2015

Knowledge Sharing in Computerization Projects: An Approach Based on Boundary Objects

Mouna Ben Chouikha
InterVPNC Lab., University of Jendouba, Tunisia, mouna.benchouikha@dauphine.fr

Salem Ben Dhaou Dakhli
CERIA Lab., Paris-Dauphine University, France, Salem.Dakhli@dauphine.fr

KNOWLEDGE SHARING IN COMPUTERIZATION PROJECTS: AN APPROACH BASED ON BOUNDARY OJECTS

Completed Research

Ben Chouikha, Mouna, InterVPNC Lab., University of Jendouba, Faculty of Law, Economics, and Management, Jendouba, Tunisia, mouna.benchouikha@dauphine.fr

Dakhli, Salem Ben Dhaou, CERIA Lab., Paris-Dauphine University, France, Salem.Dakhli@dauphine.fr

Abstract

Computerization projects play a critical role in providing applications that meet the information needs of organizations and facilitate organizational change. They functions as a channel for the transmission of knowledge from individuals, methods, and past learning. As a result, knowledge management is a prerequisite to the success of such projects. Nevertheless, sharing knowledge held by organizational actors involved in computerization projects remains a difficult goal to achieve. In this paper, we analyze the obstacles to knowledge sharing in computerization projects in terms of knowledge boundaries. Apart from this analysis, the main contribution of this paper is the proposal of an approach to knowledge sharing in computerization projects, which relies on a typology of boundary objects. A case study based on development project of a reporting tool in an insurance company, has allowed us to verify the relevance of the approach proposed in this paper, and highlight its main managerial contributions..

Keywords: Computerization project, knowledge, knowledge boundary, boundary object, organizational actor, project space..

1 Introduction

The role of the information systems in the development and the survival of modern organizations is constantly growing (Henderson and Venkatraman, 1993; Henderson and Venkatraman, 1994; Hirschheim and Sabherwal, 2001; Sabherwal and Chan, 2001; Chan, 2001). Nowadays, information systems contribute to organizations efficiency and effectiveness and help them achieve difficult goals through providing an essential support to complex and interdependent organizational processes (Gurbaxani and Whang, 1991). The high frequency of changes induced by the external environment of organizations results in changes in the organizational processes and IT applications that support them. Not only such changes of information systems are difficult to implement but they are often related to severe cost, time and quality constraints. Moreover, the increasing share of the services economy (Horio and Watanabe, 2008) leads to institutional and legal changes that make value creation difficult without innovation of organizations catalogs of products and services and sales and delivery practices (Kotha, 1995; Kakati 2002). Innovation often results either in the creation of new organizational processes or in the modification of existing ones. To support efficiently innovative organizational processes, information systems must be agile and scalable. Moreover to be permanently effective and efficient, information systems evolves either by adding new applications or by modifying existing applications. Therefore, computerization projects play a critical role in providing applications that meet the information needs of organizations and facilitate organizational change. A computerization project is a temporary organization with unique and interdependent activities with a start date, an end date, a budget, and a goal consisting in the development of a software application to support partially or completely one or more organizational processes. The computerization projects can be considered as sets of organizational measures that enable modern organizations to reconcile short-term flexibility and long-term evolution, and constitute effective means to take into account new situations. These temporary organizations are complementary with permanent organizations that create stability (Ekstedt et al, 2002; Engwall, 2003; Manning, 2008; Kaulio, 2008). The success of computerization projects is a prerequisite to ensure that information systems play their full role in supporting organizational processes. It is measured by its efficiency and effectiveness. In other words, a computerization project can be considered a success if it results in an application that meets the needs of the organization, while respecting the constraints of cost, delay, and quality. However, successful computerization projects are rare as shown by many authors who noted that there is a software crisis born since the 60s of last century (Brooks, 1987, 1995; Toffolon, 1996; Toffolon, 1999; Reich, 2008). The software crisis has ramifications for both economic and social (Dakhli, 1998). Many solutions have been proposed in the literature to overcome the software crisis and reduce its impact. These solutions fall into two categories. The first category includes tools-based solutions aimed at improving the efficiency of computerization projects (Finlay and Mitchell, 1994; Coupe and Onodu, 1996). The second category comprises methods-based solutions intented to improving the effectiveness of computerization (Zachman, 1987; Sowa and Zachman, 1992; Boehm, 1988; Toffolon, 1996; Dakhli, 1998; Humphrey, 1999; Toffolon and Dakhli, 2002, Fayad et al., 2002; Longépé, 2006; Ross, 2006; de Vaujany, 2009). While these solutions have made significant improvements both in terms of efficiency and effectiveness of computerization projects, they not eliminate all the symptoms and causes of the software crisis which persists and seems to be worsening. This failure is in our opinion due to the fact that the solutions proposed to date have ignored that knowledge is the essence of both the computerization process and artifacts produced by these processes. So far, computerization projects were analyzed as temporary organizations where the action is paramount and tasks, budgets, people, and schedules must be controlled to achieve the expected results. Despite its contributions, this perspective is gradually replaced by a complementary perspective which considers that computerization projects are temporary organizations where learning and knowledge are essential. According to this new perspective based on both theories of knowledge management and organizational learning, a computerization project functions as a channel for the transmission of knowledge from individuals, methods, and past learning (Walz et al., 1993; Faraj and Sproull, 2000;

Bredillet, 2004; Reich, 2007; Reich et al., 2008). During computerization projects life cycles, knowledge is shared, created, integrated, and operated to generate added value for organizations. Learning takes place for all stakeholders involved in a project of computerization: the sponsor, the project manager, business teams, and technical teams. Hence, the most important task of the project manager consists in combining knowledge from heterogeneous sources and mobilizing them to achieve the objectives of the computerization project he manages. As a result, management of the knowledge related to computerization projects is a prerequisite to deal with the software crisis. In particular, knowledge sharing by organizational actors involved in computerization projects is critical to the success of such projects. In this paper, we propose a knowledge sharing approach based on boundary objects. The goal of our work is to answer the following research questions:

1) What are the boundaries of knowledge in IT projects?
2) How can we manage the boundaries of knowledge in a computerization project?

To achieve this goal, we draw on the software global model proposed by Toffolon and Dakhli (2002) that we supplement by focusing on knowledge sharing. The motivation behind this choice of this model is related to its generic and structuring nature and its ability to describe the various aspects of computerization. Our paper is organized as follows. Section 1 explains why knowledge is essential for computerization projects. In Section 2, we present the theoretical foundations of our work comprising on the one hand, a synthetic description of the software global model and on the other hand, a presentation of the boundary object concept. In Section 3, we analyze how the management of knowledge boundaries contributes to knowledge sharing. Section 4 uses the software global model to identify the boundaries of knowledge in computerization projects. The boundary objects to manage these boundaries are described in Section 5. In Section 6, we present our research methodology, which includes a case study described and analyzed using the approach proposed in this paper. Section 7 concludes the paper by listing the contributions and limitations of the proposed approach, and future research directions.

2 Computerization projects knowledge

The computerization projects are based on knowledge-intensive activities and result in informational artifacts (applications, documents, models, ...) that can be considered as an accumulation of knowledge shared by the organizational actors involved in these projects (Baetjer, 1998). According to this author, the effectiveness of these artifacts increases with the amount of shared knowledge between their stakeholders. Koskinen et al. (2003) and Newell et al. (2004) have confirmed that knowledge is created and shared in computerization projects. Other authors consider that computerization projects are temporary organizations that rely heavily on knowledge held by their team members to create informational artifacts (Lundin and Soderholm, 1995; Packendorff, 1995; Sodedund, 2002; Turner and Muller, 2003). Computerization projects team members learn both individually and collectively, transfer their knowledge to others, and create new shared concepts. Thus, computerization projects management and knowledge management are two nested processes (Leseure and Brookes, 2004). Moreover, Becker (2001) noted that the success of computerization project requires integrating knowledge distributed among several individuals and groups within the organization. Other authors consider that social capital is a possible channel for integration of organizational knowledge dispersed across individuals (Newell et al, 2004. Chiu et al, 2006). According to Adler and Kwon (2002), social capital is the capital sympathy resulting from the construction of social relations that can be mobilized for supporting action. Putnam (2000) identified two types of social capital: bonding social capital and bridging social capital. The former is a glue that strengthens the cohesion of teams through internal relationships between their members while the later relies on relationships between organizational actors to enhance the cohesion of social groups who do not know each other before. Newell et al. (2004) analyzed the impact of social capital on the integration of knowledge throughout a computerization project. The results of their work distinguish the initial phase of the other phases of a computerization project. During the initial phase of a computerization project, the members of this project team use bonding social capital to weld the team and strengthen its internal cohesion by creating a common understanding of the project objectives and sharing their knowledge. As as the project progresses, the members of the project team mobilize bridging social capital to acquire external knowledge distributed among several social groups in order to achieve the project objectives. The results of conceptual work emphasizing the importance of learning and knowledge in computerization projects have been confirmed by several empirical studies. For example, Faraj and Sproull (2000) pointed out that the coordination of experts facilitates project members learning and improves their performance. Yoo and Kanawattanachai (2001) concluded that the success of a computerization project depends firstly on the knowledge held by each member of the project team about the expertise areas of other team members and secondly on the ability of each member of the project team to mobilize knowledge to achieve the project objectives. Knowledge relevant to computerization projects fall into four categories identified in the literature related to organizational learning and knowledge management (Reich and Wee, 2006; Reich, 2007): process knowledge, domain knowledge, institutional knowledge, and cultural knowledge. The process knowledge refers to knwledge held by computerization project stakeholders about the project structure, tasks, milestones, and methodology (Chan and Rosemann, 2001; Meehan and Richardson, 2002). It is a body of knowledge that makes computerization project stakeholders aware of their contribution to the project and what is expected of them in terms of deliverables, and helps them organize themselves to achieve the project's goals. Domain knowledge includes business knowledge, organization's context knowledge (opportunities, problems, potential solutions, …), and products related knowledge (Chan and Rosemann, 2001). The domain knowledge is dispersed throughout the organization. For example, the sponsor of a computerization project often holds the most important knowledge about the business, the problem to solve and its potential solutions, and the existing opportunities. Technical experts within and outside the organization have the knowledge related to the most appropriate technologies to support the possible solutions to the computerization problem. Likewise, members of the project team have a deep understanding of the organization and its organizational processes impacted by the computerization project. Institutional knowledge are a mixture of stories, information about power structures, and values of the organization. They don't relate to facts but focus on how to interpret and understand facts. This category of knowledge is essential especially for a technology provider or an external project manager to understand the computerization and business problems, their solutions, and the decisions taken as part of the computerization project. Cultural knowledge is based on both the areas of expertise and the national cultures of the computerization project team members.

3 Theoretical foundations

In this section, we present synthetically the boundary object concept and the software global model which constitute the theoretical foundation of our work. In particular, we use the software global model to identify the computerization projects knowledge boundaries.

3.1 The software global model

Based on the economic agency and transaction costs theories (Coase, 1937; Williamson, 1981; Alchian and Demsetz 1972; Jensen and Meckling, 1976) and the theory of dimensions of the software (Toffolon, 1999), the software global model is a framework for software production in modern organizations (Toffolon and Dakhli, 2002). According to this model, a computerization project is a temporary organization characterised by a structure where a group of individuals perform tasks in accordance with a production technology to achieve a set of objectives (Leavitt, 1963). This model considers that the structure of a computerization project is based on four project spaces: the problem space, the solution space, the construction space, and the operation space. The computerization project stakeholders are represented by four types of organizational actors: the customer, the architect, the developer, and the end user. Each type of actor is associated with a project space where he plays the principal role and may play a secondary role in other project spaces. The customer is associated with the problem space and plays a secondary role in the solution and the operating spaces. The architect is associated with the solution space and plays a secondary role in the problem and the construction spaces. The developer is associated with the construction space and plays a secondary role in the solution and the operating spaces. The end user is associated with the operation space and plays a secondary role in the construction and the problem spaces. The software global model describes computerization projects as network of producer-consumer contracts between organizational actors belonging to the four types listed above. These contracts are realized according to a meta-iterative process called PACO (Problem - Solution - Construction - Operation) which characterizes the dynamic part of a computerization project i.e. the tasks performed by the members of the project team. A computerization project starts from an organizational problem identified and resolved in the problem space. The organizational solution is submitted by the customer to the architect who defines, in the solution space, a software solution (software architecture) to computerize the organizational solution. This solution is implemented in the construction space by the developer who delivers a software which supports the activities carried out by the end users in the operation space. In this paper, we complete the dynamic part of the software global model by adding a project management process implemented by a project manager who plays the principal role in a project management space. The project manager plays a secondary role in the four project spaces which characterize the static part of the original software global model. Likewise, the four types of organizational actors involved in a computerization project play a secondary role in the project management space. The project manager is the contact for all other organizational actors involved in the computerization project, and the guarantor of the projects efficiency and effectiveness. Finally, in each of the five project spaces, the organizational actors involved in a computerization project use a production technology to accomplish their tasks and achieve the project goals. This technology consists of material resources, methods, techniques, tools, and existing procedures within the organization. The following diagram (Figure 1) illustrates the computerization project model used in this paper.

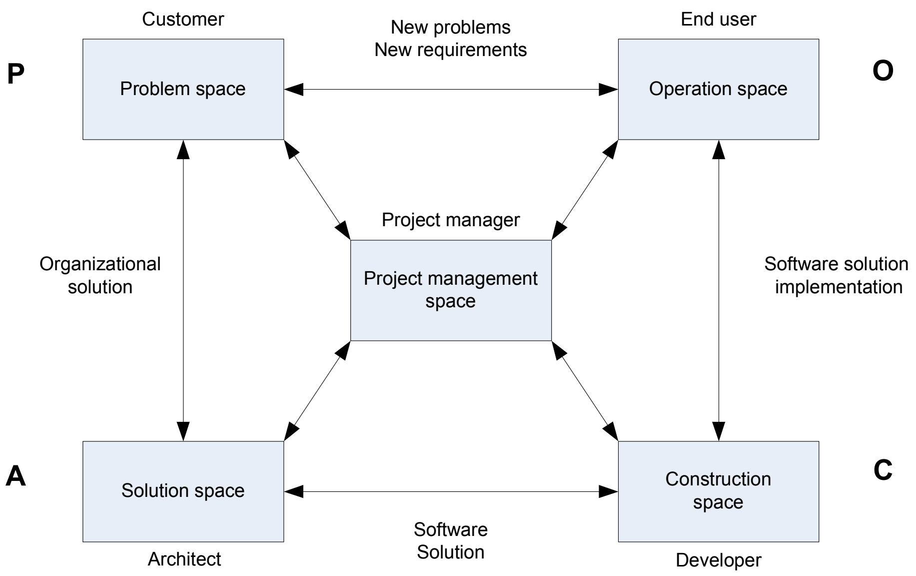

Figure 1. The computerization project model

3.2 The boundary object concept

The concept of boundary object was introduced by Star and Griesemer (1989) as an instrument of cohesion of many social worlds around a single shared goal. According to these authors, a boundary object is positioned at the crossroads of several social worlds to take into account the needs and constraints of each world. It is an instrument with a common set of shared knowledge that facilitates understanting and cooperation, and minimizes conflicts between organizational actors who have different representations of a problem. Thus, the boundary objects are characterized by their ability to federate, around a common goal, organizational actors with heterogeneous identities and different or conflicting interests. In other words, boundary objects are used for information collection, and knowledge sharing between organizational actors belonging to different social groups. Whether general or specific to a social world, a boundary object has three essential characteristics identifiable by all the social groups using it. First, it is flexible enough to adapt to the needs and constraints of organizational actors from different social worlds. Secondly, it is robust enough to maintain a common identity among many social worlds. Finally, a boundary object is not very structured in order to make its use easier in different social worlds. In addition to the three essential characteristics listed above, Vinck (2009) considers that a boundary object must be modular, multipurpose, standard, and enough abstract. In particular, the multipurpose nature of a boundary object means it can be used in the context of many different activities and practices. Star and Griesemer (1989) distinguish four categories of boundary objects: repositories, ideal types, multi-perspective objects, and standard forms. Repositories contain data that can be used by different organizational actors based on the social world they come from and their own practices. Databases and intranets are examples of repositories. Ideal types are boundary objects whose content is not identical for all social groups who adapt information inherent in these objects according to their needs and their particular contexts. Ideal types include notably computerization methods which are customized by organizations applying them. Multi-perspective boundary objects are shared by different social groups but their contents and their roles are perceived differently by these groups. For example, a software architecture model is a multi-perspective boundary object which is considered as a constraint and a source of delay by the project managers, and as a prerequisite for the quality and sustainability of the software solution provided by the information system architects. The standard forms are tools for the collection, aggregation, and analysis of data in a format shared by different organizational actors. The dashboard of a computerization project is an example of standard form shared by all project stakeholders. Briers and Chua (2001) suggest a fifth boundary objects category characterized by a high level of legitimacy and shared by a significant number of organizational actors either intra or inter-organizations that can use them specifically. Business rules, best practices shared by many managers, and organizational culture fall into this category. Organizational culture is a category of boundary object can be generalized to include moral and cultural values shared by organizational actors belonging to different social groups.

3.3 The management of knowledge boundaries'

The work of Carlile (2002, 2004) about the role of boundary objects in knowledge sharing is among the main contributions in this field. The significance of boundaries between different social groups and different disciplines has been highlighted by several authors (Leonard-Barton, 1995; Brown and Duguid, 2001). Based on the findings of Star and Griesemer (1989), Carlile (2002) identified three relational properties of knowledge at the boundaries between different fields. These properties are the difference, the dependence, and the novelty (Rebentisch and Carlile, 2003). The difference results from of the specialization areas and for the amount or type of accumulated knowledge. The concept of dependence between two entities defined by Litwak and Hylton (1962) as the condition requiring the two entities to consider each other to achieve their goals. The third relational property of knowledge at boundaries is related to the degree of novelty which impact the relations between actors belonging to different social worlds and involved in a project. Carlile (2002) noted that the emergence of a novelty at the knowledge boundaries between different disciplines or different social groups can result in the lack of a common language for representing the differences and dependencies between actors involved in a project.

According Carlile (2002, 2004, 2005), the management of knowledge boundaries is necessary to facilitate knowledge sharing in order to achieve the common goals of project team. This author distinguishes three types of knowledge boundaries (syntactic, semantic, and pragmatic) and proposes three approaches to manage efficiently Carlile (2004, 2005).

The syntactic boundary is characterized by the stability and the control of the differences and the dependencies between the organizational actors. Carlile (2004, 2005) stresses that knowledge transfer is the most appropriate approach for manager syntactic boundaries. Such an approach - called oriented information processing - is based on the development of a common language to share and evaluate knowledge at the boundaries.

The transition between the syntactic and the semantic boundaries occurs when a novelty makes ambiguous at least partly knowledge dependencies, differences or interpretations. To manage the knowledge semantic boundaries, Carlile (2004) suggests a translation approach called interpretative approach for creating common interpretations that constitute adequate means to share and evaluate knowledge at the boundaries.

The transition from a semantic pragmatic boundary to a pragmatic boundary occurs when an important novelty generates conflicting interests between actors. These interests constitute barriers to sharing and assessment of knowledge across boundaries. Indeed, in case of conflicting interests, knowledge developed in one domain often generates negative consequences for other domains, materialized by costs incurred for the actors belonging to these domains. Because of these costs, the actors are less willing to make the changes induced by an important novelty. According Carlile (2004), the management pragmatic boundaries is based on a transformation approach - called political approach - which facilitates the development of common interests through the transformation of actors conflicting interests and knowledge. These common interests constitute adequate means to share and evaluate knowledge at boundaries.

Carlile (2004) suggests that knowledge sharing at boundaries requires the creation of boundary objects common to organizational actors involved in a project. The knowledge representation using boundary objects (Star 1989) and the knowledge management process at boundaries that these objects facilitate is the key to the development of an effective context common to organizational actors who hold different types of knowledge. According to Carlile (2002), effective boundary objects establish a common language for representing knowledge and provide a concrete method to specify differences and dependencies of knowledge at boundaries. Boundary objects can be technical specialists (individuals), mutually accepted methods, or shared artifacts. Whatever their category, they not only facilitate the representation of knowledge, but also its translation and transformation. Moreover, the boundary objects provide guidance regarding the roles of the actors involved (who defines the knowledge representation? Who can modify it?). Ben Chouikha (2010) confirmed Carlile's conclusions (2002, 2004) while analyzing the design of earning organizations and validated these conlusions using a longitudinal case study focusing on several project teams in a context merger between two international companies of telecommunications services. Ben Chouikha and Dakhli (2011) completed Carlile's work through a study knowledge sharing in a virtual organization.

The creation of boundary objects for management of knowledge at boundaries leads to transformation of knowledge held by different actors interacting at boundaries. Such transformation starts when relevant sources of knowledge have been identified. If the degree of novelty is low or the dependencies between specialization domains are stable with defined boundaries or interfaces, the transfer of knowledge may be sufficient for knowledge sharing between individuals, groups, or organizations (Argote, 1999; Winter and Szulanski, 2001). However, the innovation degree increases, the differences and dependencies between individuals and groups of individuals often generate negative effects and problems that must be solved jointly. Thus, the specialized knowledge and expertise to be integrated must be transformed in order to eliminate the negative effects, and build a collective solution. The challenge of creating new knowledge is not limited to the transition from tacit knowledge to explicit knowledge through externalization as noted by Nonaka and Takeuchi (1995) but consists in negociating, redefining, transforming knowledge used by different groups to create a new solution (Carlile, 2002).

We note that the representation of knowledge, which plays an important role in this step, takes place at two levels: intra-specialization and inter-specializations. In particular, the bottleneck of the transformation of knowledge is through the different specialization domains (inter-specializations) since experts in different domains lack a common language or - in the case of complex situations – a common method of negotiation and trade-off between various alternative possibilities that enable different sources of knowledge to contribute to the construction of a new product or service. In other words, experts who hold different knowledge types need a shared context offering tangible and relevant common method and means to specify their differences and their dependencies. In particular, since the knowledge transformation process is iterative, this common method must improve the experts ability to represent, specify and negotiate their knowledge transformation throughout the iterations (Carlile, 2003). As a result, the integration of knowledge held by several organizational actors presents many difficulties and depends on the context in which these actors operate (Becker, 2001; Maaninen-Olsson et al., 2008; Carton and Farastier, 2012).

4 The knowledge boundaries in computerization projects

In this section, we first show that the difficulty to share knowledge in computerization projects is due to the existence of many knowledge boundaries between actors involved in such projects. Then we use the software global model to develop a typology of knowledge boundaries in computerization projects.

4.1 Knowledge sharing barriers in computerized projects

Becker (2001) has highlighted the difficulty to manage knowledge distributed among several individuals and groups within an organization. In particular, the computerization projects involve actors whose roles are multiple and involvement is variable. The interactions between these actors - needed to achieve the project objectives - are impacted by the specific nature of the organizational context of the project (cost, quality, time, ...). The dispersion of organizational knowledge may result in inadequate mobilization of critical knowledge due to differences in status between the project team members, the physical distance separating them, or the lack of familiarity related to differences in culture, thinking patterns, and languages. This is confirmed by Reich (2007) who has established that there is a lack of common understanding of knowledge management in the context of computerization projects and that the project team knowledge is difficult to mobilize to achieve the project objectives because of the absence of a knowledge management approach implemented within the organization (Arthur et al., 2001). According to Reich (2007), the knowledge management in the context of a computerization project consist in applying a set of principles and processes designed to bring significant knowledge to the project team. Thus, the effective management of knowledge facilitates the creation and integration of knowledge, minimizes the amount of lost knowledge, and reduces knowledge gaps between the members of the project team.

Hardy et al. (2005) defined the effective ccoperation as a process based on the differences between the participating actors to produce innovative solutions that represent a consensus among the concerns and viewpoints of these actors. This process is facilitated by the existence of an identity and shared practices (Hardy et al., 2005; Levina and Vaast, 2005; Feng et al., 2011). However, the differences in status between actors involved in cooperation constitute barriers for this process (Hoegl and Gemuenden, 2001; Levina, 2005; Metiu, 2006).

The ineffectiveness of cooperation in computerization projects is due to the existence of social or organizational boundaries between the members of the project team, which impede the creation of a common identity and shared practices and reinforce the status differences. Spatial or temporal physical distance between the members of project team is an example of organizational boundary (Levina and Vaast, 2005, 2008). Similarly, differences between spoken and written language, national or professional cultures, and the positions held are social boundaries that can both exacerbate inequality and reinforce status differences between the members of project team (Lam 1997; Espinosa et al., 2003; Krishna et al 2004;. Levina and Vaast, 2005; Levina and Vaast, 2006; Birnholtz and Finholt, 2007; Cramton and Hinds, 2007; Levina and Vaast, 2008).

Apart from organizational and social boundaries, Cramton and Hinds (2007) and Walsham (2002) emphasized that the boundaries resulting from the differences in practices between the members of project team are the most important barriers that inhibit the cooperation between these actors. Cooperation in a computerization project is made even more difficult by the fact that a significant part of the knowledge necessary to the project is tacit or embedded into the practices of the different actors involved in this project (Lam, 1997; Cramton, 2001; Hinds and Mortensen, 2004; Metiu, 2006, Levina and Vaast, 2008). Metiu (2006) pointed out that the status differences between computerization project team members can emerge from the context of project: conflicts related to design patterns or computer programs property,… Furthermore, the differences between the professional and industrial practices result in more knowledge boundaries that can reinforce status differences by creating new power relationships, and prevent the integration of new members or external stakeholders in a computerization project team (Bourdieu, 1984; Carlile, 2004; DiMaggio, 1991; Montgomery and Oliver, 2007).

4.2 Typology of knowledge boundaries in computerization projects

The software global model (Toffolon and Dakhli, 2002) shows that computerization projects take place in five project spaces (four areas of production and project management space) in which five types of organizational actors play either a key role either a secondary role. Software production is consistent with the iterative metamodel designated by the acronym PACO (Figure 1). Thus, a computerization project consists of three parts: static, dynamic, and organizational. This results in three types of knowledge boundaries:

- inter-espaces production knowledge boundaries,
- intra-spaces production knowledge boundaries,
- knowledge boundaries between between the project management space ant the organization including the production spaces.

The production inter-spaces knowledge boundaries result from the fact that the four types of project actors associated with the four spaces have specific and specialized knowledge. Thus, customer knowledge is related to his profession and to the problem and organizational solution to be computerizes. We note that even if the customer has knowledge about information and communication technologies, this knowledge is often too general. Similarly, the architect has only shallow and general knowledge of other businesses including those of the customer and the end user. Many architects know only little about development languages and databases. This is the case, for example, of functional architects. Moreover, developers usually ignore architectural rules and often consider them as constraints. They also generally have difficulty understanding the requests and questions of end-users who try to communicate with them without using a computer language.

The production intra-spaces knowledge boundaries result from the fact that a type of organizational actors associated with a production space includes a heterogeneous population of organizational actors holding different and specialized knowledge. For example, the "architect" type includes all organizational actors providing transverse support for computerization projects like functional architects, technical architects (software architecture specialists, infrastructure architects), experts in methodology, quality engineers, and databases experts. This is also the case of "developer" type which includes analysts, designers, programmers, and software integrators. Similarly, the "end user" type can include organizational actors belonging to different hierarchical levels and specialized in different businesses Finally, the "customer" type refers to a heterogeneous population consisting of middle managers, senior managers, and executives of an organization.

The knowledge boundaries between the project management space and the rest of the organization are related to the needs of communication and coordination of the project manager not only with organizational actors from the four production spaces, but also with the computerization project stakeholders. First, the project manager must interact with the customers to understand their needs and to report on the progress of the project. Second, he must also interact with the architect who helps him define a sustainable IT solution supporting the organizational solution. Third, the project manager must monitor the progress of their project, coordinate the project team and manage priorities, conflicts and resources. Finally, he must report to the project governance authorities. Thus, the project manager is in constant interaction with a heterogeneous population of organizational actors he does not share the specialized knowledge and expertise. It follows that computerization projects knowledge boundaries may be syntactic, semantic or pragmatic. In the remainder of this paper, the knowledge boundaries between the project management space and the organization are designated by the term "project management boundaries". The following table (Table 1) summarizes the computerization project types of knowledge boundaries.

Type	Boundaries list
Inter-spaces	- Customer - Architect - Architect - Developer - Developer – End user - End user - Customer
Intra-space	- Architect - Architect - Developer - Developer - End user – End user - Customer - Customer
Project management	- Project manager - Customer - Project manger – Architect - Project manager – Developer - Project manager – End user - Project manager – Governance authorities

Table 1. Typology of knowledge boundaries in computerization projects

5 The management of computerization projects knowledge boundaries

To manage boundaries knowledge in computerization projects, various boundary objects can be defined. Prior to listing examples of boundary objects, we propose a classification of these objects according to their main characteristics.

First, the three properties of boundary knowledge - the difference, the dependence, and the novelty - identified by Carlile (2002) impact the producer/consumer contracts characterizing the relationships between organizational actors involved in computer projects. As a result, the boundary objects facilitating boundary knowledge management play an important role in improving the development and implementation of these contracts. Thus, a boundary object can have four roles: transfer, clarification, production, and communication.

In a computerization project, a transfer boundary object provides the actors involved in a syntactic knowledge boundary with a common language which helps them transfer the knowledge necessary to carry out the contract between them. A set of specifications for the development of accounting software is an example of transfer boundary object. A clarification boundary object reduces uncertainty and ambiguity inherent in a contract between organizational actors involved in a semantic boundary. This is the case, for example, of an informative prototype produced by the requirements engineering process (Toffolon and Dakhli, 2002). We note that a clarification boundary object may be an organizational actor. For example, a facilitator is a clarification boundary object as it helps the architect and the developer reduce the ambiguity inherent in the organizational solution to computerize

Similarly, an architect immersed in a computerization project is a clarification boundary object as it helps developers understand the solution to implement, and the rules and architectural constraints to be taken into account. A production boundary object is a final artifact accepted by organizational actors participating in a knowledge boundary and preserved during the software lifecycle. For example, an architecture report validated by the customer and accepted by the project manager and the developer may be considered as a production boundary object. This is also the case of a software version accepted by the customer and the end-user who implements it in the operating space. Generally, production boundary objects permit managing pragmatic knowledge boundaries. However, such boundary objects can be used to manage a syntactic boundary in particular in the case of lack of novelty.

A communication boundary object enables managing project management boundaries which involve the project manager. Indeed, the project manager must communicate with various organizational actors concerned with the computerization project. Considering the heterogeneity of these actors and knowledge they hold, the project manager uses various communication boundary objects for management of project boundaries. For example, a simplified Gantt chart is a suitable communication boundary object for mangement of the Project manager – Customer boundary while a detailed GANTT chart, or a PERT chart are communication boundary objects for management of the Project manager – Developer boundary. Similarly, a dashboard describing synthetically the progress of the project and explaining the delays is a communication boundary object which help manage the Project manager border - governance authorities boundary. We note that the communication boundary objects are generally standardized and help manage the three kinds of knowledge boundaries.

Second, the boundary objects are derived from the three approaches proposed by Carlile (2004) depending on the knowledge boundaries to be managed (Feng et al., 2011). Thus, it is obvious that the transfer boundary objects are derived from the knowledge transfer approach. Similarly, the clarification boundary objects result from the interpretive approach. Moreover, the production boundary objects are often issued from the political approach. Indeed, such boundary objects are finished products which must reconcile the conflicting interests of organizational actors involved in a pragmatic boundary. However, if there is no novelty, a production boundary object can be derived from a knowledge transfer approach. Communication boundary objects may be derived from one of from the three approaches proposed by Carlile (2004). Indeed, since the content of communication boundary objects often includes indicators or instruments related to intangible artifacts, these objects should be consensual to reflect the conflicting viewpoints of the computerization project stakeholders. In this case, they may be considered the result of the political approach. In other cases, the communication boundary objects are derived from the interpretive approach. This is the case, for example, if an organizational actors involved in a knowledge frontier does not understand the meaning of the content of a communication boundary object. However, many standards of communication boundary objects exist and are accepted by most of the organizational actors involved in computerization projects. The relationships between knowledge boundaries management approaches (Carlile, 2004) and boundary objects are summarized in Table 2.

Finally, a boundary object can be described using three dimensions: boundary, role, and knowledge boundaries management approach. This provides a typology consisting of a large number of boundary objects associated with a computerization project. Certainly, all these boundary objects do not have the same weight in the management of knowledge boundaries of a computerization project. Nevertheless, that their number is important is an indicator of the complexity of knowledge boundaries and the difficulty of sharing knowledge in a computerization prject team. Table 3 provides examples of boundary objects used in computerization projects.

Boundary object	Management approach of knowledge boundaries
Transfert boundary object	- Knowledge transfert approach
Clarification boundary object	- Interpretive approach
Production boundary object	- Political approach - Knowledge transfert approach
Communication boundary object	- Political approach - Interpretive approach - Knowledge transfert approach

Table 2: *Relationships between knowledge boundaries management approaches (Carlile, 2004) and boundary objects*

Boundary	Example of boundary object	Role
Customer - Architect	- Specifications - Facilitator - Architecture report	- Transfert - Clarification - Production
Architect - Developer	- List of architecture rules to be applied - Architect immersed in the project team - Access to architecture FAQ - Architecture report	- Transfert - Clarification - Clarification - Production
Developer – End user	- Informative prototype - Mock up - Training session - Final software version	- Clarification - Clarification - Clarification - Production
End user - Customer	- Requirements list - Software evolution requests list	- Transfert - Transfert
Project manager – Governance authorities	- Project status dashboard	- Communication
Project manager - Developer	- Detailed GANTT chart - Reference schedule - Meeting report	- Communication - Communication - Communication
Developer - Developer	- Data model - Tested program to be integrated - Access to development FAQ - Access to a repository of coding problems and their solutions	- Production - Production - Clarification - Clarification

Table 3: Examples of boundary objects used in computerization projects

6 Research methodology

In this section, we briefly present the case study, the data collection methodology, and the analysis of results.

6.1 The case study description

The empirical validation of the approach presented in this article is based on a case study based on a development project of a reporting tool in an international insurance group. This group was formed through many mergers and acquisitions involving French and foreign insurance companies. It now brings together about 30,000 employees in several countries. The strategy of this group is to significantly increase its share in the savings market. The target customer segment consists of affluent customers able to save between 100,000 € and 500,000 € excluding real estate. To capture these customers, the insurance group sets up a specific commercial approach and dedicated sales forces. In order to enable financial advisors to offer these customers the best quality of service, it is planned to provide them with a comprehensive reporting tool for monitoring saving and customers assets. The purpose of this tool - designated in the remainder of this paper by the acronym NREP (New Reporting) - is to provide financial advisors with a consolidated view of the assets of each customer, facilitate financial analysis of the customer assets portfolio, and provide each customer with a reporting on the status of his account. Similarly, the insurance group's marketing department has expressed interest in NREP tool and expect that this tool provides indicators that facilitate the implementation of marketing campaigns. In other words, the main functions of the reporting NREP tool conist in:

- providing customers with a high quality after-sales service,
- enabling the financial advisors and the customers share the same objective vision of saving contracts,
- enabling a delegation – to each the customer - of some acts of management related to its contracts (payment, buyout,...),
- help financial advisors preparing their interviews with customers,
- reduce the customer’s telephone calls to his financial advisor,
- facilitate the framing and focus of marketing campaigns.

The NREP tool will be accessible to the customers so that their contracts are visible at any time. The process to be computerized describes the commercial approach followed by financial advisors to capture affluent customers targeted by the insurance group strategy. This process has four steps. The first step consists in making a diagnosis of the customer’s wealth. This diagnosis is the basis from which the financial advisor defines the advice that he will propose to the customer. The second step perform simulations are performed to convince the customer by demonstrating the relevance and appropriateness of this advice. The third step consists in developing a reporting accessible at any time by the customer and the financial advisor in order to ensure the transparency of the relationships between the advisor and the customer. The fourth step is carried out by financial advisors and consists in monitoring the customers assets portofolio. Currently, financial advisors of the insurance group do not have a single, comprehensive view of all savings contracts of their customers. Certainly, many reporting tools are available within this insurance group and used by financial advisors. However, these tools do not allow financial advisors consolidate the results of their customers portfolios, present the following several analysis axes, and querying the customers portfolios and contracts using several criteria.

The tool NREP to be developed is a data container for financial advisors, customers and savings contracts. Data is managed by existing IT applications belonging to the insurance group's information system.

After a preliminary study, the insurance group to develop the NREP tool in house by its French subsidiary Information Systems Departement composed of an Architecture unit and a Development unit. The Architecture unit consists of functional architects, technical architects, methods experts, quality engineers, databases engineers, and experts in reusable technical components. These human resources are either employees of the insurance group or external consultants. The Development unit consists of analysts and designers who are either employees of the group or external consultants. Implementation, unit testing and integration testing of applications are carried out by an Indian "off-shore" company. Software applications developed in house must respect the Information System architecture and urbanization rules published in a guide available to the Development unit. The computerization projects should be conducted according to a development method defined by the methods engineers. Despite this method is based on the waterfall lifecycle (Boehm, 1976, 1984), it enables software prototyping during the user’s requirements definition phase and allows iterative development according to the spiral model (Boehm, 1988). Apart from the geographical and cultural dispersion of the NREP computerization project design and implementation teams, this project has two important challenges. On the one hand, users have not formally expressed their needs and requirements On the other hand, to take into account the diversity of savings contracts proposed by the insurance group, the NREP tool must be able to exchange informational flows with all applications that manage these contracts. It follows that the NREP tool must take into account the technical constraints (database structure, data format, interface constraints, ...) and the operational constraints (information availability, frequency of exchange, ...) of these applications.

6.2 Data collection and results analysis

The data collected is issued from two sources among those identified by Yin (1994). On the one hand, we have consulted the documents made available by the project manager or published on the project intranet, the documents available on the intranet site of the French subsidiary of the insurance group or in the documentation repository, and the documents provided by the Architecture unit. On the other hand, we have collected information on the project progress and key events that have affected it (delays, technical problems, organizational problems,...) through a non-participant observation that consisted in attending important meetings organized by the project manager (team meeting, steering committee) and exchanging informally with the members of the project team, the contributors to the project belonging to the Architecture unit, or other project stakeholders. The approach adopted to carry out our case study is an inductive qualitative method where field data are used to highlight concepts representative of the studied phenomenon (Thiétart, 1999).

Analysis of the information collected has confirmed the interest of this case study both at the level of the theoretical model evaluation, and in terms of its managerial implications. First, the organization of Information Systems Department of the the insurance group French subsidiary studied in this paper has many similarities with the organization suggested by the computerization project model presented above. Indeed, apart from the project management space represented by the project manager, the Architecture unit can be likened to the solution space while the construction space consists of the Development unit and the Indian "offshore"company. The problem space is composed of representatives of the NREP customers who are either members of the marketing department, or financial advisors. End users of NREP tool are grouped in the operation space. It follows that the computerization project model is appropriate to model the Information Systems Department which carries out the development of the NREP tool within the insurance group. Second, information collected allowed us to verify the existence of the knowledge boundaries identified by the theoretical model proposed in this paper.

On the one hand, the fact that the definition of the users needs and requirements is not described formally confirms the existence of a knowledge boundary between the problem side (problem space, operation space) and the solution side (solution space, construction space). This boundary is the source of users requirements definition problems which result in the failures of many compouterization projects. The solutions proposed in the literature to manage this boundary was not fully effective as they have analyzed this boundary holistically. In accordance with Brooks analysis (Brooks, 1987), our model considers that that the knowledge boundary between the problem and the solution sides may be broken down into four types: Customer – Architect, Project manager - Customer, Project manager - End user, Developer - End user. This decomposition permits suggesting appropriate means to handle each type knowledge boundary.

On the other hand, the diversity of professions within the solution side confirms the existence of knowledge boundaries both between the Architecture and Development units and within each unit. Indeed, the Architecture unit is peopled of experts who do not use the same practices and the same professional concepts while performing different activities. For example, the concepts and practices used by functional architects and database experts are different. Similarly designers and developers do not always understand the usefulness of architectural rules and methodological and quality assurance standards, imposed by the Architecture unit.

Furthermore, within the problem side, the diversity of end users and their representatives shows the existence of knowledge boundaries between end users and their representatives, between end-users, or between representatives of end users. For example, the concepts, practices, and interests of financial advisors differ significantly from those used by the marketing department members. Therefore, our model facilitates the detailed identification of knowledge boundaries in computerization projects. As a result, the application of this model to information collected as part of our case study permit us verifying the existence of the knowledge boundaries listed above (Table 1). In the rest of this section, we have focused on a subset of Knowledge borders whose analysis summarizes the main findings of our case study. These are the following knowledge boundaries.

In the remainder of this section, we have focused on a subset of knowledge boundaries whose analysis summarizes the main conclusions of our case study. These knowledge boundaries are the following:

- a knowledge boundary between the customer representatives and the functional architect involved in the project,
- a knowledge boundary between the functional architect and the technical architect involved in the project,
- a knowledge boundary between the functional and technical architects and the designers,
- a knowledge boundary between the designers and the offshore programmers,
- a knowledge boundary between the methods engineers and the project team,

Knowledge boundaries management took place through a series of decisions taken by the project manager and validated by the governance authorities. These decisions have impacted both the project team that the forms of collaboration between the various project stakeholders. Thus, a facilitator intervened during the early stages of the project to help the functional architect to understand all the business aspects to be taken into account in defining the architecture of the NREP tool. This facilitator is a former financial advisor who joined in recent years Information Systems Department as a manager of the computer systems technical and functional problems. This decision helps manage the knowledge boundary between the customer representatives and the functional architect involved in the project. Based on the information we have collected, the facilitator was found to be effective by the actors involved in this knowledge boundary between the customer representatives and the functional architect involved in the project. Therefore, the facilitator can be considered as a clarification boundary object which permits managing the Customer – Architect inter-spaces boundary. However, a business dictionary uniquely defining the different business concepts can also be considered as an additional clarification boundary object complementary to the facilitator boundary object.

The knowledge boundary between the functional architect and the technical architect involved in the project is an intra-space boundary that materializes a misunderstanding between these two actors due to the lack of a shared professional culture. Indeed, the technical architect is a former programmer who does not master the software application architecture (components, connectors) and has only vague notions of the functional architecture and business intelligence architecture. The functional architect has no significant experience in defining the software architecture of an application and does not master all the constraints related to the interfaces between components or between applications. To prevent these divergences of views between the technical and the functional architect, the project manager suggested that the functional architect introduces the technical architect to the basic concepts of decisional architecture. Although this training is deemed positive by the technical architect, his divergences with the technical architect persisted throughout our observation period. These divergences of views were compounded by a lack of trust between the two actors. They often surfaced in the meetings of the project team and working sessions with designers. Despite these viewpoints differences, the software architecture of the NREP tool has been developed on the basis of the functional architecture. Also, the training provided by the functional architect played only partially the role of clarification boundary object

The knowledge boundary between the architects (functional and technical) and the designers is an inter-spaces boundary that has been managed in three stages. First, a training session was conduced by the functional architect and the technical architect in order to familiarize the two designers with the basic concepts and constraints of the information systems architecture, and convince them that these concepts are useful. This training can be considered as a clarification boundary object unlike architectural guides, published by the Architecture unit, which was considered incomprehensible by the two designers. Second, prototypes of the NREP tool functional and software architectures were developed iteratively by both architects and discussed by the two designers over several working sessions. The functional and software architectures NREP tool have been defined following these sessions. Third, it was decided to immerse part-time a technical architect in the development teams (design, coding) to help designers and programmers implement the NREP tool functional and software architectures and verify that all features of this solution was taken into account. Apart from training, four boundary objects can be identified at this stage. There are two clarification boundary objects and two objects production boundary objects. The NREP tool functional and software architectures prototypes and the technical architect associated with the development team are clarification boundary objects. The NREP tool functional and software architectures are both production boundary objects.

The knowledge boundary between designers and offshore programmers is a complex intra-spaces boundary difficult to manage. The complexity of this boundary is due to the existence of three types of dispersions: geographic, linguistic, and cultural. The geographical dispersion reflects the remoteness of design and implementation teams. This remoteness was compounded by the linguistic dispersion caused by the problems of spoken and written languages of programmers and designers. Thus, the communication between the two parties - which took place by telephone, mail, or video conferencing - has been difficult and resulted in many misunderstandings. The cultural dispersion, caused by differences in programmers and designers national cultures has fostered a climate of mistrust between the two parties which constituted a barrier to knowledge sharing. The communication difficulties between the programmers and designers, and the lack of English translation of documents and design standards to be respected explain the fact that there was no boundary object contributing the knowledge boundary management. In order to cope with the risks of failure of the project, the project manager has proposed to call upon an international consulting company composed of information systems consultants mastering both the English and French languages, including indian consultants. Following the approval of the project manager decision by the governance authorities, the international consulting company became the interlocutor of the project team and the architects and the mediator between them and the programmers of the Indian offshore company. This company has performed the English translation of the documents provided by the designers and the standards to be applied, and reported on the progress of programming work to the project manager. Similarly, it has realized the French translation of the documents prepared by the Indian programmers and provided a French version of the NREP tool to the Information Systems Departement for integration tests. At this stage, many boundary objects can be identified. For example, the French version of the NREP tool, the implementation documents translated into French, and the design documents translated into English are transfer boundary objects. We note that among these boundary objects, some will be packaged and delivered to end users. In this case, they will become production boundary objects. The status and progress reports provided by the international consulting company are communication boundary objects. Finally, the international consulting company can be considered as a clarification boundary object.

The knowledge boundary between methods engineers and the project team reflects the resistance of the designers to the application of the development defined and recommended by the Architecture unit. Although this method enables software prototyping and iterative development, it is perceived by designers as being too restrictive and a source of inconsistencies and long delays. To circumvent the designers resistance, the project manager has proposed to the methods engineers to cooperate with the designers in order to identify the methodological aspects and documentary standards whose application will have a significant added value for the project progress. This decision has not been approved by the governance authorities who feel that its implementation will be costly and cause delays. Thus, no boundary object was built to manage this knowledge boundary.

This case study presents three advantages. On the one hand, it allowed us to verify the interest of the approach proposed in this paper to identify the knowledge boundaries in a computerization project, and analyze the boundary objects that help manage them. Indeed, the existing studies of computerization projects knowledge boundaries are often descriptive and do not provide sufficient information to facilitate the knowledge sharing and help the project governance authorities take the appropriate decisions. On the other hand, it has provided information confirming the need to clarify the concept of boundary object. Since boundary objects are intended to facilitate knowledge sharing, such objects can take many forms. They can be informational artifacts, human beings with multiple skills, or organizational entities. This confirms some of the results obtained by Levina and Vaast (2005) and Carton and Farastier (2012) who have introduced the boundary actor concept. Moreover, to be effective, boundary objects in the form of informational artefacts must be built collectively by organizational actors involved in knowledge boundaries. This explains the positive role played by the training sessions, the informational prototypes in the management of knowledge boundaries in our case study. This also explains why the guides and standards built by a group of actors are not always effective boundary objects. Finally, we use this case study to highlight the following managerial implications:

- To manage inter-spaces knowledge boundaries, one can effectively use human actors with multiple skills as boundary objects.
- To manage the knowledge boundaries resulting from differences in national cultures or spoken or written languages, organizational entities can be effective boundary objects.
- To manage a knowledge boundary with a boundary object in the form of an informational artifact, all organizational actors involved in this boundary must participate in the construction of this boundary object.
- Setting up a business dictionary can help manage knowledge boundaries between the problem side and the solution side.

7 Conclusion and future reserach directions

In this paper, we proposed a theoretical framework to analyze the knowledge boundaries in a computerization project and identifying the boundary objects that help manage them. The computerization project model used in this work is a generic computerized approach and not a particular computerization method. It is therefore open to all computeraization approaches and methods whatever their nature. The identification of computerization projects knowledge boundaries and boundary objects that help manage them is the first contribution of our article that provide an answer to the first research question. The interest of this analysis is related on the one hand, to its independence of existing computerized methods, and on the other hand, in the consideration - in each identified knowledge boundary – of all organizational actors who participate to it. For example, the Customer – Architect knowledge boundary takes into account not only the customer and the architect types but also the developer and end user types who play a secondary role respectively in solution space and the problem space. That each type of computerization project knowledge boundary includes many heterogeneous organizational actors who hold specialized knowledge permits highlighting the complexity of such boundaries and the difficulty to manage them. In addition, we have answered the second research question by defining a typology of boundary objects in the context of a computerization project and providing examples of such objects. The second contribution of our paper is related to the accuracy of the boundary object concept. Indeed, we have established that a boundary object can be either an informational artifact, an organizational actor, or an organizational entity. The third contribution of our paper results from the managerial implications of the approach we have proposed for the knowledge boundaries management. However, the analysis in this article should be more detailed and enough thorough to describe the production boundary objectsissued from the political approach. This is to specify in particular the role of national and organizational cultures in the shaping of production boundary objects if innovation and instability are not limited to technology but include social and cultural factors. This is the case, for example, of computerization projects in virtual organizations or post-merger organizations. Another important direction of research is to use the findings of this paper to build instruments that permit assessing the ability of computerization methods and approaches to promote the knowledge sharing.. Finally, the findings from the case study presented in this paper should be enhanced and consolidated with other case studies. This is a third research direction.

References

Adler, P. and S. W. Kwon (2002). “Social capital: Prospects for a new concept.” *Academy of Management Review* 27 (1), 17-40.

Alchian, A. A. and H. Demsetz (1972). “Production, Information Costs and Economic Organization.” *American Economic Review* 62 (5), 777-795.

Argote, L. (1999). *Organizational Learning: Creating, Retaining, and Transferring Knowledge*. Norwell, MA: Kluwer.

Arthur, M., De Fillippi, R. and C. Jones (2001). “Project-based learning as the interplay of career and company nonfinancial capital.” *Management Learning* 32 (1), 97-11.

Baetjer, H., Jr. (1998). “Software as Capital: An Economic Perspective on Software Engineering.” Piscataway, New Jersey: The Institute of Electrical and Electronics Engineers.

Becker, M. C. (2001). “Managing dispersed knowledge: organizational problems, managerial strategies, and their effectiveness.” *Journal of Management Studies* 38 (7), 1037-1051.

Ben Chouikha M. and S. Ben Dhaou Dakhli (2011). “Knowledge Sharing Enablers and Barriers: The Case of Virtual Organizations.” In: *Proceedings of the IFIP-AIS MCIS’2011 conference*, Limassol, Cyprus, September 3-5, 2011.

Ben Chouikha, M. (2010). “Design de l’Organisation Apprenante : Etude de Cas Longitudinale des Equipes de Projet NOKIA SIEMENS NETWORKS. ” Thèse de doctorat, Université de Paris-IX Dauphine, Paris.

Birnholtz, J. and T. Finholt (2007). “Cultural Challenges to Leadership in Cyberinfrastructure Development.” In: *Leadership at a Distance: Research in Technology- Supported Work*, S. Weisband (ed.), New York: Lawrence Erlbaum Associates, 195-207.

Boehm B. W. (1988). “A Spiral Model of Software Development and Enhancement.” *IEEE Computer* 21 (5), 61-72.

Boehm, B. W. (1976). “Software Engineering.” *IEEE Transactions on Computers* C25 (12), 1226-1241.

Boehm, B.W. (1984). “Verifying and Validating Software Requirements and Design specifications.” *IEEE Software* 1 (1), 75-88.

Bourdieu, P. (1984). *Distinction: A Social Critique of the Judgment of Taste*. Cambridge, MA: Harvard University Press.

Bredillet, C. N. (2004). “Projects: Learning at the edge of organization.” In: P. W. Morris and J. K. Pinto (Eds.), *The Wiley Guide to Managing Projects*. Hoboken: John Wiley & Sons.

Briers M. and W. F. Chua (2001). “The role of actor-networks and boundary objects in management accounting change: a field study of an implementation of activity-based costing.” *Accounting. Organizations and Society* 26, 237-269.

Brooks, F. P. Jr. (1987). “No Silver Bullet-Essence and Accidents of Software Engineering.” *IEEE Computer* 20 (4), 10-19.

Brooks, F. P. Jr. (1995). *The Mythical Man Month: Essays on Software Engineering*. Reading, M.A.: Addison-Wesley.

Brown, J. S. and P. Duguid (2001). "Knowledge and organization: A socialpractice perspective." *Organization Science* 12, 198–213.

Carlile, P. R. (2002). "A Pragmatic View of Knowledge and Boundaries: Boundary Objects in New Product Development." *Organization Science*, 13 (4), 442-455.

Carlile, P. R., and C. Østerlund (2005). "Relations in Practice: Sorting Through Practice Theories on Knowledge Sharing in Complex Organizations." *Information Society* 21 (2), 91-107.

Carlile, P. R. and E. S. Rebentisch E. S. (2003). "Into the Black Box: The Knowledge Transformation Cycle." *Management Science* 49 (9), 1180-1195.

Carlile, P. R. (2004). "Transferring, Translating, and Transforming: An Integrative Framework for Managing Knowledge across Boundaries." *Organization Science* 15 (5), 555-568.

Carton, S. and A. Farastier (2012). "Intégration des connaissances client dans un projet en systèmes d'information : influence de l'environnement de connaissance du projet." *Revue Systèmes d'Information et Management* 17 (2).

Chan, R. and M. Rosemann (2001). "Managing knowledge in enterprise systems." *Journal of Systems and Information Systems* 5 (2), 37–53.

Chan, Y. E. (2001). "Why Haven't We Mastered Alignment? The Importance of the Informal Organization Structure." *MIS Quarterly Executive* 1 (2), 97-112.

Coase, R. (1937). "The Nature Of The Firm." *Economica* 4, 386-405, Traduction Française: *Revue Française d'Economie* 2 (1), 133-163.

Coupe, R. T. and N. M. Onodu (1996). "An empirical evaluation of the impact of CASE on developer productivity and softw are quality." *Journal of Information Technology* 11 (2), 173-181.

Cramton, C. D. and P. J. Hinds (2007). "Intercultural Interaction in Distributed Teams: Salience of and Adaptations to Cultural Differences." In: *Proceedings of the Academy of Management Annual Meeting, Best Papers*. G. Salomon (ed.), Philadelphia, PA, August 3-8.

Cramton, C. D. (2001). "The Mutual Knowledge Problem and Its Consequences for Dispersed Collaboration." *Organization Science* 12 (3), 346-371.

Dakhli, S. (1998). "Le Prototypage." Thèse de doctorat, Université de Paris-IX Dauphine, Paris.

De Vaujany, F.-X. (2009). *Les grandes approches théoriques du système d'information*. Paris Editions Hermès-Lavoisier.

DiMaggio, P. J. (1991). "Social Structure, Institutions, and Cultural Goods: The Case of the United States." In: *Social Theory for a Changing Society*. P. Bourdieu and J. S. Coleman (eds.), Boulder, NY: Westview Press, pp. 133-155.

Ekstedt, E. (2002). "Contracts of work in a project based economy." in Sahlin Andersson, K., Söderholm, A. (Eds.), *Beyond project management – New perspectives on the temporary – permanent dilemma*. Liber, Malmö: pp. 59-77.

Engwall, M. (2003). "No project is an island: linking projects to history and context, Research Policy." 32 (5), 789-808.

Espinosa, J. A., Cummings, J. N., Wilson, J. M. and B. M. Pearce (2003). "Team Boundary Issues across Multiple Global Firms." *Journal of Management Information Systems*. 19 (4), 157-190.

Faraj, S. and L. Sproull (2000). "Coordinating expertise in software development teams." *Management Science* 46 (12), 1554–1568.

Fayad, M., Henry D. and D. Bougali (2002). "Enterprises Frameworks." *Software Practice and Experience* 32, 735-786.

Feng; Y., Ye, H. and S. L Pan (2011). "Delivering Knowledge Across Boundaries: A Case Study of Bankco's Offshoring Projects." *Pacific Asia Journal of the Association for Information Systems* 3 (3), 57-79. Available at: http://aisel.aisnet.org/pajais/vol3/iss3/4.

Finlay, P. N. and A. C. Mitchell (1994). "Perceptions of the Benefits From the introduction of CASE: An Empirical Study." *MIS Quarterly* 18 (4), 353-370.

Gurbaxani V. and S. Whang (1991). "The Impact of Information Systems on Organizations and Markets." *Communications of the ACM* 34 (1), 59-73.

Hardy, C., Lawrence, T. B. and D. Grant (2005). "Discourse and Collaboration: The Role of Conversations and Collective Identity." *Academy of Management Review* 30 (1), 58-77.

Henderson, J. C. and N. Venkatraman (1994). "Strategic Alignment: A Model for Organizational Transformaton via Information Technology." In: Allen, T. J. and Scott Morton, M. S. (Eds.), *Information Technology and the corporation of the 1990s*. Research Studies. Oxford: Oxford University Press.

Henderson, J. C. and N. Venkatraman (1993). "Strategic alignment: Leveraging information technology for transforming organizations." *IBM Systems Journal* 32 (1), 472-484.

Hinds, P. J. and M. Mortensen (2004). "Understanding Conflict in Geographically Distributed Teams: The Moderating Effects of Shared Identity, Shared Context, and Spontaneous Communication." *Organization Science* 16 (3), 290-307.

Hirschheim, R. and R. Sabherwal (2001). "Detours toward Strategic Information Systems Alignment." *California Management Review* 44 (1), 87-108.

Hoegl, M. and H. G. Gemuenden (2001). "Teamwork Quality and the Success of Innovative Projects: A Theoretical Concept and Empirical Evidence." *Organization Science* 12 (4), 435-449.

Horio, H. and C. Watanabe (2008). "The Paradox of a Service-Oriented Economy for Sustainability: Co-Evolution between Innovation and Resources Effectuation by a Global Complement." *Journal of Services Research* 8 (1), 155-176.

Jensen, M. C. and W. H. Meckling (1976). "Theory of the Firm: Managerial Behavior, Agency Costs and Ownership Structure." *Journal of Financial Economics* 3 (4), 305-360.

Kakati, M. (2002). "Mass customization – needs to go beyond Technology." *Human Systems Management* 21, 85–93.

Kaulio, M. A. (2008). "Project leadership in multi-project settings: Findings from a critical incident study. *International Journal of Project Management* 26 (4), 338-347.

Koskinen, K. U., Pihlanto, P. and H.Vanharanta (2003). "Tacit Knowledge Acquisition and Sharing in a Project Work Context." *International Journal of Project Management* 21 (4), 281-290.

Kotha, S. (1995). "Mass customization: Implementing the Emerging Paradigm for Competitive Advantage." *Strategic Management Journal* 16, Special Issue, 21–42.

Krishna, S., Sahay, S. and G. Walsham (2004). "Cross-Cultural Issues in Global Software Outsourcing." *Communications of the ACM* 47 (4), 62-66.

Lam, A. (1997). "Embedded Firms, Embedded Knowledge: Problems of Collaboration and Knowledge Transfer in Global Cooperative Ventures." *Organization Studies* 18 (6), 973-996.

Leavitt, H. J., (ed.) (1963). *The Social Science of Organizations, Four Perspectives*. Englewood Cliffs, New Jersey: Prentice-Hall.

Leonard-Barton, D. (1995). *Well Springs of Knowledge: Building and Sustaining the Sources of Innovation*. Boston, MA: Harvard Business School Press.

Leseure, M. J., and N. J. Brookes (2004). "Knowledge management benchmarks for project management." *Journal of Knowledge Management* 8 (1), 103-116.

Levina, N., and E. Vaast (2008). "Innovating or Doing as Told? Status Differences and Overlapping Boundaries in Offshore Collaboration." *MIS Quarterly* 32 (2), 307-332.

Levina, N., and E. Vaast (2005). "The Emergence of Boundary Spanning Competence in Practice: Implications for Implementation and Use of Information Systems." *MIS Quarterly* 29 (2), 335-363.

Levina, N., and E. Vaast (2006). "Turning a Community into a Market: A Practice Perspective on It Use in Boundary-Spanning." *Journal of Management Information Systems* 22 (4), 13-38.

Levina, N. (2005). "Collaborating on Multiparty Information Systems Development Projects: A Collective Reflection-in-Action View." *Information Systems Research* 16 (2), 109-130.

Litwak, E. and L. F. Hylton (1962). "Interorganizational Analysis: A hypothesis on coordination agencies." *Administrative Science Quarterly* 6, 395–420.

Longépé, C. (2006). *Le Projet d'Urbanisation du SI*. Paris: Editions Dunod.

Lundin, R. A. and A. Soderholm (1995). "A Theory of the temporary organization." *Scandinavian Journal of Management* 11 (4), 437-455.

Maaninen-Olsson, E., Wismén, M. and A. A. Carlsson (2008). "Permanent and temporary work practices: knowledge integration and the meaning of boundary activities." *Knowledge Management Research & Practice* 6 (4), 260-273.
Manning, S. (2008). "Embedding projects in multiple contexts – a structuration perspective." *International Journal of Project Management* 26 (1), 30-37.
Meehan, B. and I. Richardson (2002). "Identification of software process knowledge management." *Software Process Improvement and Practice* 7 (2), 47–55.
Metiu, A. (2006). "Owning the Code: Status Closure in Distributed Groups." *Organization Science* 17 (4), 418-435.
Montgomery, K. and A. L. Oliver (2007). "A Fresh Look at How Professions Take Shape: Dual-Directed Networking Dynamics and Social Boundaries." *Organization Studies* 28, 1-27.
Newell, S., Tansley, C. and J. Huang (2004). "Social Capital and Knowledge Integration in an ERP Project Team: The Importance of Bridging AND Bonding." *British Journal of Management* 15, 43-57.
Nonaka, I. and I. Takeuchi (1995). *The Knowledge-Creating Organization*. Oxford: Oxford Press.
Packendorff, J. (1995). "Inquiring into the temporary organization: New directions for project management research." *Scandinavian Journal of Management* 11 (4), 319-333.
Putnam, R. (2000). *Bowling alone: The collapse and revival of American community*. New York: Simon and Shuster.
Reich, B. H. and S. Y. Wee (2006). "Searching for knowledge in the PMBOK® guide." *Project Management Journal* 37 (2), 11-25.
Reich, B. H., Gemino, A. and C. Sauer (2008). "Modeling the Knowledge Perspective of IT Projects." *Project Management Journal* 39, Supplement, S4–S14.
Reich, B. H. (2007). "Managing Knowledge and Learning in IT Projects: A Conceptual Framework and Guidelines for Practice." *Project Management Journal* 38 (2), 5-17.
Ross, J. W., Weill, P. and D. Robertson (2006). *Enterprise architecture as strategy: Creating a Foundation for Business Execution*. Boston, MA: Harvard Business Press.
Sabherwal, R. and Y. E. Chan (2001). "Alignment Between Business and IS Strategies: A Study of Prospectors, Analyzers, and Defenders." *Information Systems Research* 12 (1), 11-33.
Star, S. and J. Griesemer (1989). "Institutional Ecology, 'Translations', and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-1939." *Social Studies of Science* 19 (3), 387-420.
Thiétart, R. A. (1999). "Méthodes de recherche en management." Paris: Editions Dunod.
Toffolon, C. and S. Dakhli (2002). "The Software Engineering Global Model." In: *Proceedings of the COMPSAC'2002 Conference*, Oxford, United Kingdom. Los Alamitos, CA: IEEE Computer Society Press, (pp. 47-53)
Toffolon, C. (1996). "L'Incidence du Prototypage dans une Démarche d'Informatisation." Thèse de doctorat, Université de Paris-IX Dauphine, Paris.
Toffolon, C. (1999). "The Software Dimensions Theory." In: Joaquim Filipe (Ed.), *Enterprise Information Systems, Selected Papers Book*, KLUWER ACADEMIC PUBLISHERS.
Vinck, D. (2009). "De l'objet intermédiaire à l'objet-frontière: Vers la prise en compte du travail d'équipement." *Revue d'anthropologie des connaissances* 3 (1), 51-72.
Walsham, G. (2002). "Cross-Cultural Software Production and Use: A Structurational Analysis." *MIS Quarterly* 26 (4), 359-380.
Walz, D. B., Elam, J. J. and B. Curtis (1993). "Inside a software design team: Knowledge acquisition, sharing and integration." *Communications of the ACM* 36 (10), 63–77.
Williamson, O. E. (1981,). "The Modern Corporation: Origins, Evolution, Attributes." *Journal of Economic Literature* 19, 1537-1568.
Winter, S. and G. Szulanski (2001). "Replication as strategy." *Organization Science* 16, 730–743.
Yin, R. K. (1994). *Case Study Research: Design and Methods*. Beverly Hills, CA: Sage Publishing.