved in the vertical relations. Specifically, EMF is part of the Eclipse project[7], whose goal is to provide a highly integrated tool platform. With EMF it is possible to explicitly define the domain model and this helps to provide clear visibility of it. Indeed, EMF has a distinction between the metamodel and the actual model: the metamodel describes the model structure (*System* and *Technology Metamodel* in Fig. 17(a)) while an actual model is a concrete instance of this metamodel (*System* and *Technology Model* in Fig. 17(a)). Another benefit is that EMF allows to persists the data model; the default implementation uses a data format called XML Metadata Interchange (XMI) that is a standard for exchanging meta-data information via Extensible Markup Language (XML). The EMF integration in the platform offer the advantage to perform different kind of transformations. For example, both the model-to-model and the model-to-code transformations, or a combination of them. This leads to modularity improvement, indeed, instead of making a single big transformation it can be divided into smaller once increasing, also, the overall process maintenance.

Therefore, in the Zachman vertical dimension, model transformations play a central role since they represent the glue between the several levels of abstraction and enable the generation of: *(i)* different artifacts for learning purposes using ATL[8] in *Model2Model* (see Fig. 17(a)) transformation languages and *(ii)* the generation of implementation code [9] by means of Acceleo[9] in *Model2Code* transformation (see Fig. 17(b)).

5 Related Work

Many efforts have been done in order to support the integration of models, tools and techniques used to describe various aspects of a complex organization.

[20] tackle the issue of integration of all the concepts and modelling techniques used by architects to describe their architectural domains, presenting an enterprise modelling approach. In this approach several abstract layers are integrated combining several existing languages. Unlike the work presented in

[6] Implemented ATL and Acceleo transformations resources can be respectively found in the repository: https://github.com/LearnPAd/learnpad/blob/master/lp-model-transformer/src/main/resources/transformation/ado2xwiki.atl and https://github.com/LearnPAd/learnpad/blob/master/lp-model-transformer/src/main/java/eu/learnpad/transformations/model2text/main/generate.emtl.

[7] https://eclipse.org/.

[8] https://eclipse.org/atl/.

[9] https://eclipse.org/acceleo/.

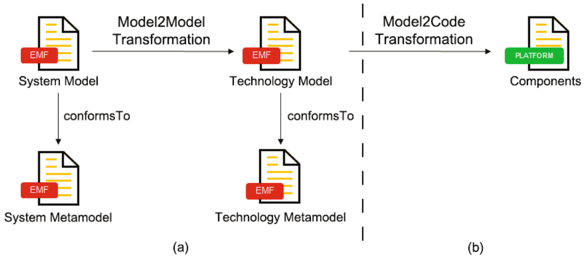

Fig. 17. The vertical Zachman transformation chain.

this paper, they propose a workbench for enterprise architecture that supports the integration of models in existing modelling languages and the integration of existing modelling tools. We choose to perform a similar integration using the Zachman Framework mainly because we are aware that the communication is important.

Indeed, thanks to the Framework's perspective, which allows us to answer the *what*, *how*, *where*, *who*, *when*, and *why* questions, we are able to create different descriptive representations (i.e., models), which translate from higher to lower perspective. This guidance is both clear and complete and as result these perspectives, in relation with these questions, determine a communication matrix. Furthermore, the Zachman Framework permits us to understand where completeness lies, and how to asses when we've achieved it. Indeed, "*Zachman's framework suggests that an architecture can be considered a complete architecture only when every cell in that architecture is complete. A cell is complete when it contains sufficient artifacts to fully define the system for one specific player looking at one specific descriptive focus*" [26].

Although we do not use the tools which they have defined, we still followed the method defined in [23]. In the article, in fact, they propose a method to achieve an Enterprise Architecture Framework based on the Zachman Framework Business. Furthermore, the authors identify a new concept related to this framework defined as "anchor cell" that defines the semantic relationships existing between cells on any of the framework's perspectives. In our work, we developed this "anchor cells" that represents vertical relationships with model transformations that transform a model in a perspectives in another model in another perspective. Moreover we have horizontal relationships through the rows (dimensions) using the weaving model [12].

6 Conclusion

In this paper, we presented a model-based architecture that fosters an *informative* learning approach based on simulation and monitoring besides the learning-by-doing paradigm. This enables complex organization employees also a *procedural* learning by accessing and studying organized business process models

and related material. However, the enriched business models might not convey enough information to support on the one side the enactment of the represented complex organization process, and on the other side the training of the civil servant who is assigned to the tasks. Thus, it is of great relevance to be able to trace and relate all the models and the informative artifacts that structure and represent information with the specific tasks to which they refer. This is done by means of advanced model-driven techniques able to keep aligned different views (i.e., models specified at the same level of abstraction) and to manage multi-scale models (i.e., models in which parts of the system are specified at different level of detail) by means of bidirectional transformations [11] and uncertainty management [13]. However, these approaches testify the benefits and advantages of applying theory and results from MDE on learning [19].
The inherit complexity arising using these models both in horizontal and vertical dimension is managed through the Zachman Framework adoption that helps in the in the models organization through the definition of the relations among them.

Acknowledgements. We thank our colleague Barbara Re from University of Camerino who provided us model fragments in the scenario discussed in Sect. 2. We also thank professor Alfonso Pierantonio for the precious comments that greatly improved the manuscript.

This research was supported by the EU through the Model-Based Social Learning for Public Administrations (Learn Pad) FP7 project (619583) For further informations visit the website: http://www.learnpad.eu. It is possible to find the repository hosting the platform implementation, concerning the part described in this article, here: https://github.com/LearnPAd/learnpad/tree/master/lp-model-transformer.

References

1. Business Process Model OMG. Notation (BPMN) 2.0. Object Management Group: Needham, MA, 2494:34 (2011)
2. OMG. Case Management Model and Notation (CMMN), V 1.0. Technical report, Object Management Group OMG (2013)
3. OMG. Business Motivation Model (BMM). Technical report, Object Management Group OMG (2014)
4. Akehurst, D., Kent, S.: A relational approach to defining transformations in a metamodel. In: Jézéquel, J.-M., Hussmann, H., Cook, S. (eds.) UML 2002. LNCS, vol. 2460, pp. 243–258. Springer, Heidelberg (2002). doi:10.1007/3-540-45800-X_20
5. Pierantonio, A., Rosa, G.: Design and initial implementation of metamodels for describing business processes in public administrations. Deliverable D3.2 - EU FP7 Project Learn PAd
6. Pierantonio, A., Rosa, G.: Domain Analysis of business processes in public administrations. Deliverable D3.1 - EU FP7 Project Learn PAd
7. Re, B., Sergiacomi, A.: Demonstrators BP and Knowledge models. Deliverable D8.1 - EU FP7 Project Learn PAd

8. Basciani, F., Ruscio, D., Iovino, L., Pierantonio, A.: Automated chaining of model transformations with incompatible metamodels. In: Dingel, J., Schulte, W., Ramos, I., Abrahão, S., Insfran, E. (eds.) MODELS 2014. LNCS, vol. 8767, pp. 602–618. Springer, Cham (2014). doi:10.1007/978-3-319-11653-2_37
9. Bézivin, J.: On the unification power of models. Softw. Syst. Model. **4**(2), 171–188 (2005)
10. Chen, K., Sztipanovits, J., Abdelwalhed, S., Jackson, E.: Semantic anchoring with model transformations. In: Hartman, A., Kreische, D. (eds.) ECMDA-FA 2005. LNCS, vol. 3748, pp. 115–129. Springer, Heidelberg (2005). doi:10.1007/11581741_10
11. Czarnecki, K., Foster, J.N., Hu, Z., Lämmel, R., Schürr, A., Terwilliger, J.F.: Bidirectional transformations: a cross-discipline perspective. In: Paige, R.F. (ed.) ICMT 2009. LNCS, vol. 5563, pp. 260–283. Springer, Heidelberg (2009). doi:10.1007/978-3-642-02408-5_19
12. Del Fabro, M.D., Bézivin, J., Jouault, F., Valduriez, P., et al.: Applying generic model management to data mapping. In: Proceedings of the Journées Bases de Données Avancées (BDA 2005) (2005)
13. Eramo, R., Pierantonio, A., Rosa, G.: Managing uncertainty in bidirectional model transformations. In: Proceedings of the 2015 ACM SIGPLAN International Conference on Software Language Engineering, SLE 2015, pp. 49–58, New York, NY, USA. ACM (2015)
14. Hausmann, J.H., Kent, S.: Visualizing model mappings in UML. In: Proceedings of the 2003 ACM Symposium on Software Visualization, pp. 169–178. ACM Press (2003)
15. Hay, D.C.: The Zachman Framework: An Introduction. The Data Administration Newsletter, Issue 1. Essential Strategies Inc. (1997)
16. Heisig, P.: Process modelling for knowledge management
17. Hinkelmann, K., Merelli, E., Thönssen, B.: The role of content and context in enterprise repositories. In: Proceedings of 2nd International Workshop on Advanced Enterprise Architecture and Repositories - AER 2010 (2010)
18. Inmon, W.H., Zachman, J.A., Geiger, J.G.: Data Stores, Data Warehousing and the Zachman Framework: Managing Enterprise Knowledge. McGraw-Hill Inc., New York (1997)
19. Laforcade, P., Choquet, C.: Next step for educational modeling languages: the model driven engineering and reengineering approach. In: Null, pp. 745–747. IEEE (2006)
20. Marc, M., Lankhorst, M.M.: Enterprise architecture modelling-the issue of integration. Adv. Eng. Inf. **18**(4), 205–216 (2004)
21. Melnik, S., Rahm, E., Bernstein, P.A.: Rondo: a programming platform for generic model management. In: Proceedings of International Conference on Management of Data, pp. 193–204. ACM Press (2003)
22. Oh, S., Sandhu, R.: A model for role administration using organization structure. In: Proceedings of the Seventh ACM Symposium on Access Control Models and Technologies, pp. 155–162. ACM (2002)
23. Pereira, C.M., Sousa, P.: A method to define an enterprise architecture using the Zachman framework. In: Proceedings of the 2004 ACM Symposium on Applied Computing, pp. 1366–1371. ACM (2004)
24. Pierantonio, A., Rosa, G., Silingas, D., Thönssen, B., Woitsch, R.: Metamodeling architectures for business processess in organizations. Projects Showcase@ STAF 2015, p. 27 (2015)

25. Reiter, T., Kapsammer, E., Retschitzegger, W., Schwinger, W.: Model integration through mega operations. In: Accepted for Publication at the Workshop on Model-driven Web Engineering (MDWE 2005) (2005)
26. Tupper, C.: Data Architecture: from Zen to Reality. Elsevier, Amsterdam (2011)
27. Yu, E., Strohmaier, M., Deng, X.: Exploring intentional modeling and analysis for enterprise architecture. In: 10th IEEE International Enterprise Distributed Object Computing Conference Workshops, EDOCW 2006, pp. 32–32. IEEE (2006)
28. Zachman, J.A.: The framework for enterprise architecture-cell definitions. ZIFA report (2003)
29. Zachman, J.A.: The Zachman Framework For Enterprise Architecture. A Primer For Enterprise Engineering And Manufacturing (2012)