TOWARDS A METHOD FOR ENTERPRISE INFORMATION SYSTEMS INTEGRATION
(Extended version)

Silveira, R. W.; Pastor, J.A.; Mayol, E.
Facultat d'Informàtica de Barcelona, Universitat Politècnica de Catalunya
{silveira; pastor; mayol}@lsi.upc.edu

Keywords: Enterprise Integration, Enterprise Information Systems Integration, Enterprise Application Integration (EAI), Business Process Management (BPM), Information Systems, IC-BPM suites.

Abstract: Enterprise information systems integration is essential for organizations to fulfil interoperability requirements between applications and business processes. To carry out most typical integration requirements, traditional software development methodologies are not suitable. Neither are enterprise package implementation methodologies. Thus, specific ad-hoc methodologies are needed for information systems integration. This paper proposes the basis for a new method for enterprise information systems integration in order to facilitate continuous learning and centralized management during all the integration process. This method has been developed based on the ontology defined in ISO/IEC 24744.

1 INTRODUCTION

Nowadays, it is common that enterprises develop specific projects for the integration of both disparate information systems (IS) within one enterprise and between IS from several enterprises. For such a task, it is a good practice that facilitates project success to apply an adequate method. Despite the fact that traditional software engineering methods alone are not adequate for enterprise information system integration (Themistocleous & Irani, 2006), previously published methods, specific for enterprise IS integration, have not yet had a great level of popularity. Therefore, two research lines could be proposed: first, to analyse why these methods are not so popular and second, to define new enterprise IS integration methods more focused on usefulness.

In this paper we propose the bases for a new method for enterprise IS integration, which we are designing with an aim towards continuous learning and with a centralized management as a differential feature. Moreover, we propose explicit evaluation phases to improve the quality of the integration task to be undertaken.

The paper is structured as follows. We begin with an introduction to enterprise IS integration projects. Section 3 describes the method proposed. In section 4 we present a real enterprise IS integration project from which some of our ideas are drawn. In section 5 we establish relationships with previous research on integration methods, and in section 6 we present our conclusions and ideas for further work.

2 ENTERPRISE IS INTEGRATION

Enterprise information systems integration may refer in some contexts to different concepts as Information Technology Integration, Information System Integration, Application Integration, Business Integration and Data Integration, etc. However, the main task of all of them relies to integrate some element of enterprise.

Over the past times, enterprises have the need to integrate their data, processes, applications or systems. At the end of the 90's, the high development costs, the trust in the reliable operation of robust legacy systems, and the need for the quick integration of new kinds of information systems such as eBusiness applications, all together motivated the rise of Enterprise Application Integration (EAI). EAI

was defined by Linthicum (Linthicum, 1999) as the unrestricted sharing of data between two or more organization applications, where a group of technologies allow information flow and exchange among different applications and business processes, of the same enterprise or between different enterprises.

To satisfy that need, different software vendors came out with several proprietary tools to simplify different integration tasks. Integration projects lead to be the development of a great number of interfaces for applications. Thus, the main benefit of using EAI tools is to reduce the required effort to build and to maintain all those interfaces. Also, these interfaces become useful for obtaining more integrated architectures to substitute point-to-point communications between applications, which was the typical solution prior to EAI and to enterprise IS packages such as ERP, CRM or SCM.

Much more recently, the adoption of web-services technology and Service-Oriented Architectures (SOA), have changed the applications development and integration paradigm. This new approach supports high levels of system artefact reuse and can frequently result in dramatically reduced coding for new application functionality. At the same time, this approach brings similarities between integration and component-based new application development. This confluence has promoted a new service market, as the convergence of the previously independent tool markets for EAI, Workflows and Document Management, and Business Process Management tools (Meta Group, 2003). By the end of 2006, Forrester defines this new service market as that one of the Integration Centric Business Process Management (IC-BPM) suites (Vollmer & Peyret, 2006). It consists on a group of distributed applications oriented to business process management but centred on integration. IC-BPM suites have evolved primarily out of the EAI space and continue to represent the leading edge in integration capabilities.

Over the past six years, the main IC-BPM suites vendors have transformed a market dominated by proprietary and closed-framework solutions to one that is now almost exclusively based on SOA and more committed with every passing day to standards-based integration technology (Vollmer & Peyret, 2006).

Next, we describe the most common reasons that justify enterprise IS integration projects, and later we emphasize the need for a methodological approach for such projects.

2.1 Rationale of IS Integration Projects

Enterprise IS integration projects usually involve, among other tasks, the acquisition of a suit of tools to simplify the integration (Silveira & Pastor, 2006), the replacement of inefficient legacy applications and the development of services to connect intra- or inter-enterprise processes through the integration of their information systems.

Today, rather than stable application portfolios, often uncertainty and continuous evolution are the natural state of enterprise information systems. For these reasons, IS integration projects do not begin and finish like traditional projects, but rather they are more open and continuous projects. Therefore, it is necessary to foresee potential business changes and to prevent and adapt IS integration while those changes arise.

Following this trend, enterprises become interested in implementing a platform to integrate current and future information systems when a business opportunity arises that implies interconnecting its information systems with those of its clients and/or suppliers. Another reason that often justifies enterprise IS integration projects is the establishment of joint ventures or enterprise mergers and takeovers, situations that require relating in a fast and simple way their different information systems.

In fact, due to fast evolution and constant changes in business domains and industries, such as those implied by globalization and competition, information systems do have to adapt constantly to those circumstances, in parallel with the enterprises hosting them.

2.2 Need for New Methods for Enterprise IS Integration

Given the dynamic nature of the scenario presented above, having an integration platform is not enough. Enterprises would benefit from using methodologies first for the effective selection of appropriate integration tools, and later for the implementation of new functionalities into current information systems or for the ongoing evolution of the already integrated information systems.

To carry out an enterprise IS integration project, we could try to adapt and use some prior development approach, such as a conventional software development methodology or some proprietary enterprise package implantation methodology. However, none of these alternatives

fits in a natural and easy way with the situations arising in enterprise IS integration projects.

Typical software development methodologies are designed for the bespoke construction of software solutions from anew, while in enterprise IS integration projects we find a varied set of legacy applications that are to be integrated along some pre-existing or newly designed business process. Software development methodologies are mainly built over specific functional requirements, while the integration of information systems must be built also over the technical requirements of underlying technologies. This calls for the procurement of specific tools that simplify the interconnection tasks, and for the negotiation with IS technicians of other enterprises.

Enterprise IS integration is far different from implementing an enterprise package. Legacy applications are heterogeneous in several ways, while enterprise packages are much more homogenous and their implementation methodologies and tools are adapted to such a state.

Thus, it seems appropriate to think of new ad-hoc methodologies for enterprise IS integration. Although these could reuse some features from other types of methodologies, they must be designed for the particularities of enterprise IS integration. Along this line, we agree with Themeistocleous et al. (Themistocleous & Irani, 2006) when they state the importance and need for more research on enterprise IS integration methodologies.

As far as we know, there is no standard or widely distributed method designed specifically to manage enterprise IS integration projects, provided by integration tools vendors or integration service providers such as consulting firms. So far only two academic proposals have been published, respectively by Themeistocleous et al. (Themistocleous & Irani, 2006) and Lamb et al. (Lam & Shankararaman, 2004).

3 OUR IS INTEGRATION METHOD

To define a new method it is convenient to refer to previous method engineering results. Fortunately, Method Engineering is not a new research area and since the 90s many articles and proposals have been published (Brinkkemper et al., 1999) (Weerd et al., 2007). Our goal is to organize artefacts and activities that can be found in a typical integration project by using an ontology that will allow us to formalize our method in a more straightforward way. To accomplish this goal, we have based our initial proposal on the ontology defined by the ISO/IEC 24744 (González et al., 2007).

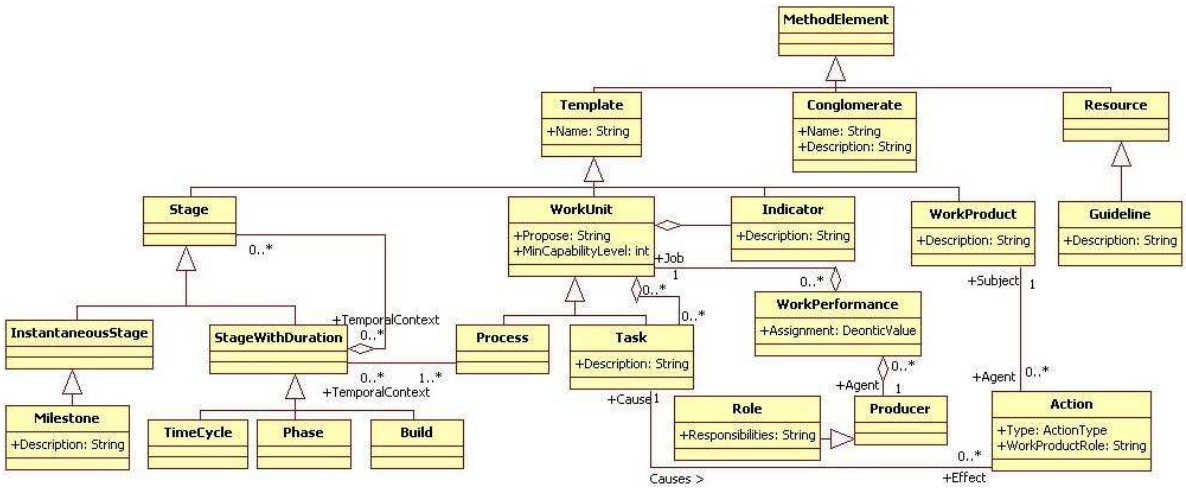

Figure 1: Class diagrams of methodological elements

The ISO/IEC 24744 standard introduces the Software Engineering Metamodel for Development Methodologies, the aim of which is to define methodologies in information-based domains (ie. areas characterized by their intensive reliance on information management and processing), such as software, business or systems engineering. ISO/IEC 24744 is an instrument that is suited to our needs and that also provides a quality and diffusion framework.

Additionally, our method is based on an iterative approach and bears continuous evaluation tasks in the end of each integration stage. It promotes the acquisition of integration knowledge through several evaluation stages. These stages allow us to learn from past successful actions as well as mistakes, and to identify causes of deviations with respect to initially-planned duration, budget or integrated functionality. At the beginning of each new cycle, if necessary, objectives may be adjusted based on results on previous cycles. Moreover, we take also into account several lessons learned from our participation in a real enterprise IS integration project within a big Spanish insurance enterprise.

Figure 1 shows the domain of usual method elements of enterprise IS integration and their relationship. We consider the following four main elements:

- Stages. Different development activities are scheduled in a generic level, through an integration life cycle, where we distinguish its stages and phases.
- Work units. Activities which should be done during the development.
- Producers. Enterprise roles with the responsibility to do those work units.
- Work Products. Set of artefacts considered during the work units execution.

Given the ontology described above, we define a life cycle for enterprise IS integration with three basic stages: Procurement, Implementation and Use. For each stage we define four phases (see Figure 2). As central elements in this cycle there are two kinds of artefacts, those that make up the integration (applications, processes, systems, tools) and those that supports project management activities (management tools such as balanced scorecard). In each stage, we distinguish two specific phases that interact with the management artefacts, a first phase that includes planning tasks, and a second one that includes evaluation tasks. Management tools propose initiatives to be considered during planning phases, while evaluation phases generate feedback that is used as data input for these management tools. In this way, our method promotes continuous learning.

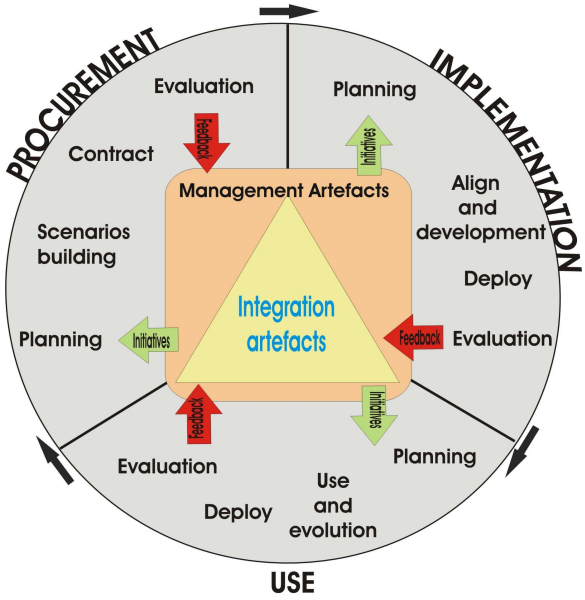

Figure 2: Method for managing enterprise IS integration

Next, we describe the mains elements of our method: life cycle, integration artefacts, and management artefacts, among other important elements.

3.1 Life Cycle

We propose a life cycle for an integrated enterprise information system defined in terms of three main stages, each one decomposed in four phases (Figure 2). Our life cycle stages are:

- Procurement: Composed by the phases named Planning, Scenarios building, Contract and Evaluation.
- Implementation: Composed by the phases named Planning, Align and Development, Deploy and Evaluation.
- Use: Composed by the phases named Planning, Use and Evolution, Deploy and Evaluation.

Below, we explain stages and phases in detail

3.1.1 Procurement

In this stage, enterprise needs for the development of the future integrated IS are studied and a conceptual integration solution is designed. This solution involves the conceptual description of applications to be integrated and the interfaces between them. Phases of this stage are the following:

- **Procurement planning**. At this phase, business processes and requirements are analyzed; integration requirements are compiled; risks, benefits and organizational impacts are evaluated; integration strategies are established;

available tools and technological alternatives are analyzed; and possible integration developments (interconnection or glue code) are identified. Moreover, perspective and business initiatives are considered; knowledge from integration experience is used.

- **Scenarios building and business process reengineering**. Possible scenarios are built by considering tools, technologies and architectures. Design and development efforts are forecasted for each scenario. Besides, it is an opportunity to do appropriate business process reengineering. The scenario most convenient is selected according to enterprise goals. The evaluation of different IC-BPM suites may be done through the quality model for EAI tools proposed by Silveira et al. (Silveira & Pastor, 2006). Plus, other non-technical system factors, such as organizational issues (barriers, benefits, costs, etc.), should be considered (Themistocleous & Irani, 2001).
- **Contracting**. Finally, the chosen IC-BPM suite is acquired, the implementation and maintenance team is contracted, and the organizational benefits and impacts are estimated.
- **Procurement evaluation**. The quality of the procurement process is evaluated. The results are analyzed and stored in a base knowledge.

3.1.2 Implementation

After designing a conceptual solution and having the tools and the technical team to carry on with it, the implementation stage may start with the following phases:

- **Implementation planning**. According to the business initiatives and strategy of the hosting enterprise or enterprises, the implementation of the solution is planned, the implementation teams are coordinated and the deployment strategy is defined. Moreover, knowledge from integration experience is used.
- **Alignment and development**. In this phase, the architecture is deployed, the integration tools are customized and the interconnecting code is designed and developed. Unitary and integrated tests are designed, populated and applied.
- **Deployment**. Finally, data migrations are executed, the new applications and interfaces are distributed, the business processes are deployed, and the system operators are trained.
- **Implementation evaluation**. The quality of the integration process is evaluated. The results are analyzed and stored in a knowledge base.

3.1.3 Use

After the resulting integrated information system is deployed, it begins to be used and eventually will need to be maintained.

- **Usage planning**. System usage and its evolution (maintenance) are planned taking into account enterprise strategy. Moreover, perspective and business initiatives are considered; knowledge from integration experience is used
- **Use and evolution**. The system is used and its evolution maintenance strategy is followed.
- **Deploy patches and new versions**. New patches and versions are proposed to redress system behaviour when appropriate.
- **Usage evaluation**. At this phase, the alignment between the system integration and the enterprise strategy is evaluated. The results are analyzed and stored in a knowledge base. Moreover, it is also considered how the system use fulfils the enterprise goals.

To describe each phase, we mention some of the activities performed into its temporal context; this can be explicit through a relationship between phases and processes. Figure 3 shows an example of instantiation of this relationship, where different process such as scenario building and scenario effort forecast are run within a temporal context, that we name Scenarios building and business process reengineering phase.

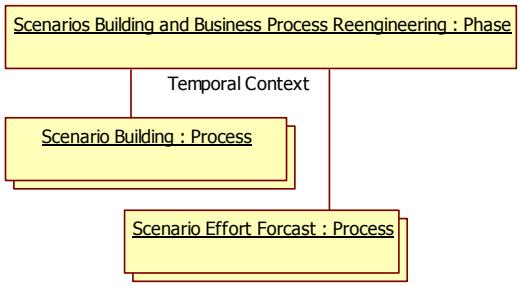

Figure 3: Example of relationship between a phase and process

3.2 Integration Artefacts

According to our ontology, we can say that integration artefacts are Work Products. A work product is an artefact of interest to the development effort. Many work products are created during the development effort, but many others are procured outside and are modified during the development. Clearly, the IC-BPM suites are examples of artefacts that are modified or customized during the development while “glue code” is an example of a product that is created during the development. All

of these are artefacts that compose the centre of integration and we call them Integration artefacts.

Our method revolves around a group of integration artefacts that compose the integrated IS. These artefacts are the business processes, applications and enterprise systems to be integrated, the tools that simplify integration tasks and the overall architecture where integration is going to be founded (see Figure 4).

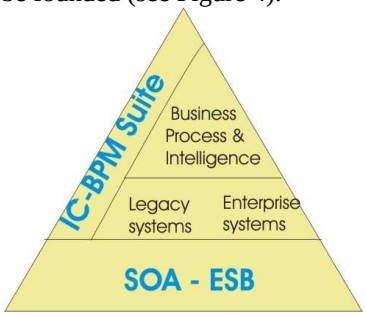

Figure 4: Integration artefacts

As explained in section 2, currently many integrated systems are built upon service-oriented architectures, the most popular of these being the Enterprise Services Bus (ESB). Tools that simplify our enterprise IS integration are those classified under IC-BPM suites. Information systems to be integrated may be legacy applications and enterprise package systems; business processes may be of an intra- or inter-enterprise nature; and we may also have business intelligence applications.

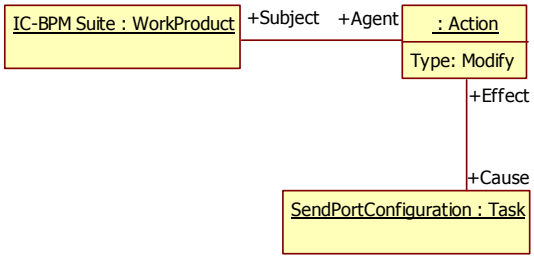

Figure 5: Example of relationship between Integration Work Products and Tasks

3.3 Management Artefacts

Apart from integration artefacts, we distinguish another type of work product, which are generated in the development of the project in order to be used as tools in the integration project management. We call them Management artefacts. Over the work products, events are executed through the concrete action of specifics tasks.

In our method, we include the development of a model for the management of the project. For example, management tools popular as balanced scorecard may be essential to align business strategies with information integration strategies (Figure 6). Moreover, these may be the key in continuous learning through cycles of evaluation and analysis of the results that are used as input for future iterations.

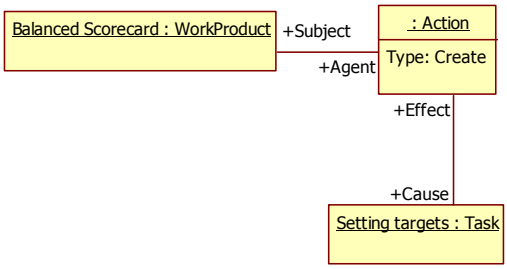

Figure 6: Example of relationship between Management Work Products and Tasks.

3.4 Others Key Methodological elements

To provide a complete method it is necessary to describe some additional methodological elements that so far we have not described in our domain such as: Work Units and Roles. In this paper, we have not deepened its description, but we present a brief overview.

3.4.1 Work Units

Work units describe the main activities that must be done in each phase; these activities can be processes or specific tasks. Typically, a process is described associating it to a set of tasks. Figure 7 shows this association with an example between an IC-BPM Suite Evaluation work unit and the Evaluates characteristics tasks.

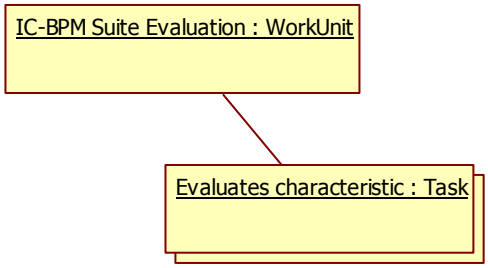

Figure 7: Example of relationship between a work unit and task

3.4.2 Roles

We use "roles" to describe agent responsibilities; in our case, stakeholders are agents, and three groups of responsibilities are identified:

- Responsible people for Business Activities performed by CTO, CEO, business sponsors, business partners, among others.
- Responsible people for Technical Activities performed by project managers, IT architects, end-users, among others.
- Responsible people for Organizational Activities performed by operations staff, maintenance staff, end-users, among others.

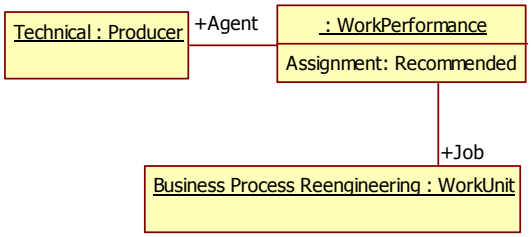

Figure 8: Example of relationship between a producer and a work unit

We call work performance to the assignment of a stakeholder to a work unit, and the type of assignment is specified there (Mandatory, Recommended, Optional, Discouraged, Forbidden). For example, Figure 8 shows the recommendation of a Technical worker assigned to Business Process Reengineering work unit.

4 EXPERIENCE FROM A REAL EAI PROJECT

Our inspiring experience study is an integration project of a Spanish corporation which, being one of the leading companies in the Spanish insurance industry, during the last years has been absorbing other smaller regional insurance companies.

Obviously, these acquisitions have created the necessity to integrate some of the information systems of the regional companies with those of the corporation. Initially, without an adequate integration platform and without a methodological approach that would help with this process, the change pace in IS integration did not meet the speed imposed by the business.

This demonstrated the need to address the integration tasks as an enterprise IS integration project, which would efficiently integrate the enterprises that form the group through the integration of some of their information systems, and that in the future would ease the IS integration coming from new acquisitions.

Previous methodological proposals for the management of IS integration (or EAI) projects do not consider explicitly some tasks that we now consider as key for the overall success of an integration project. For example, nothing is mentioned about the contracting phase or about tasks dealing with the evaluation of the evolution of the integration efforts throughout the project.

In our experience study, one of the serious mistakes incurred at the beginning, and that delayed the project in eight months, was the decision of contracting the integration project to a consulting company recommended by an important vendor of IC-BPM suites. The consulting enterprise was contracted on the basis of this recommendation without really evaluating its experience specifically in integration projects.

Another source of problems in the considered experience study was that the enterprise adopted, without much analysis, a strategy to add to the integration platform the various functional areas one to one, in a sequential way. Therefore, once the project was launched, this strategy forced to manage situations where, in a specific time, some business areas were in implementation stage while others were in evolution stage while the implicit methodological approach was linear. Consequently, we think that enterprise IS integration projects are, in general, cyclical or iterative in nature rather than linear. This does not preclude the project from being managed in a rather centralized and homogeneous way.

With this iterative approach in mind, including evaluation tasks will allow and help project managers to monitor, control and learn from the gained experience and the successes and mistakes happened in a prior cycle. For example, in our experience study, the area of car insurances was the first to be integrated, but it was not until the evaluation of the usage of the integrated IS that a high number of data mistakes were detected from one of the sources. While that lack of information quality could have been addressed before, at least its resolution has helped in the more recent integration efforts of business areas and processes dealing with home insurances working with the same source, since contingency processes have been designed to prevent and correct similar problems.

5 RELATIONSHIPS WITH PRIOR EAI RESEARCH

In contrast with the relevance of the topic in industry, so far there is not much research on EAI project management or in IS integration methods. As far as we know, only two contributions have appeared out of academic research: Lam and Shankararaman (2004) and Themistocleous and Irani (2006). Next we present them and compare our proposal with them. Table 1 relates the stages of the methods presented in the above references with the stages and phases of our proposal, and tries to make a mapping between the stages of the three proposals.

Lam and Shankararaman (2004) were the first who proposed a methodology for enterprise IS integration. Their proposal called it Enterprise Integration Methodology (EIM) and consists of five stages (Lam & Shankararaman, 2004).

The steps defined in the five stages proposed by Lam et al. are also considered in our method. To understand the end-to-end business process is the first step of the analysis, which we considered in our procurement planning phase. Their tasks named Map the process onto components and derive the requirements stages, are considered in our Scenarios Building phase, because of the each scenario building we must map the process onto components and we deal with the integration requirements analyzed in previous phases. Their task Produce the architecture is one of the steps that we include in our Align and development phase. Finally, their stage named Plan the integration corresponds to our beginning phase of the Implementation stage.

The methodology proposed by Lam et al. defines general lines but does not detail or describe specific points in integration projects. They do not deal in their procurement stage with issues such as scenario building and evaluation. Regarding implementation stage, unless they do not identify in early instances the necessities of new developments, they do not consider their implementation either. Finally, no usage stage is considered at all, thus leaving out any integration evolution evaluation phases, nor the evaluation of the usage of integrated IS.

EIM has some limitations, such as the lack of consideration of systems restructuring or the necessity to develop new software. Trying to cover these limitations, they propose another methodology of eight stages (Themistocleous & Irani, 2006).

The sequence of steps defined by Themeistocleous et al. is distributed over our structure of stages and phases. Planning stage covers activities that aim towards the study of the factors that affect the process of adopting an EAI approach, such as barriers, costs or benefits. We make these studies when evaluating the different scenarios, because costs, barriers, benefits or organizational issues in general affect in different ways each scenario. Their Scenarios building and evaluation stage, Business Process reengineering stage and Systems restructuring stage corresponds with our Scenarios building and business process reengineering phase While we build our scenarios, we are taking advantage of the opportunity to propose initiatives in business process reengineering and Systems restructuring. We locate their Requirements analysis stage in a different position, because we believe that integration requirements must be clear before the building of scenarios. Their Filling the gap and New systems development stages are included in our Align and development phase within the Implementation stage. Their steps of Integration and testing are distributed in our Align and development and Testing and evaluation stages. Finally, their Operation and maintenance stage corresponds with our Use and evolution phase.

On the other hand, Themeistocleous et al. do not consider hiring subjects, which we inserted in a phase between the selection of the suitable scenario and the beginning of the planning implementation phase. The results obtained in the Hiring stage, often affect technically the development of a project, so we considered them sufficiently important to dedicate a phase where we can align the technological strategy with the business strategy. They do not propose any management tool to help manage business initiatives or technical adjustments derived from the continuous evaluation.

Table 1: Ours vs. prior methodologies

Stages	Phases	Work units	Lam and Shankararaman (2004)	Themeistocleous and Irani (2006)
PROCUREMENT	Planning	Integration requirement analysis. Impact analysis. Risk analysis. Procurement planning.	A. Understand the end-to-end business process. B. Map the process onto components. C. Derive the requirements.	I. Planning II. Scenarios building and evaluation. III. Business process reengineering. IV. Systems Restructuring. V. Requirements analysis.
	Scenarios building and Business process reengineering	To find IC-BPM Suite and architecture. Effort forecast in design and implementation for each scenario. IC-BPM Suite evaluation. Business process reengineering.		
	Contract	Buy of Suite. Contract the development teams for integration and new developments. Benefits forecast.		
	Evaluation	Procurement evaluation		
IMPLANTATION	Planning	Implementation planning. Coordinate developments. Define implantation strategies.	D. Produce the architecture E. Plan the integration.	VI. Filling the gap – New systems development VII. Integration and testing
	Align and development	IC-BPM Suite configuration Interface development. Unitary testing. Integration testing.		
	Deploy	Data migration. Deploy New applications and interfaces. Deploy integration process. Operators training.		
	Evaluation	Implantation evaluation.		
USE	Planning	Maintenance planning. Evolution planning.		VIII. Operation and maintenance
	Use and evolution	Use. Evolution development.		
	Deploy	Deploy patches. Deploy new versions.		
	Evaluation	Use evaluation. Evolution evaluation.		

6 CONCLUSIONS AND FURTHER WORK

It is not yet usual that important failures in IS integration projects are published, but through interviews with experts in integration, we know several fiasco cases in Spain and other countries. In these cases, often after great investments in integration platforms, at the end they have ended up in great overruns and low satisfaction, with interconnections have ended up being implemented point to point to hide the project failure.

From these observations, we assume that it is not enough to have implemented an integration platform to take advantage of the benefits promoted from IS integration, or from EAI projects.

In this paper we have presented our bases for a new method for enterprise IS integration, constructed on prior research on EAI topics, and from the analysis of mistakes and successes in a real rich experience study, from interviews with experts in integration projects, other case studies published, and the analysis of the methodologies previously proposed.

The ideas and artefacts incorporated in our proposal must be formalized within the tenets of prior research from other related areas, such as Method Engineering from software and IS engineering. Similarly, prior results on best practices recognized in the management of other similar projects within information systems, such as ERP, CRM or SCM implementation projects, should be taken into account in the refinement of our method for IS integration projects.

REFERENCES

Brinkkemper, S., Saeki, M., Harmsen, F., 1999. Meta-Modelling based assembly techniques for situational method engineering. *Information Systems*, Vol. 24, No. 3, p. 209-228.

González, C., Larrucea, X., Bediaga, A., Gortazar, A., 2007. Metamodelo para Metodologías de Desarrollo. Retrieved September, 2007, from http://meta.dsic.upv.es

Lam, W. & Shankararaman, V., 2004. An Enterprise Application Methodology, *IT Profesionals*, 6(1): pp. 40-48.

Linthicum, D., 1999. *Enterprise Application Integration*, Addison-Wesley, Massachusetts, USA.

Meta Group, 2003. Enterprise Application Integration METAspectrumSM Evaluation.

Silveira, R. & Pastor, J., 2006. A model for enterprise application integration tools evaluation. *Proceedings of the European and Mediterranean Conference on Information Systems (EMCIS)*.

Themistocleous, M. & Irani, Z., 2001. Benchmarking the Benefits and Barriers of Application Integration, *Benchmarking: An International Journal*. Vol. 8. No. 4, 317-331.

Themistocleous, M. & Irani, Z., 2006. Towards a Methodology for the Development of Integrated IT Infrastructures. *Proceedings of the 39th Hawaii International Conference on System Sciences*.

Vollmer, K. & Peyret, H., 2006. The Forrester Wave: Integration-Centric Business Process Management Suites, Q4 2006".

Weerd, I., Brinkkemper, S., Versendaal, J., 2007. Concepts for incremental method evolution: Empirical exploration and validation in requirements management. *Proceedings of CAiSE*, pp. 469-484.