



HAL
open science

Open source software adoption for safety-critical information systems design: the Thales case study

Nordine Benkeltoum

► **To cite this version:**

Nordine Benkeltoum. Open source software adoption for safety-critical information systems design: the Thales case study. 2017, 10.3917/sim.164.0071 . hal-04191308

HAL Id: hal-04191308

<https://hal.science/hal-04191308v1>

Submitted on 30 Aug 2023

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

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

Open source software adoption for safety-critical information systems design: the Thales case study

*Nordine Benkeltoum**

*University of Lille Nord de France, Central School of Lille

Abstract

Information Systems (IS) literature shows a huge interest for open source technologies as well as from IT (Information Technology) strategies, business models or organization. Nevertheless, research on the application of open source in safety-critical information systems is particularly scant. By means of an exploratory study, this article describes the reasons why Thales, a firm specialized in safety-critical fields (defence, aircraft industry and security), has adopted open source software technologies. From a theoretical perspective, this article relies on the technology, organization, environment (TOE) framework and the literature on technology and innovation adoption. This research suggests a model of barriers and motivations to open source software adoption for safety-critical information systems design.

Keywords

Open source software, innovation adoption, mission-critical, safety-critical, TOE

Important information

The translation of this article (dedicated to non-French speakers) is largely based on automated translators (*Google Translate and DeepL*). All errors remain my own. For citation purpose, please refer to the French edited version published in *Système d'Information et Management* available here:

<https://aisel.aisnet.org/sim/vol21/iss4/4/>

Introduction

Open source software provides freedom to run, study, improve, and distribute. These technologies have become essential in the field of information systems (IS) whether in theory or in practice (Benkeltoum 2013; Poba-Nzaou *et al.* 2014). Described yesterday as a social movement (O'Mahony 2003; Von Krogh et Spaeth 2007), open source is nowadays more business oriented (Meissonier *et al.* 2010). In reality, for certain products, the community contribution is sometimes negligible or even non-existent. Companies are even mobilizing open source software as technological bricks that they combine for value creation. In practice, the publication of modifications is not systematic. Some companies source code parts that are considered as sensitive to their competitiveness and keep sensitive parts secret (Henkel 2006). Open source adoption within organizations is increasingly common regardless of the sector of activity. According to a recent study conducted on a sample of 1,300 companies, 78% of firms base all or part of their operations on open source components. Additionally, 55% of companies lack an overarching policy defining rules for adopting these technologies (Blackduck 2016). This observation is also supported by academic research. Indeed, the criteria on which organizations base themselves to adopt or reject open source are highly uncertain. With the massive diffusion of these technologies, specialists are pushing organizations to formalize criteria for adoption (Marsan *et al.* 2012). Thus, more work is needed to explain adoption in general and adoption by organizations in particular (Li *et al.* 2013; Poba-Nzaou *et al.* 2014).

The IS literature asserts that open source has characteristics that promote research (Von Krogh et Spaeth 2007). Studies have thus focused on development methods (Agerfalk et Fitzgerald 2008; Haefliger *et al.* 2008), organizational structures and their dynamics (Benkeltoum 2011a; Stuermer *et al.* 2009) or even business models (Bonaccorsi *et al.* 2006; Lisein *et al.* 2009; Mouakhar et Tellier 2013; Välimäki 2003). On the other hand, few studies have addressed the question of the use of open source for the design of *safety-critical information systems*. An IS is qualified as critical because of the sensitivity of the activities it supports and when a failure of the latter can endanger lives of human beings (Gary *et al.* 2011).

This research focuses on the elements that motivated Thales, a company specializing in the fields of defence, aeronautics and security, to adopt open source. Indeed, open source technologies have undeniable advantages for the design of safety-critical IS such as technological independence or improved responsiveness. In this article, open source adoption will be considered as a case of technological innovation adoption. As a result, research on innovation adoption (Rogers 1983) and technology adoption (Thong 1999) are mobilized.

The literature argues that organizations for which IT criticality is high and which have a large IT department tend not to adopt open source (Li *et al.* 2013). By applying this proposition to the Thales case, a paradox appears: why did a company evolving in a highly critical sector and having a large IT department adopt open source? Furthermore, Li *et al.* emphasize the need to highlight the success of the application of open source in government, military or financial organizations (Li *et al.* 2013). Thus, few works have focused on the use of open source in critical or sensitive areas requiring a high degree of reliability. This research will address the following question: *What are barriers and motivations to open source adoption in the field of safety-critical information systems?*

This article is structured as follows. The first section provides a state of knowledge on open source adoption. The second defines the concept of safety-critical IS and offers a literature review on the use of open source technologies for safety-critical IS. The third addresses research methodology. The fourth describes the barriers and motivations for open source adoption by

Thales. Finally, the last section returns to the contributions of this case study and discusses its theoretical and practical implications.

1. Open source adoption by organizations

Before developing the theoretical base of this research, it is necessary to justify the non-use of some popular frameworks in IS. The literature on the adoption of information technologies (IT) is based on different theoretical frameworks, including the organizing vision model and the absorptive capacity model. The model of the organizing vision is part of the neo-institutional current on managerial fads. This framework aims to analyse fads in the field of IT by focusing more particularly on managerial discourse. The purpose of the discourse is to facilitate the interpretation, legitimization and mobilization of innovation by adoption units (Carton *et al.* 2003; Swanson et Ramiller 1997). If this model is particularly suitable for assessing the adoption of an IT by different organizations and thus measuring imitation phenomena (Lesca *et al.* 2015; Swanson et Ramiller 1997), it seems less relevant to analyse the adoption of an innovation by a single organization. In addition, Thales is one of the pioneering companies in the open source adoption.

The absorptive capacity model describes the ability of a company to learn from its environment through a process of identification, assimilation and exploitation (Cohen et Levinthal 1989; Cohen et Levinthal 1990) aiming at value creation (Amabilé *et al.* 2012). However, this research aims to study the factors that motivated open source adoption by Thales. It is not a question of studying the role of absorptive capacity in the IT adoption process as has been widely done in the literature (Ben Youssef *et al.* 2015).

1.1. The adoption of innovation and technologies

Innovation refers to the introduction of a novelty or a significant improvement in terms of a product, service, process, distribution or organization (OECD/Eurostat 2005). Adoption designates the action of making a deliberate choice *vis-à-vis* a thing in order to appropriate it for a specific use¹. From an academic view, adoption consists in deciding to use an innovation because the individual (or the adoption unit) considers that it is the best alternative for the perimeter concerned by the adoption. Abundant IS research study the link between adoption of innovation and adoption of technology. The innovation itself is not necessarily new itself, but it must be perceived as such by the adopting unit (Thong 1999). Conversely, rejection refers to the non-use of the innovation (Rogers 1983). There are two types of rejection: *active rejection*, where the innovation has been used or tried before being rejected; *passive rejection*, where the innovation is rejected without implementation (Eveland 1979; Rogers 1983). Currently, there is no rejection technology theory. Nevertheless, the most frequently cited reasons for rejection are cost, lack of capability, and lack of market acceptance (Goode 2005).

Innovation adoption can be analysed on an individual as well as on an organizational level. It follows a process comprising five phases: (1) knowledge, the organization learns of the existence of the innovation and gathers information about it; (2) persuasion, it formulates a favourable or unfavourable attitude towards the innovation; (3) the decision, it engages in

¹The concept of adoption proposed is based on the entries “*adopt*” and “*adoption*” proposed by the CNRTL (National Center for Textual and Lexical Resources). This definition is however a proposal of the author.

activities aiming at adopting or rejecting the innovation; (4) implementation, it uses innovation; (5) confirmation, it confirms or denies the adoption (Rogers 1983).

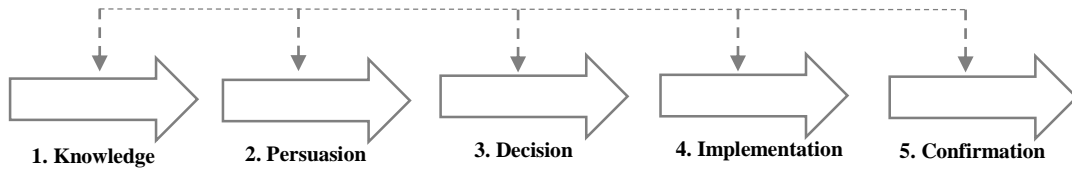


Figure 1: Adoption of innovation (diagram adapted from Rogers (1983))

In technology innovation, adoption can be summarized in three stages: (1) initiation, where the organization gathers and evaluates information about the technology, (2) adoption, where the decision about acceptance or rejection of technology is formulated and (3) implementation, where it is about integrating the technology into the organization (Thong 1999).

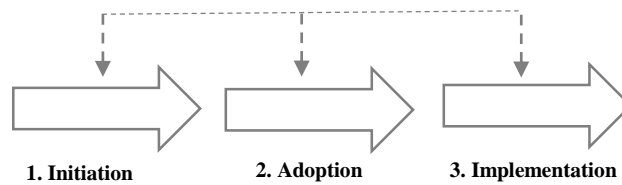


Figure 2: Technology Adoption

In the IS literature, the study of innovation adoption is often coupled with the TOE (Technology, Organization, Environment) framework (Huy *et al.* 2012) which refers to the technological, organizational and environmental context (Tornatsky *et al.* 1983). The technological context is concerned with the characteristics of the technologies held by the company and by the way in which these can influence adoption. The organizational context focuses on the characteristics of the organization in terms of size, structure or resources that may constrain or, on the contrary, facilitate adoption (Chau et Tam 1997). The environmental context considers competitors, suppliers, or relationship to states, which may also play a role in adoption (Zhu *et al.* 2003). The TOE framework has been widely used to analyze the factors of adoption of new technologies such as electronic data interchange (Iacovou *et al.* 1995), open systems (Chau et Tam 1997) or even e-business (Zhu *et al.* 2003). This theoretical framework offers a structured characterization of the adoption environment (Chau et Tam 1997). More recently, the TOE framework has been extended by two dimensions. The first dimension focuses on the characteristics of managers in terms of IT knowledge and attitude towards innovation (Huy *et al.* 2012). The second dimension takes into consideration the absorptive capacity of companies (Ben Youssef *et al.* 2015). The TOE framework will be used to classify the obstacles and motivations to open source adoption for the design of safety-critical IS.

1.2. Open source adoption by organizations

Although there is theoretical and practical controversy over the innovative character of open source software as a product (Benkeltoum 2013; Fitzgerald 2006; Fuggetta 2003), many academics consider open source to represent radical innovation (Bonaccorsi *et al.* 2006) or rupture (Rossi 2009; Spinellis et Giannikas 2012) which includes different technological

artefacts and families of technologies (Marsan *et al.* 2012). Adoption of these technologies is considered a case of technological innovation adoption (Li *et al.* 2011). It should be emphasized that IS innovations can have a limited impact while others can have a global impact on the organization (Chau et Tam 1997). In the case of open source technologies, the transformation caused can be variable (Grand *et al.* 2004).

Existing work on open source technology adoption (TA) falls into two streams. The first current focuses on the adoption or rejection of the individual point of view (Alexy *et al.* 2013; Li *et al.* 2011). For instance, this work suggests that identifying as open source community has a positive impact on adoption (Gwebu et Wang 2011). The second stream looks at adoption or rejection at the organizational level. Open source TA can be achieved to varying degrees. Thus, it is considered that a company can have four levels of resource allocation in order to exploit the opportunities offered by this software: (1) use, the company does not intervene in the development process of software; (2) offering free software as a complement; (3) adopting open source as a production method; (4) adaptation of the business model to the specificities of these technologies (Grand *et al.* 2004). Moreover, participation in communities, the fact of offering open source products or the use of development practices specific to open source, do not necessarily imply organizational adoption for internal use (Marsan *et al.* 2012). Since this research focuses on open source adoption from an organizational point of view, the literature review will focus on this point. Based on the TOE theoretical framework, it will be necessary to classify barriers and motivations to open source TA.

1.2.1. Motivations for adoption

Technological motivations. The main driver for open source adoption is the low cost of acquisition and ownership (Li *et al.* 2013; Poba-Nzaou *et al.* 2014; Spinellis et Giannikas 2012). In fact, such software has a negligible (Goode 2005; Macredie et Mijinyawa 2011) or even zero (Dedrick et West 2003) acquisition cost (Goode 2005; Macredie et Mijinyawa 2011). They are also adopted because of their high reliability (Dedrick et West 2003; Li *et al.* 2013; Qu *et al.* 2011), quality (Capra *et al.* 2011), stability (Spinellis et Giannikas 2012) and rapid bug fixing (Bitzer et Schröder 2005). Elements more related to the characteristics of software also play a role in terms of adoption, including: innovation (Dedrick et West 2003), compatibility (Macredie et Mijinyawa 2011), *trialability* (capacity of an innovation to be tested within a limited scope) (Dedrick et West 2003; Qu *et al.* 2011), improvement in the ability to adapt software, reduction in the risk of *lock-in* to a single proprietary provider (Li *et al.* 2013).

Organizational motivations. The literature has shown that the open source adoption can be motivated by the improvement of image and reputation (Bonaccorsi *et al.* 2006). Companies tend to adopt open source when they have technical skills (Dedrick et West 2003; Qu *et al.* 2011). The quality of human capital is therefore of central importance when it comes to adoption (Colombo *et al.* 2014; Li *et al.* 2013). Indeed, human skills improve the ability of companies to take advantage of these technologies (Harison et Koski 2010). The literature maintains that adoption is favoured when it is in line with the company's business (Qu *et al.* 2011) or when the organization considers that it can improve its performance (Macredie et Mijinyawa 2011; Marsan *et al.* 2012). Similarly, firms lacking financial resources tend to adopt more readily (Macredie et Mijinyawa 2011; Qu *et al.* 2011).

Environmental motivation. The existence of professional technical support promotes the adoption of free software (Dedrick et West 2003; Spinellis et Giannikas 2012). In fact, the most

popular software based on large communities is supported by service organizations. Conversely, less popular software sometimes suffers from lack of support (Marsan *et al.* 2012). From a cultural point of view, countries where the culture of individualism is developed, i.e. countries where the individual seeks his individual well-being rather than that of the community, tend to adopt open source (Qu *et al.* 2011). Second, a country's IT skill level also has a positive impact on adoption. The more people are proficient in IT, the more companies are able to adopt this software. Finally, countries with culturally accepting individuals adopt more easily (Qu *et al.* 2011). Furthermore, public policies can also have a positive impact on adoption (Deodhar *et al.* 2012). For example, the Gendarmerie migrated all of its workstations to the Openoffice.org office suite and in 2010 adopted a specific distribution called GendBuntu for nearly 85,000 machines.

1.2.2. Barriers to adoption

Technological barriers. Generally speaking, it is the lack of reliable information regarding the concept of open source that slows adoption in organizations (Marsan *et al.* 2012). Research indicates that software immaturity has a negative impact on adoption (Nagy *et al.* 2010). Indeed, study carried out by professional organizations shows that certain areas including office automation, workstations and business applications lack maturity (CIGREF 2011). Academic literature has confirmed this observation since these software are perceived as being more suited to technical than to business aspects (Marsan *et al.* 2012). In a sample of Swedish companies, most companies adopted infrastructure software (Lundell *et al.* 2010). Moreover, the fear that the source code of the adopted software will be used to create a derivative work (or *forking*) hinders adoption (Nagy *et al.* 2010). From a legal perspective, open source licenses lead to the creation of uncertainty regarding adoption. Even specialized lawyers have difficulty advising organizations in this area (Marsan *et al.* 2012). In addition, the fear of being unable to integrate these technologies into existing IS (Nagy *et al.* 2010) in particular because of problems related to non-compatibility with technologies from proprietary monopolies and the unavailability of drivers for some hardware (Macredie et Mijinyawa 2011) hinder adoption. Opponents to open source point out that such software can harm the organization due to its unreliability, security vulnerabilities that become identifiable, and higher cost of ownership than other types of software (Marsan *et al.* 2012). Additionally, due to breach, security, and maintenance risks, organizations with high information technology (IT) criticality will tend not to adopt open source. In addition, CIOs (Chief Information Officers) and IS managers consider that this software is not suitable for critical business applications. Organizations adopting open source have a lower level of IT criticality than non-adopters (Li *et al.* 2013).

Organizational barriers. Having worked with proprietary software (Qu *et al.* 2011) and previous investments made in closed software (Nagy *et al.* 2010) hinder adoption. Other works point out that companies reject open source mainly for the lack of relevance to the activity of the company and for the absence of business demand (Goode 2005). The lack of internal knowledge for the integration of these technologies also plays an important role in terms of barriers to adoption (Nagy *et al.* 2010) but also the lack of adequate resources (Macredie et Mijinyawa 2011). Finally, the more a company has a large IT department (Li *et al.* 2013) and the more it has a large IT budget, the less it will tend to adopt open source (Dedrick et West 2003).

Environmental barriers. Insufficient external technical support (Goode 2005; Li *et al.* 2013; Qu *et al.* 2011), the perception of uncertainty in software services and support are major barriers to adoption (Li *et al.* 2013). Uncertainty around services and support explains why CIOs consider open source inappropriate for *business applications* (Li *et al.* 2013). Governmental pressures for the adoption of closed source software have a negative effect on adoption (Qu *et al.* 2011). From a more general point of view, the more a company is established in a country with a high level of development, the more adoption is slowed down. Second, companies located in countries with regimes where the distance between those in power and citizen is high tend not to adopt open source (Qu *et al.* 2011). The table below summarizes the literature on the obstacles and motivations to open source adoption (Table 1).

	Motivations for adoption		Barriers to adoption	
	Explanations	References	Explanations	References
Technology				
Cost	Low cost of acquisition and ownership	(Li <i>et al.</i> 2013; Poba-Nzaou <i>et al.</i> 2014; Spinellis and Giannikas 2012)	Higher cost of ownership	(Marsan <i>et al.</i> 2012)
Reliability and quality	High reliability and stability	(Li <i>et al.</i> 2013; Qu <i>et al.</i> 2011; Spinellis and Giannikas 2012)	Lack of reliability	(Marsan <i>et al.</i> 2012)
	Quick bug fixes	(Bitzer and Schröder 2005)		
	Quality	(Capra <i>et al.</i> 2011)		
Forking			Fear of forking	(Nagy <i>et al.</i> 2010)
Information			Lack of reliable information	(Marsan <i>et al.</i> 2012)
Innovation	Innovation	(Dedrick and West 2003)		
Interoperability	Adaptability	(Li <i>et al.</i> 2013)		
	Compatibility	(Qu <i>et al.</i> 2011)		
			Inability to integrate	(Nagy <i>et al.</i> 2010)
Licenses			Licensing complexity	(Marsan <i>et al.</i> 2012)
Supplier lock-in	Supplier lock-in	(Li <i>et al.</i> 2013)		
Maturity			Immaturity	(Nagy <i>et al.</i> 2010)
Role of IT			Criticality of IT	(Li <i>et al.</i> 2013)
Security			Identifiable security vulnerabilities	(Marsan <i>et al.</i> 2012)
Trialability	Trialability	(Qu <i>et al.</i> 2011)		
Organization				
Activity	Fit with business	(Qu <i>et al.</i> 2011)	Low relevance to activity	(Goode 2005)
	Performance improvement	(Marsan <i>et al.</i> 2012)	Lack of business demand	(Goode 2005)
Proprietary software track record			Adoption of proprietary software	(Qu <i>et al.</i> 2011)
			Closed Software Investments	(Nagy <i>et al.</i> 2010)
Characteristics of the IT service			Size of IT service	(Li <i>et al.</i> 2013)
Internal skills	Adequate technical skills	(Qu <i>et al.</i> 2011)	Insufficient internal knowledge	(Nagy <i>et al.</i> 2010)
Financial resources	Low financial resources	(Qu <i>et al.</i> 2011)		
Environment				
Country characteristics	Acceptance of novelties	(Qu <i>et al.</i> 2011)		
	IT competence	(Qu <i>et al.</i> 2011)		
			High distance with power	(Qu <i>et al.</i> 2011)
			Culture of individualism	(Qu <i>et al.</i> 2011)
			Level of development	(Qu <i>et al.</i> 2011)
			State pressures for closed software	(Qu <i>et al.</i> 2011)
Support	Existence of support	(Spinellis and Giannikas 2012)	Insufficient external support	(Goode 2005; Li <i>et al.</i> 2013; Qu <i>et al.</i> 2011)
			Perception of uncertainty support and services	(Li <i>et al.</i> 2013)

Table 1: summary of barriers and motivations

2. Open source and safety-critical systems: state of knowledge

2.1. Safety-critical information system

An IS is said to be *safety-critical* because of the sensitivity of the activities it supports and when it performs operations for which a blockage or breakdown can have significant or even dramatic consequences such as the loss of life. Such an IS must therefore have a high degree of reliability and correctness (Gary *et al.* 2011), high fault tolerance, ensure important services in extreme conditions even if part of it is compromised due to an external attack (Min et Choi 2004). Safety-critical IS should be distinguished from *mission-critical applications* where the critical aspect designates the fact that this software supports business operations such as ERP (Enterprise Resource Planning) as opposed to infrastructure software (Poba-Nzaou *et al.* 2014).

It is important to differentiate between the notions of safety-critical IS, extreme situation and crisis. A management situation is qualified as *extreme* when it has three characteristics: scalability, uncertainty and risk (Bouty *et al.* 2011; Lebraty 2013; Lièvre et Gautier 2009). Scalability refers to the rapid, dynamic and discontinuous changes faced by actors. Uncertainty relates to the discontinuity and rupture of the observed changes. Risk refers to the significant probability of harm to the actors involved. In an extreme situation, the potential effect of the risk is critical but the actors have the impression of mastering it (Lebraty 2013). A *crisis* is an unstable or even dangerous situation triggered by an event. Three elements distinguish the extreme situation from the crisis: the generation of ambiguous situations, the multiple consequences and the disruption of existing frames of reference (Evrard Samuel et Ruel 2013).

The notions of extreme situation and crisis relate to the characteristics of a management situation, while the notion of safety-critical IS describes the characteristics of an IS used in a management situation, whether standard, extreme or in crisis. It can be emphasized that the failure of a safety-critical IS due to the occurrence of an event can lead to the creation of a crisis (Lebraty 2013). Furthermore, extreme situations increase the likelihood of crises (Arena *et al.* 2013). Safety-critical IS is one of the means to prevent a situation from deteriorating and becoming unstable or even dangerous. Safety-critical IS are found in various sensitive areas such as air traffic management, railway switching, financial information (Arena *et al.* 2013), space missions (Norris 2004), surgery (Gary *et al.* 2011) or the defence (Godé *et al.* 2012).

In the defence field, a *safety-critical system* or *mission-critical computer system* is defined as a set of equipment, software and services whose operation involves: (1) intelligence activities, (2) data relating to national security, (3) the control of military resources, (4) equipment integrated in whole or in part in an armament device or which is (5) critical for the performance of a military activity or for intelligence (Bloom et Chung 2001). For example, Liaison 16, the network-centric decision support system embedded in some French military aircraft, is a typical case of safety-critical IS since it is on the latter that aerial bombardment decisions are based (Godé *et al.* 2012). Since the beginning of the 1980s, the incorporation of computer code into military systems has experienced dizzying growth (Austin et Larkey 1992) which means that the issue of IS security has received greater interest. Similarly, in the aeronautical sector, the rate of equipment that requires IT to function increased from 20 to 80% between 1980 and 2000 (Arena *et al.* 2013). In this area, safety-critical IS are governed by strict standards. For instance, the DO178C standard distinguishes five levels of criticality for an on-board component in the field of avionics ranging from level A (a failure can jeopardize flight safety) to level E (a defect will have no effect on flight safety) (Ferrell et Ferrell 2001). Development constraints (documentation, traceability, etc.) are associated with each level of criticality. More generally, the safety of people relies on this type of system (Gary *et al.* 2011).

2.2. Open source at the service of safety-critical information systems

The massive use of open source in business (IDC 2009) has shown that there are significant disparities in the level of maturity depending on the domain. While development, administration, internet technologies, web or application servers have a high degree of maturity, other areas such as business applications and security are much less mature (CIGREF 2011). In addition, the use of open source for safety-critical IS is the subject of little work even if in practice this phenomenon is known (Lawton 2002; Lundell *et al.* 2010) and is far from negligible (Gartner Inc. 2011).

The application of open source in such systems comes up against major obstacles. First, because of the opening of the source code, some consider that the IS is more subject to the identification of security flaws or *backdoors*. It is ultimately the integration of unknown components that poses the most problem for security professionals and not the fact that the software is open source (Lawton 2002). Second, releasing sensitive application code is hardly acceptable to practitioners or organizations such as NASA (Norris 2004). In these structures, *security by obscurity*, i.e. keeping the source code of a program secret in order to protect an IS, is an integral part of the culture of these organizations (Hansen *et al.* 2002). And this despite the significant limitations highlighted by various specialists. Many security vulnerabilities are discovered in closed software. For example, security flaws discovered on the iPhone made it possible to send data without the knowledge of the user (Le Boulout 2011). In addition, *reverse-engineering* or retro-engineering (moving from object code to source code) is a common practice among computer specialists, which considerably limits the scope of source code secrecy practices (Hansen *et al.* 2002).

2.3. Cases of safety-critical information systems from the literature

A study on open source adoption within the American defence (DoD) illustrates well the enthusiasm existing within public administrations. For example, the National Security Agency (NSA) has designed its own version of Linux called Security Enhanced Linux (SEL) which aims to support critical applications. Similarly, the US NAVY has adopted open source components to build applications and has implemented a policy aimed at eliminating any proprietary technology (Le Texier et Versailles 2009).

Feedback on the American *Mars Exploration Rover mission* explicitly addresses the subject. Indeed, the *Science Activity Planner* (SAP), a statement analysis tool, is based on open source components such as Castor, Java Expression Parser, VRML97 or the best known MySQL. NASA classifies SAP as a critical application because a malfunction in this program could compromise part of the operations. SAP was clearly designed on the basis of open source technologies to achieve development savings. Thanks to specific agreements with the authors of the open source components, NASA managed not to publish the SAP source code. According to the SAP project manager, the open source components had a higher level of quality than their non-free counterpart. Several elements should guide the choice of components to integrate: the maturity of the software (stage of development), the longevity (size of the team) and the flexibility of the project (reactivity of the team) (Norris 2004) to which one can add scalability (ability of the component to respond to increasing stress) (Sarrab et Rehman 2014).

Another feedback combines open source and agile methods for medical image-guided surgery software. This software is distributed under a permissive license which means that it can be modified, linked or simply integrated with free or closed source code. The objective of this software is to serve as a technological base for the creation of applications for surgery. It is indeed a safety-critical IS for the safety of people or *safety-critical* (Gary *et al.* 2011). At the

European level, a consortium of companies (Airbus, Thales, Turbomeca, etc.) coordinates the development of *Topcased*, a set of tools intended to be used for safety-critical IS and in particular in the air domain.

3. Research problem and methodology

3.1. Positioning in the adoption process

Drawing on the relevant literature regarding technology adoption and the TOE theoretical framework, this study examines the factors that account for open source adoption by Thales. The diagram presented below specifies the phases of the adoption process investigated in this research (Figure 3).

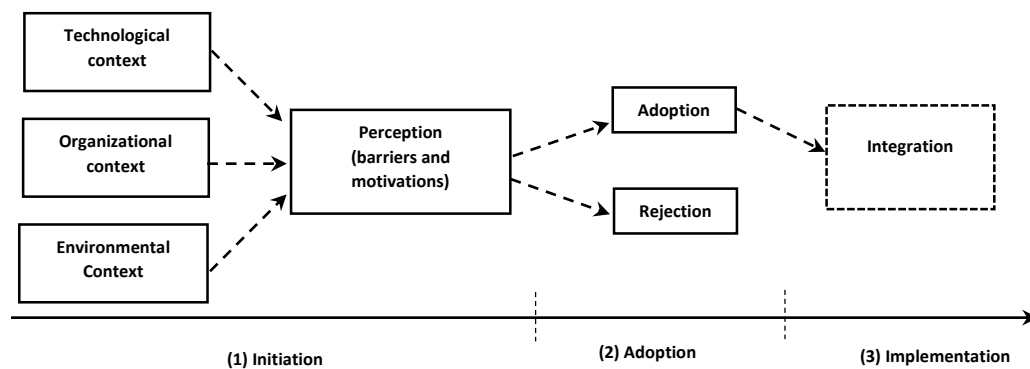


Figure 3: positioning in the adoption process

The adoption process is divided into three stages (Thong 1999). First, the initiation which aims to study the perception of the technological, organizational and environmental contexts (TOE framework) (Tornatsky *et al.* 1983) which condition the barriers and motivations to adoption. In the case of Thales, it will focus on the essential elements of safety-critical IS. For the technological aspect, we will consider: security, reliability, correction of malfunctions, tolerance to failure and external attacks (Gary *et al.* 2011; Lawton 2002; Min et Choi 2004). For the organizational aspect, we will analyse the sensitivity of the activities (Bloom et Chung 2001) and the reactivity. For the environmental aspect, we will focus on government pressures and client organizations (Le Texier et Versailles 2009; Norris 2004). Then, it will be a question of studying the elements related to the decision to adopt open source. On the other hand, no observations were made during the decision-making process. Therefore, this research will focus on the main elements that influenced adoption. Finally, the implementation phase which will not be studied because it does not concern the elements that motivated the adoption but its use within the adoption unit.

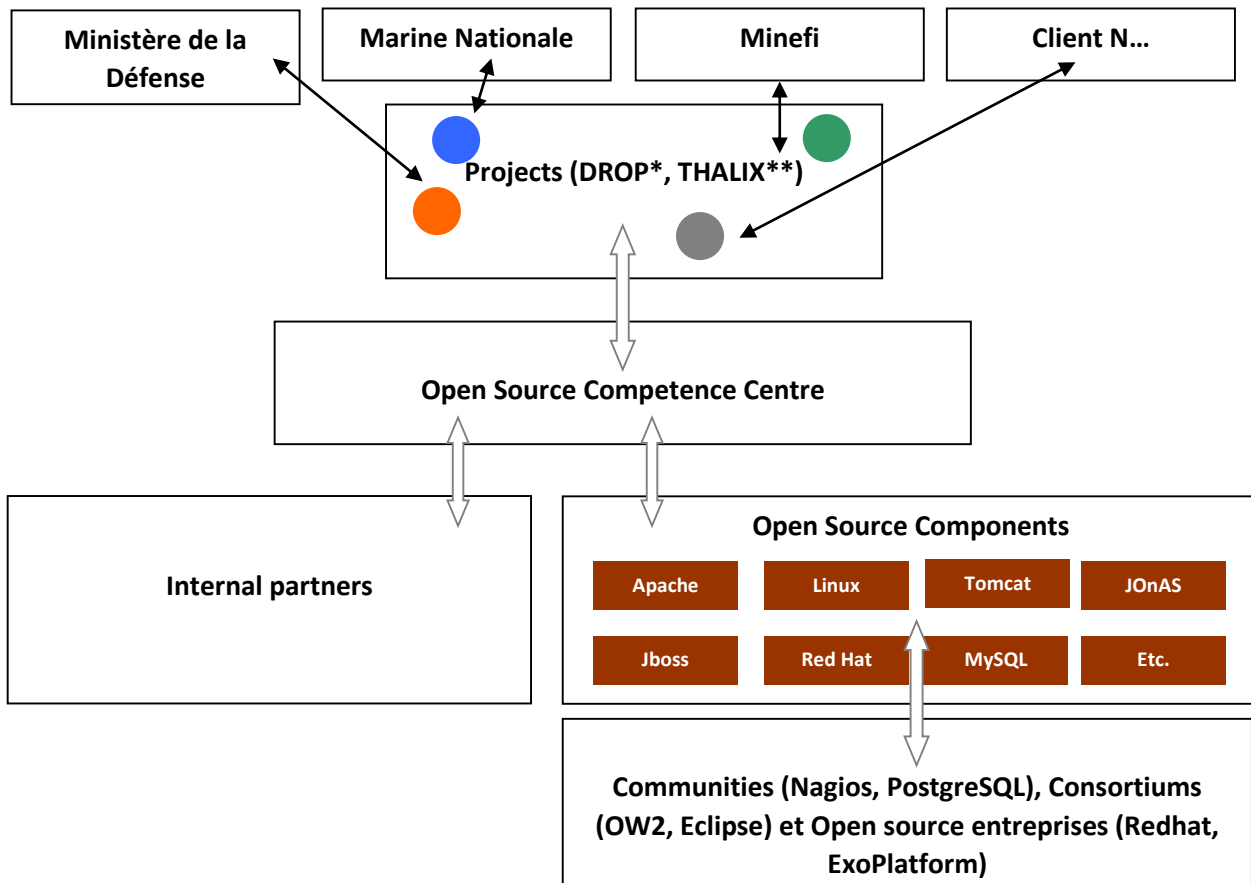
3.2. Presentation of the case

3.2.1. General presentation

Thales is an electronics and systems group that specializes in defence, aeronautics, and security. With a global workforce of approximately 70,000 employees, more than a third of whom are researchers, Thales invests almost a fifth of its activity in research and development annually, reflecting the firm's expertise in high technologies. Thales collaborates with academic institutions and shares multiple research laboratories. For example, Albert Fert, Director of the joint CNRS-Thales physics research unit, was awarded the Nobel Prize in Physics in 2007 for his research on magnetoresistance.

3.2.2. The competence centre

In order to ensure open source adoption, Thales created an *Open Source Competence Centre in 2005* responsible for developing the necessary knowledge and getting involved in the various communities. It was initiated in a context of research and development and responded to an anticipation of the evolution of development methods: from a closed logic to an open logic applied to the field of IS and services. Its role is to explore the field of open source and to build a network of strategic partnerships with companies (based on the words of Interviewee 3, Thales). Thales D3S is the most advanced part of the group in terms of open source. Indeed, this division cooperates with all the other divisions to promote open source technologies. The centre serves as an interface between projects commissioned by customers (National Navy, Ministries, etc.) and open source components developed within the framework of communities or industrial consortia. The diagram au-dessous offers a simplified vision of the role of the centre (diagram adapted from official Thales documents).



*DROP is a dematerialized system project for spare parts requests (radars, missiles, etc.) designed from free components for the French Navy.

**THALIX is an air traffic management platform based on open source components.

Figure 4: The role of the Open Source Competence Center

The centre ensures the sustainability of the software architecture and applications. In other words, Thales creates a service-oriented architecture where open source components can evolve or even be replaced. This transformation is essentially conditioned by technology and customer needs. In the IT field, technology and standards evolve very quickly, which requires extensive technical flexibility. Similarly, the architecture can evolve according to business and customer needs without calling into question the operation of the system.

3.3. Methodological approach

3.3.1. Exploratory study based on a case

Several elements justify the use of the exploratory case study. First, this research tries to understand the *why* of a phenomenon (Leonard-Barton 1990). Then, the paucity of work on the use of open source for the design of critical or sensitive IS, the emerging nature of this type of practice, further justify the use of this method. Indeed, the case study can be used to describe, test a theory or generate a theory. This research strategy is based on the selection of one or more cases and numerous levels of analysis (Eisenhardt 1989). This or these cases must particularly be characteristic of a phenomenon to be observed (Leonard-Barton 1990).

Some authors even claim that the validity of theories based on a limited number of cases have the same validity as those based on a large sample (Christensen 2006; Fitzgerald 1997). It can also be said that the case study is now a unanimously recognized methodology (Eisenhardt

1989; Eisenhardt et Graebner 2007). Case studies combine data that can vary in nature (qualitative, quantitative) and according to their provenance (archives, interviews, etc.) (Eisenhardt 1989; Leonard-Barton 1990).

3.3.2. Data source

Thales Group was studied as part of a national research project to analyse the use of open source in business. More specifically, the author had the opportunity to witness the emergence and development of an open source skills centre within the Thales Division Services (Thales D3S) division between 2006 and 2009, which corresponds to a longitudinal case study (Pettigrew 1990).

Thales has been analysed through several data sources (Eisenhardt 1989; Pettigrew 1990). The first source corresponds to semi-structured interviews and meetings (duration ranging from one to two hours) face to face, by Internet or telephone carried out with key players (Pettigrew 1990 (Pettigrew 1990)). These interlocutors held the following positions: Director of Research and Innovation, Director of Research and Technology Asia, Manager of the open source skills centre, Open source Architect, Open source Engineer. Since open source is an extreme case of *open innovation* (Dahlander et Wallin 2006), it was relevant to extend the analysis outside the firm (Chesbrough 2003) in order to enrich the analysis with partners. These data were transcribed manually. The second source was based on informal exchanges within the framework of seminars. Among the latter is participation in an international conference dedicated to open source for three days with a European project manager. The third source corresponds to internal documents, press releases, public information or even oral or written presentations.

The fourth corresponds to email exchanges (about 200) on specific points. This last source was particularly rich. These data sources are summarized below (Table 2).

	Interviews	Emails	Conferences	Internal and official documentation
Thales Group	7	X	2	X
Partners	4	X	1	X

Table 2 : data sources

3.3.3. Data analysis

The methodological approach of this research is a longitudinal case study (Pettigrew 1990) where the adoption of Thales open source was analysed. The interpretation of qualitative data was based on inductive reasoning and more particularly on *Grounded Theory* (Glaser et Strauss 1967). In this analytical framework, the data codification process is based on a dynamic relationship between data and interpretation. To do this, data coding software (Nvivo) was used. The coding procedure was broken down into three steps (Figure 5).

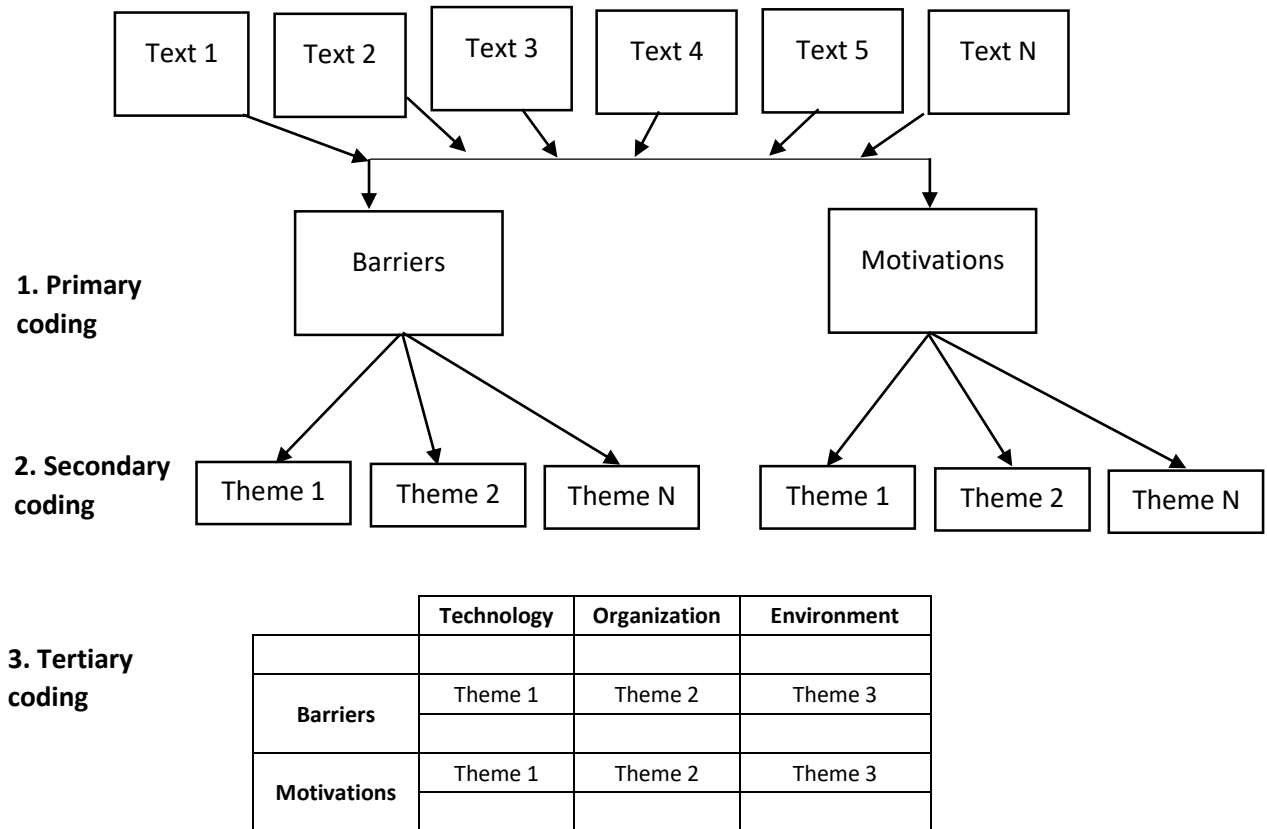


Figure 5: Coding procedure

Primary coding. From the various texts and based on the literature on adoption, the identification of the obstacles and motivations of the Thales group was carried out. To do this, a coding by lexical field was chosen. The first lexical field, dedicated to obstacles to adoption, grouped together all the elements mentioning:

- risks: *“Risk assessment, liability”* ;
- doubts: *“There is also a question of technology quality of projects. »* ;
- negative or pejorative expressions and perceptions: *“Free licenses were created by technicians, alter-globalists. »*

The second lexical field, centred on motivations for adoption, brought together expressions referring to:

- perceived benefits: *“Transfer of innovation from the Open Source world. »* ;
- strategic intentions: *“It is therefore a question of a desire for technological independence. »*

Secondary coding. On the basis of the first coding making it possible to make the dichotomy between the elements slowing down the adoption on the one hand and the elements motivating the adoption on the other hand, a second coding was operated. The latter aimed to identify themes maximizing the phenomena of reversal, extension and replication (Eisenhardt et Graebner 2007).

Tertiary coding. The last coding phase is based on the one hand on the data previously classified as brakes and motivations (primary coding) and themes (secondary coding) and on the other hand on the TOE analysis framework taken from the literature.

4. The obstacles and motivations to open source adoption in the field of safety-critical IS

If open source has aroused many fears inside and outside (customers, partners) of Thales, these technologies have also presented many interests for the design of safety-critical IS. The table below summarizes the results of the data coding procedure and provides a summary of the main themes grouping the obstacles and motivations that motivated the adoption of open technologies within Thales (**Table 3**).

	Technology	Organization	Environment
Brakes	Quality	Business models	Support
	Code exposure		
	Interoperability and integration		
	Lack of security		
	Compatibility		
	Immaturity		
	Licenses		
	Responsibility		
Motivations	Innovation	Associated services	Reactivity
	Technological independence		
	Cost reduction		

Table 3: summary of the themes of the coding procedure

The thematic classification grouping the obstacles and motivations to open source adoption within Thales clearly shows an imbalance of concerns. Whether on the side of the brakes or whether on the side of the motivations, the main concerns of the group are at the technological level. Organizational and environmental aspects play a much less important role. Therefore, the first two propositions follow:

Proposition 1a: In the field of critical information systems, the main obstacles to open source adoption are at the technological level.

Proposition 1b: In the field of critical information systems, the main motivations for adopting open source are at the technological level.

By comparing the obstacles and motivations from the literature and those identified in the data analysis via the TOE framework, four types of findings can be made: 1) confirmation of proposals from the literature, 2) extension of proposals general to the field of safety-critical IS, 3) the contradiction of the literature and finally 4) the suggestion of new proposals specific to safety-critical IS. The table below provides a summary of the theory comparison with the Thales case and highlights the contributions (**Table 4**).

		T	VS	Contributions	Thales case	Proposals	
Category	Technology						
Cost	Low cost of acquisition and ownership	+/-	+	Confirmation	" Cost containment. »		
Reliability and quality	High reliability and stability	+	+	Extension	"Open Source components are now mature enough, stable and robust to be implemented in any mission critical application. »	4	Some open source components are sufficiently reliable, stable and mature to be integrated into safety-critical IS.
	Quality	+	-	Contradiction	"That poses a problem for manufacturers, namely: there are really very good apps but there is no support, there is no quality, we don't know what it's worth. What happens if in three years we stop working on it. »	6	In the field of safety-critical IS, the quality required for open source adoption software is higher than the average quality expected for professional (non-critical) use.
	Quick bug fixes	+	+	Extension	"So we participate in the community to be much more responsive. »	5	In the field of safety-critical IS, open source adoption aims to improve responsiveness.
Innovation	Innovation	+	+	Confirmation	"Detection of new technologies and uses. »		
Interoperability	Compatibility	+	-	Contradiction	"How should an Open Source component be integrated, modified and distributed?" »	8	In the field of safety-critical IS, concerns related to interoperability and integration are above the average expected for professional (non-critical) use.
	Adaptability	+	+	Confirmation	"Flexibility of architecture and components. »		
Licenses	Licensing complexity	-	-	Extension	"One of the obstacles to the promotion and propagation of OSS components stems from the fears generated by the profusion of free licenses. »	2	In the field of safety-critical IS, licensing issues are a major barrier to adoption.
Technological independence	Technological independence		+	Suggestion	"Open Source is seen as the way to guarantee Europe's software independence and therefore its software security. »	12	In the field of safety-critical IS, open source is a means of regaining technological independence.
Maturity	Immaturity	-	-	Extension	"For security, open source is mature but with a small downside to the database and security components. authentication, identification, etc. »	3	The lack of maturity of certain security components in open source hinders adoption in the field of safety-critical IS
Responsibility	Responsibility		-	Suggestion	"Risk assessment, liability. »	10	In the field of safety-critical IS, the issues related to liability in the event of failure is greater than in other fields, which constitutes a brake on adoption.
Role of IT	Criticality of IT	-	+	Contradiction	"PoSSIS is a suite of 40+ Open Source Components leveraging an SOA Platform for Critical Information Systems. »	7	In the field of safety-critical IS, some open source components are perceived as capable of supporting sensitive operations.
Code exposure	Code exposure		-	Suggestion	"There was a fear: exposing certain codes. »	11	In the field of safety-critical IS, the fear of exposure of the source code of certain sensitive components is holding back adoption.
Organization							
Activity	business uncertainty		-	Suggestion	"Thales wants to do open source but what does it pay? »	9	For some industries known to be users of closed software, open source presents uncertainty in the business model, which hinders adoption.
Environment							
Support	Insufficient external support	-	-	Confirmation	"The customer wants quality of service regardless of the integrated system. He is entitled to demand the same quality of service that a software publisher can provide him. In the world of publishers, when you have a contract with IBM and you have a problem, you can turn to IBM. But when it's a community it's different! Who to turn to? »		

Table 4: confrontation theory (T) / Thales case (C) / contributions (A)

4.1. Confirmation of proposals from the literature

The literature argues that the cost of acquisition and ownership is the main factor driving adoption (Li *et al.* 2013; Poba-Nzaou *et al.* 2014; Spinellis et Giannikas 2012). Other authors claim that the cost of ownership is also one of the major obstacles (Marsan *et al.* 2012). In the case study, the issue of cost reduction is obviously present.

A distinction should be made between the cost of acquisition, which favors open source software because of its low cost (Goode 2005) and the cost of ownership (Fitzgerald 2006), which takes into consideration other factors such as: training, adaptation to needs, costs related to data migration, etc. We can however explain this phenomenon by asserting that the cost of adoption (acquisition and ownership) of open source software is not uniform in organizations. When a company has adequate technical skills internally (Qu *et al.* 2011), the cost is lower than in organizations with insufficient internal knowledge (Nagy *et al.* 2010).

The adaptability of components (Li *et al.* 2013) and taking advantage of innovation (Dedrick et West 2003) are also among the reasons for adopting open source.

“Transfer of innovation from the Open Source world. (Interviewee 5, Thales)

From the point of view of obstacles, the lack of external support (Goode 2005; Li *et al.* 2013; Qu *et al.* 2011) also constitutes an obstacle to adoption within Thales.

“In the world of publishers, when you have a contract with IBM and you have a problem, you can turn to IBM. But when it's a community it's different! Who to turn to? (Interviewee 4, Thales)

4.2. The extension of general proposals to the field of safety-critical IS

The literature points out that the complexity of licenses (Marsan *et al.* 2012) is a major obstacle to adoption. Within Thales, licensing concerns relate not only to the complexity of licensing but also to other related factors. Open source and hybrid licenses are particularly numerous and complex. They required the development of specific skills in order to create IS respecting licensing rules and at the same time protecting portions of code with high added value.

The open source licenses have also raised internal reservations about their applicability in French and European law. Indeed, the fact that free licenses are written in a foreign language poses legal problems. Furthermore, some licenses such as the General Public License (GPL) have a power of *contamination* that is poorly understood. For Thales, the issues related to licenses go well beyond the observation of complexity. Since the group raises items related to applicability and code contamination. Therefore, the following proposition emerges from the case study.

Proposition 2: In the field of safety-critical IS, licensing issues are a major obstacle to adoption.

Some works defend the idea that the immaturity of components hinders adoption (Nagy *et al.* 2010). In the case study, the lack of maturity is indicated for the area of security.

Hence the following proposition.

Proposition 3: The lack of maturity of certain security components in open source hinders adoption in the field of safety-critical IS.

The literature further highlights that adoption is driven by the high reliability and stability of open source components (Li *et al.* 2013; Qu *et al.* 2011; Spinellis et Giannikas 2012). These

qualities are also seen as vectors of adoption for Thales. However, an important nuance must be made. Not all open source components have the same reliability and stability. Thales selected the components according to criteria related to safety-critical IS.

Hence the following proposition.

Proposition 4: Some open source components are sufficiently reliable, stable and mature to be integrated into safety-critical IS.

Moreover, the literature indicates that the rapid correction of bugs (Bitzer et Schröder 2005) constitutes an element motivating adoption. Within Thales, this point is mentioned, it is even a factor that makes it possible to improve the company's responsiveness.

In addition, a blocking bug affecting a safety-critical IS must be corrected as quickly as possible due to the criticality of the operations supported. The community model has much greater bug-fixing capabilities.

“Minefi² had a blocking bug on its tax payment site with Jboss³. The problem was solved in 9 hours! There was an official version that took the patch in 9 hours, it's unheard of. Even the big publishers don't have this responsiveness. (Interviewee 1, Thales)

Hence the proposal below.

Proposition 5: In the field of safety-critical IS, open source adoption aims to improve responsiveness.

4.3. The contradiction of literature

By comparing the obstacles and motivations to adoption in the literature and the results of the Thales case, strong contradictions emerge. In terms of technology, open source software is adopted for its quality (Capra *et al.* 2011) but it is also rejected for its lack of reliability and security (Marsan *et al.* 2012). This contradiction is also found in the data of the case since the lack of quality is mentioned as an element leading to rejection.

Therefore, the following proposition can be realized.

Proposition 6: In the field of safety-critical IS, the quality necessary for open source adoption software is higher than the average quality expected for professional (non-critical) use.

Some authors point out that the criticality of IT is negatively correlated with open source adoption (Li *et al.* 2013). The Thales case study shows exactly the opposite phenomenon. Since Thales operates in areas where IT is highly critical (air traffic management, defence, security). The company has selected components with a view to their integration into safety-critical IS.

Hence the proposal below.

Proposition 7: In the field of safety-critical IS, some open source components are perceived as capable of supporting sensitive operations.

Furthermore, open source software is mentioned for its compatibility (Qu *et al.* 2011). Yet in the case of Thales, the conditions of interoperability and integration raised questions.

Therefore, the following proposition is suggested.

²Ministry of Economy and Finance

³Jboss is an open source application server.

Proposition 8: In the field of safety-critical IS, concerns related to interoperability and integration are above the average expected for professional (non-critical) use, which hinders open source adoption software.

4.4. The suggestion of new proposals specific to safety-critical IS

Some obstacles and motivations emanating from data analysis do not appear in the literature. First, the question of the valuation of open source software has raised fears.

“Thales wants to do open source but what does it pay? (Interviewee 1, Thales)

The proposition below stems from this observation.

Proposition 9: For certain sectors reputed to be users of closed software (such as safety-critical IS), open source presents uncertainty at the level of the business model, which hinders adoption.

Liability issues also posed problems. For example, Thales experts pointed out that the *GPL* contains a clause that states that the distributor cannot be held liable for any damage caused by the software. However, in French law this clause is void because it is not possible to release one's responsibility by means of a clause (contrary to US law): whoever distributes the software is legally responsible for it. In addition, the question of remote work and the associated responsibility raised questions.

“Is it possible for us to donate code in the name of Thales? For the technicians, it is the problem of the ecosystem, if there is no one who gives the ecosystem will not be able to continue. But if we reverse the code in the name of Thales, this means that the responsibility of Thales is engaged. In the end, there was no opposition, neither from the "Corp" side nor from the "Division" side. (Interviewee 1, Thales)

Hence the proposal below.

Proposition 10: In the field of safety-critical IS, the issues related to liability in the event of a failure is greater than in other fields, which constitutes an obstacle to adoption.

Exposing the code of certain *sensitive* components is mentioned among the obstacles to adoption. This is in opposition to certain elements motivating adoption, such as improving the company's responsiveness in terms of bug resolution. This exposure raised a number of fears for Thales's principals, the Ministry of Defence in particular.

Hence the following proposition.

Proposition 11: In the field of safety-critical IS, the fear of exposure of the source code of certain sensitive components hinders adoption.

Ideas about the lack of security of open source components have also been debated.

“Security issues have not been addressed by the communities because they conflict with the philosophy of open source, the spirit of sharing. (Interviewee 1, Thales)

In the design of safety-critical IS, the exposure of *sensitive code* is a real fear because notions such as *security through obscurity* are an integral part of the culture of this type of organization (Hansen *et al.* 2002).

For Thales, open source adoption is primarily motivated by the acquisition of technological independence. For example, in the field of air traffic management, the company initially worked with editors who controlled the installed IS.

“Historically, the servers used before the arrival of these machines under Thalix, were “Stratus”, whose hardware and operating system were controlled by a single supplier. This results in the problems of available skills, of responsiveness of the single hotline, in particular in the face of hardware breakdowns of which almost no one had experience. (Interviewee 3, Thales)

In terms of security, some Thales experts consider that closed software presents problems related to the presence of security flaws (voluntary or not). In the field of defence, the integration of therefore unknown compiled components is no longer accepted. Thales is therefore seeking to control all the components that go into the design of safety-critical IS, which until now were under the control of companies controlled by foreign powers. These technologies constitute a credible alternative in order to regain the technological independence of the company.

“In the proprietary environment there was an American dependency. This dependence is technological, it's not for “cocorico”...” (Interviewee 4, Thales)

Hence the following proposition.

Proposition 12: In the field of safety-critical IS, open source is a means of regaining technological independence

Based on the proposals resulting from the confrontation between the literature and the data from the Thales case, a modelling of the main obstacles and motivations to open source adoption for the design of safety-critical IS is proposed (Figure Figure 6.

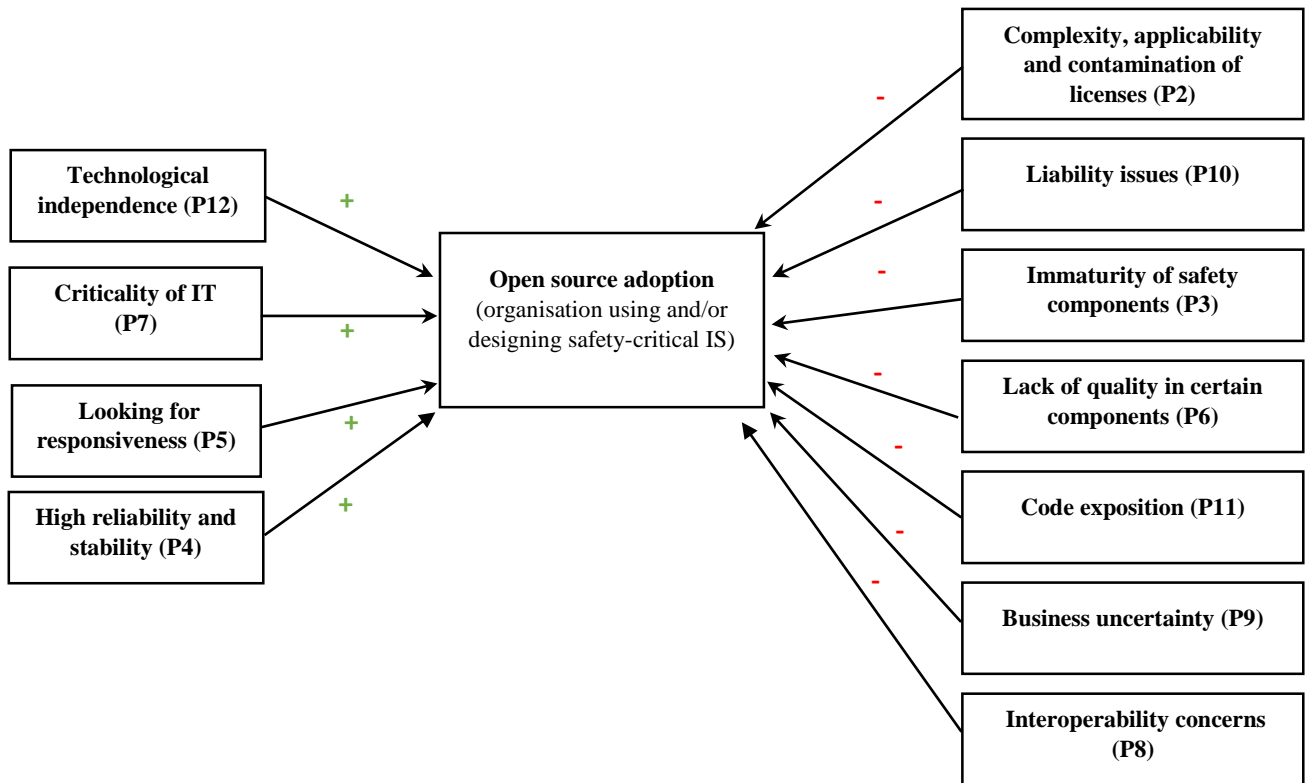


Figure 6: modelling of the barriers and motivations to open source adoption for safety-critical IS

Conclusion and implications

The literature has undeniably documented the different facets of open source and its application in business. She therefore distinguished between open (Dahlander 2005; Jullien 2003), hybrid (Bonaccorsi *et al.* 2006; Mouakhar et Tellier 2013; Välimäki 2003)and closed (Benkeltoum 2011b; Lisein *et al.* 2009); describes models of intervention in communities (Dahlander et Magnusson 2008; Dahlander et Magnusson 2005) and allocation of resources to take advantage of open source (Grand *et al.* 2004); drawn up the advantages and disadvantages of these technologies in terms of structure (Capra *et al.* 2011), modularity (MacCormack *et al.* 2006) or even reactivity (Paulson *et al.* 2004)without, however, explaining the reasons pushing companies designing SI critical to adopting or rejecting open source.

The literature has also pointed out that companies that have adopted proprietary software and made high investments in it tend to reject open source (Nagy *et al.* 2010; Qu *et al.* 2011). However, Thales had adopted many closed components and had developed a large number of partnerships with publishers, which did not prevent the group from adopting these technologies. The literature has mainly studied open source adoption for infrastructure software (Lundell *et al.* 2010)and very often in the organizations studied, adoption is a tactical and not a strategic choice (Spinellis et Giannikas 2012). Tactical adoption only aims to improve a short-term element such as costs or image. While strategic adoption requires the organization to transform its technology strategies (transformation of component development and selection practices),

business (creation and capture of value, change in terms of partnership, involvement and risk sharing with communities and consortia) and organizational (deployment of internal forging, development of specific skills). The Thales case shows how open source adoption can be integrated into a *strategic intention*. For Thales, the search for technological independence is the main driving force behind open source adoption for the design of safety-critical IS. Therefore, we can consider that organizations are divided into two categories: those who consider that open source is not relevant to the activity and / or trades of the company (Goode 2005) and another category of companies that see open source as a way to improve performance (Marsan *et al.* 2012) and business (Qu *et al.* 2011).

This research contributes to the literature on open source adoption for companies designing safety-critical IS. It describes more specifically the reasons why a company operating in demanding fields in terms of quality, stability and security, has changed its development perspective by integrating free technologies and by sharing technologies that are not critical to its competitiveness. A Thales partner points out that:

"Open-source has an innate power to transform even the most closed and the most policed organizations: once it has entered, things change, and it gets out of control (the remark applies to Thalès ... as for the whole of China) . (Experlog, OW2 Partner)

The Thales case proves that security is not necessarily a question of *security through obscurity* (Hansen *et al.* 2002) *black box* type software whose content is unknown. In this field, open source adoption components is mainly explained by elements related to technology, including the search for technological independence, technological criticality or the improvement of technical responsiveness. In the same way, adoption comes up against mainly technological obstacles, including the immaturity of certain security components or the fear of exposure of certain portions of source code.

This research responds to the request of some practitioners (Blackduck 2016) and researchers who mentioned the need to study the factors explaining open source adoption in general (Marsan *et al.* 2012; Poba-Nzaou *et al.* 2014) and in governmental or military organizations in particular (Li *et al.* 2013). It offers a set of specific proposals for companies using and/or designing safety-critical IS. However, the validity and robustness of these propositions need to be tested through a multiple case study (Eisenhardt et Graebner 2007; Leonard-Barton 1990) or even the validation of these criteria via a sample most important companies. Finally, the type of activity is not really taken into account as a variable to explain the adoption or rejection of open source. However, it is obvious that certain obstacles and motivations seem specific to sectors of activity, as illustrated by the Thales case. Therefore, it is necessary to study adoption and rejection by looking more closely at the influence of activity on barriers and motivations to adoption.

References

- Agerfalk, P. J. & Fitzgerald, B. (2008) "Outsourcing to an Unknown Workforce: Exploring Opensourcing as a Global Sourcing Strategy." *Management Information System Quarterly*, Vol. 32, n°2: p. 385-409.
- Alexy, O., Henkel, J. & Wallin, M. W. (2013) "From Closed to Open: Job Role Changes, Individual Predispositions, and the Adoption of Commercial Open Source Software Development." *Research Policy*, Vol. 42, n°8: p. 1325-1340.
- Amabilé, S., Meissonier, R., Haller, C. & Boudrandi, S. (2012) "Capacité d'absorption des informations et pratiques de veille stratégique dans les PME: une étude sur des domaines vitivinicoles provençaux." *Systèmes d'Information et Management*, Vol. 17, n°3: p. 111-142.
- Arena, L., Oriol, N. & Pastorelli, I. (2013) "Systèmes d'information et gestion du couple performance/sécurité : trajectoires comparées de trois situations extrêmes." *Systèmes d'Information et Management*, Vol. 18, n°1: p. 87-123.
- Austin, R. & Larkey, P. (1992) "The Unintended Consequences of Micromanagement: The Case of Procuring Mission Critical Computer Resources." *Policy Sciences*, Vol. 25, n°1: p. 3-28.
- Ben Youssef, A., Hadhri, W. & Meharzi, T. (2015) "Adoption of Cloud Computing in Emerging Countries: The Role of the Absorptive Capacity." *Systèmes d'Information et Management*, Vol. 20, n°4: p. 117-142.
- Benkeltoum, N. (2011a) *Gérer et comprendre l'open source*. Paris: Presses des Mines.
- Benkeltoum, N. (2011b) "Regards sur les stratégies de détournement dans l'industrie open source." *Vie et sciences de l'entreprise*, Vol., n°187: p. 72-91.
- Benkeltoum, N. (2013) "Évaluation de l'innovation des logiciels open source." *Système d'Information et Management*, Vol. 18, n°3: p. 37-84.
- Bitzer, J. & Schröder, P. J. H. (2005) "Bug-fixing and Code-writing: the Private Provision of Open Source Software." *Information Economics and Policy*, Vol. 17, n°3: p. 389-406.
- Blackduck. (2016) "The Ninth Annual Future of Open Source Survey."
- Bloom, P. C. & Chung, Q. B. (2001) "Lessons Learned from Developing a Mission-critical Expert System with Multiple Experts through Rapid Prototyping." *Expert Systems with Applications*, Vol. 20, n°2: p. 217-227.
- Bonaccorsi, A., Giannangeli, S. & Rossi, C. (2006) "Entry Strategies Under Competing Standards: Hybrid Business Models in the Open Source Software Industry." *Management Science*, Vol. 52, n°7: p. 1085-1098.
- Bouty, I., Drucker-Godard, C., Godé, C., Lièvre, P., Nizet, J. & Pichault, F. (2011) "Les pratiques de coordination en situation extrême." *Management & Avenir*, Vol., n°1: p. 387-393.
- Capra, E., Francalanci, C., Merlo, F. & Rossi-Lamastra, C. (2011) "Firms' involvement in Open Source projects: A trade-off between software structural quality and popularity." *Journal of Systems and Software*, Vol. 84, n°1: p. 144-161.
- Carton, S., de Vaujany, F.-X. & Romeyer, C. (2003) "Le modèle de la Vision Organisée: un essai d'instrumentation." *Systèmes d'Information et Management*, Vol. 8, n°4: p. 3-30.
- Chau, P., Y. K. & Tam, K. Y. (1997) "Factors Affecting the Adoption of Open Systems: An Exploratory Study." *MIS Quarterly*, Vol. 21, n°1: p. 1-24.
- Chesbrough, H. (2003) *Open Innovation: The New Imperative for Creating and Profiting from Technology*: Harvard Business School Press.
- Christensen, C. M. (2006) "The Ongoing Process of Building a Theory of Disruption." *Journal of Product Innovation Management*, Vol. 23: p. 39-55.
- CIGREF. (2011) "Maturité et gouvernance de l'Open Source - La vision des grandes entreprises." CIGREF, Paris.
- Cohen, W. M. & Levinthal, D. A. (1989) "Innovation and Learning: The Two Faces of R & D." *The Economic Journal*, Vol. 99, n°397: p. 569-596.
- Cohen, W. M. & Levinthal, D. A. (1990) "Absorptive Capacity: A New Perspective on Learning and Innovation." *Administrative Science Quarterly*, Vol. 35, n°1: p. 128-152.

- Colombo, M. G., Piva, E. & Rossi-Lamastra, C. (2014) "Open Innovation and Within-industry Diversification in Small and Medium Enterprises: The case of Open Source Software Firms." *Research Policy*, Vol. 43, n°5: p. 891-902.
- Dahlander, L. (2005) "Appropriation and Appropriability in Open Source Software." *International Journal of Innovation Management*, Vol. 9, n°3: p. 259-285.
- Dahlander, L. & Magnusson, M. (2008) "How do Firms Make Use of Open Source Communities?" *Long Range Planning*, Vol., n°41: p. 629-649.
- Dahlander, L. & Magnusson, M. G. (2005) "Relationships between Open Source Software Companies and Communities: Observations from Nordic Firms." *Research Policy*, Vol. 34, n°4: p. 481-493.
- Dahlander, L. & Wallin, M. W. (2006) "A Man on the Inside: Unlocking Communities as Complementary Assets." *Research Policy*, Vol. 35, n°7: p. 1243-1259.
- Dedrick, J. & West, J. (2003) "Why Firms Adopt Open Source Platforms: a Grounded Theory of Innovation and Standards Adoption." Pp. 236-257 in *Proceedings of the workshop on standard making: A critical research frontier for information systems: Seattle, WA.*
- Deodhar, S. J., Saxena, K. B. C., Gupta, R. K. & Ruohonen, M. (2012) "Strategies for Software-based Hybrid Business Models." *The Journal of Strategic Information Systems*, Vol. 21, n°4: p. 274-294.
- Eisenhardt, K. M. (1989) "Building Theories from Case Study Research." *The Academy of Management Review*, Vol. 14, n°4: p. 532-550.
- Eisenhardt, K. M. & Graebner, M. E. (2007) "Theory Building from Cases: Opportunities and Challenges." *Academy of Management Journal*, Vol. 50, n°1: p. 25-32.
- Eveland, J. D. (1979) "Issues in Using the Concept of "Adoption of Innovations"." *The Journal of Technology Transfer*, Vol. 4, n°1: p. 1-13.
- Evrard Samuel, K. & Ruel, S. (2013) "Systèmes d'information et résilience des chaînes logistiques globales." *Système d'Information et Management*, Vol. 18, n°3: p. 57-85.
- Ferrell, T. K. & Ferrell, U. D. (2001) "RTCA DO-178B/EUROCAE ED-12B." in *The Avionics Handbook*, edited by C. r. Spitzer: CRC Press LLC.
- Fitzgerald, B. (1997) "The Use of Systems Development Methodologies in Practice: a Field Study." *Information Systems Journal*, Vol. 7, n°3: p. 201-212.
- Fitzgerald, B. (2006) "The Transformation of Open Source Software." *Management Information System Quarterly*, Vol. 30, n°3: p. 587-598.
- Fuggetta, A. (2003) "Open Source Software an Evaluation." *Journal of Systems and Software*, Vol. 66, n°1: p. 77-90.
- Gartner Inc. (2011) "Gartner Survey Reveals More than Half of Respondents Have Adopted Open-Source Software Solutions as Part of IT Strategy" February 8
<http://www.gartner.com/it/page.jsp?id=1541414>
- Gary, K., Enquobahrie, A., Ibanez, L., Cheng, P., Yaniv, Z., Cleary, K., Kokoori, S., Muffih, B. & Heidenreich, J. (2011) "Agile Methods for Open Source Safety-critical Software." *Software: Practice and Experience*, Vol. 41, n°9: p. 945-962.
- Glaser, B. G. & Strauss, A. L. (1967) *The Discovery of Grounded Theory: Strategies for Qualitative Research*: Transaction Publishers.
- Godé, C., Hauch, V., Lasou, M. & Lebraty, J.-F. (2012) "Une singularité dans l'aide à la décision: le cas de la Liaison 16." *Systèmes d'information & management*, Vol. 17, n°2: p. 9-38.
- Goode, S. (2005) "Something for Nothing: Management Rejection of Open Source Software in Australia's Top Firms." *Information & Management*, Vol. 42, n°5: p. 669-681.
- Grand, S., von Krogh, G., Leonard, D. & Swap, W. (2004) "Resource Allocation Beyond Firm Boundaries." *Long Range Planning*, Vol. 37, n°6: p. 591-610.
- Gwebu, K. L. & Wang, J. (2011) "Adoption of Open Source Software: The Role of Social Identification." *Decision Support Systems*, Vol. 51, n°1: p. 220-229.
- Haefliger, S., von Krogh, G. & Spaeth, S. (2008) "Code Reuse in Open Source Software." *Management Science*, Vol. 54, n°1: p. 180-193

- Hansen, M., Köhntopp, K. & Pfitzmann, A. (2002) "The Open Source Approach — Opportunities and Limitations with Respect to Security and Privacy." *Computers & Security*, Vol. 21, n°5: p. 461-471.
- Harison, E. & Koski, H. (2010) "Applying open innovation in business strategies: Evidence from Finnish software firms." *Research Policy*, Vol. 39, n°3: p. 351-359.
- Henkel, J. (2006) "Selective Revealing in Open Innovation Processes: The Case of Embedded Linux." *Research Policy*, Vol. 35, n°7: p. 953-969.
- Huy, L. V., Rowe, F., Truex, D. & Huynh, M. Q. (2012) "An Empirical Study of Determinants of E-Commerce Adoption in SMEs in Vietnam: An Economy in Transition." *Journal of Global Information Management*, Vol. 20, n°3: p. 23-54.
- Iacovou, C. L., Benbasat, I. & Dexter, A. S. (1995) "Electronic Data Interchange and Small Organizations: Adoption and Impact of Technology." *MIS Quarterly*, Vol. 19, n°4: p. 465-485.
- IDC. (2009) "Open Source Software Market Accelerated by Economy and Increased Acceptance From Enterprise Buyers, IDC Finds" *Business Wire*, 29 July
<http://www.businesswire.com/news/home/20090729005107/en/Open-Source-Software-Market-Accelerated-Economy-Increased>
- Jullien, N. (2003) "Le marché francophone du logiciel libre." *Système d'Information et Management*, Vol. 8, n°1: p. 77-100.
- Lawton, G. (2002) "Open Source Security: Opportunity or Oxymoron?", pp. 18-21.
- Le Boulout, E. (2011) "Une importante faille de sécurité découverte dans les iPhone" *01Net*, 8 octobre
<http://www.01net.com/editorial/546126/une-importante-faille-de-securite-decouverte-dans-les-iphone/>
- Le Texier, T. & Versailles, D. W. (2009) "Open Source Software Governance Serving Technological Agility: The Case of Open Source Software within the DoD." *International Journal of Open Source Software and Processes (IJOSSP)*, Vol. 1, n°2: p. 14-27.
- Lebraty, J.-F. (2013) "SI et situations extrêmes." *Systèmes d'information & management*, Vol. 18, n°1: p. 3-10.
- Leonard-Barton, D. (1990) "A Dual Methodology for Case Studies: Synergistic Use of a Longitudinal Single Site with Replicated Multiple Sites." *Organization Science*, Vol. 1, n°3: p. 248-266.
- Lesca, N., Caron-Fasan, M.-L., Loza Aguirre, E. & Chalut-Sauvannet, M.-C. (2015) "Drivers and Barriers to Pre-adoption of Strategic Scanning Information Systems in the Context of Sustainable Supply Chain." *Systèmes d'Information et Management (French Journal of Management Information Systems)*, Vol. 20, n°3: p. 9-46.
- Li, Y., Tan, C.-H., Xu, H. & Teo, H.-H. (2011) "Open Source Software Adoption: Motivations of Adopters and Amotivations of Non-adopters." *SIGMIS Database*, Vol. 42, n°2: p. 76-94.
- Li, Y., Tan, C.-H. & Yang, X. (2013) "It is all About What we have: A Discriminant Analysis of Organizations' Decision to Adopt Open Source Software." *Decision Support Systems*, Vol. 56: p. 56-62.
- Lièvre, P. & Gautier, A. (2009) "Les registres de la logistique des situations extrêmes : des expéditions polaires aux services d'incendies et secours." *Management & Avenir*, Vol. 4, n°24: p. 196-216.
- Lisein, O., Pichault, F. & Desmecht, J. (2009) "Les business models des sociétés de services actives dans le secteur Open Source." *Systèmes d'Information et Management*, Vol. 14, n°2: p. 7-38.
- Lundell, B., Lings, B. & Lindqvist, E. (2010) "Open Source in Swedish Companies: Where are we?" *Information Systems Journal*, Vol. 20, n°6: p. 519-535.
- MacCormack, A., Rusnak, J. & Baldwin, C. Y. (2006) "Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code." *Management Science*, Vol. 52, n°7: p. 1015-1030.
- Macredie, R. D. & Mijinyawa, K. (2011) "A Theory-grounded Framework of Open Source Software Adoption in SMEs." *European Journal of Information Systems*, Vol. 20, n°2: p. 237-250.
- Marsan, J., Paré, G. & Beaudry, A. (2012) "Adoption of Open Source Software in Organizations: A Socio-Cognitive Perspective." *The Journal of Strategic Information Systems*, Vol. 21, n°4: p. 257-273.

- Meissonier, R., Bourdon, I., Houze, E., Amabile, S. & Boudrandi, S. (2010) "Comprendre les motivations des développeurs de l'open source à partir de leur participation : une étude empirique sur une question de recherche inversée." *Système d'Information et Management*, Vol. 15, n°2: p. 71-133.
- Min, B. J. & Choi, J. S. (2004) "An Approach to Intrusion Tolerance for Mission-critical Services Using Adaptability and Diverse Replication." *Future Generation Computer Systems*, Vol. 20, n°2: p. 303-313.
- Mouakhar, K. & Tellier, A. (2013) "Comment concilier marchand et non-marchand : une taxonomie empirique des comportements stratégiques des SSSL." *Systèmes d'Information et Management*, Vol. 18, n°3: p. 5-36.
- Nagy, D., Yassin, A. M. & Bhattacharjee, A. (2010) "Organizational Adoption of Open Source Software." *Communications of the ACM*, Vol. 53, n°3: p. 148-151.
- Norris, J. S. (2004) "Mission-Critical Development with Open Source Software: Lessons Learned." *IEEE Software*, Vol. 21, n°1: p. 42-49.
- O'Mahony, S. (2003) "Guarding the Commons: How Community Managed Software Projects Protect their Work." *Research Policy*, Vol., n°32: p. 1179-1198.
- OECD/Eurostat, L. (2005) *Oslo Manual: Guidelines for Collecting and Interpreting Innovation Data, 3rd Edition, The Measurement of Scientific and Technological Activities: OECD Publishing.*
- Paulson, J. W., Succi, G. & Eberlein, A. (2004) "An Empirical Study of Open-Source and Closed-Source Software Products." *IEEE Trans. Softw. Eng.*, Vol. 30, n°4: p. 246-256.
- Pettigrew, A. M. (1990) "Longitudinal Field Research on Change: Theory and Practice." *Organization Science*, Vol. 1, n°3: p. 267-292.
- Poba-Nzaou, P., Raymond, L. & Fabi, B. (2014) "Risk of Adopting Mission-critical OSS Applications: an Interpretive Case Study." *International Journal of Operations & Production Management*, Vol. 34, n°4: p. 477-512.
- Qu, W. G., Yang, Z. & Wang, Z. (2011) "Multi-level Framework of Open Source Software Adoption." *Journal of Business Research*, Vol. 64, n°9: p. 997-1003.
- Rogers, E. M. (1983) *Diffusion of Innovations*. London: The Free Press.
- Rossi, C. (2009) "Software Innovativeness. A Comparison Between Proprietary and Free/Open Source Solutions Offered by Italian SMEs." *R&D Management*, Vol. 39, n°2: p. 153-169.
- Sarrab, M. & Rehman, O. M. H. (2014) "Empirical Study of Open Source Software Selection for Adoption, Based on Software Quality Characteristics." *Advances in Engineering Software*, Vol. 69: p. 1-11.
- Spinellis, D. & Giannikas, V. (2012) "Organizational Adoption of Open Source Software." *Journal of Systems and Software*, Vol. 85, n°3: p. 666-682.
- Stuermer, M., Sebastian, S. & von Krogh, G. (2009) "Extending Private-collective Innovation: a Case Study." *R&D Management*, Vol. 39, n°2: p. 170-191.
- Swanson, E. B. & Ramiller, N. C. (1997) "The Organizing Vision in Information Systems Innovation." *Organization Science*, Vol. 8, n°5: p. 458-474.
- Thong, J. Y. L. (1999) "An Integrated Model of Information Systems Adoption in Small Businesses." *Journal of Management Information Systems*, Vol. 15, n°4: p. 187-214.
- Tornatsky, L. G., Eveland, J. D., Boylan, M. G., Hetzner, W. A., Johnson, E. C., Roitman, D. & Schneider, J. (1983) *The Process of Technological Innovation: reviewing the literature*. Washington, D.C.: National Science Foundation.
- Välimäki, M. (2003) "Dual Licensing in Open Source Software Industry." *Systemes d'Information et Management*, Vol. 8, n°1: p. 63-75.
- Von Krogh, G. & Spaeth, S. (2007) "The Open Source Software Phenomenon: Characteristics that Promote Research." *Journal of Strategic Information Systems*, Vol. 16, n°3: p. 236-253.
- Zhu, K., Kraemer, K. & Xu, S. (2003) "Electronic Business Adoption by European Firms: a Cross-country Assessment of the Facilitators and Inhibitors." *European Journal of Information Systems*, Vol. 12, n°4: p. 251-268.