Developing IT Systems to Leverage Information and Knowledge Management

José Viterbo, Maria Luiza d'Almeida Sanchez and Carlos Alberto Malcher

Laboratory of Management in Communication and Information Technology,
Fluminense Federal University, Niterói, Brazil
viterbo@ic.uff.br, mluiza@gteccom.uff.br
malcher@gteccom.uff.br

Abstract. *Information and knowledge management are fundamental components for helping organizations in efficiently acquiring, storing and utilizing knowledge for problem solving, dynamic learning, strategic planning and decision making. In a previous work we presented the Information and Knowledge Management Framework (MGIC), a methodology for building a model to serve as basis for the implementation of integrated information and knowledge systems that support business intelligence in public organizations. The first step of this methodology consists on building an integrated model, based in well-known modeling techniques, such as Business Use Case models and class models, which focus on the representation of the main business requirements and services provided by the organization. In this paper we discuss how such model is used to guide the requirements specification and development of new IT systems or the evolution of legacy systems.*

Keywords: Business Intelligence, Knowledge Management, Software Development

1 Introduction

The term Business Intelligence (BI), coined in the early 1990s, is widely used today to describe analytic applications, i.e., systems that combine data gathering, data storage, and knowledge management with analytical tools to present complex internal and competitive information to planners and decision makers. BI allows managers to make informed and intelligent decisions regarding the functioning of their organizations. Informed decisions lead to better, more efficient processes in the actual work environment, and help create a powerful competitive advantage.

As organizations become more automated and data-driven, the industry is evolving toward BI systems that support operational decision making at all levels in the organization. There is a growing realization that BI must be integrated into the business operations of the enterprise to enable the many knowledge workers engaged in business processes to make better and quicker decisions. This scenario imposes several new requirements on the architecture of software systems and on the back end data integration processes, such as handling a much larger number and diversity of data sources and data types, more complex analytic and reporting tools, a larger number of data mart connections, 24x7 availability, etc.

Besides data architecture, the knowlegde management is also an integral component of BI and hence decision making in the organization. Knowledge management and BI need to be considered together as necessarily integrated and mutually critical components in the management of intellectual capital. Knowledge management, which may be described as a systematic process of finding, selecting, organizing, distilling and presenting information in a way that improves the comprehension of the organization's employees in specific areas of interest, helps an organization to gain insight and understanding from its own experience. Specific knowledge management activities help focus the organization on acquiring, storing and utilizing knowledge for such things as problem solving, dynamic learning, strategic planning and decision making.

As such, information and knowledge management are fundamental components for helping organizations in efficiently acquiring, storing and utilizing knowledge for problem solving, dynamic learning, strategic planning and decision making. In a previous work we presented the Information and Knowledge Management Framework (MGIC), a methodology for building a set of models to serve as basis for the implementation of integrated information and knowledge systems that support business intelligence in public organizations [1].

The MGIC aims at identifying the whole information flow in a organization, since it is collected from internal and external sources, stored, processed and output. The first step of this methodology consists on building an integrated model, based in well-known modeling techniques, such as business use cases and class models, which focus on representing the main business requirements and services provided by the organization. In this paper we discuss how such model is used to guide the requirements specification and development of new IT systems or the evolution of legacy systems.

2 Objectives of the research

The first step of the MGIC methodology consists on building an integrated model that comprises four modules: an information flow model, an ontology model, a knowledge model and a requirements model. Each model is built based in well-known modeling techniques, such as BPMN, OntoUML [2], business use cases, etc. When the final model is ready, the MGIC comprises a set of guidelines that governs the creation, the validation, the dissemination, the storage, the use and the recovery of information and knowledge to allow the organization to fully reach its goals. In our present research we are interested in defining how this set of models may be used to guide the software development process in the organization. In particular, software development is mainly influenced by the requirements model, which focus on the representation of the main business requirements and services provided by the organization.

The RUP (Rational Unified Process) methodology, which suits the development of large projects, is partly employed to build the requirements model, in an iterative approach. Each iteration includes one or more disciplines of development, has a set of well-defined objectives and produces a partial model of the final system, using as input the partial model of the previous iteration to evolve and refine the system to get to the final product, complying with all rules and requirements. To deal with the iterative development, RUP provides a structured approach, dividing the project into four distinct phases: conception, design, construction and transition. Each of aggregate activities required to achieve a certain goal.

In the MGIC methodology, efforts are limited to the design phase. The main RUP disciplines approached in this phase are the identification of business requirements and the development of the information model. During the identification of requirements, the techniques used consist of interviews, questionnaires, group dynamics, study of documentation (legislation, existing systems, standards and business rules, etc ...) and on-site observation. The main model developed during this discipline is the set of functional requirements expressed in the form of user needs through the use case business models. Moreover, during the course of the following discipline, the information model is built. It consists of a class diagram, identifying relevant information and their associations to comply with the provisions of the use case model, thus proposing an information architecture. This model is supported by textual descriptions that facilitate the understanding of systems developers. It is important to stress that this model is a conceptual data model, i.e., it does not incorporate characteristics of a particular implementation. All models built are validated by the users who provided the information for their preparation.

In the next section, we discuss how such model must be used to guide the requirements specification and development of new IT systems or the evolution of legacy systems.

3 Scientific contributions

As a further step to MGIC, we propose a software development process (SDP) as a methodology for developing software coherently with the model previously described. This methodolgy is also based on RUP and aims at standardizing the life cycle of a software development project. One of the benefits of adopting a well-defined process is to increase the level of productivity of the technical teams involved in projects, because it formalizes the distribution of activities and assignments for each role. Besides being a mechanism for obtaining a high quality product, the SDP also aims at supporting the definition of agreements on future contracts for service software development. Our proposed SDP comprises six stages: (a) preliminary study, (b) initiation, (c) elaboration, (d) construction, (e) transition and (f) acceptance. It involves the Information Management team (IM), Knowledge Management team (KM), the Information Technology team (IT), the client (any unit or member in the organization) and the software development company contracted (contractor).

3.1 Preliminary study

In the preliminary study, the feasibility of implementing a functional demand is analyzed in the following steps:

1. IM receives from the Client a request for implementing a particular functionality.
2. Based on the MGIC models, IM analyzes the request in order to verify:
   - If this functionality or other one similar is already implemented in any subsystem or if it fits in an existing subsystem. In this case, such subsystem will undergo an evolutive maintenance.
   - If this functionality must be implemented in a new subsystem. In this case, the development of such subsystem will follow this process.
3. If necessary, IM and KM update the MGIC models. Any new business use case model must be ready before the start of the system analysis (next stage). IM prepares (or updates) a requirements model, defining business use cases, information model and business rules.
4. IM prepares the preliminary study, a document that defines to which services specified in MGIC these features are associated, if new services/business requirements will have to be updated in the MGIC model and gguidelines for the system development (system that will be changed, requirements priorities, requirements to be met).
5. IT prepares the preliminary project containing a brief description of the project, a technical feasibility study considering the costs-benefits relation associated, project estimation, macro schedule and technology options/software architecture.
6. IT prepares the Project Opening Document, validated by IM.

In this stage, steps 2 to 4 are additions to the traditional software development process, in order to provide for the use of the MGIC model.

3.2 Initiation

The initiation stage aims at reaching consensus on the project goals among all stakeholders, and absorb and refine the information presented by the preliminary study. It comprises the following steps:

1. IT prepares (or reviews, in case of maintenance) the vision document for development projects.
2. IT prepares (or reviews, in case of maintenance) the project glossary, maintaining consistency with the ontology model.
3. IM validates the vision document and the glossary and determines whether it is necessary to maintain the ontology model.

4. IT details the requirements model, defining system use cases.
5. IT prepares the detailed schedule of the project and a list of implementation risks.

In this stage, step 3 is an addition to the traditional software development process to garantee that MGIC model is consistently updated.

3.3 Elaboration

The goal of the elaboration stage is to define a baseline for the system architecture, aiming at providing a stable base for the effort of the construction stage. It comprises the following steps, each validated with IM and the Client, and, when a software development company is contracted, with IT, ensuring the quality of each product:

1. IT refines the use cases with sequence diagrams and defines the class model, consistently with the logic information model.
2. IT refines the software architecture, considering non-functional requirements.
3. IT elaborates the interaction model and storyboard.
4. IT prepares preliminary user manual and training plan.
5. IT prepares the data model, which must be consistent with the information model (classes that access persistent data and their access methods).
6. IT prepares the acceptance tests plan (and homologation).
7. IT develops a map of function points.

3.4 Construction

The purpose of the construction stage is to complete the system development based on the architecture previously defined. It comprises the following steps:

1. The contractor implements the use cases.
2. The contractor prepares the system interaction manual, which complements the user manual.
3. The contractor prepares a maintenance test plan and test programs.

3.5 Transition

The transition stage focus on ensuring that the software is made available for its end users. This includes testing the product in preparation, comprising the following steps:

1. The contractor prepares the implementation plan.
2. The contractor prepares the installation manual.
3. The contractor and IT perform acceptance testing and generate an approval report. If there are outstanding issues, a plan is established to solve the problems raised.
4. The contractor deliveries the product.

3.6 Acceptance

In the acceptance stage, the main goal is to assess the results and service levels, based on the user's feedback. In this stage, steps 1, 4 and 5, that would tipically be executed by IT, are due to IM instead. Besides, KM is in charge of step 3, adhering to MGIC methodology, as follows:

1. IM and the client perform tests in the homologation environment to verify compliance with the specified requirements.
2. IT deploys the system in a production environment (client).
3. KM trains the users of the system for deployment.
4. IM conducts a satisfaction survey with the client.
5. IM issues the project closure term.

4 Conclusions

MGIC comprises a framework that governs the creation, the validation, the dissemination, the storage, the use and the recovery of information and knowledge to allow an organization to fully reach its goals. In our present research we are interested in defining how this set of models guide the development of new software functionalities to give support to some organizational business requirement. We proposed a software development process (SDP) to guide the development of software coherently with the MGIC models. One of the benefits of adopting this well-defined process is to increase the level of productivity in the organization. In particular, this SDP assures that new software functionalities will meet the general business requirements previously modeled. Moreover, the model will be maintained in order to incorporate any new requirements that may be identified.

5 Ongoing and future work

.

As an ongoing work, MGIC framework is being applied to create a model for the Brazilian public agency responsible for the regulation of terrestrial transpost services. While the model is being built, legacy systems are being mapped to check which business requirements they satisfy. The SDP has just being proposed and the implementation of new software functionalities is being discussed to serve as a case study.

References

1. Rezende, L., Malcher, C., Lobo, A., Castro, J., Burmann, C., Merino, L.: A Gesto do Conhecimento na Cadeia de Valor Colaborativa para o Desenvolvimento Sustentado do Transporte Terrestre no Brasil. In: Proc. of Knowledge Mangement Brasil. (2012)
2. Benevides, A., Guizzardi, G.: A model-based tool for conceptual modeling and domain ontology engineering in ontouml. In Filipe, J., Cordeiro, J., eds.: Enterprise Information Systems. Volume 24 of Lecture Notes in Business Information Processing. Springer Berlin Heidelberg (2009) 528–538