



Defining the Responsibility Space for the Information Systems Evolution Steering

Jolita Ralyté, Wanda Opprecht, Michel Léonard

► To cite this version:

Jolita Ralyté, Wanda Opprecht, Michel Léonard. Defining the Responsibility Space for the Information Systems Evolution Steering. 9th IFIP Working Conference on The Practice of Enterprise Modeling (PoEM), Nov 2016, Skövde, Sweden. pp.179-193, 10.1007/978-3-319-48393-1_13 . hal-01653521

HAL Id: hal-01653521

<https://inria.hal.science/hal-01653521>

Submitted on 1 Dec 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution 4.0 International License

Defining the Responsibility Space for the Information Systems Evolution Steering

Jolita Ralyté¹, Wanda Opprecht², Michel Léonard¹

¹Institute of Information Service Science, University of Geneva, Carouge, Switzerland

²Ensemble Hospitalier de la Côte, Morges, Switzerland

jolita.ralyte@unige.ch, wanda.opprecht@ehc.vd.ch, michel.leonard@unige.ch

Abstract. Information System (IS) evolution is today a continuous preoccupation of every modern organization that aims to have a perfect support for its constantly changing business ecosystem. However, the task of IS evolution is not anodyne, it presents several risks towards IS sustainability as well as towards enterprise activity. Taking a decision related to any IS change can be a troublesome responsibility because of the amount of information that has to be processed and the uncertainty of the impact of the change on the enterprise and its IS. To reduce this uncertainty, we have developed a conceptual framework for IS evolution steering. This framework allows to capture the information on how enterprise IS supports its activities and regulations, and then to extract the information relevant for realizing IS evolution activities, simulating their impact, and taking appropriate decisions. This second part of the framework is called the responsibility space of IS evolution steering and is the main subject of this paper.

Keywords. Information system evolution, IS evolution steering, IS steering model, Responsibility space of IS evolution steering

1 Introduction

The constantly changing business and technology environment of modern organizations implies the necessity for continuous evolution of their Information Systems (IS) that are expected to fit it perfectly. However, the task of IS evolution is not anodyne, it presents several risks towards IS sustainability as well as towards enterprise structure and activity [1]. Taking a decision related to any IS change can be a troublesome responsibility. First, because of the amount of information that has to be processed while this information is not always available or easy to find, which creates the feeling of uncertainty. Second, because of the uncertainty of the impact of the change on the enterprise and its IS. Some of the problems that could appear are: the undetected inconsistency between the organization's activity and the IS functionality, the loss of regulatory compliance, conflicting IS evolutions, the impossibility to undo IS/Organizational changes, the loss of information, and the need to change the whole system when only part of it is impacted. Therefore, a tool supporting IS evolution steering is indispensable to guide and to reassure the officers responsible for this task.

In our previous work [2] we have introduced a conceptual framework for IS evolution steering. This framework includes several conceptual models each of them taking into account a particular perspective of IS evolution steering:

- the information on how enterprise IS supports its activities and regulations is captured in the *IS Steering Model* (IS-SM);
- the notion of IS evolution including its structure, lifecycle and impact assessment is represented in the corresponding *Evolution Models*;
- the responsibility of IS evolution steering actors on (1) the consistency of enterprise information (data) and on (2) the enterprise compliance with regulations governing its activities is formalized in *Ispace* and *Rspace* models respectively; and
- the guidance to use the aforementioned models, and so to help the actors in charge of IS evolution steering, is formalized in the *Evolution Steering Method*.

The framework constitutes the foundation for developing an Informational Steering Information System (ISIS). Up to now we only presented the general overview of the framework and its IS-SM model [2], and partially discussed the IS evolution models [3]. In this paper we focus our attention on the third part of our framework that deals with the notion of responsibility in IS evolution steering. In particular, we explain how to extract the information that is necessary to understand the scope of a particular IS evolution, to assess its impact, and finally to take appropriate decisions. We call this type of information the *Responsibility space*. As mentioned above, we distinguish two types of responsibility: the one over the enterprise information space that we call *Ispace*, and the other over the regulation space that we call *Rspace*.

The rest of this paper is organized as follows: in section 2 we discuss the responsibility issues in the domain of IS evolution. Section 3 provides an overview of the fundamental model of our framework – the IS-SM. Section 4 presents the main contribution of this paper – the models for defining the responsibility space of IS actors. In section 5 we briefly discuss the related works and we conclude in section 6.

2 Responsibility Issues in IS Evolution Steering

Information systems evolution is closely related to the changes undertaken in the organization itself and in its environment. A decision to move the organization from a current situation (ASIS-Org) to a new one (TOBE-Org) generally implies a more or less important change in its information system – the move from a current IS (ASIS-IS) to the new one (TOBE-IS). At each increment of ASIS-IS evolution towards TOBE-IS, the IS evolution steering officers have to take important decisions that could have more or less important impact on the TOBE-IS and therefore on the TOBE-Org. Indeed, they are directly responsible for the quality and sustainability of the TOBE-IS as well as its fitness to the TOBE-Org business. Besides, they are indirectly responsible for the future sustainability of the organization and its business, which depend on the result of the IS change. Therefore, the task of IS evolution steering is not so simple and is characterized by a high level of uncertainty, and the main reasons of that are as follows:

- Besides its operational importance, an information system has also a strategic significance for the organization. Indeed, it holds key information for the organization, and represents a strategic resource which underpins its key functions and decision-making processes.
- IS evolution may be triggered by business needs, legislation and/or strategic technological decisions. An organization, its business activities and its information systems are interwoven, so changes to one of them are likely to affect the others [4].
- The decisions on how to change the organization and how to change its IS are not taken at the same level and not by the same people.
- The decisions at organization's strategic level are generally made in situations distinguished by their uniqueness and uncertainty (e.g. business innovation). That makes the decisions at IS level even more risky. These decisions may have serious consequences and could jeopardize the sustainability of the organization [5].
- IS evolution requires knowledge [4] about IS structure, about dependencies between IS components and applications and how they support business activities, about how people work with the IS, and what are their rights and responsibilities. Having this knowledge is of prime importance in the IS evolution steering as it provides the basis for the action [6]. Knowledge deficiency, in the contrary, creates the situation of uncertainty.
- The complexity of IS evolution is due to the fact that several IS dimensions have to be taken into consideration [7, 8]: the *information dimension* responsible for the availability and integrity of data, the *regulatory dimension* ensuring IS compliance with the enterprise regulatory framework (standards, laws, regulation policies), the *activity dimension* supporting enterprise business activities, and the *technology dimension* dealing with the implementation and integration solutions.
- Finally, there are many different ways to realize an IS evolution, each of them having a different impact on the current and future condition of the organization's IS, and therefore, of the organization itself.

To reduce the uncertainty level that IS evolution steering actors have to face we need to guarantee that they possess all the required knowledge allowing to observe IS changes, to understand their impact, and to identify potential risks on the TOBE-IS and TOBE-Org. We agree with [4] that models are means to record this knowledge and make it accessible. Hence, to capture this knowledge, we have developed an Information System Steering Model (IS-SM) that allows to define how exactly enterprise activities and regulations are implemented in its information system. Actually, IS-SM is the underpinning model for the development of a meta-IS (IS upon IS in [9]) that we call ISIS – Informational Steering Information System. Enterprise IS and ISIS are not at the same level. IS is at the operational level, where actors (IS users) can query/create/delete/modify objects of IS classes, and trigger/control/stop operations on these objects according to their access rights. ISIS is at the IS steering level, where actors (IS steering officers) can query/create/delete/modify the design of classes, operations on these classes, integrity rules, processes, and access rights of the IS.

When a particular IS evolution is at stake, only a part of the information available in ISIS is needed – the ISIS entities that are directly or indirectly concerned by the

evolution. They represent the *responsibility space* of this IS evolution steering. Identifying this space contributes to reduce the risk of information overload [10], which could lead the IS steering actors to confusing estimations and inappropriate decisions. The main role of the responsibility space is to assist the evolution impact analysis.

Inspired by [11] and [12], we define the responsibility space as a set of ISIS instances that represent accountabilities and capabilities of an actor to perform a task. We distinguish two perspectives of responsibility:

1. *Ispace* representing the responsibility over information elements (i.e. objects, operations, and integrity rules implemented in the enterprise IS), and
2. *Rspace* representing the responsibility over regulatory elements (i.e. laws and regulation policies governing enterprise activities supported by its IS).

Formally, the Ispace/Rspace model is defined as a part of IS-SM. With Ispace/Rspace, we create sub-sets of information, extracted from ISIS, that inform the IS steering actors about the changes caused by an evolution affecting the responsibility of IS users. They allow to simulate IS evolutions and to identify potential risks. In the next section we overview IS-SM, and then in section 4 we define mechanisms to extract the Ispace and Rspace from IS-SM.

3 Overview of the Information System Steering Model

The role of the Information System Steering Model (IS-SM) consists in defining the information that can be obtained from the enterprise IS in a generic way. In particular, it allows representing the information related to the enterprise structure (organizational units and their composition), positions in an organizational unit, persons' assignments to one or several positions, their responsibility over different information elements, etc. Fig. 1 depicts a simplified version of IS-SM. More details about IS-SM can be found in [1], and its complete version is available online [13]. IS-SM contains three main parts: activity, information, and regulatory. In this section we provide a short summary of each of them.

The *activity part* of IS-SM (the left side in Fig. 1) describes the organization of the enterprise activity. In particular, it allows to capture the information on how different *persons* are related to the *organizational units* through *positions* they hold, which *activities* they can perform in these positions, how activities are combined in larger *business processes* and what are the *business rules* that control the execution of activities and business processes.

The *information part* of IS-SM (the right side in Fig. 1) reflects the usual definition of IS concepts such as *class*, *operation*, *integrity rule*, and their interrelations. In addition, this part takes into account the fact that usually an enterprise has several more or less interdependent information systems and many services can be built upon an IS. Therefore, IS-SM also includes the concept of *IS* and the concept of *service*, and relates all the elements of its information part with the IS and services where they are present. For readability purposes, these two concepts are not shown in Fig. 1; the complete model can be found in [13].

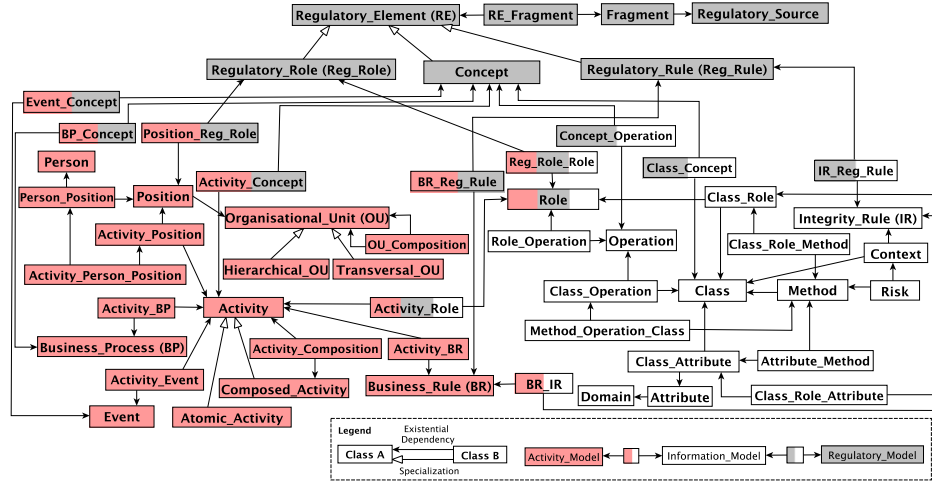


Fig. 1. Simplified version of IS-SM, see [13] for details

The main interrelations between the activity and information parts are established through the concept of *role*, which is a pivotal concept of IS-SM. It allows to connect activities and, consequently, persons who may perform these activities, to the IS classes and operations by defining the appropriate access rights. Other interrelations concern business rules from the activity part and integrity rules from the information one. A business rule is related to an integrity rule if the latter was created from the former.

The *regulatory part* of IS-SM (the top part in Fig. 1) expresses knowledge about science, techniques, standards, skills, laws, policies and regulations that are independent of the enterprise but govern enterprise activities. As a consequence, enterprise IS, supporting these activities, has to comply with them. This knowledge is formalized in terms of *regulatory elements* that can be *concept*, *regulatory role* and *regulatory rule*.

The interrelations between the activity and information part elements on one hand and the regulatory part elements on the other hand are established if the former are founded on the latter. Indeed, laws or other regulatory instruments can impose the creation of particular classes, operations, business rules, or positions. Generally, this information is not made explicit in the traditional IS. However, it is very important for the IS evolution steering to trace regulatory elements inside the IS [8]. In case they change (e.g. a change of a law), a conforming IS evolution must be triggered.

To conclude, we claim that an ISIS based on the IS-SM allows to capture all the information necessary to deal with IS evolution steering. However, this information is still too large when considering a particular ASIS-IS transformation into TOBE-IS. In the following sections we define mechanisms to extract the information that is really at stake for a particular IS change – the responsibility space of the change.

4 Defining the Responsibility Space from IS-SM

We study the notion of responsibility from the enterprise activity point of view, i.e. how any change in the organization (e.g. a hospital) affects the responsibility of a particular role in this organization (e.g. prescribing drugs to patients), an activity where this role has to be played (e.g. visit patients), a position to which this activity is allocated (e.g. doctor) and finally a person that is allocated to this position (e.g. Dr Laennec). In the following sub-sections we formally define how the Ispace/Rspace can be calculated for different IS-SM activity part elements. We illustrate these definitions with a simple example from the hospital domain.

4.1 Responsibility on the Information Space

The information space of a particular IS-SM activity part element x (i.e. role, person, position, or activity), retrieved from IS-SM and denoted $Ispace(x)$, represents the space of information accountability and capability of x . For example, if the Ispace of a person p , $Ispace(p)$, includes a class cl , then p has accountability over the objects of cl , and p is expected to have capabilities thereof (e.g. create, update, delete the objects of cl).

The formalization of $Ispace(x)$ and $Rspace(x)$ is inspired by the relational algebra in the following way:

$$Ispace/Rspace(x) = (Cl_A * Cl_B * Cl_C)[Cl_A = x][Cl_B]$$

where $*$ expresses the join between IS-SM classes, $[Cl_A = x]$ represents the selection of the object x from the class Cl_A , and $[Cl_B]$ expresses the projection on the class Cl_B .

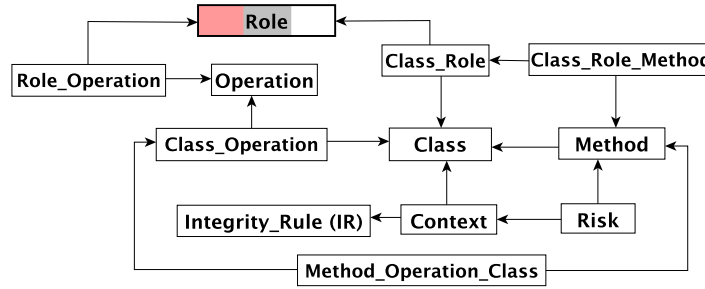


Fig. 2. The part of IS-SM allowing to retrieve the Ispace of a role.

Formally, Ispace of an element x is defined as a powerset $Ispace(x) = \langle Cl(x), Op(x), IR(x) \rangle$ that includes a set of classes $Cl(x)$, a set of operations $Op(x)$ and a set of integrity rules $IR(x)$ accessible from x and defined in IS-SM.

For example, the Ispace of a role r is defined as $Ispace(r) = \langle Cl(r), Op(r), IR(r) \rangle$. The part of IS-SM allowing to retrieve $Ispace(r)$ is shown in Fig. 2. Here, $Cl(r)$ represents the set of classes that r can access, and, therefore, for which r carries accounta-

bility (probably shared with other roles) and holds capability to execute methods on these classes. This set of classes can be accessed through the operation:

$$Cl(r) = (Role * Class_Role * Class) [Role = r][Class]$$

$Op(r)$ represents the set of operations that can be executed by the role r , and for which r carries accountability and holds the necessary capability. This set of operations can be accessed through the operation:

$$Op(r) = (Role * Role_Operation * Operation) [Role = r] [Operation]$$

Finally, $IR(r)$ includes the set of integrity rules that can be accessed by the role r via the methods and operation of the classes that are in $Cl(r)$ of its Ispace. Therefore, r carries accountability to validate these rules when this validation is not completely automatized and holds capability for that. The set of integrity rules of a role can be accessed through the operation:

$$IR(r) = (Role * Class_Role * Class_Role_Method * Method * Risk * Context * Integrity_Rule) [Role = r] [Integrity_Rule] \cup (Role * Role_Operation * Operation * Class_Operation * Method_Operation_Class * Method * Risk * Context * Integrity_Rule) [Role = r] [Integrity_Rule].$$

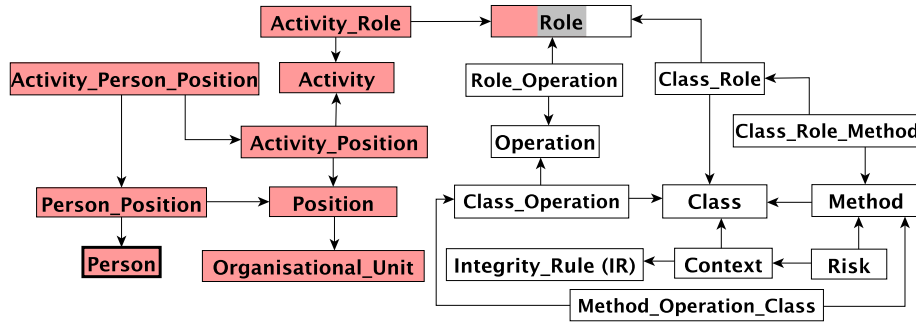


Fig. 3. The part of IS-SM allowing to retrieve the Ispace of a person.

The Ispace of a person p , $Ispace(p)$, represents the information elements (classes, operations and integrity rules defined in the organization's IS) that p needs to access in order to perform the activities related to her position(s) in the organization. Fig. 3 shows the part of IS-SM allowing to retrieve the $Ispace(p)$. In the course of evolution, if a person leaves the organization, her information space should be identified and transferred to another person to ensure the continuity of the business activities. On the opposite, if a new person enters the organization, she should be trained to use her information space. Because a person may have several roles, each of them related to the activities for which she has the duty in her position(s), $Ispace(p)$ is defined by the union of the $Ispace(r)$ of each of its roles:

$$Ispace(p) = \cup_{r \in Role(p)} Ispace(r), \text{ where}$$

$$\text{Role}(p) = (\text{Person} * \text{Person_Position} * \text{Activity_Person_Position} * \text{Activity_Position} * \text{Activity} * \text{Activity_Role} * \text{Role}) [\text{Person} = p] [\text{Role}]$$

In a similar way, Ispace is defined for positions and activities.

4.2 Responsibility on the Regulatory Space

The regulatory space of a particular IS-SM activity part element x (i.e. role, person, position, and activity), denoted $Rspace(x)$, is also retrieved from IS-SM. It represents a space of regulatory accountability and capability of x , i.e. how x is related to the regulatory elements (i.e. concepts, regulatory rules and regulatory roles) of the regulatory part of IS-SM. For example, if the Rspace of a person p , $Rspace(p)$, includes a concept c , then p has compliance responsibility over the objects of c , and possesses the capability thereof, i.e. the required knowledge.

Rspace of an element x is defined as a powerset $Rspace(x) = \langle Co(x), RRole(x), RRule(x) \rangle$ including a set of concepts $Co(x)$, set of regulatory roles $RRole(x)$, and a set of regulatory rules $RRule(x)$ accessible from x and defined in IS-SM. An IS-SM activity part element x may access a regulatory element in three different ways (see Fig. 1):

- Via a direct relation, e.g. *Activity* has a direct link to *Concept* via *Activity_Concept*.
- Indirectly via intermediate activity part elements, e.g. *Activity* can be related to *Regulatory_Role* through *Position* and *Position_Reg_Role*, or it can be related to *Concept* via *Event* and *Event_Concept*.
- Indirectly via intermediate information part elements, e.g. *Role* can be related to *Concept* via *Class*, and to *Regulatory_Rule* via *Class* and *Integrity_Rule*, where *Class* and *Regulatory_Rule* are elements of the information part of IS-SM.

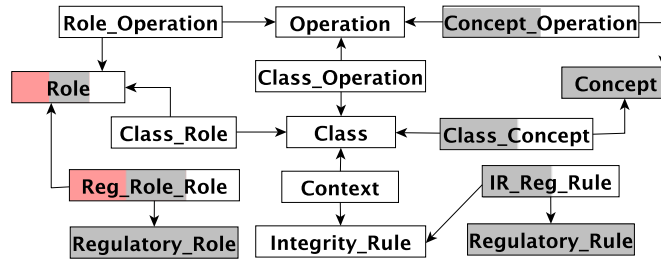


Fig. 4. The part of IS-SM that allows to retrieve the Rspace of a role.

The total Rspace of x will take into consideration all possible ways x can access the regulatory elements.

Fig. 4 shows the Rspace model for a role. The Rspace of a role r is defined as: $Rspace(r) = \langle Co(r), RRole(r), RRule(r) \rangle$.

$Co(r)$ includes a set of concepts (instances of the class *Concept*) that represent the regulatory foundation of the role r . These concepts can be implemented in the enter-

prise IS as classes or operations to which the role has access. To calculate $Co(r)$ we use the following operation:

$$Co(r) = (Role * Class_Role * Class * Class_Concept * Concept) [Role=r][Concept] \cup (Role * Role_Operation * Operation * Concept_Operation * Concept) [Role=r][Concept]$$

$RRole(r)$ represents the set of regulatory roles that are the "raison d'être" of a role r . In the course of IS evolution, if it happens that a regulatory role has no more corresponding role, we can deduce that either the IS is not compliant with the regulatory framework of the organization, or this regulatory role is now out of the IS scope. The set of regulatory roles of a role r can be accessed through the operation:

$$RRole(r) = (Role * Reg_Role_Role * Regulatory_Role) [Role = r][Regulatory_Role]$$

Finally, $RRule(r)$ represents a set of regulatory rules that the role r has to be compliant with. This compliance is implemented through the integrity rules over the classes to which this role has access. The operation to calculate $RRule(r)$ is:

$$RRule(r) = (Role * Class_Role * Class * Context * Integrity_Rule * IR_Reg_Rule * Regulatory_Rule) [Role = r][Regulatory_Rule]$$

In a similar way we define the Rspace for activities, positions and persons.

4.3 Illustrating Example

To illustrate the Ispace and Rspace definitions proposed above we use an example of the hospital ROL. Fig. 5 depicts a small part of the kernel of its information system schema.

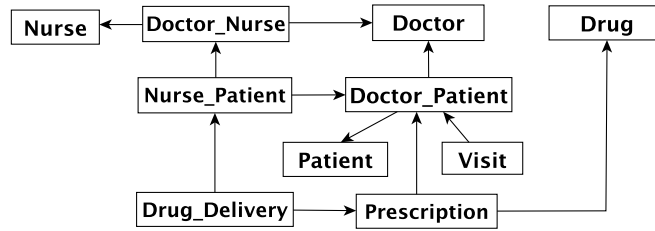


Fig. 5. A small part of the IS schema of the hospital ROL.

In this example we will consider:

- one organizational unit: the general medicine department,
- two positions: the doctor and the nurse,
- two activities of a doctor: a_1 concerning the care of patients (visit, diagnostic, prescription) and a_2 concerning the management of the nurses working in her team.

The activity a_1 is associated with two roles: r_{11} for the visit of patients, and r_{12} for the prescription of drugs to patients. To calculate the Ispace of the role r_{11} we apply the formulas provided above.

$Ispace(r_{11}) = \langle Cl(r_{11}), Op(r_{11}), IR(r_{11}) \rangle$:
 $Cl(r_{11}) = \{Doctor, Patient_Doctor, Patient, Visit, Prescription, Drug_Delivery, Nurse_Patient\}$ is the list of classes that the role r_{11} should access;
 $Op(r_{11}) = \{(\forall cl \in Cl(r_{11}), read(cl)), create(visit)\}$: r_{11} can read the objects of any class of $Cl(r_{11})$ and can create an object of the class $Visit$;
 $IR(r_{11})$ is empty.

Therefore, the Ispace of the doctor Laennec related to the role r_{11} is:

$Ispace(Laennec / r_{11}) = (Cl(r_{11}), Op(r_{11}), \emptyset) [Doctor = 'Laennec']$.

This selection, as in the relational model, means that any object accessible by Laennec must be reachable by join operation with the object of the class *Doctor* identified by Laennec. So, for instance, Laennec can create an object of *Visit* only for his patients and can access to the objects of *Patient* only for his patients.

Similarly, we calculate the Ispace of the role r_{12} :

$Ispace(r_{12}) = \langle Cl(r_{12}), Op(r_{12}), IR(r_{12}) \rangle$:
 $Cl(r_{12}) = Cl(r_{11})$;
 $Op(r_{12}) = \{(\forall cl \in Cl(r_{12}), read(cl)), create(prescription)\}$; $IR(r_{12}) = \emptyset$.

The activity a_2 is associated with two roles: r_{21} and r_{22} – the arrival and the departure of a nurse in/from the team of a doctor. Indeed, in the ROL hospital, a doctor has the responsibility to accept/refuse/fire a nurse in her team. The Ispace of the roles r_{21}/r_{22} is as follows:

$Ispace(r_{21}) = \langle Cl(r_{21}), Op(r_{21}), IR(r_{21}) \rangle$:
 $Cl(r_{21}) = \{Doctor, Docor_Nurse, Nurse\}$;
 $Op(r_{21}) = \{(\forall cl \in Cl(r_{21}), read(cl)), create(doctor_nurse)\}$; $IR(r_{21}) = \emptyset$.

$Ispace(r_{22}) = \langle Cl(r_{22}), Op(r_{22}), IR(r_{22}) \rangle$.
 $Cl(r_{22}) = \{Doctor, Docor_Nurse, Nurse\}$;
 $Op(r_{22}) = \{(\forall cl \in Cl(r_{22}), read(cl)), delete(doctor_nurse)\}$; $IR(r_{22}) = \emptyset$.

To illustrate an evolution case, let's suppose that now the ROL hospital wants to get into improving patients safety. To this end, it proposes to introduce the concept of skill. Following international standards, some particular skills will be required for the delivery of drugs to patients. The doctor will be in charge to guarantee that nurses of her team have sufficient competences for administrating all the drugs she can prescribe. So, the IS of ROL must evolve, especially by introducing new classes such as *Skill*, *Nurse-Skill* associating a nurse to her skills, *Required-Skill* associating a drug to the skills required to administrate it, and the class *Doctor_Drug* associating a doctor to any drug she can prescribe. The new IS schema is shown in Fig. 6.

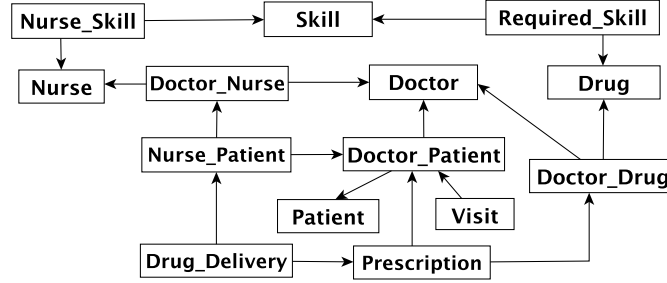


Fig. 6. The IS schema of the hospital ROL after the evolution.

After this evolution, the position “doctor” has still the same activities a_1 and a_2 , and the activity a_1 has the same roles r_{11} and r_{12} . But, for the activity a_2 , the roles r_{21} and r_{22} , must be extended into r_{211} and r_{221} . Furthermore, this activity has additional roles r_{23} and r_{24} related to the skills and the drugs. Here are the definitions of the Ispace of r_{211} , r_{221} , r_{23} and r_{24} :

r_{211} : when dealing with the arrival of a new nurse, the doctor can check if the nurse has sufficient skills given the skills of the other team nurses.

$Ispace(r_{211}) = \langle Cl(r_{211}), Op(r_{211}), IR(r_{211}) \rangle$:

$Cl(r_{211}) = \{Doctor, Doctor_Nurse, Nurse, Nurse_Skill, Skill, Doctor_Drug, Drug, Required_Skill\}$;

$Op(r_{211}) = \{(\forall cl \in Cl(r_{211}), read(cl)), create(doctor_nurse)\}$;

$IR(r_{211}) = \emptyset$.

r_{221} : the doctor can check if the departure of the nurse has serious consequences given the skills of the other team nurses.

$Ispace(r_{221}) = \langle Cl(r_{221}), Op(r_{221}), IR(r_{221}) \rangle$:

$Cl(r_{221}) = \{Doctor, Doctor_Nurse, Nurse, Nurse_Skill, Skill, Doctor_Drug, Drug, Required_Skill\}$;

$Op(r_{221}) = \{(\forall cl \in Cl(r_{221}), read(cl)), delete(doctor_nurse)\}$; $IR(r_{221}) = \emptyset$.

r_{23} : the doctor can add or remove skills to a nurse.

$Ispace(r_{23}) = \langle Cl(r_{23}), Op(r_{23}), IR(r_{23}) \rangle$:

$Cl(r_{23}) = \{Doctor, Doctor_Nurse, Nurse, Nurse_Skill, Skill, Doctor_Drug, Drug, Required_Skill\}$;

$Op(r_{23}) = \{(\forall cl \in Cl(r_{23}), read(cl)), delete(nurse_skill), create(nurse_skill)\}$;

$IR(r_{23}) = \emptyset$.

r_{24} : the doctor can add/remove a drug to/from her list of drugs.

$Ispace(r_{24}) = \langle Cl(r_{24}), Op(r_{24}), IR(r_{24}) \rangle$:

$Cl(r_{24}) = \{Doctor, Doctor_Drug, Drug\}$;

$$Op(r_{24}) = \{(\forall cl \in Cl(r_{24}), read(cl)), create(doctor_drug), delete(doctor_drug)\};$$

$$IR(r_{24}) = \emptyset.$$

With this example we can also briefly show the relevance of the Rspace. The class *Skill* can be considered as a concept of the regulatory space because its origin can come from an international standard, which is independent of the hospital. Thanks to this concept, a nurse can position herself in her work domain. For example, if she wishes to move to another team, she should compare her present skills with the required skills of this team. To make the transfer possible, she needs to know the list of the skills required by this team. Consequently, she must be allowed not only to access all the skills of her team (through the required skills of the drugs of her doctor) but also all the skills of the other teams. This situation emerges only because of the regulatory space, not at all because of the operational activities of the hospital. Hence, this situation shows how the knowledge of the enterprise IS and its different dimensions can be useful not only for operational purposes but also to support IS evolution.

5 Related Works

The subject of IS evolution was largely discussed in the literature taking into account various perspectives and positions. Most of the works consider a particular aspect of evolution (e.g. changes in data schema, requirements, technology, and architecture, business reengineering and enterprise reorganization) without trying to provide a holistic approach and tool support for its steering. Though, IS evolution steering is at the crossroads of several research areas such as: evolution models, Business/IT alignment, Enterprise Architecture (EA) and Enterprise Modelling (EM), IS governance and risk management. Here, because of the space limit, we will mention only a few works, mainly with the aim to demonstrate the position of our contribution.

Comyn-Wattiau et al. [14] emphasize that IS evolution and the mechanisms for supporting it highly depend on its different facets or dimensions, like the nature of change, the time frame and the importance. Similarly, various IS change dimensions and taxonomies are proposed in the domains of data management [15] and business process management [16] taking into consideration various aspects of change like its subject, cause, type, extent, effect, swiftness, temporal and spatial issues. We agree with the idea that IS evolution is multifaceted. Our framework for IS evolution steering includes three interrelated dimensions (i.e. activity, information and regulation) each of them being a potential trigger and a subject of the change. Besides, we cope with different IS evolution perspectives: its structure, lifecycle, impact and responsibility.

Let's look now how our framework could be situated with regards to the EA/EM contributions. Based on the literature review, Niemi and Pekkola [17] define EA as 'an approach to managing the complexity of an organization's structures, information technology (IT) and business environment, and facilitating the integration of strategy, personnel, business and IT towards a common goal through the production and use of structural models providing a holistic view of the organization'. They also claim that

‘because of this scope, EA can be approached from a number of viewpoints’. Indeed, most of the EA frameworks (e.g. Zachman Framework [18], TOGAF [19], CIMOSA [20]) and EM languages (e.g. EKD-CMM [21], DEMO [22], ArchiMate [23]) acknowledge the need for multiple views and abstraction levels. They are necessary to cope with IS/enterprise architecture complexity and separation of concerns, and to address the different life spans of the architecture elements [24, 25]. These approaches expose best practices and generic principles that support the creation of ‘blueprints for shared vision’ [24] of the current (ASIS-Org and ASIS-IS) and desired (TOBE-Org and TOBE-IS) situations. However, they fail to offer a formal evolution steering method and tool supporting continuous IS changes. We acknowledge that they provide a detailed EA picture and propose approaches for EA evolution [26]. Though, they do not offer means to measure the impact of the changes on the IS instances and on the organisation activities, and the regulation dimension is clearly missing.

Another perspective we would like to address is the notion of risk related to the IS evolution. IS evolution is the entanglement of organizational, information, regulatory and technology issues and hence, the nature of IS evolution risk is as complex as the setting they stem from. In their study of IS risks Alter and Sherer [27, 10] review a large amount of dedicated literature from which they identify risk components (e.g. financial, functionality, organisational, technology, security, people, information, political, etc.), organize risk factors and propose a general and adaptable model of IS risk. From this and other related studies [28, 29] we retain that risk is related to uncertainty. We argue that a way to cope with uncertainty in IS evolution steering is by providing the precise information about the IS elements that are at stake. This is the main role of our framework.

To conclude, we propose to discuss the notion of responsibility, which is, to our best knowledge, the less considered topic in the domain of IS evolution. In [11, 12] Feltus et al. propose a metamodel formalizing the notion of responsibility of IS actors. In this model the responsibility of an actor is related to a business role she plays in the organization, and represents her accountabilities on a business task and the required capabilities and rights to perform those accountabilities. Our framework for IS evolution steering is compliant with this responsibility definition. Furthermore, with the Ispace and Rspace definition mechanisms, we provide a powerful tool for assessing and managing the responsibility of IS actors, and hence, the responsibility of IS evolution steering officers.

6 Conclusion

Every modern organization understands that its prosperity largely depends on the quality and fitness of its information systems. Because of the constant changes in the organization and its environment, IS evolution is permanently at stake. Actors, responsible for IS evolution steering, are challenged to take decisions having impact on the enterprise business. To be able to take these decisions, they must have a thorough knowledge of the situation allowing to assess the impact of changes. The aim of our work is to support IS evolution steering with conceptual tools allowing to reduce the

uncertainty inherent to IS evolution and the complexity pertaining to the multiple IS dimensions. The main model of our conceptual framework, named IS-SM, represents a foundation for developing an information system for IS evolution steering, that we call ISIS. After a brief review of IS-SM, we present the main contribution of this paper, which deals with identifying the responsibility space of IS actors. This responsibility is considered from two complementary points of view: information (Ispace) and regulation (Rspace). The Ispace and Rspace defined systematically on the current IS (ASIS-IS) and future IS (TOBE-IS) allow to measure the delta of the IS evolution from information and regulatory points of view respectively. They offer a view on how the responsibility space will change for each actor of the organization. A person can be granted with new responsibilities (e.g. creating, deleting, modifying objects) over an existing or newly created information space. Is she ready to assume them? Does she have capabilities? Such changes at IS level require decisions at organization's steering level, while the information necessary to support the decision taking can be obtained at the IS level. We are convinced that the implementation of ISIS would be helpful not only for IS evolution steering but also when dealing with enterprise reorganization.

References

1. Mukherji, A.: The Evolution of Information Systems: their Impact on Organizations and Structures. *Management Decision*, 40(5), 497 – 507 (2002)
2. Opprecht, W., Ralyté, J., Léonard, M.: Towards a Framework for Enterprise Information System Evolution Steering. Frank, U. et al. (eds.), *PoEM 2014, LNBIP 197*, pp. 118–132, Springer (2014)
3. Ralyté, J., Opprecht, W., Léonard, M.: Reorganizing an Enterprise thanks to its Information System. *Proc. of ICEIS 2016*, vol. 2, pp. 567-574, ISBN: 978-989-758-187-8, SCITEPRESS (2016)
4. Hofer, S.: Modeling the Transformation of Application Landscapes. Grabis, J. et al. (Eds.) *PoEM 2013, LNBIP 165*, pp. 101–113, Springer (2013)
5. Lesca H. and Lesca N.: *Strategic Decisions and Weak Signals: Anticipation for Decision-making*, Wiley USA, 144 p (2014)
6. Davis, G. B.: Information Systems Conceptual Foundations: Looking Backward and Forward. R. Baskerville et al. (eds.), *Organizational and Social Perspectives on Information Technology*, IFIP vol. 41, pp. 61–82, Springer (2000)
7. Estier T., Falquet G., Guyot J. and Léonard M.: Six Spaces for Global Information Systems Design. TC8 (Ed.), *Proc. of IFIP Working Conference on The Object Oriented Approach in Information Systems*, (1991)
8. Léonard, M., Khadraoui, A. and Ralyté J.: Regulation in Information Systems at the Level of Tunement. *Proc. of Workshop on Regulations Modelling and their Validation and Verification (REMO2V)*, at CAiSE'06, *CEUR-WS* vol. 241, (2006)
9. Le Dinh, T.: Towards a New Infrastructure Supporting Interoperability of Information Systems in Development: the Information System upon Information Systems. Konstantas, D., et al. (Eds.) *Interoperability of Enterprise Software and Applications*, pp. 385-396, Springer (2006)
10. Sherer, S. A. and Alter, S.: Information System Risks and Risk Factors: Are They Mostly About Information Systems. *Communications of the AIS*, 14(2), 29–64 (2004)

11. Feltus, C., Petit, M., Dubois, E.: ReMoLa: Responsibility Model Language to Align Access Rights with Business Process Requirements. *Proc. of RCIS 2011*, pp. 1–6 (2011)
12. Khadraoui, A., Feltus, C.: Service specification and service compliance: How to consider the responsibility dimension? *Journal of Service Science Research*, 4(1), 123–142 (2012)
13. IS-SM, <http://cui.unige.ch/~rallyte/ISIS/IS-SM.html>
14. Comyn-Wattiau, I., Akoka, J. and Lammari, N.: A Framework for Database Evolution Management. *Workshop on Unanticipated Software Evolution (USE 2003)*, in conjunction with ETAPS'03, Warsaw, Poland, (2003)
15. Roddick, J. F., Al-Jadir, L., Bertossi, L., et al.: Evolution and Change in Data Management: Issues and Directions. *SIGMOD Record*, 29(1), 21–25 (2000)
16. Regev, G., Soffer, P. and Schmidt, R.: Taxonomy of Flexibility in Business Processes. In: *Proc. of the CAISE'06 Workshop on Business Process Modelling, Development, and Support (BPMDS'06)*, CEUR-WS vol. 236, (2006)
17. Niemi, E. and Pekkola, S.: Enterprise Architecture Quality Attributes: A Case Study. *Proc. of the 46th Hawaii International Conference on System Sciences (HICSS)*, pp. 3878–3887 (2013)
18. Zachman, J.A.: A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3), 276–292 (1987)
19. The Open Group: TOGAF version 9.1, An Open Group Standard. <http://pubs.opengroup.org/architecture/togaf9-doc/arch/>, [accessed April 2016] (2012)
20. ESPRIT Consortium AMICE (Ed): *Open Systems Architecture for CIM*, 2nd. Revised and Extended Version. Springer, Berlin (1993)
21. Barrios, J. and Nurcan, S.: Model Driven Architectures for Enterprise Information Systems. Persson, A. and Stirna, J. (eds.), *CAiSE 2004*, LNCS 3084, pp. 3–19. Springer, Heidelberg (2004)
22. Dietz, J.L.G.: DEMO: Towards a Discipline of Organisation Engineering. *European Journal of Operational Research*, 128, 351–363 (2001)
23. ArchiMate: <http://www.archimate.nl/>
24. Armour, F. J., Kaisler, S. H. and Liu, S. Y.: A Big-Picture Look at Enterprise Architectures. *IT Professional*, 1(1), 35–42 (1999)
25. Greefhorst, D., Proper, E.: *Architecture Principles - The Cornerstones of Enterprise Architecture*. The Enterprise Engineering Series 4, Springer (2011)
26. Gaaloul K., Guerreiro, S.: A Decision-Oriented Approach Supporting Enterprise Architecture Evolution. *Proc. of the 24th IEEE International Conference WETICE 2015*, pp. 116–121, IEEE (2015)
27. Alter, S. and Sherer, S.: A General, But Readily Adaptable Model of Information System Risk. *Communications of the AIS*, 14(1), 1–28 (2004)
28. Barki, H., Rivard, S. and Talbot, J.: Toward an Assessment of Software Development Risk. *Journal of Management Information Systems*, 10(2), 203–225 (1993)
29. Copas, J.: Statistical Modelling for Risk Assessment. *Risk Management*, 1(1), 35–49 (1999)