Chapter 15
Handling the Evolution of Information Systems: An Overview of Challenges and Prospective Solutions

Michel Léonard and Jolita Ralyté

Abstract Evolution is characteristic to all enterprise information systems (IS) because of continuing changes in its environment. It is also a necessary condition for guaranteeing IS fitness to the organizational needs and requirements. Nonetheless, each IS evolution presents several risks towards their sustainability and requires an accountable steering. In this chapter we consider two major challenges related to the IS evolution: the way to design and implement legacy IS evolution and the why to govern it. We look for responses to those challenges in existing literature and we review our previous and on-going work. In particular, we promote the use of service-oriented paradigm to deal with the complexity, interoperability and evo-lution of legacy IS, and we propose the concepts of information service and in-formation services system (ISS) as well as different ways to design an ISS. Con-cerning the second challenge, we propose a framework for IS evolution steering that aims to guide the actors responsible for this complex task by providing the in-formation necessary to realise IS evolution activities and to simulate their impact.

15.1 Introduction

Business and information technology innovation are two important evolution drivers in today's organizations. They lead them to take new forms, to reengineer their business processes and update technologies, and they also imply the creation of new types of inter-organizational and networked information systems (IS) and to offer online services. These changes are necessary and permanent at all enterprise lev-

Michel Léonard
University of Geneva, Institute of Information Service Science, Switzerland,
e-mail: michel.leonard@unige.ch

Jolita Ralyté
University of Geneva, Institute of Information Service Science, Switzerland,
e-mail: jolita.ralyte@unige.ch

J. Cabot et al. (eds.), *Conceptual Modeling Perspectives*,
https://doi.org/10.1007/978-3-319-67271-7_15

els: strategic development, business management and operation, and information systems.

In this paper we consider enterprise IS sustainability as one of the major issues in enterprise evolution. It is clear that it is not possible to replace existing IS by new ones for each enterprise business and/or organizational change – legacy IS have to evolve together with enterprise changes. IS evolution can take different forms: the integration of new components from the market or custom-made, the development of services on top of the existing IS, the establishment of interoperability between two or more IS, etc. An inappropriate way to do that would lead to the IS fragmentation, and therefore to the redundancy between different IS parts. IS redundancy entails a need for permanent validation of the consistency of data, processes and rules.

In this context, service-oriented approaches emerge as prospective ones to deal with IS fragmentation, interoperability and evolution situations [6, 15, 32], and as a support for inter-organizational IS development [24, 25, 31]. However, the shift from a conventional IS architecture to a service-oriented one is not an easy task despite of the various service design approaches proposed in the literature. In section 2 of this chapter we discuss the notion of information service as a fundamental concept for designing service-oriented IS that we call Information Services Systems (ISS). Then, we review three different approaches to design an ISS from scratch or by reuse of legacy IS.

IS evolution is a necessary condition for guaranteeing IS fitness to the enter-prise business needs and requirements. However, each IS evolution presents several risks towards its sustainability and further changes. Therefore, another im-portant issue in IS evolution is the impact and the responsibility of its steering.

Every change in enterprise organization, business activity, or regulation inevitably entails a chain of evolutions of its IS and information services. Actors, responsible for IS evolution steering, have to take important decisions those impact on the enterprise business and legacy IS can be devastating. To be able to make these decisions, they must have a thorough knowledge of the situation. In our work, we claim that this information can be extracted from enterprise IS. In section 3 we discuss different issues related to the IS evolution steering and overview some related works. Then, in section 4 we outline our work in this domain and, in particular, we describe our proposal for a framework supporting IS evolution steering.

15.2 A Service-Oriented Perspective to IS

Today, service orientation is considered as a new design paradigm for increasingly complex IS engineering which promises to improve their flexibility and changeability. The literature review demonstrates the advent of proposals to redesign conventional IS architectures into the service-oriented ones [6, 15, 24, 31]. Recently we have introduced the notions of information service [9, 10] and information services system [33] and proposed several approaches applying the service paradigm to support IS evolution [10, 33].

In order to fit the IS context, an information service has to support inter-organizational and/or intra-organizational business activities through a collaborative creation, transformation and transmission of information. An ISS, on its turn, aims to ensure the consistency of enterprise information by supporting its creation, management and sharing through the use of information services. At the same time, it improves the modularity, agility and interoperability of IS architecture. Below, we summarize these two notions and overview three approaches to design ISS.

15.2.1 Information Service

The notion of information service [9, 10] is built upon the concept of IS component [42], and is defined as "a component of an information system representing a well defined business unit that offers capabilities to realize business activities and owns resources (data, rules, roles) to realize these capabilities". In other words, it is de-fined over classes, methods, integrity rules, processes, roles and events that constitute a semantic unit where several actors aim to achieve a common goal. Consequently, an information system can be seen as built of a collection of interoperable information services.

The particularity of the information service definition (in comparison to the web-service definition) consists in requirement for the service to be transparent. In the IS context it is not sufficient to consider services as black boxes with only inter-face part available for their selection and composition purposes. It is essential to make explicit the information concerning service structure, processes, rules and roles, and to be able to identify what is shared with other services. Fig. 15.1 shows the simplified metamodel of the information service where only the main concepts are represented (see [9] for the detailed version). As shown in this figure, the information service definition is composed of four interrelated information spaces: static, dynamic, rules and roles.

The *static space* of the service defines its data structure in terms of classes, relationships between classes and attributes. The notion of *hyperclass* (introduced in [41] to specify IS components) is used to represent complex domain concepts by putting together the corresponding set of classes. Classes are linked only via existential dependencies and specialization relationships. An existential dependency is materialized via an attribute with mandatory and permanent constraints.

The *dynamic space* defines service capabilities in terms of actions and their effects on service classes. An action is triggered by an event that occurs in the service information space and is described by a process to be executed by one or several actors having the responsibility on this action. An action produces one or more effects on the static space trough primitive methods (e.g. create an object of a class, modify an attribute). The notion of effect is used to characterize the result of the action and allows to evaluate the impact of the action on the static space.

The *rule space* deals with service regulation policies that are formalized as integrity rules validating service data, and pre-, post-conditions controlling service actions (not shown in Fig. 15.1). An integrity rule is associated to a context and to a set of risks that represent all the methods of different service actions that could transgress the rule.

Finally, the *role space* describes the roles the actors have on service actions, depending on the responsibilities they assume in the organization. Altogether, the four spaces compose a consistent and complete view of an in-formation service and establish a foundation for different information service and information services systems engineering approaches.

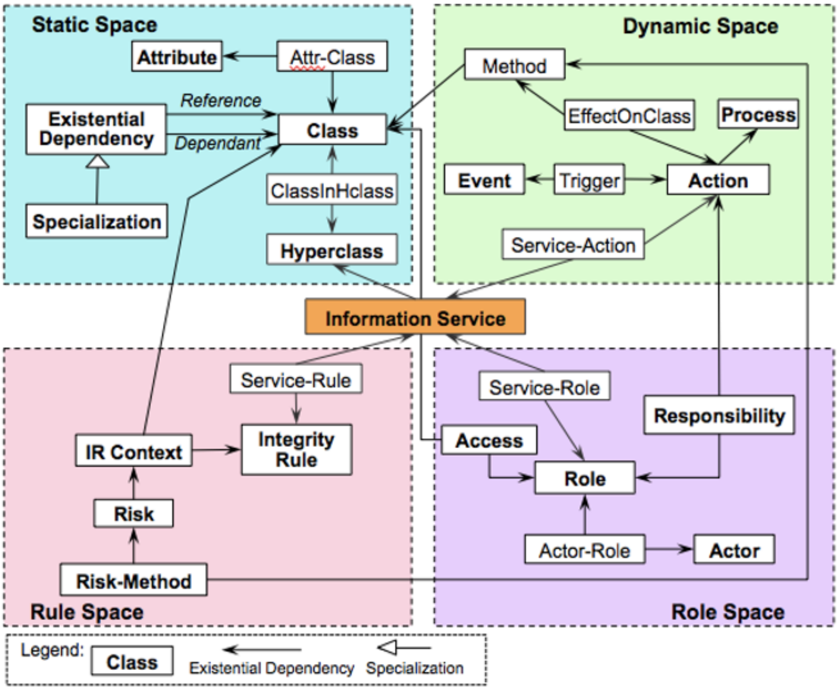

Fig. 15.1 Simplified metamodel of the information service

15.2.2 Information Services System

We define an Information Services System (ISS) [33] as a collection of interoperable information services as presented above. This definition takes inspiration from works by Spohrer et al. [38, 39] in the domain of Service Science. They define a service system as "a value-coproduction configuration of people, technology, other internal and external service systems, and shared information (such as language, processes, metrics, prices, policies, and laws)" with the aim to create a mutual value.

The notion of value coproduction is also key in the domain of information services systems. An ISS aims to provide services that allow actors to co-execute business activities by means of service actions and to coproduce shared information. The scope of actors' behavior inside an ISS depends on the responsibilities assigned to them. It is explicitly described by means of roles that allow to en-act service actions in compliance with the rules embedded in the corresponding services. The main challenge of ISS consists in transforming an integrated and rather rigid IS architecture into a more flexible, modular and sustainable one providing facilities to easily modify existing services and/or integrate the new ones.

15.2.3 Designing Information Services Systems

Shifting from conventional IS to the service-oriented ISS is not an obvious task, especially then various legacy IS are at stake. Such transition needs to be carefully designed and governed. It has to take into consideration not only technical implementation but also conceptual design and business strategic issues.

The review of related literature reveals that the number of approaches for service-oriented IS engineering is growing, however many of them consider only technical integration or migration of legacy IS to the service-oriented technology [6, 17, 21, 44], and propose to reuse legacy code to provide web services [27, 37]. Nevertheless, at design level, a few conceptual frameworks have been proposed including the framework for designing service-oriented inter-organizational IS [24] and the one for service modeling in a network of service systems [25]. A model-driven approach for service oriented IS development introduced in [19] mainly focuses on mapping rules from BPMN models to SOAML diagrams. The goal modeling technique i*, adapted to the service-oriented business modeling, is underpinning in the reference catalogue approach to design an SOA system [28]; it guides the selection of reference business models from the catalogue and their adaptation to the particular case. Finally, at business strategic level, the adoption of service-oriented paradigm also turns to be a real challenge. A few publications discuss how to assess legacy IS for the evolution towards service-oriented architectures [35, 36] and analyze the impact of SOA on enterprise systems [13]. In order to determine whether the introduction of SOA justifies the effort, [40] propose a value-driven approach to design service-oriented IS based on business process modeling and cost/benefit analysis. Other research works define critical success factors of service orientation in IS engineering [5], discuss strategies for service-oriented IS design [4] and how service-oriented design should be applied in an organization in order to adopt SOA for IS engineering [15].

In our research group, we have explored three generic approaches guiding the design of information services systems while taking into consideration the evolution of enterprise legacy IS [33]. Each approach deals with a particular organizational context and ISS design situation, as well as legacy IS reuse. We summarize them below.

15.2.3.1 Services upon Legacy IS

This approach, originally introduced in [23], guides the definition of new information services upon various existing legacy IS by reusing their data, processes, rules, and roles. It aims to bring some flexibility and modularity to the rather monolithic and fragmented legacy enterprise IS without inflicting to them any major transformation. The approach consists in identifying for each new service the existing resources that are potentially scattered in different IS and to guarantee that the execution of the service will keep these legacy IS in a consistent state, i.e. will ensure data consistency and will not violate their rules and responsibilities. The key step of this approach (sketched in Fig. 15.2) consists in defining a common base on top of a set of existing IS. The role of this common base consists in (1) specifying the overlapping information available in different IS, (2) offering each service the access to the precise and consistent information distributed in those IS, and (3) guaranteeing service compliance with a particular organizational context and with the enterprise legal frame, which is a composition of laws and regulation policies that govern enterprise activities.

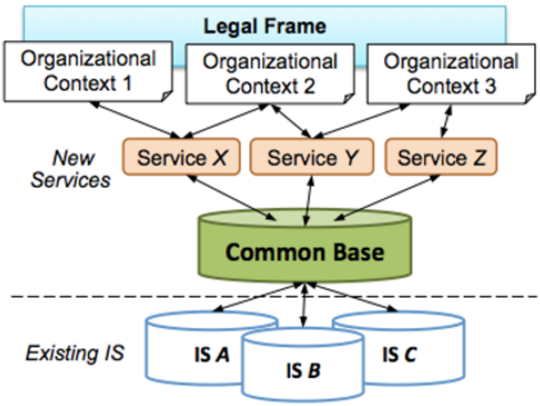

Fig. 15.2 An overview of the approach "Service upon Legacy IS"

15.2.3.2 Fully Service-Oriented ISS

This approach, developed in [10, 32], in the contrary, considers an information system as a composition of information services, where each service provides a sup-port for a particular business or administration activity. Therefore, it requires a preliminary ?decomposition? (at least at conceptual level) of the existing IS into a collection of information services, and defining the overlap (common data, activities, roles, rules) between them. Information overlap management represents the biggest challenge when including new services into an existing ISS. In fact, the overlap between information services can exist in the four information spaces (static, dynamic, rule and role), and the integration of each new service creates new overlap situations (see the idea in Fig. 15.3). Therefore, this approach is based on the analysis and resolution of overlap inconsistencies between legacy and new services.

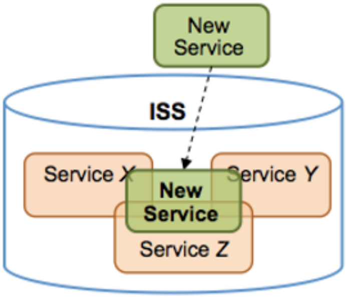

Fig. 15.3 An overview of the approach "Fully Service-Oriented ISS"

15.2.3.3 Information Kernel-Based ISS

This approach proposes an architecture based on a core IS and information services as its extensions. The core IS captures the kernel information – the most stable data, processes and rules – while information services offer capabilities for business activities that are subject to change. In this type of architecture the main challenges consist in (1) defining the information kernel, which is formalized as a collection of kernel services, and (2) preserving this kernel when adding new services to the ISS.

The approach argues that the information kernel can be obtained from the enterprise legal frame that includes laws and other regulation policies governing enterprise activities. Such documents generally define concepts, rules and constraints related to the institutional activities, and represent a rich source of knowledge for the ontological information extraction and the information kernel conceptualization. Therefore, this approach consists in constructing the ontological level model based on the analysis of different legal sources, and then, mapping this model into the conceptual model representing the kernel ISS (see the overview in Fig. 15.4). The extension of the kernel ISS with new services can then follow the Fully Service-Oriented ISS engineering approach. More details about this approach can be found in [33], while some examples of the ontological model construction from the legal frame are given in [22].

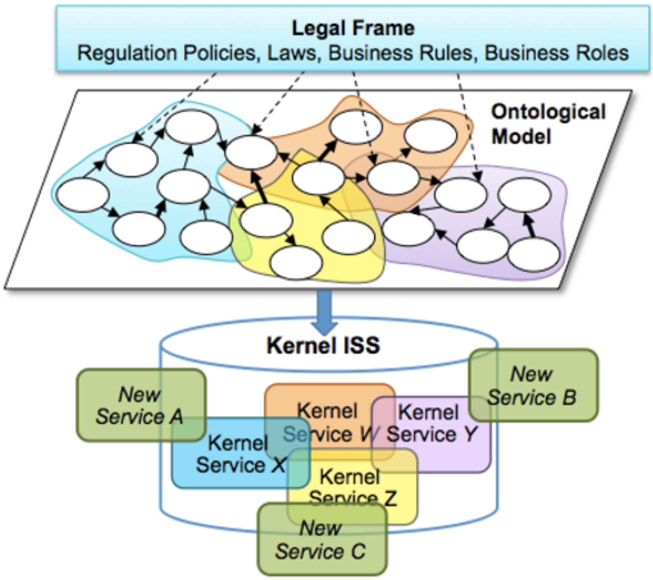

Fig. 15.4 An overview of the approach "Information Kernel-Based ISS"

15.3 IS Evolution Steering

As said in the introduction, evolution is inherent to every IS. Even more, evolving is its permanent condition because of its ever-changing environment where contingencies may arise from various dimensions such as: enterprise structure (e.g. reorganization of business units, merger or buyout of companies) business activity (e.g. establishment of new business processes), technology (e.g. introduction of new hard or soft technology), or regulation (e.g. law abrogation, modification or creation, adoption of new industrial standards). In order to ensure IS sustainability, its evolution must be understood and supported, i.e. steered.

The main challenge of the IS evolution steering is to cope with the proliferation and complexity of enterprise IS as well as with the uncertainty of the impact of their changes on the organization itself. The proliferation and overlap of IS are generally due to the inconsistent management of their evolution. IS complexity, in the contrary, is a characteristic by definition caused by the entanglement of multiple dimensions such as regulation (laws and rules governing enterprise activities), responsibility (organizational units and roles), information (its structure and provisioning), activity (business processes and collaborations), and the underpinning technology. While IS evolution is necessary, it also presents several risks related to the enterprise business. For example, if not all significant information is available during a particular IS change, the evolution can fail to fit business activities or to comply with the enterprise regulatory framework.

In our research, we assume that in every organization several IS are potentially at stake during IS evolution steering. Either wholly (or partly) dependent or in-dependent from each other, they support activities of the organization at different organizational levels (i.e. strategic, tactic, operational). Some of them have been developed and evolved in silos, and therefore testify to the consequences of the organizational restructuring, changes of the organization activities, or the involvement of the organization into new collaborations. This situation causes important issues regarding IS interoperability at the information, technical and organizational levels, and is particularly critical when the organization aims to adopt a service-oriented paradigm.

The responsibility of the IS steering officer is to ensure IS sustainability at each step of its evolution which can be more or less complex. This challenging task needs a methodological and tool support providing the necessary and precise in-formation and the means to simulate IS change and to evaluate its impact before its actual realization. In our work we consider that such information is available in the enterprise IS, and we define a framework for IS evolution steering that allows to obtain this information from enterprise IS and to handle the IS evolution. Concerning the state of the art in the domain, there is no consensus on the definition, goals, models and methods of IS evolution steering. This domain is at the crossroads of several IS research areas such as: Enterprise Architecture (EA) and Enterprise Modeling, Business/IT alignment, IS Governance and Risk Management.

Today, the domain of Enterprise Architecture is rich in EA frameworks (e.g. TOGAF [2], GERAM [12], etc.) and dedicated modeling languages (e.g. DEMO [18], ArchiMate [1], MEMO [20]). Most of those frameworks acknowledge the need for multiple views (e.g. business, function, information, infrastructure) in or-der to manage enterprise complexity, to separate concerns and to address different life spans of EA elements [8]. These frameworks expose best practices and generic principles, and propose modeling notations, but fail to offer a formal steering method. Quite abundant literature is available in the domain of Business/IT alignment proposing various approaches to measure the fitness between enterprise business and its supporting IT, their respective strategies, infrastructures and processes. A systematic review of Business/IT alignment is presented in [43].

Finally, in the domains of IS Governance and Risk Management, risks are generally considered from the perspective of IS security (e.g. [36]) or from the perspective of software development and software project management (e.g. [11]). A literature review and comparison of risk management approaches is proposed in [7]. There is also a large amount of literature dealing with software change impact analysis (see a review in [26]) but rather from software maintenance point of view – the impact of IS change on the organization and its business is not considered. For several authors (e.g. [7, 11, 16]) risk is related to uncertainty. In our on-going work [30], we attempt to provide a holistic approach for IS evolution steering that would allow not only to deal with IS changes but also to measure the impact of the-se changes on the enterprise ant its business. The main objective of this framework is to reduce the uncertainty that IS evolution steering actors are facing at each step of IS change. We present this work in the next section.

15.4 A Framework for IS Evolution Steering

In IS and software engineering the evolution techniques are mostly based on models (e.g. [14, 3]). These works mainly address the problem of structural evolution (e.g. changing a class hierarchy, adding a new class). Their intention is to support the change propagation in order to allow the automation of data migration, to evaluate the impact of metamodel changes on models, to support forward-, reverse-, and re-engineering techniques or to record the model history. However, these models are not designed for IS evolution steering purposes and are not considered as means to support decision making in IS evolution, which is the purpose of our framework for IS evolution steering [30]. The construction of this frame-work is based on the following assumptions:

- the IS evolution steering requires understanding the underpinning IS domain, and vice versa, enterprise IS contain an accurate information about enterprise structure, activities, information and regulation,
- the impact of IS evolution is difficult to predict, so the simulation could help to take evolution-related decisions, and
- the guidance for IS evolution steering is almost non-existent, and therefore needs to be developed.

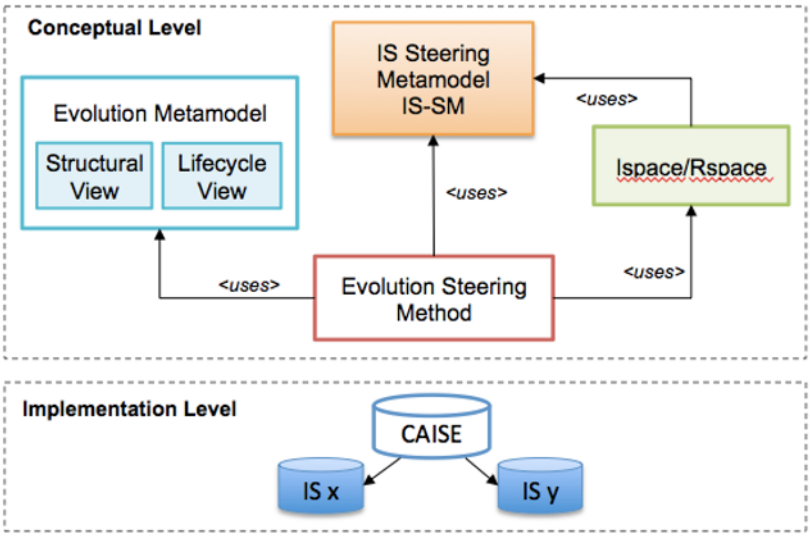

Fig. 15.5 An overview of the framework for IS evolution steering

Therefore, we propose a framework for IS evolution steering based on several models as shown in Fig. 15.5. The aim of this framework is to provide the foundation for the development of a Computer-Aided Information Steering Environment

(CAISE). This decision supporting tool uses existing enterprise information systems as source of information and guides the IS evolution steering actors in the IS change process by providing evolution simulation and impact identification facilities. At the conceptual level, the main element of the framework is the metamodel for IS evolution steering (IS-SM) which homogeneously integrates enterprise activity, regulation and information dimensions. It is complemented with the evolution models and the evolution steering method that provides guidelines for extracting the necessary information and simulating the evolution.

15.4.1 IS-SM

The *IS Steering Metamdel* (IS-SM) represents an information kernel, generic to any organization, and supporting the evolution steering of several IS in the organization. As show in Fig. 15.6, it consists of three models: activity, regulation and information. The *Activity Model* reflects enterprise business structure: *business units*, *positions*, *activities*, *rules*, *roles* and *responsibilities* that different persons hold when are assigned to a particular position. The *Regulatory Model* reflects how the enterprise complies with different laws, policies and other regulations by modeling their structure and relationship with different elements form activity and information models. Finally, the *Information Model* is composed of three sub-models: the *Generic IS Model*, *IS Model* and *Service Model* each of them representing the corresponding information level. The Generic IS Model represents an integrated view of the IS level which can consist of several IS. It allows inter-relating the Information model with the Activity and Regulatory models and defines the generic concepts such as *class*, *role*, *operation*, and *integrity rule*. The IS Model defines the information elements relating to the composition of the enterprise IS – which IS supports which activity in the organization. Finally, the Service Model defines how information services are implemented through the existing IS, knowing that a service can be based on one or several IS.

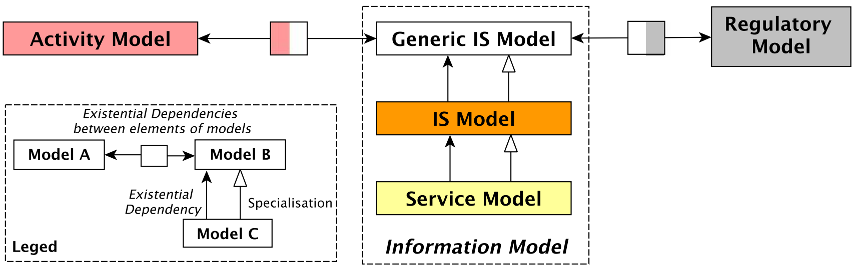

Fig. 15.6 A general structure of IS-SM

15.4.2 Evolution Metamodel

The *Evolution Metamodel* has two views, named structural and lifecycle, that are designed to support respectively the decomposition of a complex IS evolution and to guide the evolution process. In particular, the *Structural View* allows to capture the complexity of an evolution – to define the schema of an evolution that can be composed of several parts each of them being an evolution too. The *Lifecycle View* represents different possible states of an evolution (e.g.: ready, triggered, succeeded, failed) together with the conditions (transactions) allowing the evolution to pass from one state to another. In case of a failure, it allows to identify its cause.

15.4.3 Ispace/Rspace

Each IS evolution has an impact on the IS actors, more exactly on their information space (*Ispace*) and their regulatory space (*Rspace*), that can be extended or reduced. The Ispace and Rspace are based on the notion of responsibility that is a key concept for the impact analysis of an evolution. The responsibility is defined here as a set of information entities that represent the accountabilities and the capabilities of an actor (or group of actors) to perform a task. Therefore, the Ispace of an IS actor represents her responsibility over information elements, i.e. objects, operations, and integrity rules implemented in the IS that she can read/create/modify/delete. The Rspace represents her responsibility over regulatory elements, i.e. laws and regulation policies governing her activities in the organization supported by the IS. The Ispace and Rspace are defined as parts of IS-SM (see [34] for details). They allow to obtain sub-sets of information that inform the IS steering actor about the changes caused by an evolution affecting the responsibility of IS users. Together they allow to simulate the impact of IS evolutions on different IS actors responsibility and to identify potential risks.

15.4.4 Evolution Steering Method

Any change in the enterprise IS implies a shift from a known to an expected, but at the same time unknown, situation. Actors, responsible for IS evolution steering, are accountable for the decision making under a certain level of uncertainty, because the information, necessary to assess the evolution situation, can be incomplete or, in the contrary, overloaded. Consequently, IS steering actors need guidance for obtaining all relevant information, identifying risks, taking decisions about their handling and finally handling them.

Any IS evolution may fail due to its complexity. Guidance for IS evolution steering is essential for understanding and taking into account the various and interrelated components that constitute the complexity of the evolution situations. Therefore, the last, but not least, component of the framework is the Evolution Steering Method that aims to provide guidelines on how to use the aforementioned models in an efficient way and to support the mission of the actors in charge of IS evolution steering.

15.5 Conclusions

In this chapter we consider IS evolution as an ordinary situation in every modern organization. Furthermore, any change in the enterprise IS has impact not only on their own sustainability but also on the enterprise business activity and governance. In this context, we identify risks and challenges related to the enterprise IS evolution and we overview our and related work on this topic.

First, we discuss the adoption of service-oriented paradigm in IS engineering and demonstrate how IS-specific service-oriented architectures can be elaborated to deal with legacy IS evolution towards service-oriented IS that we call ISS. The notion of information service is defined as underpinning building block in the ISS construction. In particular, we briefly overview three ISS construction approaches, namely Services upon Legacy IS, Fully Service-Oriented ISS and Information Kernel-Based ISS. Each of them is dedicated to cope with a particular situation of ISS construction and can be combined with the other two approaches.

In the second part of the paper, we discuss the challenges related to the IS evolution steering and we introduce a framework dedicated to help the actors responsible for IS evolution steering to take critical decisions. The framework aims to ad-dress IS sustainability issues and to reduce the uncertainty by proving clear and complete information allowing to simulate IS changes and to assess their impact. The framework is composed of several models each of them representing a particular IS evolution perspective such as: the related information structure, the evolution lifecycle, the impact on the organization and its IS, and the responsibility. Besides, it provides guidance to use these models. In our future work we aim to ex-tend this framework with the technology dimension and with the support to the potential security risk analysis caused by an evolution.

Finally, it is important to note that all these works, like many others, relating to important subjects in the development of information systems and now services, are based on conceptual models. They were discovered, and established by the work of a few pioneers who established such solid and durable bedrock. Antoni Olivé is one of these pioneers as evidenced by his masterful article [29].

References

1. ArchiMate, `http://www.opengroup.org/subjectareas/enterprise/archimate`
2. TOGAF version 9.1, `http://pubs.opengroup.org/architecture/togaf9-doc/arch/`

3. Aboulsamh, M.A., Davies, J.: Towards a model-driven approach to information system evolution. In: ISD 2009. pp. 269–280. Springer (2009)
4. Aier, S.: Strategies for establishing service oriented design in organizations. In: ICIS 2012. Association for Information Systems (2012)
5. Aier, S., Bucher, T., Winter, R.: Critical success factors of service orientation in information systems engineering – derivation and empirical evaluation of a causal model. Business & Information Systems Engineering 3(2), 77–88 (2011)
6. Almonaies, A.A., Cordy, J.R., Dean, T.R.: Legacy system evolution towards service-oriented architecture. In: International Workshop SOAME 2010. pp. 53–62 (2010)
7. Alter, S., Sherer, S.A.: A general, but readily adaptable model of information system risk. CAIS 14, 1 (2004)
8. Armour, F.J., Kaisler, S.H., Liu, S.Y.: A big-picture look at enterprise architectures. IT Professional 1(1), 35–42 (1999)
9. Arni-Bloch, N., Ralyté, J.: MISS: A metamodel of information system service. In: Papadopoulos, G.A., Wojtkowski, W., Wojtkowski, G., Wrycza, S., Zupancic, J. (eds.) Information Systems Development: Towards a Service Provision Society. pp. 177–186. Springer US (2009)
10. Arni-Bloch, N., Ralyté, J., Léonard, M.: Service-driven information systems evolution: Handling integrity constraints consistency. In: PoEM 2009. LNBIP, vol. 39, pp. 191–206. Springer (2009)
11. Barki, H., Rivard, S., Talbot, J.: An integrative contingency model of software project risk management. J. of Management Information Systems 17(4), 37–70 (2001)
12. Bernus, P., Nemes, L.: A framework to define a generic enterprise reference architecture and methodology. Computer Integrated Manufacturing Systems 9(3), 179 – 191 (1996)
13. Bieberstein, N., Bose, S., Walker, L., Lynch, A.: Impact of service-oriented architecture on enterprise systems, organizational structures, and individuals. IBM Systems Journal 44(4), 691–708 (2005)
14. Burger, E., Gruschko, B.: A change metamodel for the evolution of mof-based metamodels. In: Modellierung. LNI, vol. 161, pp. 285–300. GI (2010)
15. Chua, F.: Adoption of service-oriented architecture by information systems. IJAACS 2(4), 317–330 (2009)
16. Copas, J.: Statistical modelling for risk assessment. Risk Management 1(1), 35–49 (1999)
17. Cuadrado, F., García, B., Dueñas, J.C., Parada, H.A.: A case study on software evolution towards service-oriented architecture. In: AINA Workshops at AINAW 2008. pp. 1399–1404. IEEE (2008)
18. Dietz, J.L.: Demo: Towards a discipline of organisation engineering. European Journal of Operational Research 128(2), 351 – 363 (2001)
19. Fazziki, A.E., Lakhrissi, H., Yétongnon, K., Sadgal, M.: A service oriented information system: A model driven approach. In: SITIS 2012. pp. 466–473. IEEE (2012)
20. Frank, U.: Multi-perspective enterprise modeling (memo): Conceptual framework and modeling languages. In: HICSS 2002. pp. 1258–1267 (2002)
21. Khadka, R., Reijnders, G., Saeidi, A., Jansen, S., Hage, J.: A method engineering based legacy to SOA migration method. In: ICSM 2011. pp. 163–172. IEEE (2011)
22. Khadraoui, A., Léonard, M., Thi, T.T.P., Helfert, M.: A framework for compliance of legacy information systems with legal aspect. AIS Transactions on Enterprise Systems 1, 15–26 (2009)
23. Khadraoui, A., Opprecht, W., Léonard, M., Aïdonidis, C.: Service specification upon multiple existing information systems. In: RCIS 2011. pp. 1–11. IEEE (2011)
24. Le Dinh, T., Nguyen-Ngoc, A.V.: A conceptual framework for designing service-oriented inter-organizational information systems. In: SoICT 2010. ACM International Conference Proceeding Series, vol. 449, pp. 147–154. ACM (2010)
25. Le Dinh, T., Pham Thi, T.T.: A conceptual framework for service modelling in a network of service systems. In: IESS 2010. LNBIP, vol. 53, pp. 192–206. Springer (2010)
26. Lehnert, S.: A review of software change impact analysis. Technische Universitat Ilmenau (2011)

27. Liu, Y., Wang, Q., Zhuang, M., Zhu, Y.: Reengineering legacy systems with restful web service. In: COMPSAC 2008. pp. 785–790. IEEE (2008)
28. Lo, A., Yu, E.S.K.: From business models to service-oriented design: A reference catalog approach. In: Conceptual Modeling - ER 2007. LNCS, vol. 4801, pp. 87–101 (2007)
29. Olivé, A.: Conceptual schema-centric development: A grand challenge for information systems research. In: CAiSE 2005. LNCS, vol. 3520, pp. 1–15. Springer (2005)
30. Opprecht, W., Ralyté, J., Léonard, M.: Towards a framework for enterprise information system evolution steering. In: PoEM 2014. LNBIP, vol. 197, pp. 118–132. Springer (2014)
31. Ralyté, J.: Applying transdisciplinarity principles in the information services co-creation process. In: RCIS 2012. pp. 1–11. IEEE (2012)
32. Ralyté, J., Arni-Bloch, N., Léonard, M.: Information systems evolution: A process model for integrating new services. In: AMCIS 2010. p. 431. Association for Information Systems (2010)
33. Ralyté, J., Khadraoui, A., Léonard, M.: Designing the shift from information systems to information services systems. Business & Information Systems Engineering 57(1), 37–49 (2015)
34. Ralyté, J., Opprecht, W., Léonard, M.: Defining the responsibility space for the information systems evolution steering. In: PoEM 2016. LNBIP, vol. 267, pp. 179–193. Springer (2016)
35. Ransom, J., Sommerville, I., Warren, I.: A method for assessing legacy systems for evolution. In: CSMR 1998. pp. 128–134. IEEE (1998)
36. Reddy, V.K., Dubey, A., Lakshmanan, S., Sukumaran, S., Sisodia, R.: Evaluating legacy assets in the context of migration to soa. Software Quality Journal 17(1), 51–63 (2009)
37. Sneed, H.M.: Integrating legacy software into a service oriented architecture. In: CSMR 2006. pp. 3–14. IEEE (2006)
38. Spohrer, J., Maglio, P.P., Bailey, J., Gruhl, D.: Steps toward a science of service systems. Computer 40(1), 71–77 (2007)
39. Spohrer, J., Vargo, S.L., Caswell, N., Maglio, P.P.: The service system is the basic abstraction of service science. In: HICSS 2008. p. 104. IEEE Computer Society (2008)
40. Thomas, O., vom Brocke, J.: A value-driven approach to the design of service-oriented information systems – making use of conceptual models. Information Systems and e-Business Management 8(1), 67–97 (2010)
41. Turki, S., Léonard, M.: Hyperclasses: Towards a new kind of independence of the methods from the schema. In: ICEIS 2002. pp. 788–794 (2002)
42. Turki, S., Léonard, M.: IS components with hyperclasses. In: OOIS Workshops 2002. LNCS, vol. 2426, pp. 132–141. Springer (2002)
43. Ullah, A., Lai, R.: A systematic review of business and information technology alignment. ACM Trans. Manage. Inf. Syst. 4(1), 1–30 (2013)
44. Umar, A., Zordan, A.: Reengineering for service oriented architectures: A strategic decision model for integration versus migration. Journal of Systems and Software 82(3), 448–462 (2009)