Exploration of Managers' Experiences about the Diffusion of Software Innovations within the Solaris Operating System Engineering department of Oracle Corporation

~~~

Yannick NGHIEM-XUAN

MSc in Management

National College of Ireland

~~~

Submitted to the National College of Ireland, September 2014

Yannick NGHIEM-XUAN — Exploration of Managers' Experiences about the Diffusion of Software Innovations within the Solaris Operating System Engineering department of Oracle Corporation.

I. Abstract

This qualitative research project explores the experience of managers regarding projects of Diffusion of Software Innovations in a very specific environment of the Information Technology industry: the Solaris Operating System Engineering department of Oracle Corporation. The outcome of this paper is a list of insights and recommendations that can be used by actors involved in similar projects of the Information Technology industry.

The investigation is performed through a research process composed of several steps: a process of data collection through individual interviews, a synthesis and analysis of the data using a thematic analysis methodology, and finally an analysis of the data through the flagship theory of the Diffusion of Innovation by Roger.

This paper emphasises the core topics and questions that the interviewed managers regard as being critical to the success or failure of Projects of Diffusion of Innovation, it questions the nature of the uncertainty induced by the innovation and the way it appeals to the use of intuition, finally it explores the nature possible causes of failure of these projects, and if they can be avoided.

II. Declaration

I declare that the work being submitted for examination is wholly my own work and the materials consulted and ideas garnered in the process of researching the dissertation have been properly and accurately acknowledged.

Yannick NGHIEM-XUAN

III. Acknowledgements

I am using this opportunity to express my gratitude to everyone that helped and supported me throughout the course of this MSc in Management project. I am thankful for their guidance, their recommendations, their advices and their patience during the research work.

I would like to express my sincere thanks to Mr. Damien Farnham for encouraging me to take the courage to do this MSc course at part-time.

I would like to thank my project supervisor, Dr. Armendariz for his guidance, and to all the lecturer and staff of the National College of Ireland for their contribution to the exceptional educational experience during these two years. Thanks also to my classmates, with whom classes have been very enjoyable moments.

I would also like to thanks all the managers that have accepted to participate to this qualitative research, and to thank them for their enthusiasm and their useful insights.

Finally, I would like to thank my friends Mirella, Thomas, Vivien, Andrew, Swann, Estelle and my family that were always there for me with their moral support and to cheer me up during difficult times.

Table of Contents

IV. Introduction

Nowadays, technological innovation and diffusion within a company are key success factors in the industry, a strategic aspect for any organisation is to continuously innovate its production processes (D'Aniello, Masone, Tammaro 2006). The rising complexity and subsequent costs for developing new technologies forces companies to manage the research and development in a way to minimise redundant work by spreading innovation as widely as possible in the organisation (Schuh, Aghassi & Valdez 2013). Subsequently, the innovation network has become the most important theme of industrial economy (Zeng & Chuanrong 2009).

However technological innovation only brings value to the company if the technology and its benefits are properly transferred through the enterprise. Despite the progress made by the introduction of methods and techniques, unfinished projects, project overruns, and system failures are still the norm (Kautz & Nielsen 2000, Damsgaard 1994), and despite of huge investments in system development methods and tools, these innovations are not necessarily widely used (Mustonen-Ollila & Lyytinen 2003). There is strong evidence in the literature that innovations are very often not diffused within organisations to achieve improvement (Lundblad 2003). This qualitative research proposition is an exploration of managers experiences of both successful and unsuccessful projects of diffusion of software innovation in a software company. It aims at treating the gaps identified by the literature review in this document.

A. Problem to be addressed

The goal of the introduction of a new Technology within a business is to improve the productivity, the efficiency (cost in time, money, and other resources) with which the work is performed, and to improve the overall production processes (D'Aniello et al. 2006). Indeed, technical leadership is a critical business differentiator (Tidd, Bessant, Pavit 2005). Continuous innovation and improvement of the production processes are crucial aspects in the strategy of any actor in the Industry. The adoption of new software innovations is the realisation at the cost of breaking the habits and the familiar processes in place. It requires staff training, changes in practice and general commitment from anyone impacted (users, managers, etc.) (D'Aniello et al. 2006).

Unfortunately, the process of diffusion of innovation is quite complex and needs to take many different factors in account, of both human and technical nature (Attar & Shahabi 2014). Innovation is not always seen and accepted since it introduces a change of techniques and methods that are ingrained in the production processes. It is often judged as too high of a risk for management as the exact costs of transition and consequent benefits are often very difficult to evaluate (D'Aniello et al. 2006). The adoption is unimaginable without the stakeholders to perceive the advantages and benefits of the project (Rogers 2003). For these reasons, this type of project cannot be conducted in a blink of an eye, it has to be done gradually, step by step, and a migration transition from the old process to the new one has to be closely monitored, every phase has to be supervised under a close control, to avoid any failure (D'Aniello et al.

2006).

The development of software innovation is also known to be a complex task, and as Kautz & Nielsen (2000) assess:

> *despite the progress made by the introduction of methods and techniques, unfinished projects, project overruns, and system failures are still the norm. The limited success of technology-driven approaches has led to a stronger focus on organizational and process-oriented aspects of software development. The organizational implementation of such approaches is an important and problematic issue of knowledge and technology transfer (Kautz and Nielsen 2000).*

Software-engineering technology has evolved rapidly but has not significantly affected the state of the practice of software-development (Raghavan, Chand 1989).

B. Objectives of the proposed dissertation research

This qualitative research proposition is an exploration of managers experiences of successful and unsuccessful projects of Diffusion of Software Innovation in the Solaris Operating System Engineering department of Oracle Corporation. Its ultimate aim is to give insights and to propose a list of recommendations for people to get involved in this type of projects.

C. A few words about the researcher and the proposal

The technological diffusion of software innovation problem, as any problem involving software development, must include the human as well as the technical dimensions (Tidd et al. 2005, Green & Hevner 2000). Research, technology and innovation are now the driving forces behind economic growth in different nations. This led as a result to a growth of the importance of technology management (Schuh et al. 2013).

This research has therefore the advantage to give a glance at a phenomenon which is at the core of the economic grow of nations, and also to target an area of investigation, that is at the intersection of different worlds:

- Practical and Theoretical
- Industrial and Academic
- Technological and human (Managerial)

The problematic of Diffusion of Software Innovation is, for the researcher, a familiar subject. After having spent about ten years in the Software Industry, he had the chance to work on many projects and to witness dozens of projects of Diffusion of Software Innovation in different companies. He personally strongly agrees with Kautz & Nielsen (2000) in their statement that unfinished projects, project overruns, and system failures are still the norm. This statement remains true nowadays. The hope is that this research project will enable to bring some light on this very common phenomenon in the

Software Industry, to enlarge the researcher's understanding on the subject, and to provide useful insights to stakeholders involved in projects of diffusion of software innovation.

The target research environment is the department in which the researcher presently works: the Solaris Operating System Engineering department of Oracle Corporation. Oracle is a large software company that sells a large range of business softwares, and this particular division works on the company's main Operating System product: Solaris, a Unix Operating System. An operating system is a software that manages computer hardware, it is the core program that runs in any computer, it serves as a platform for the applications. Windows is a popular example of Operating System, but while Windows intends to be mainly used on desktop computers, Solaris is meant to be used on very large servers in data-centres, as a part of big information systems such as in banking companies and insurance companies, etc.

V. Literature review

This section aims at giving an overview of the state of the art of the literature related to the Diffusion of Innovation. First, it defines the common vocabulary of the research domain, then it gives a glance at the diversity of the theories and frameworks in the literature. Finally, it concludes by identifying the gaps, and by providing synthesis of the main topics identified in the literature.

A. Terms of reference

The adoption of innovation has been studied for over the last three decades, in a broad variety of disciplines such as technology, industry, political science, communications, economics, and education (Sahin 2006). One of the most popular theoretical frameworks regarding the model of adoption of innovations is Rogers' Diffusion of Innovation theory, or D.O.I. (Oliveira, Thomas and Espadanal 2014, Sahin 2006, Lundblad 2003, Mustonen-Ollila and Lyytinen 2003). It is useful to define and clarify the vocabulary in use in this research, as there is a variety of interpretations and usages associated with the terms used by different researchers and practitioners (Lundblad 2003).

According to Roger, a ***technology*** is "a design for instrumental action that reduces the uncertainty in the cause-effect relationship involved in achieving a desired outcome" (Roger 2003). It is composed of two components: hardware and software (Sahin 2006, Roger 2003). The hardware is "the tool that embodies the technology in the form of a material or physical object" and the software is "the

information base for the tool" (Sahin 2006, Roger 2003). Software technology refers to the development tools, methods, knowledge, and skills that aid in performing the software development tasks (Nahar, Käkölä, Huda 2002). The ***adoption*** is a decision in which an innovation gets "full use as the best course of action possible" (Roger 2003). The ***diffusion*** is a "process in which an innovation is communicated thorough certain channels over time among the members of a social system" (Roger 2003).

Roger (2003) defines an innovation as "an idea, practice, or project that is perceived as new by an individual or other unit of adoption". The ***software technology diffusion*** refers to the transfer of a software technology and of the knowledge about how to use it from a producer to a user who utilises it in order to get a productive value from it (Nahar et al. 2002). Roger (2003) describes the ***uncertainty*** as being a consequence of the process of adoption, it defines the consequences that "occur in an individual or social system as a result of the adoption or rejection of an innovation". An ***innovation*** is "an idea, practice, or project that is perceived as new by an individual or other unit of adoption" (Roger 2003). In fact, an innovation can be unknown by a group of individual and know by others, what matters for the theory is whether the individuals perceive it as new or not (Sahin 2006). The ***communication channels*** are one of the main elements in the diffusion of innovation process in Rogers' theory (Sahin 2006). Rogers (2003) defines communication through the channels as "a process in which participants create and share information with one another to reach a mutual understanding". The ***social system*** in which the diffusion

process takes place is defined by Roger (2003) as “a set of interrelated units engaged in joint problem solving to accomplish a common goal”.

B. Theories and perspectives in the field of the Diffusion of Innovation

The diffusion of innovation has been studied by scholar in different domains, depending on the targeted use they were intended for: sociological and organisational, managerial, diffusion networks, economical, historical, and business. These approaches cover different aspects of the subject and are complementary, they complete the perspectives of each other (Hall 2003). Studies from many economists on the subject tend to observe the process from a cumulative or aggregate result of individual projects (Hall 2003).

The managerial approach focuses on aspects such as the use of networks of suppliers, customers and others outside the company to influence and stimulate the innovation, on the roles of knowledge and knowledge management to sustain it, on systematic frameworks aiming at being used in practice by Management, on analytical integration of technology, market and organisational change, on practical ways to handle rapid change, extreme volatility implied by technology, how to deal with high uncertainty, etc. (Tidd et al. 2005). The business perspective aims at looking at how innovation diffusion can serve profitability, and sustainable growth (Al-Makim 2010). Also it provides performance evaluation frameworks for innovation diffusion (Al-Makim 2010).

Sociological and organisational perspectives study the influence of the social environment, the norms, politics, standardisation, culture, the degree of interconnected-ness, the extend to which decisions are made individually or collectively, the communication channels (Mann & Chan 2011, Schein 2010, Roger 2005). Sociological factors can have a major influence on the innovation diffusion as it can lead to suboptimal choices, due in part to human cognitive limits (Mann & Chan 2011, Roger 2005), as testified by the example of the QUERTY keyboard design, originally conceived to minimise the key clashes on typewriters but which is still widely adopted for computer age keyboards (Mann & Chan 2011). The sociological and organisational perspective also look at the conditions that can help create a social climate for he innovation and its diffusion, and also how organisational structure changes can be performed in that matter (Schein 2010, Roger 2005).

Theory is needed to identify features that help to predict the success or failure of a diffusion of software innovation project. However, understanding the way in which the diffusion process unfolds requires models (Hall 2003). Attar & Shahabi (2014) assess that there are two traditions of models: rational and scientific on one side, and non-rational and artistic models on the other. The former, the model is regarded as a very methodical and rational approach, describing the innovation as a tightly controlled process with precise steps, and strict controls between the steps, like a 'development funnel' (Attar & Shahabi 2014). The later class of models, outlines the process as a non-rational, social-technical practice that reveal the influence of politics, rituals, norms, organisational behaviour factors

into account (Attar & Shahabi 2014). Roger (2003) also provides a framework, in the last versions of his book, that belongs to this second class of models.

C. Flagship of the Diffusion of Innovation Theory: Rogers' model

The core of literature on the diffusion of innovation is represented by Rogers' work in a book titled 'Diffusion of Innovations', the latest and fifth edition has been published in 2003 (Roger 2003). Most of the authors in the literature refer to his work. In his book, Rogers reviews the subject primarily from a sociological perspective, but one that is informed by research on organisations, the role of economic factors, and the strategies of firms and development agencies (Hall 2003). Rogers' theoretical framework is often used as a reference and as a base, in large and small business environments, for the diffusion of new information technologies and their adoption (Oliveira et al. 2014, Sahin 2006). Rogers' theoretical framework will be exposed in more detail in the following section.

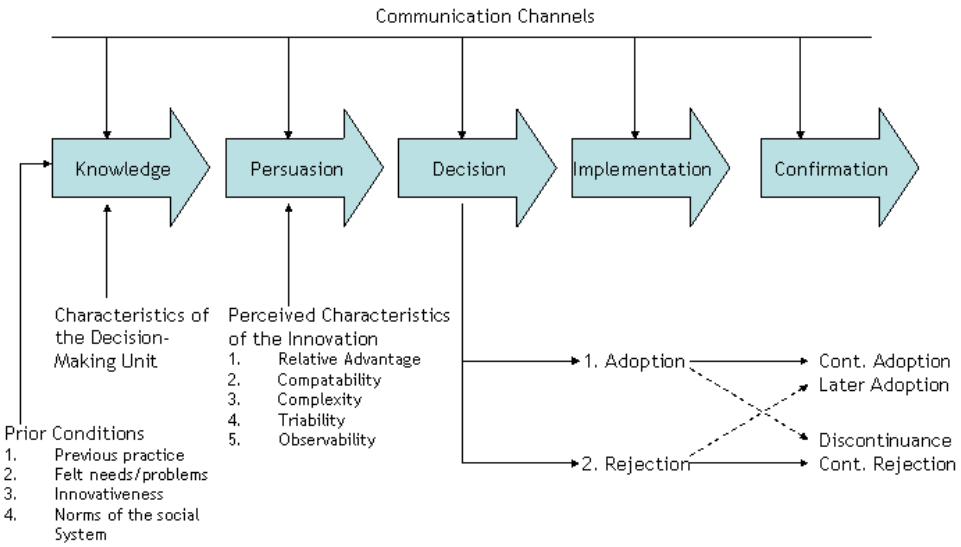
D. Rogers' theoretical framework: Diffusion of Innovation (DOI)

This section of the document aims at giving to the reader an overview of Rogers' Diffusion of Innovation theoretical framework, which is a prominent adoption model used in Information Technology research (Oliveria et al. 2014). A large part of the research in the field of diffusion of innovation is around technology, so Roger often uses the words “innovation” and “technology” to designate the same things (Sahin 2006). The essential elements of Rogers' theory are the Innovation-Decision Process model, which segments the diffusion

process through different stages, the adopters categorisation, which explains how different categories of individuals enter the process of adoptions, and finally the criteria and traits of an innovation that can impact its diffusion (Roger 2003).

1. The innovation-Decision Process

Roger suggests that innovation is a communication process using the various channels within a social system (Oliveria et al. 2014). He describes this process as "an information-seeking and information-processing activity, where an individual is motivated to reduce uncertainty about the advantages and disadvantages of an innovation" (Roger 2003). According to Roger, the Innovation-Decision Process is sub-divided into five steps: knowledge, persuasion, decision, implementation and confirmation (Sahin 2006).



Stages of the Innovation-Decision Process (Roger 2003)

a. The knowledge stage

The knowledge stage is the first stage of the Innovation-Decision Process. In this step, the individuals learn about the existence of the innovation and seek to get an understanding of it, about what is its nature, what it is meant to do, and how it is meant to be used (Sahin 2006).

b. The persuasion stage

In this stage, the individuals adopt an attitude toward the innovation, which can be favourable or unfavourable, this is often is a very affective way, rather than rational (Sahin 2006). This stage involves social interaction (discussions, exchanges of opinions) which are likely to influence the individuals opinions and beliefs about the innovation (Sahin 2006).

c. The decision stage

The decision stage is the stage at which individuals decide to either adopt or reject the innovation (Roger 2003). As Sahin (2006) notes in his review of Rogers' book, if an innovation has a partial trial basis, it is usually adopter more quickly, since most individual wish to test the innovation in their own personal use-cases to evaluate how it fits their needs and as if it is likely or not to bring them an added-value compared to what was used beforehand.

d. The implementation stage

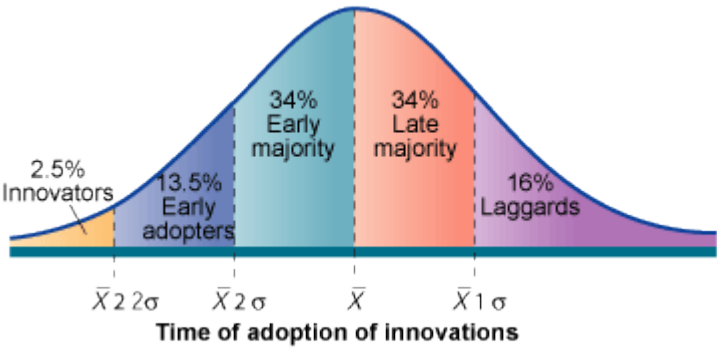
At the implementation stage, the innovation is put into practice and is used by the individuals (Sahin 2006). At this stage, uncertainty can still be an issue, and the users might need support and technical assistance (Sahin 2006).

e. *The confirmation stage*

In the confirmation stage, users look for support for their decision to have adopted the innovation, but this decision can still be reversed if the individual is "exposed to conflicting messages about the innovation (Roger 2003). Sahin (2006) assesses that in general, at this stage of the diffusion, individuals tend to seek supportive messages and try to stay away from the opposition to confirm their decision.

2. The adopters categories

Roger (2003) classifies the adopters into five categories. He defines them as "the classification of members of a social system on the basis of innovativeness" (Roger 2003). This classification groups individual regarding their innovativeness, that is "the degree to which an individual or other unit of adoption is relatively earlier in adopting new ideas than other members of a system" (Roger 2003). The innovativeness is a relatively stable characteristic that indicate users willingness to change their habits, and familiar practices (Sahin 2006).



Adopter Categorisation on the basis of innovativeness (Roger 2003)

a. Innovators

According to Roger (2003), innovators are willing to experience new ideas. They may not be respected by the other members of the social system because of their venturesomeness (Sahin 2006).

b. Early adopters

Roger (2003) argued that early adopters are more limited with boundaries than innovators in the social system, and are likely to have leadership roles inside the social system. Thus, they have an important role in the diffusion process.

c. Early majority

As Roger (2003) claimed, the early majority often have good interactions with the members of the social system, then do not have leadership roles though. The early majority adopt deliberately the innovation, but their decision usually takes more time than the innovators and early adopters (Sahin 2006, Roger 2003).

d. Late majority

The late majority wait until most of their peers have adopted the innovation, and tend to be skeptical about it and about its benefits (Sahin 2006).

e. Laggards

The laggards are traditionalists and are the most skeptical individuals about the innovation, they tend to adopt it when they are forced to, and when they have no other choice (Sahin 2006, Roger 2003).

3. Criterion impacting the diffusion of innovation

For Roger (2003), the innovation-diffusion process in "an uncertainty reduction process". The traits he defines have a direct impact on the diffusion process and Roger stated that "individual's perceptions of these characteristics predict the rate of adoption of innovations" (Roger 2003). Studies, such as the one conducted by Mustonen-Ollila and Lyytinen (2003) demonstrate that several of the D.O.I factors strongly affect the Information System and technologies process of adoption. The D.O.I. theory classifies the domains that impact the diffusion of innovation into five categories (Sahin 2006, Hall 2003, Roger 2003):

a. Relative advantage

The relative advantage of an innovation is the advantage this innovation has compared to the one it is supposed to supersede (Roger 2003).

b. Compatibility

According to Roger (2003), the "compatibility is the degree to which an innovation is perceived as consistent with the existing values, past experiences, and needs of potential adopters". As Sahin (2006) points out, for Information Technologies, the lack of compatibility with users needs may impact negatively the usage of the innovation.

c. Complexity

Roger (2003) defines the complexity as "the degree to which an innovation is perceived as relatively difficult to understand and use". In the contrarily of the other features of an innovation, this one may have a negative impact on the diffusion process (Sahin 2006).

d. Trialability

The trialability of an innovation is the extend to which the innovation can be tested by the targeted adopter (Roger 2003). Roger (2003) defines the trialability as “the degree to which an innovation may be experimented with on a limited basis”. As Sahin (2006) assesses, this factor positively correlates with the rate of adoption.

e. Observability

T h e observability of an innovation is the ease with which the innovation can be evaluated after trial (Roger 2003). Roger (2003) describes the observability as “the degree to which the results of an innovation are visible to others”. Similarly to the relative advantage, the compatibility and the trialability, the observability correlates positively to the rate of adoption of the innovation (Sahin 2006).

Hall (2003) assesses that most of these attributes are often undertaken by researchers, though sometimes with different names (e.g. trialability and observability for the level of uncertainty faced by a potential adopter). In addition to these criteria, Roger also emphasises the importance of the social conditions (norms, culture, degree of interconnectedness, the extend to which decisions are made individually or collectively, etc.) and communication channels (promotion efforts, etc.) and their impact of the diffusion process (Roger 2003).

E. Similar theories

Deshipande (1983) has reviewed and compared three popular books on the theory of the diffusion of innovation, each of which is focused on a very different perspective:

- Innovation Diffusion: A New Perspective by L. A. Brown. New York Metbuen & Company, 1981
- Diffusion of Innovation, third edition by Everett M. Roger. New York: The Free Press, 1983
- Patterns of Technological Innovation by D. Sahal MA: Addison Wesley Publishing Company, 1981.

His aim was to figure out how these theories could be used for the field of Marketing. While Brown studies the diffusion of innovation from a spacial perspective, Roger from a sociological and communicative perspective, and Sahal focuses on technological, economical and technological forecasting perspectives (Deshipande 1983). Deshipande (1983) finds that despite of their very different focus, the authors tend to have complementary theories that do not contradict each other, and that even tend to complement each other. He finds that despite the fact each of these books focuses on a different perspective, they do not seem to contradict, and they present harmonious and compatible theoretical paradigms, as their theoretical purpose and their theories seem to fit nicely with each other (Deshipande 1983). As Deshipande (1983) assesses, despite of the fact there is no general theory on the Diffusion of Innovation, the fragmentation in the literature does not seem to be explained by

contradiction in the theories, but in fact it seems to be a consequence of the multitude of approaches of the authors explained by the differences in the perspectives they focus on.

F. Criticism of Rogers' theory in the literature

The most recurrent criticism regarding the limitations of the applicability of Rogers' theory is that too generalist to be applied to specific cases or fields, such as a particular corporate environment, or the field of Marketing (Deshipande 1983) or Information Technology (Oliveira et al. 2014, Lundblad 2003, Mustonen-Ollila et al. 2003). Different domains have very different criterion impacting the Diffusion of Innovation, just as different industries have different drivers, priorities and criterion (Oliveira et al. 2014, Lundblad 2003), and organisational type, and size are not considered in D.O.I., though they are very likely to affect the adoption (Lundblad 2003). As Oliveira et al. (2014) assess, D.O.I. does not consider key factors such as cost optimisation and security concerns, that are critical to firm's adoption of Software Innovations. While Rogers' theoretical framework covers the diffusion of innovation within the organisations, variables and their interactions when innovation get diffused are less well described in and across organisations, and as a consequence, the interactions in the social network in business environments are less well defined too (Lundblad 2003). The boundaries of D.O.I theory is a social system, which can be anything from a neighbourhood to a profession, or a country, and, as Lundblad (2003) assesses, Roger sees the organisation as the social system in the case of business environments, but it can also be a team or a department. Thus, the specific issues related to the interactions between teams or departments within the organisation are not

address, nor how they impact the adoption (Lundblad 2003). Rogers' theory is a base, a foundation that can guide and help actors, stakeholders, or theorist in most domains that involve the study of diffusion of innovation in their respective context, but its generality, and its lack of precision for specific cases must be kept in mind when it is used in particular contexts. These specificities of a very particular domain such as the Software Industry studied in this research, are likely to have very significant impacts on the adoption or the processes of diffusion of innovation and the rules can diverge greatly from Rogers' observations in corporate environments.

G. Use of Rogers' theory in the field of Information Technologies

Rogers' Diffusion of Innovation theory is well established and widely used in Information Technologies (Mustonen-Ollila & Lyytinen 2003), and is still in 2014 the dominant theoretical framework in the sector (Oliveira et al. 2014), as it has been for the past five decades (Mustonen-Ollila & Lyytinen 2003). The validity of D.O.I theory in explaining innovation adoption for Information System has been confirmed (Mustonen-Ollila & Lyytinen 2003), however, as seen in the section about Rogers' criticism in the literature above, Rogers' theoretical framework is too generalist to cover expansively the high tech industry and does not considers costs saving factors and security concerns that are critical to the firms (Oliveira et al. 2014), and incorporate resources and time restrictions (Mustonen-Ollila & Lyytinen 2003). Mustonen-Ollila & Lyytinen (2003) assess that D.O.I. criterion are difficult to observe in organisations, thus their comprehensiveness and the reliability of data, and the classification into innovative types is imprecise. Moreover, some results obtained

in an organisation may not be applicable to others because of firms specificities (Mustonen-Ollila & Lyytinen 2003). Practitioners and academics alike still have little knowledge about the underlying mechanisms of the Diffusion of Innovation processes that influence the adoption of new Software Technologies in the Information Technology Industry (Mustonen-Ollila & Lyytinen 2003), and while Rogers' DOI framework can be used as a theoretical foundation for Diffusion of Software Innovation projects, it has to be used with great care and stakeholders involved need keep in mind that this framework is not exhaustive in regard of the specifics of the field.

H. Conclusion

Rogers' D.O.I. theoretical framework cannot be overlooked and constitutes a generalist theoretical foundation to the understanding of the diffusion of innovation process. Nevertheless, the very generalist aspect of the theory also means that, while it can provide guidance for more specific cases, it will never be able to answer in depth to more focused and specialised contexts like the one targeted in this research. Indeed, Roger (2003) has studied the diffusion of innovation phenomenon at a general level, that occurs when an innovative product such as a new mobile phone hits the market. In these cases, individuals are free to adopt or reject the innovation, it is their own choice, and most of the social interactions as described by Rogers' framework assume this freedom. In a corporate environment, the rules are different, end users are rarely those who decide to adopt or reject the innovation. The decision tends to be less emotional, and more rational, and factors such as cost optimisation and security concerns are taken in account (Oliveira et al. 2014), whereas in a general context, these criterion are less likely to have an influence on

the choices. While Rogers' framework provides a few elements of theory regarding professional environments, they are very succinct, and constitute some sort of overviews about what can happen in a business environment. Moreover, the conditions, social interactions, constraints and priorities are likely to vary greatly from a professional environment to an other, and what may apply in the textile industry for example, is likely do diverge greatly from what applies in the software industry. This means that even the few elements in Rogers' model that apply to the businesses also have to be regarded as generalist guidelines. However, Rogers' theory cannot be ignored, and the authors in the literature seem to agree that his model constitute the baseline for the core diffusion of innovation theory, even if it does not cover extensively very specific areas such as the one targeted in this research, that is a particular environment of the Information Technology industry.

I. Core themes in the literature

This section contains a synthesis of the themes that the articles studied for this research identify as having an incidence on the Diffusion of the Innovations. While the framework used in this research is Rogers' model, a cross thematic analysis of the different articles permitted the identification of some gaps, which consequently are involved in the definition of the research questions. The following table categorises the topics and factors having an impact on the research, as well as the corresponding references.

	Topics and factors of the Diffusion of Innovation	References
I	***Environment***	
1	Social, cultural, political, rituals and norms	Attar & Shahabi 2014 Oliveria et al. 2014 Hall 2013 Mann & Chan 2011 Al-Hakim 2010 Schein 2010 Tidd et al. 2005 Roger 2003
2	Professional norms, institutionalisation, organisational behaviour, nature of the industry	Attar & Shahabi 2014 Oliveria et al. 2014 Hall 2013 Mann & Chan 2011 Al-Hakim 2010 Schein 2010 Tidd et al. 2005 Roger 2003
II	***Innovation***	
1	Prediction of the benefits of the innovation, and evaluation of the performances of the innovation	Al-Hakim 2010 Roger 2003 Kautz & Nielsen 2000
2	Recognition of the need to innovate	Mann & Chan 2011 D'Aniello et al. 2006 Roger 2003 Damsgaard 1994
3	Empowerment of the users by the innovation	Attar & Shahabi 2014 Oliveria et al. 2014 Roger 2003
III	***Diffusion process***	
1	Nature of the new solution (complexity, compatibility, observability, trialability)	Attar & Shahabi 2014 Roger 2003
2	Communication channels	Schuh et al. 2013 Mann & Chan 2011 Zeng et al. 2009 Roger 2003 Green 2000
3	Agent of promotion effort	Schuh et al. 2013 Mann & Chan 2011 Roger 2003

		Green 2000
4	Prediction of time and resources needs	Hall 2013 Roger 2003
5	Dealing with the uncertainty	Attar & Shahabi 2014 Hall 2013 Roger 2003
IV	***Aftermath***	
1	Understanding the causes of project failures	Attar & Shahabi 2014 Kautz & Nielsen 2000
2	Use of intuition under uncertainty	Attar & Shahabi 2014 Hall 2013
3	Use of theory and of models in the field	Attar & Shahabi 2014 Hall 2013

Core topics and factors as identified by the articles and impacting the process of Diffusion of Innovation

This table has permitted to clarify the fact that most of the authors covered the impacts of the environment, of the innovative process and about the way the diffusion process itself is handled, to explain their interactions and their consequences onto the Diffusion of Innovation process. Attar & Shahabi (2014), Hall (2013) and Kautz & Nielsen (2000) seem agree that the aftermath of the Diffusion processes have very little coverage in the literature. Theories in the field of diffusion of innovation give very few explanation for the causes of failures (Attar & Shahabi 2014, Kautz & Nielsen 2000), as to the use of intuition induced by the uncertainty related to the adoption or rejection of the innovation (Attar & Shahabi 2014, Hall 2013), and to the extent of the coverage of the theories in the field, and their restrictions (Attar & Shahabi 2014, Hall 2013). These gaps are detailed in the following section.

J. Gaps in the literature

This section aims at depainting the gaps identified in the literature, it is important to note that only a selection of them are to be treated by the research questions.

1. Lack of completeness in the perspectives of the literature

Attar & Shahabi (2014), Oliveira et al. (2014), Hall (2013) and Lundblad 2003 assess that the existing frameworks and theories seem to focus each on the specific perspectives they were intended for, that is the economists will focus primary on the economical perspectives, the sociologists on the sociological ones, and so on for other domain such as political and technological domains. However, for a stakeholder such as a Manager that needs to get involved in a Diffusion of Innovation project and that has to mind several of these aspects at the same time, the theories do not synthesise the problematics of the different aspects of different nature in a unique model (political, sociological, etc.). There is a lack of information regarding which of these aspects he or she should focus his or her attention on, and about which of them should be considered as being core to the diffusion process. There does not seem to have a theory that synthesises all the perspectives in a unified and practical way, which could really help the stakeholders involved, to keep track of the important aspects of diverse nature that can impact the ultimate outcome of the project. This research will hopefully permit to bring some light on the perspectives they really perceive as being of importance.

2. No clear boundaries for the theories

Second, the models fail to give precise information regarding their limitations and their field or validity and applicability. As Attar & Shahabi (2014) argue, neither of the rational and non rational models give much emphasis to the restrictions, problems, position between the theory and practice of the instruments they suggest. Being able to evaluate the environment and the factors precisely, and being able to predict their impacts, is needed in order to frame properly a Diffusion of Innovation project, and to be able to foresee the outcomes. If the theories fail to define the condition in which they apply, and if they fail to state with precision the environmental factors that need to be met to for their validity, it makes it very hard for Managers involved to use these theories in the field, as they need to reach high level of certainty, which are required by these projects in the large businesses. Thus, by their lack of precision, the existing methodologies fail to provide frameworks that permit a meticulous control and management of the diffusion projects with the high standards needed in the large corporate environment of the Information Technology industry. However the research will not cover this particular area by the need to stay focused on a restraint number of issues.

3. No coverage over the use of intuition under uncertainty

Neither of the theories has much to say either about managers use of intuition to make decision under uncertainty, nor about how to behave in problematic situations (Attar & Shahabi 2014). As a reminder, uncertainty is, as per Rogers' (2003) definition, a repercussion of the process of adoption, which defines the consequences that “occur in an individual or social system as a result

of the adoption or rejection of an innovation". Innovations, by definition and by nature, bring novelty, and novelty brings uncertainty as to what the actual capabilities of the innovation are, what are its limitations, as to which extend it will be able to fulfil the role it has been designed for, and about the unforeseen issues can be encountered with the innovation and during the diffusion process. The theories are very evasive regarding how Managers can react to the uncertainty, how they take decisions when they are confronted to contingency, that can happen directly as a result of the nature of the innovation, but also from the environment in which the diffusion happens. This aspect is to be treated in this research, as it aspires to get insights about the nature of these situations, and about the way Managers address them in the field.

4. No explanation for the firm failures

Finally, theories and models around the Diffusion of Innovations in general seem to fail to give explanations about firms failures (Attar & Shahabi 2014, Kautz & Nielsen 2000), and as Deshipande (1983) assesses, innovative products failure appear to be a neglected topic. The causes of failure of the Diffusion of Innovation projects are not surprisingly not furthermore developed in Rogers' theoretical framework that the mechanisms specific to the Diffusion of Innovations in business environments. Indeed, as discussed previously, Rogers' model has a very succinct coverage of the business environment. Developing an understanding of the causes of failures can bring judicious information about Diffusions of Innovations in a particular environment, it can provide a vision to aspects of the process that are required to achieve success, and it can also give useful insights on aspects that could first appear as being

unimportant but that the experience emphasises as more critical than it can be initially thought of. As Bill Gates once said, *"It's fine to celebrate success but it is more important to heed the lessons of failure"* (Gray 2006). The explanation for the firm failures is to be covered by this research, this is an area where experienced managers should be able to provide some light with useful insights for anyone that can be actively involved in a Diffusion of Innovation project.

VI. Research Questions and Goals

A. Overall objective

The aim of this research is to explore Managers' experiences of the Solaris Operating System Engineering department of Oracle Corporation regarding the projects of Diffusion of Software Innovation. The goal of this research to address the selected gaps identified in the literature review, and to provide, for each of them, insights and a list of recommendations, that could help stakeholders involved in similar projects.

B. Sub-Objectives

1. First Sub-Objective

- *To identify the main themes and questions that managers recognise as being core to the success or failure of Software Innovation Diffusion.*

The literature review of this document has highlighted the fact that the theorists focus each on a different perspective, however Managers that are involved in projects of Diffusion of Software Innovation are confronted to several of these aspects at the same time (social, economical, political, etc.). The aim of this question is to clarify the aspects that matter the most to achieve success, from their perspective and their experience, in the Solaris Operating System Engineering Department of Oracle Corporation.

2. Second Sub-Objective

- *To explore the problematic situations induced by uncertainty, about their nature, about the way they can be dealt with, and to question the use of intuition in these situations (Attar & Shahabi 2014, Hall 2013).*

According to Roger (2003), and as seen previously in the terms of reference section of this document, uncertainty is an outcome of the process of adoption and it defines the consequences that occur in the social system as a result of the adoption or rejection of an innovation. In the literature review section, we have seen that theory has little coverage regarding the nature of the situations induced by uncertainty, and in regard of how manager deal with them. This sub-objective aims at exploring manager experiences in order to get useful and practical insights on these questions.

3. Third Sub-Objective

- *To explore the reasons for project failures, and to try to identify trends in the signs that could potentially have warned Management about the possibility of failure.*

Failures of Project of Diffusion of Software Innovation is an area that theory have a very thin coverage for (Attar & Shahabi 2014, Kautz & Nielsen 2000) even though such failures are still very common in the industry (Kautz & Nielsen 2000, Damsgaard 1994). This sub-objective aims at exploring the possible reasons for these failures, and to appreciate if they occur suddenly or if they are preceded by warning signs, and also if failure could have been avoided. The experience of Managers on past project will be able to give clarifications and a better understanding of the firm failures.

C. Questions presented during the interviews

This sub-section lists the question as presented to the interviewees to collect the data. They cover the three themes of the research questions and are voluntarily open questions in order to avoid influencing as much as possible the interviewees, in order to provoke a minimum amount of interference with the production of the data.

1. Main themes and questions

- What are the main themes and questions that you identify as being core to the success or failure of Software Innovation Diffusion projects?

2. Nature of the Uncertainty and dealing with it

- What is the nature of the problematic situations induced by uncertainty you have encountered in these projects?

- How did you deal with them?

- To what extent do they appeal to the use of intuition?

3. Investigation on the causes of failure

- According to your experience, what are the most common causes of failures for this type of project?

- Were there signs that could have warned Management about the possibility of failure?

- Do you think something could have been done to prevent these failures?

VII. Research Methodology and Design

A. Rationale for the study

This research is an exploration of experiences that is conducted through personal interviews of managers who have been involved in projects of Diffusion of Software Innovation. Its aim is to get the perspectives of managers regarding the projects of Diffusion of Software Innovations for a particular department in a particular company: the Solaris Operating System Engineering department in Oracle Corporation. It seeks to explore the perceptions, experiences and points of view to gain a better understanding of the particularities of the research topic in this specific business environment. A qualitative research conducted by the bias of interviews is more appropriate than a quantitative approach, as qualitative methods generally aim to understand the experiences and attitudes of persons (Bryman 2012, Gill, Stewart, Treasure and Chadwick 2008, DiCicco-Boom and Crabtree 2006, Quinn Patton 2002). Such an approach encourages a much greater interest in interviewees' points of view and own perspectives (Gill et al. 2008), and to go off at tangents, which helps to get insights into what the interviewees perceive as being important and relevant (Bryman 2012). Also, a quantitative approach fits nicely when the subject can be broad or when little is known about the domain in the literature such as in a very specific environment, and it can help the researcher with the generation of hypothesis (Quinn Patton 2002), it also helps understanding different perspectives on the topic (Quinn Patton 2002).

Qualitative methods aiming to understand and explore the meaning and perception in a conceptual and theoretical way, and having for purpose to generate insights and a list of recommendations, need to be semi-structured and organised around a set of predetermined open-ended questions, with other unplanned questions that emerge from the discussions between the interviewer and the interviewee (DiCicco-Bloom et al. 2006). Semi-structured interviews consist of several key questions that help defining the areas to be explored, to provide the interviewee with some guidance, but with a lot of flexibility (Gill et al. 2008). This approach allows the discovery of ideas and information that have previously not been thought of (Gill et al. 2008), while aiming at at helping the understanding of complex problematic, potentially involving many factors (DiCicco-Bloom et al. 2006). Moreover, qualitative researches are meant to help the understanding of some aspect of the social life and its methods (Gill et al. 2008, Quinn Patton 2002).

B. Research environment

1. Oracle Corporation

Oracle is a large american multinational software company that sells a large range of business softwares, it is well know for its database application: Oracle database, but also sells a wide range of application such as Enterprise Resource Planning softwares, Customer Relationship Management softwares. Oracle also proposes an Operating System product to its customers: Solaris, a Unix Operating System. As written earlier in this document, an Operating System is a software that manages computer hardware, it is the core program that runs in any computer, it serves as a platform for the

applications. Windows is a popular example of Operating System, but while Windows intends to be mainly used on desktop computers, Solaris is meant to be used on very large servers in data-centres, as a part of big information systems such as in banking companies and insurance companies, etc. The structure is a divisional structure, which means the entities of Oracle are organised around the products.

2. The Solaris Operating System Engineering department and the line of command

The Solaris Operating System Engineering department is the place where this research takes place. This department is responsible for the constant development and improvement of the Solaris Operating System. As many other departments within Oracle, the teams that compose this department are spread all over the world. The communication within the teams in mainly performed through the Internet, using remote conferencing technologies such as the Voice Over Internet Protocol, and electronic mails. The structure within the department is a classic hierarchical structure, where most of the teams have an average of ten members, and where managers also have an average of ten direct reports. The department contains several thousands of employees, but the exact organisation chart remains confidential.

3. Nature of the Software Innovation in the Solaris Operating System Engineering department

The innovations diffused within the Solaris Operating System Engineering department are for the vast majority (about 80% according to the Management) Software Innovations produced internally, inside the same exact department, in order to improve the

internal workflow and to increase productivity. Indeed, the needs within this department are often very specific, and very specialised, which explains the fact that most of the time, adopting an innovation from the outside world, that is produced by an other company, or developed for an other department, is rarely an option. The expression of the needs at the source of the innovation often comes from the stakeholders that aim to be themselves the final users of the innovation. In general, what happens is that these stakeholders warn the hierarchy about the need, and in return, if the hierarchy is convinced that the investment in the project is worth it, they will initiate the Diffusion of the Software Innovation project to address the need.

C. *Limitations of the research*

1. Restriction on the interviewees

The research will be conducted only with managers having directly been involved in a project of Diffusion of Software Innovation, within Oracle, and in the Solaris Operating System Engineering department. There is no other restriction of sex or age or nationality, but men represent the vast majority of the staff, as it is often the case in software companies, which implies that the sample has a masculine majority.

2. Restriction of the Diffusion of Software Innovation projects

The Diffusion of Software Innovation projects often contain several iteration, which means what once the Software Innovation has been diffused within the department, there can be other versions of the innovation coming later on to supersede the pervious iteration in

order to to satisfy some new needs, or to adapt the innovation to changes in the environment. This research will only focus on the really first iteration of the Diffusion of the Software Innovations, that is from the recognition of the initial need to the moment where users are using it in their daily routine.

D. Data collection

1. Selection of the questions

Questions have to be likely to yield as much information about the study phenomenon as possible, and have to be able to get the information needed by the research (Gill et al. 2008). They have to be open-ended, and have to invite the interviewees to get the freedom of covering the areas through their perspective, but they need to be precise and clear enough to avoid too much divergence and to stay focused (Gill et al. 2008, DiCicco-Bloom 2006). They need to drive the discussion for a period from about half an hour, to sometimes several hours (DiCicco-Bloom 2006), but in this particular case the interviews are unlikely to last more than an hour, because of the professional imperatives of the interviewees. DiCicco-Bloom (2006) assesses that the first question in particular should be broad and on-pen ended, it should reflect the nature of the research and be non-threatening to invite the participant into the topic.

2. Pilot interview

A pilot interview has to be conducted in order to allow the researcher to test the question, that is to determine if they are understandable, if they need to be clarified, and if they permit to get the information needed by the research (Gill et al. 2008).

3. Selection of the interviewees sample and conditions and method of interview

a. The sample

Quinn Patton (2002) assesses that while a quantitative sample has needs of statistical representativeness, the samples in qualitative research are purposive. The implication is that participants are selected because of their likeliness to generate useful data for the project (Quinn Patton 2002), key informants are to be selected their knowledge and their experience (DiCicco-Bloom 2006). Quinn Patton (2002) suggests that the sample sizes are typically small in qualitative researches, and that the researcher should continue to interview more people until nothing new comes from the data, a point called 'saturation'. The selection of the sample will be done through different bias, i.e. through the researchers network of relations, directly and indirectly, and through the internal forums and social networks of the company, by probing for voluntary participants.

b. Ideal participants

The ideal participants are Managers who have been involved in one or more projects of Diffusion of Software Innovation within the target research environment. They are to located in different countries such as the United States, countries of Western Europe such as Ireland, and Asia such as China. There should not be any particular difficulty to interview some of the ideal participants in the context of this research.

c. Conditions of the interviews

The interviews are to be individual rather than in group, because

individual interviews encourage in-depth discussions to delve deeply into social and personal matter (DiCicco-Bloom 2006). While in group, people tend to avoid going into too much personal matter, because of the public nature of these meetings (DiCicco-Bloom 2006). The location in which the interviews take place has an impact on the answers, there need to be enough privacy to encourage the interviewees to feel comfortable and to give honest answers (Quinn Patton 2002). The interviews will be conducted in areas free from distractions and at time and locations that are suitable for the interviewee (Gill et al. 2008). The question are submitted to the interviewees about a day in advance, so they have time to think about the subject, and so that they feel more relaxed and less stressed, and know what is expected from them, which increases the likelihood of honesty (Gill et al. 2008). Sending the questions in advance also permits to both the interviewer and interviewee to establish an implicit schedule of the interview meeting, in order to cover the totality of the questions (Gill et al. 2008). In some cases, the interviewees will not be able to be physically in the presence of the researcher, in these cases, interviews will be arranged over the phone or through the Voice Over IP technologies on the Internet.

d. Recording

The conversations are to be recorded, with the approval of the interviewee, and are to be transcribed verbatim afterwards, as it protects against bias and as it keep a record of what has been said, and of what has not been said (Gill et al. 2008). Any sensible information such as person names and project names are to be substituted to preserve the confidentiality. Once the records have been transcribed, they need to be erased to preserve the anonymity of

the participants. Notes are written by the interviewer during the interview process, and are to be added as field notes to the transcripts (DiCicco-Bloom 2006).

4. Skill to expect from the interviewer

The researcher has to be able to listen attentively to what is being said, to leave the interviewees to recall their experience in depth (Gill et al. 2008). A positive relationship needs to be developed during the interviews, with essentially conditions of trust and respect for the interviewee and the information he shares (DiCicco-Bloom 2006). Gill et al. (2008) assess that it is important for the researcher to adopt an open and emotionally neutral body language in order to influence the interviewee as little as possible. Nodding and smiling encourage a climate that favours the responses and encourages the participant to talk more (Gill et al. 2008). To develop some of the topics furthermore, the interviewer can make probing remarks, when appropriate, to seek some clarifications (Gill et al. 2008). At the end of the interview, the participant will be ask if they would like to add anything, this can help to raise issues they could have thought about and that have not been dealt with by the interviewer, this can lead to the discovery of unanticipated information (Gill et al. 2008). The interviewer has to be able to refocus the discussion gently if it does diverge from the topic for some time, with simplicity and courtesy (Gill et al. 2008).

5. Ethical considerations

For confidentiality reasons, the names of the persons and the name of project will be substituted with false ones. These pieces of information are not necessary to conduct the research, so there

should not have a particular ethical issue with this research project. If required by the interviewees, non disclosure agreements can be signed, however the questions will aim at gathering general information about the experiences, which should preserve confidentiality. The recordings of the interviews are to be temporarily saved, and are to be erased as soon as their transcription has been performed to preserve the anonymity and the confidentiality of the participants.

E. Data analysis

1. Thematic and synthetic data analysis

The analysis of the data is a thematic analysis, that looks across all the data to identify the major themes, trends and common issues that recur, and that summarise all the views collected during the research (Quinn Patton 2002). The data analysis aims at generating an understanding of the data, and to identify the patterns for organising text elements (DiCicco-Bloom 2006). According to Quinn Patton (2002), the key stages are:

- reading and annotation of the transcripts: to write down the preliminary observations

- identification of the themes: highlights the themes refereed to by the interviewees and identify their essence

- development of a coding scheme: listing the themes and 'codes' that are to be applied to the data, such as context of the project, constraints, conditions. Codes can be divided in sub-codes to sub-categorise even further

- coding the data: this consists to apply the codes to the whole data,

to offer a new vision on the data through the codes

DiCicco-Bloom (2006) also encourages the coding approach, which is called the 'template approach' as it involves applying categories based on prior research and theoretical perspectives. This helps the researcher to make interpretative statements during the process of themes and patterns identification (DiCicco-Bloom 2006). Factors such as level of enthusiasm, and attitudes of the participants are also be relevant information and need to be taken into account during the data analysis phase (Quinn Patton 2002).

2. Analysis of the data through Rogers' frameworks

The thematic and synthesis analysis output data will then be analysed through Rogers' Diffusion of Innovation framework, which is considered by scholar, and as seen previously in this document, as the flagship of the Diffusion of Innovation theory, and that is used very frequently in the field of Information Technologies. This step of the analysis aims at giving a new understanding of the data through Rogers' model, it will permit to observe the particularities of the data, and their similitudes with the theory of Diffusion of Innovation in the general, as Rogers describes it.

3. Limits of analysis

The analysis performed during the study is an exploration of the experience of a few managers only, it will not permit to conclude any general rule regarding the Software Innovation Diffusion, even in potentially similar environments (DiCicco-Boom 2006, Quinn Patton 2002). However, it can provide useful insights and a list of recommendations to stakeholders involved in projects of similar nature (Gill et al. 2008).

F. Steps of the research methodology and planning

Here is a summary of the steps that compose the research methodology:

Data Collection	Data Collection
Analysis – step 1	Thematic and synthetic data analysis
Analysis – step 2	Analysis of the data through Rogers' model
Conclusion	Summarised answers to the questions and generation of the list of insights and recommendations

Summary of the steps of the research methodology

The research project extends from the 17th of March until the last week of August. The roadmap is pretty straightforward:

May and June	July	August
- conduction of the interviews - transcribing of the interviews - preparation of the Dissertation structure	- analysis of the transcriptions, identification of the themes and codes, etc. - drafting	- writing - finalisation of the Dissertation

Research planning

VIII. Data analysis

A. *Profile of the interviewees and course of the interviews*

This section aims at providing details about the profile of each individual interviewee, and about their attitude during the interviews, that according to Quinn Patton (2002) can be taken in account for the analysis of the data.

1. Number of the interviewees

Quinn Patton (2002) suggests that the sample sizes are typically small in qualitative researches, and that the researcher should continue to interview more people until nothing new comes from the data, a point called 'saturation'. At the forth and fifth interviews, no real new information was coming in the data so the researcher has decided to stop interviewing after the fifth interview.

2. Actual selection of the interviewees

The interviewees are the managers that have been selected randomly from the organisational chart in the department targeted by the research, they have been contacted by email and the interviews have been performed with the managers having accepted to participate to the interviewing process. The questions were attached to the emails so that the candidate knew perfectly what would be discussed.

3. Profile of interviewee A

The interviewee is a male subject of approximatively thirty five years old, and works in the Republic of Ireland, and has been working within the company for about six years. The interviewee had

prepared the interview in advance, and had notes ready for the moment of the interview, he was very eager to share his views about the subject, and the record lasted about forty minutes. The interview has been performed in person, in the office of the interviewee.

4. Profile of interviewee B

The interviewee is a male subject of approximatively thirty three years old, and works in Italy, and has been working within the company for about seven years. This interviewee also had prepared the interview in advance, and also had notes ready for the moment of the interview. The interviewee was visibly enthusiastic about the subject, he has been very dynamic and has gone through the questions in about thirty minutes. The interview has been performed by phone at the office hours.

5. Profile of interviewee C

The interviewee is a female subject of approximatively thirty eight years old, and works in the United States of America, and has been working within the company for about four years. She was apparently stressed at the moment of the interview, probably because of work pressure, she had a quick read to the question beforehand but she has been quickly going through the question for a total of about twenty minutes. The interview has been performed by phone at the office hours.

6. Profile of interviewee D

The interviewee is a male subject of approximatively forty five years old, and works in the United States of America, and has been working within the company for about fifteen years. The interviewee was very

eager to share his past experiences, he has gone through the questions in about forty minutes. The interview has been performed by phone at the office hours.

7. Profile of interviewee E

The interviewee is a male subject of approximatively fifty years old, and works in the Republic of Ireland, and has been working within the company for about fifteen years. The interviewee was welcoming and glad to share his experience, the interview has lasted for about half an hour. The interview has been performed in the office of the interviewee.

8. Conclusion

Most of the interviewees have been enthusiastic to share their experience, and the course of the interviews has been very interesting, insightful and dynamic. As we will see through the analysis of the data, the individual interviews seem to have very few contradictions with the others, they actually seem to complete each others very well.

B. Thematic and synthetic data analysis

1. Coding scheme

This section describes the process of elaboration of the coding schemes, and lists in a table the themes and code used in the process of analysis of the data.

a. Process of code creation

The analysis of the data is a thematic analysis as described by Quinn Patton (2002), it consists to insert markers in the data contained in

the transcripts, in order to be able to sort their content by themes, which are themselves sub-divided into codes. The first step of the codification process has been to generate an initial version of the table of codes and themes, based on the table of "*Topics and factors of the Diffusion of Innovation*" from this document, which contains the core themes in the literature during the literature review. The second step consisted to mark the data in the transcripts with these codes, and to add more markers to the table along the flow, as new codes and themes were identified. Finally, once the codification process was completed, the data of the interviews were fragmented into pieces, the pieces of information have been sequentially searched, code by code, inside the transcription texts, and then group together. Then, the data group by theme and code could be analyse to extract synthetic information they contain, and to confront and compare them with each other.

b. Table of Codes and Themes

Here is the table of the codes and themes used in the process of codification of the data:

	Description	Code
I	***Environment***	
	Social	ENV_SOCI
	cultural	ENV_CULT
	political	ENV_POLI
	rituals	ENV_RITU
	norms	ENV_NORM
	environmental factors related to the nature of the industry	ENV_INDU
	organisational behaviour	ENV_ORGB

	institutionalisation	ENV_INST
	Technological environment	ENV_TECH
II	***Innovation***	
	Recognition of the need to innovate	INN_NEED
	Nature of the new solution: complexity	INN_COPL
	Nature of the new solution: compatibility	INN_COMP
	Nature of the new solution: observability	INN_OBSE
	Nature of the new solution: trialability	INN_TRIA
	Empowerment of the users by the innovation	INN_EMPO
	Perceived advantage of the innovation, and evaluation of the performances of the innovation	INN_ADVA
	Problematic of the design of the innovation	INN_DESI
III	***Diffusion process***	
	Communication channels	DIF_CHAN
	Prediction of time and resources needs	DIF_RESO
	Preparation to risk factors	DIF_RISK
	Agent of promotion effort, advertising and training	DIF_ADVE
	Active implication of the stakeholders	DIF_ACST
	Dealing with the uncertainty, risks	DIF_UNCE
	Resistance to innovation	DIF_RESI
	Acceptance of the innovation	DIF_ACCE
IV	***Aftermath***	
	Understanding the causes of project failures	AFT_FAIL
	Conditions that can favour success	AFT_SUCC
	Use of intuition under uncertainty	AFT_INTU
	Reaction to warning signs that the project can fail	AFT_WARN
	Use of theory and of models in the field	AFT_THEO

Table of codes and themes used for the analysis of the data

2. Presence of the Themes and Codes in the data

The following table highlights the themes that the interviewees have judged to be the most important for the projects of Diffusion of Software Innovation. The aims of this table is to give a visual overview of the way the selected topics are spread within the data of the interviews. It is interested to notice that some themes are covered by all the interviewed managers, such as the social environment, the active involvement of the stakeholders in the diffusion process, and the problematic of resistance to change. Other themes however do not seem to be perceived as being core by all the interviewees, such as the political inter-links in the hierarchy, the complexity, observability and trialability of the innovation, or the preparation of the project to the risk factors.

Interview ->	A	B	C	D	E
Codes I			***Environment***		
ENV_SOCI	✓	✓	✓	✓	✓
DIF_ACST	✓	✓	✓	✓	✓
ENV_POLI	✓	✓			✓
ENV_INDU	✓			✓	
ENV_ORGB	✓	✓	✓		✓
ENV_INST	✓				
ENV_TECH	✓	✓		✓	✓
II			***Innovation***		
INN_NEED	✓	✓			✓
INN_COPL	✓	✓	✓	✓	
INN_COMP	✓	✓		✓	
INN_OBSE			✓	✓	
INN_TRIA				✓	
INN_EMPO	✓	✓			

INN_ADVA	✓	✓	✓	✓	✓
INN_DESI				✓	✓
III	***Diffusion process***				
DIF_CHAN	✓		✓	✓	✓
DIF_RESO	✓		✓	✓	
DIF_RISK	✓	✓			✓
DIF_ADVE	✓	✓	✓	✓	
DIF_UNCE	✓	✓	✓		
DIF_RESI	✓	✓	✓	✓	✓
DIF_ACCE	✓		✓	✓	
IV	***Aftermath***				
AFT_FAIL	✓	✓	✓	✓	✓
AFT_SUCC	✓	✓	✓	✓	✓
AFT_INTU	✓	✓			
AFT_WARN	✓				
AFT_THEO	✓	✓			

Code coverage of the data of the interviews

C. Synthesis of the data

This section of the document retraces and synthesises the main insights and recommendations found in the interviews, and for each of them, it specifies how many of the interview are in concordance and agree on them. This cross-analysis permits to identify the aspects and themes where the majority of the interviewees agree, it also emphasises some divergence in their perspectives. This section of the document is articulated around the themes and codes used for the analysis of the data.

1. Environment

Three interviewees agree that the identification of the stakeholders and their interests, understanding their likely position on the project, and the organisational context with their political interdependencies is the first and primary key problematic to be considered for any project of diffusion of an innovation, and its understanding is vital to achieve success. A diffusion project is “all about Humans” that is the social system environment, purely technical issues are often though as being much easier to deal with (four of the interviewees agree on these points). Three interviewees agree that the compatibility of the innovation with the technical environment and the existing workflow have to be carefully ensured though.

2. Innovation

a. Design of the innovation

One interviewee states that sometime there is a clear business need for an innovation but designing the solution can be difficult, two of them agree that it is often easy to identify the problem but it is usually harder to figure out the innovation to address it though. Three managers agree that often, people do not know how to realise the innovation properly, which can lead to over complexity or to a failure of the innovation to fulfil the needs properly.

b. Perceived value

The vast majority of the managers interviewed (four of them) argue that it is mandatory that the innovation brings a clear value to the business for the diffusion to succeed, and that, obviously, it must bring more value to the company than its total overall cost.

c. Complexity

Four interviewees agree that the intuitiveness of an innovation increases the ease with which the users can figure out how they can use it to their advantages, this can greatly help the acceptance and the adoption. However, one argues that it is not always easy to achieve an intuitive design. Complexity is indeed a key criteria for the adoption of an innovation, and if the innovation is too complex to use, resistance to change can be very strong. Perceived complexity can often be at least partially reduced with well build documentation and with proper training materials according to three of the managers, and with an intuitive design for one of them.

3. Diffusion process

a. Acceptance

One of the five interviewees states that a project has generally better acceptance when it is pushed by users themselves, as they understand very well their own needs for their daily routine, and therefore they are at the best place to advertise the innovation. According to four of the managers, the acceptance and for the success of the process of diffusion of the innovation, the end users and the hierarchy have to perceive clearly the relative advantage in their daily routine, and one of them adds that the project manager has to keep in mind that different categories of users will have different expectations.

b. Active implication of the stakeholders in the process

The totality of the interviewees agree on the fact that stakeholders have to be actively involved in the diffusion process, that they need to

feel listened to and they need to be empowered in the process, their acceptance is much better when their concerns and perceptions are questioned and answered to in a forth and back loop, and when there is a way for the users to give feedback during the deployment of the innovation. The fears of the end users and other stakeholders need to be addressed quickly for four of the interviewees, and all the interviewees agree that the personal advantages that each individual stakeholder will gain in their daily routine need to be highlighted to increase their acceptance.

c. Advertising and training

All the interviewees are in accordance on the fact that the advertising and the training of the end users is essential in order to increase their acceptance and to reduce their resistance, but also to reduce the perceived complexity. Similarly, three of them add that the lack of training can increase the perceived complexity of the innovation, and can therefore increase the resistance. One manager emphasise that in very dynamic industries like the Software industry, innovations are more easily accepted than in others. However, two interviewees state that if the innovation is "wrong", because it does not bring the value it is supposed to, the resistance from users will be consistent over time. One of them states that the advertising is more efficient when performed through the bias of different channels simultaneously, such as social networks and the Intranet, but three of them agree that it will be more convincing and efficient through demonstrations and workshops to the end users, and by mouth to hear channels, and all the interviewees agree that the most persuasive media is from mouth to ear from already convinced users. Four managers precise that a transition period has to be planned between the existing process or

solution and the new one, in order to give enough time to the end users to get used to the innovation, and to learn the new way to do things, two of them also highlight that a transition period can also help to address the potential issues that can come with the uncertainty induced by the innovation.

4. Aftermath

a. Use of intuition under uncertainty

According to one interviewee, the way managers use intuition in situations induced by uncertainty comes from their knowledge, past experiences and common sense, the intuition generally evolves with the experience. Three of them agree that the uncertainty can come from many places and at any time, such as reorganisations and department mergers, or accidents to important stakeholders. However, two managers argue that the problem of uncertainty can be partially addressed by writing a risk management plan of actions, to list as many risks as possible that can be encountered during the project, and to detail what needs to be done for each. Some risks are hard or impossible to anticipate though, according to three interviewees. Two of them insist of the fact that it is important not to underestimate anything and to try to test and validate everything that can be in a context as close to the target as possible, such as the tools, migrations, deployments, etc., and that assuming things will just "work" without verification is a common cause of failure.

b. Causes of failure

Three managers state that in many cases of project failure, there were diverse warning signs such as successive extensions to the planning, or disagreements between departments. Two managers argue that

some of these failure could have been addressed but it would have required a lot of social or political skills to break the taboos and to deal with the politic inter-links, and the vast majority of the interviewees agree on the fact that many managers with technical backgrounds such as in the Solaris Operating System Engineering department of Oracle do not have these skills. Three interviewees state that other warnings such as information about imminent reorganisations can sometimes be predicted, but they are hard to address, if not by interrupting the project altogether. Three interviewed managers argue that taking the courage to make the right decisions is sometimes hard because of the pressure on the project from the hierarchy, but it is often better than ignoring the warnings, even if it means putting the diffusion project to a stop. Four interviewees argue that an under-estimation of the human, time or funding resources required for a project seem to be a common cause of failure for diffusion projects, so it is important to have security margins to be able to address these unexpected issues induced by uncertainty. As one of them emphasises, in the case of resources cuts, it is important to re-negotiate a re-scaling of the project in accordance with the hierarchy, by cutting functionalities off for example, and to formalise what has been agreed. One of the other interviewees also argue that sometimes, projects are promoted for political reasons, pushed by individuals to promote themselves for example, and in these cases the benefits of the pretended innovation can be non-existent.

c. Usage of theory

One of the managers insist on the fact that theory can help reducing the uncertainty, by following very formal and precise steps in the

diffusion process, with methodologies like PMP or Prince, or with a firm-made theoretical framework built upon experience of previous similar projects and improved over time.

5. Conclusion

The codification and cross-analysis of the data shows that while the answers from the interviewees are different, they are not contradictory but in fact they complement each other. The most powerful message out-coming from the data seem to be that a Diffusion of Innovation is a very human and social process, and its success depends on the ability of the stakeholders involved to communicate properly. The managers agree that in the research environment, and in order to succeed, the Diffusion process has to very actively involve the stakeholders in the process, with a clear line of communication between the project initiators and the stakeholders, and this line has to be bi-directional. Users' inputs have to be taken in account, their fears needs to be addressed, which means they need to be listened to and answered to about their concerns, and users' have a more positive perception of the innovation if their opinion has a visible influence on the project. The other very important aspect discussed in the data is how the viability of the project of Diffusion of Innovation is important to its success and to the existence of the innovation in the first place, that is the project has to have a measurable added value over the tool or framework it supersedes, so that Management can evaluate the gains and performances. The advertising and training around the project are judged by the interviewees as being very important, the benefits that each user will gain in their daily routine have to be emphasised in order to favour their acceptance of change, while training helps

them in their transition to use the innovation, and in reducing the perceived complexity. The uncertainty is present at every stage of the Diffusion process, and while it can partially be addressed with risk management techniques, it can never be completely taken away, and unfortunately some events induced by uncertainty always remain unpredictable. Despite of the fact that sometimes events issue by uncertainty raise warning signs, in about half of the case the issues cannot be countered, and remain a fatality that lead the project to failure. Finally, the analysis of the data shows some significant differences and similarities with Rogers' model, these are detailed in the following section.

D. Analysis of the data through Rogers' theoretical framework

Even at the early attempts to fit the Software Diffusion Projects at Oracle into Rogers' Diffusion of Innovation theoretical framework, some differences and similarities start to appear very quickly and very clearly. The course of the diffusion projects as detailed in the interviews have difficulties to match strictly the steps of the innovation-decision process as defined by Roger. Also the adopters categories to be found in Rogers' model, as well as the criterion that impact the Diffusion of Innovation, do not seem to be completely adapted to the research environment. This section of the document analyses the diffusion projects described in the interviewees experiences, point by point through Rogers' D.O.I. framework, in order to perform the emphasis on the similarities and disparities between both.

1. The Innovation-Decision Process

a. The knowledge stage

This stage is the stage where stakeholders learn about the existence of the innovation and seek to get an understanding of it (Rogers 2003). According to the interviewees, there are two categories of innovations in the studied environment. In the first category, the innovation exists and has been developed by an external company or internally within an other department of Oracle. This case much closer to Rogers' model, in the way that the users learn about an existing Software Innovation and what it is meant to do and about how it is meant to be used (Sahin 2006, Roger 2003). However, in the second case, which appears to be the most common situation according to the data, the individuals start to learn about of the Software Innovation even before its existence. In this case, the future users actually start learning about a need to innovation, with the recognition of a problem to solve, and so, while the solution to the problem is being discussed by the involved stakeholders. The project starts when the initiators of the change seek to shape the solution to fulfil the need, before any development has started at all. Roger (2003) assumes that at this stage the innovation already exists, thus, the second case constitutes a very significant difference with what is described in Rogers' theory.

b. The persuasion stage

The persuasion stage is the stage where the individuals adopt a favourable or unfavourable attitude toward the innovation (Roger 2003). Again, in the research environment, we can distinct the same two situations as in the knowledge stage. The first situation is fairly close to what Roger (2003) describes about the discussion and

exchanges of opinions about the innovation. However, in the second case, the innovation does not really exist yet, it has the particularity to be only existing as a concept. Thus, the favourable or unfavourable attitude developed by the individuals is primarily based on their belief that the conceptualised solution is the right answer to the problem or not. They can only imagine if the concept could work, without actually being able to test the innovation. The majority of the interviewees consider that it is very important to actively involve all the stakeholders at this stage, and to take their inputs, their suggestions, and their fears into account, and to address them. Indeed, the individuals are more likely to have a favourable attitude toward the Software Innovation Diffusion project if they feel that the solution will help them in their own daily routines, if they feel listened to when they mention their concerns, and if they can verify that their inputs are taken into account in the while process.

c. The decision stage

The decision stage differs significantly from Rogers' model, as in the research environment, users do not have the freedom to adopt or reject the innovation, as this choice is a corporate decision. Indeed, the decision is to be taken by the Management, by the hierarchy, even though the initial need for an innovation, or for a framework improvement, is generally first identified at the end user level. According to the data, what usually happen is that the end users identify a need or an issue in their daily routine, then this issue is reported to the Management. If Management judge that solving the issue brings value to the business and to company, they decide the innovation is needed. From that point, the initiators look to realise a Software Innovation to address the need, and they seek information

to design the actual solution from the stakeholders. Three of the interviewees judge that the end users are indeed at the best place to give information about the sort of solution that is needed. The other significant difference with Rogers' model is that the decision stage does not happen right after the knowledge stage and the persuasion stage, but before these two stages. The conclusion of this section of the document recapitulates and compares the stages and their order as they happen in Rogers' model, and as they actually happen in the research environment according to the data.

d. The implementation stage

The implementation stage is the stage where the innovation is put into practice and is used by the individuals (Sahin 2006, Roger 2003). This stage happens in the research environment once the innovation has been realised by the initiators. As described in Rogers' D.O.I., during the implementation stage, also commonly called "deployment stage" by the interviewees, uncertainty can still be an issue, and users might need support and technical assistance with the innovation (Sahin 2006, Roger 2003). At this stage, especially at the beginning of it, the end users actually start to use the software innovation in their daily routine, and according to the data, they often discover unexpected issues, such as for example unexpected technical incompatibilities, or missing features that were present in the previous tool but that are not covered by the innovation. Indeed, at that stage, the innovation starts being used in the real environment, and it is the point where the theory of the innovative design meets the reality of the field and of the daily routine of the users. Success and failure of the diffusion project, at that stage, depends on the ability of the responsible Manager and his team to

rectify the issues within the constraints of resources attributed to the diffusion project, which are essentially human resources, funds, and time boundaries.

e. *The confirmation stage*

According to Roger (2003), the confirmation stage is the stage where users look for support for their decision to have adopted the innovation. In the research environment, however, which is a corporate environment, the confirmation stage is more a stage where the hierarchy seek to know if the diffusion of the innovation was a success or not. While the confirmation as described by Roger is based on an emotional response to users from their environment regarding their own choices (Roger 2003), in the corporate environment, the hierarchy and Management look for scientific and measurable proof that the innovation fulfil its role, and addresses the need. That is, to evaluate of the total cost of the project is covered by the value brought to the business by the innovation, if it permitted to reduce the costs, or to improve the productivity, depending on the context where the innovation is diffused. Thus, and due to the business environment in which the Diffusion of the Innovation occurred, the confirmation does not follow the “emotional” response from the user to his environment as described in Rogers' framework.

2. The adopters categories

Rogers' adopter categories define the “classification of members of a social system on the basis of innovativeness” (Roger 2003). This classification is simply not applicable in this corporate environment, as users are not free to adopt or not the innovation, or to choose the moment at which they do so, this decision being done at the

Management level. Indeed, the willingness of the users to change their habits and routine does not count in their adoption of the innovation.

3. The criterion impacting the diffusion of innovation

According to Roger (2003), some criterion or traits have a direct impact on the diffusion process, and Roger (2003) assesses that these have an influence on the rate of adoption. This section of the document reviews each of the criterion as defined by Roger and describes for each of them what the data teaches us about their importance and impacts in the studied environment.

a. Relative advantage

The relative advantage is the value the innovation adds over the previous tool or framework it supersedes. The interviewees seem to agree on the fact that, for the innovation to be worth being deployed, the value its brings to the business must be superior to the total cost of the project. The benefit it brings to the company has to be measurable, it can be materialised into diverse gains such as a reduction of the costs, an augmentation of the productivity, a reduction of the volume of human time to do repetitive tasks, etc. Thus the relative advantage in the research environment is perceived by the stakeholders as a very rational criteria. However, in Rogers' (2003) theory, the relative advantage is the advantage perceived by the users, which means it can also be non productive advantages such as a better aesthetic if we take the example of a new mobile phone over an older one. The relative advantage as perceived in the studied environment is compatible with Rogers' definition, it is just more precise.

b. Compatibility

Rogers (2003) defines the compatibility as “the degree to which an innovation is perceived as consistent with the existing values, past experiences, and needs of potential adopters”. Here again, in the research environment, the meaning of compatibility seems to be consistent with Rogers', however it is just more precise. According to the interviewees, the compatibility criteria of an innovation is essentially defined as its ability to fit into the existing workflow, and to work hand in hand with the technical environment. As Sahin (2006) emphases, the compatibility is a crucial criteria for the usage of Software Innovations in the field of Information Technologies.

c. Complexity

The complexity is defined as the “degree to which an innovation is perceived as relatively difficult to understand and use” (Roger 2003). The data describe the complexity level of a Software Innovation as being a fundamental factor that can lead to its success or failure. The interviews teach us several interesting things about the targeted environment: first, the complexity of an innovation has a direct impact on users' resistance, a very complex Software Innovation is likely to induce a lot of resistance, but similarly, an intuitive innovation, that is, an innovation that users can use with very few learning, can speed up its adoption greatly. Second, the complexity and the users' resistance that come along, can often be partially addressed with good quality training and documentation.

d. Trialability

The trialability of an innovation is defined by Roger (2003) as being the extent to which the innovation can be tested by the targeted

adopters. Rogers' framework assumes here again that the users are free to try and to adopt or reject the innovation, which conflicts with the imperatives of a corporate environment. It is interesting to highlight that the data contains nothing about the trialability of the innovation, none of the interviewees has approached this subject directly or indirectly. It is therefore reasonable to assume that trialability is a criteria that has an influence only in a context where users have the freedom of adoption. Thus, this is a point where Rogers' theory does not apply to the research environment.

e. Observability

The observability of an innovation is the ease with which the innovation can be evaluated after trial (Roger 2003). The data show that observability might be an even more important criteria in a corporate environment than in the general case described in Rogers' framework. Indeed, it is decisive that the performances of the Software Innovation are measurable, because it is the base criteria for the management that pushed the Diffusion of Innovation project to evaluate the viability of the whole project. Just like the observability criteria, the observability is key to control the extent to which the innovation fulfils the role it has been designed and diffused for.

4. Conclusion

The analysis of the data through Rogers' D.O.I. theoretical framework has permitted to verify the fact that while D.O.I. can provide guidance to the Managers involved in Diffusion of Software Innovation projects in the research environment, some significant differences exist with Rogers' model. These differences fall into two categories: the differences in the Innovation-Decision process, and

the differences in the criterion impacting the Diffusion of Innovation.

a. Differences in the structure of the Innovation-Decision process

First, the steps of the Innovation-Decision process and their order differ. In the data, the decision stage happens even before the knowledge stage in Rogers' theory, because in the research environment, the decision to adopt or reject the innovation comes from the hierarchy, which means that the decision is made even before the end users are involved in the process, as in a classic corporate decision. However, in Rogers' theory, the users are free to adopt or reject the innovation, so they perform their choice only after both the knowledge stage and the decision stage, which explains the different position order of the decision stage.

The other significant difference is that in Rogers' framework, Rogers assumes that the innovation exists before the diffusion process, whereas in the mainstream case as described in the data, the future users come to learn about the innovation even before its existence, when they get themselves involved in the innovation design stage, their inputs being probed by the initiators of the project. It is interesting to note that in Rogers' model, the innovation design and development stages are not included into the Innovation-Decision Process, because Roger assumes that the innovation already exists at that stage, whereas in the research environment, these two stages are completely integrated to the process as the realisation of the innovation is a part of it.

The following table recapitulates the differences in the Innovation-

Decision process between Rogers' model and the research environment:

<table>
<tr><th>Rogers' D.O.I. framework</th><th colspan="2">Actual steps as followed in the research environment</th></tr>
<tr><td>-</td><td colspan="2">recognition of the need to innovate</td></tr>
<tr><td>-</td><td colspan="2">decision stage</td></tr>
<tr><td>knowledge stage</td><td>knowledge stage</td><td rowspan="2">innovation design stage</td></tr>
<tr><td>persuasion stage</td><td>persuasion stage</td></tr>
<tr><td>decision stage</td><td colspan="2">development (realisation) stage of the innovation</td></tr>
<tr><td>implementation stage</td><td colspan="2">implementation stage</td></tr>
<tr><td>confirmation stage</td><td colspan="2">confirmation stage</td></tr>
</table>

Comparison of the Innovation-Decision Process as defined in Rogers' theory and to the steps as found in the research data

b. Differences in the criterion impacting the Diffusion of Innovation

Rogers' (2003) model defines five criterion which impact directly the Diffusion of the Innovation detailed previously in this document: the relative advantage, the compatibility, the complexity, the trialability, and the observability. In the research environment, only four of them seem to actually have an impact. The trialability of the innovation, that is the extent to which the innovation can be tried before the

adoption decision (Roger 2003), does not count in the Diffusion of Innovation in the research environment for two reasons: first the decision is generally performed by the hierarchy to create and diffuse the innovation before its existence, so the trialability does not impact the adoption. And second, still in the research environment, users are not free to adopt or reject the innovation anyway. While the other four criterion count in both cases, there is however a difference if the way the relative advantage and the observability are appreciated. In Rogers' framework, both criteria are based on the perspectives from the users, which means from what they feel, that is on their feelings about the innovation. However, in the research environment, these two criterion have to be rational, and measurable, in order for the Management to be able to measure the performances of the innovation, and the real gains and value it brings to the company. Rogers definition of the compatibility is also not really precise, as he defines it at a “degree to which an innovation is perceived as consistent with the existing values, past experiences, and needs of potential adopters” (Roger 2003). However, in the research environment, the definition of the compatibility is very precise, as it is the ability of the innovation to be integrated within the existing corporate framework and technical environment.

The following table recapitulates the importance of the criterion impacting the Diffusion of Innovation in both Rogers' framework and the research environment:

Criteria	Rogers' D.O.I.	Research environment
Relative advantage	✓ perceived	✓ measured
Compatibility	✓ perceived	✓ with corporate framework and technical environment
Complexity	✓	✓
Trialiability	✓	✗ irrelevant
Observability	✓ perceived	✓ measured

Confrontation of the criterion impacting the Diffusion of Innovation in Rogers' theory and in the data

Finally, some notions like the “time of adoption of innovations” are simply not applicable at all in the research environment, indeed such a notion assumes that the users are free to adopt the innovation at the moment they decide to, which permits to define categories of adopters. However, in the context of a corporate environment, the adoption is not free as it is imposed to users by the hierarchy.

IX. Discussion

A. Retrospective on the research process

Globally, the research process has been very smooth, no particular difficulty has been encountered during the project, accessing to the data and finding a sample of individuals to get the information from has been relatively easy to do for the researcher, probably because it was performed in his work environment, and very few managers have refused to participate to the interviews. Submitting the questions to the interview twenty four hours in advance has proven to be efficient, as the interviewees had the time to think about the questions and to ask for some clarifications when required, the data collected was pretty much in line with what was expected and there have been relatively few divergence from the research subject during the interviews. Moreover, most of the interviewees have been very enthusiastic to share their views and their experience in the domain, it was an enjoyable moment for both parties, which probably favoured a better quality for the data. The transcription process has been surprisingly very long, and even longer than thought it would be initially, and it was sometimes difficult to understand precisely the words used by the interviewees when playing back the interview records. Also, sometimes in the records, sentences were truncated, or built incorrectly, no because of technical issues but by the nature of a spoken language, the transcription process as therefore needed to perform some cleanup of the data. The codification process and the thematic analysis as been relatively easy to do, the arguments from the interviewees were pretty clear and very in line with each other, also they were very complementary and these was very little

opposition in the data. Analysing the data through Rogers' framework has permitted to understand why it is a considered as a reference in the domain of the Information Technology, even though the generalist aspect of this framework does not permit a tight control of the Diffusion of the Innovation process in the research environment, as seen in the analysis of the data, and even if the research environment has particularities that lead to some variations in the application of Rogers' model, Rogers criterion impacting the adoption of the innovation and the innovation-decision process remain for the most part valid.

B. What to think about the findings

As Tidd et al. (2005) and Green & Hevner (2000) argued, the Diffusion of Software Innovation problem must include the human and the technical dimensions. However the findings insists much more on the human aspects, and on the need of a very intense flow of communication around the diffusion project with an active involvement of the stakeholders along the whole duration of the diffusion process. The synthesis of the data has demonstrated that indeed, the interviewees seem to think that most of the technical issues can be address much more easily than social or political issues around the project, and some managers did not even bother to mention the technical environment. The findings really bring light on the gaps identified in the literature, they show how the projects of Diffusion of Software Innovation require a plurality of competences, mainly social but also technical, economical and managerial. They also give useful insights on where the uncertainty comes from and how it is address by Managers on the field. Also the return on experiences of the interviewees on the cause of failures has permitted

to highlight their diversity in natures and in the possibility to be predictable and to be solvable. Indeed, some of the cause of failures can never really be countered. The finding are very much in line with what was expected at the beginning of the research project.

X. Conclusion

This paper has given a list of insights and recommendations to stakeholders involved in Projects of Diffusion of Software Innovation in Information Technologies environments. The themes and questions that managers recognise as being core to the success or failure of Software Innovation Diffusion are, as initially expected, of very diverse natures. Costs and rentability, preparation to the uncertainty with risk management, political inter-links, a managers involved in such project has to consider many different aspects simultaneously and needs multiple competencies to be able to deal with them at the same time. The exploration of the problematic situations induced by uncertainty has permitted to get very useful insights about where the contingency comes from, and on the fact that the innovation is not the only source of uncertainty, it can also come from the contingency in the environment, with accident happening to important stakeholders of the project, with unexpected reorganisations that can put at risk the existence of the whole project, etc. The exploration of the reason for project failures has been very insightful too, it has demonstrated that there is a large number of possible reasons for failure, some are controllable and rise warning signs that permit their identification to enable corrective actions, some are hard to control and require particular skill such as political issues between different actors of the project, some others are just a

fatality and nothing can really be done to address them, like in the case of departments mergers for example. However, the majority of the interviewees seem to agree that failures are often due to under-estimation of resources, and to a lack of communication around the project.

While the findings may be regarded as being very specific in the field of the Diffusion of Innovation theory because of the singular nature of the research environment, the insights and recommendations detailed in this research project might be useful to individuals working on Diffusion of Software Innovation projects in similar Information Technology environments. Rogers model, which is the flagship of the theory of the Diffusion of Innovation, could be used in the process of analysis of the data, however it needed adaptations in regard of its generalist nature, to fit the constraints and particularities of the studied business environment. Possible area of investigation for furthermore work could be to study how Rogers' Diffusion of Innovation theoretical framework could be generalised to the Information Technology environments.

XI. References and Bibliography

Al-Hakim, L., Jin, C. (2010) *Innovation in business and enterprise: technologies and frameworks*. Hershey, New York: business science reference.

D'Aniello, A., Masone, A., Tammaro, A. (2006) 'Technological Innovation within EDS Italia Software: Experience Report', *Proceedings of the Conference on Software Maintenance and Reengineering,* Naples, Italy*,* 22-24 March 2006, 361-366.

Attar, H., Shahabi S.M.R. (2014) Beyond 'Funnel' and 'Fireworks': 'Water Ribbed Balloon' as a New Metaphorical Approach to Innovation-in-Practice;. *Iranian Journal of Management Studies (IJMS)*, 7 (1): 67-93.

Bryman, A. (2012) 'Interviewing in qualitative research'. In: *Social Research Methods*. 4th ed. Oxford: Oxford University Press: 312-333.

Damsgaard, J. (1994), 'How Information technologies penetrate organizations: an analysis of four alternative models', *In Levine L. Diffusion, Transfer and Implementation of Information Technology, Proceedings on the IFIP TC8 Working Conference*, North Holland, Amsterdam, The Netherlands, 1-22.

Deshipande, R. (1983) 'A comparative review of innovation diffusion

books', *JMR, Journal of Marketing Research*, August 1983, 20: 327-335.

DiCicco-Bloom, B., Crabtree, B. F. (2006) 'The qualitative research interview'. In: *Making sense of qualitative research*. Blackwell Publishing, 40: 314-321.

Gill, P., Stewart, K., Treasure, E., Chadwick, B. (2008) 'Method of data collection in qualitative research: interview and focus groups'. *British Dental Journal*, March 22nd, 204 (6): 291-295.

Gray, P. (2006) 'The State of Software Development'. *Information Systems Management*, 15 (1): 88-92.

Green, G., Hevner, A. (2000) 'The successful diffusion of innovations: guidance for software development organizations'. *Software, IEEE*, 17 (6), 96-103.

Hall, B. (2003) 'Innovation and Diffusion' Mowery D., and Nelson R. *NBER working paper series, working paper 10212*, National Bureau of Economic Research, Cambridge, Massachusetts, January, 2004, 4-25.

Kautz, K., Nielsen, P. (2000) 'Implementing Software Process Improvement: Two Cases of Technology Transfer'. *System Sciences*

conference in Hawaii, January 4-7 2000.

Quinn Patton, M. (2002) 'A guide to using qualitative research methodology', *Qualitative Research & Evaluation Methods*. 3Rd ed. London: SAGE Publications, 207-339.

Lundblad, J. (2006) 'A Review and Critique of Rogers' Diffusion of Innovation Theory as It Applies to Organizations'. *Organization Development Journal*, Organizational Development Institute. Washington D.C. , January 1 2003.

Mann, L., Chan, J. (2011) *Creativity and Innovation in Business and Beyond, Social Science Perspectives and Policy Implications*. New York: Routledge.

Mustonen-Ollila, E., Lyytinen, K. (2003) 'Why organizations adopt information system process innovations: a longitudinal study using Diffusion of Innovation theory'. Information System J, Lappeenranta: Finland, Blackwell Publishing Ltd, 13: 275-297.

Nahar, N., Käkölä, T., Huda, N. (2002) 'Diffusion of Software Technology Innovations in the Global Context', *International Conference on System Science in Hawaii, January 7-10 2002*, 2749-2757.

Oliveira, T., Thomas, M., Espadanal, M. (2014) 'Assessing the determinants of cloud computing adoption An analysis of the manufacturing and services sector', *Information & Management*, April 6 2014, 51: 497-510.

Tidd, J., Bessant, J., Pavitt, K. (2005) *Managing Innovation, Integrating Technological, Market and Organizational Change Third Edition*. Chichester, England: John Wiley & Sons Ltd.

Raghavan, S., Chand, D. (1989) 'Diffusing software-engineering methods'. *Software, IEEE*, July, 6 (4): 197.

Rogers, E. (2003) 'Diffusion of Innovations (5th ed.)', New York: Free Press.

Sahin, I. (2006) 'Detailed Review of Rogers' Diffusion of Innovations Theory and Educational Technology-related Studies based on Rogers' Theory', *The Turkish Online Journal of Educational Technology*, April, 5 (2) art. 3.

Schein, E.H. (2010) *Organisational Culture and Leadership, fourth edition*. San Francisco, CA: Jossey-Bass.

Schuh, G., Aghassi, S., Valdez, A. (2013) 'Supporting Technology Transfer via Web-based Platforms', *Proceedings of PICMET '13:*

Technology Management for Emerging Technologies, San Jose, CA, USA., July 28-Aug 1, 2013, 858-866.

Zeng, D., Wu, C. (2009) 'Decision Support Model and Simulation for Knowledge Transfer in Innovation Networks', *International Conference on Business Intelligence and Financial Engineering*, Beijing, 24-26 July 2009, 519-523.

XII. Annexes

A. Interview A

Yannick: What are the main themes and questions that you identify as being core to the success or failure of Software Innovation Diffusion projects?

Steven: The main criterion according to my experience would be whether project was initiated, pushed, by the users themselves or if it was a project pushed by, lets say the direction, the higher hubs (INN_NEED, INN_ADVA, DIF_ACCE). That is the main critical factor for me, because for a project where the users are the one advertising, pushing, for sure the acceptance of the software innovation is much easier (DIF_ADVE). If the software innovation is pushed by the hierarchy, if it is a top down decision as opposed to bottom up, then you can have a lot of resistance factors to change (DIF_RESI). Let's start by the easy way, with the bottom up diffusion of software innovation, where the users themselves are the initiators. That is the easy way because typically these guys, they know the work-flow, they know what they need (INN_NEED, INN_COMP, ENV_TECH), they are probably at the best place to detect, suggest software innovations, or work-flow innovations, they are at the best place to do the demonstration, the advertising of the interests of the innovation to other users (DIF_ADVE). So it is kind of graceful movements. So in that case the spreading is pretty easy, especially with the new IT technologies with the remote workplaces and social networks that kind of thing (DIF_ADVE, DIF_CHAN). That kind of project of innovation, the main resistance would come actually from

regulatory authorities. So the hierarchy who want to keep some consistency within technologies in the company (ENV_INST, DIF_RESI).

Either request for the unification of technologies, or run it behind the hierarchy and do it anyway. Of course if the software innovation is not that great, not that advantageous for users it will not be accepted by users (DIF_RESI), it is obvious. So that was the easy part. The hard part, the most frequent situation is when the innovation is pushed top-down by the direction, by the chief, the management, in that case there is a huge number of factors will spell the failure of projects. Now if I want to be positive regarding the success of the project, the main factor I would say, regarding the success or failure of the project is whether the users are empowered by the change (INN_EMPO), if they are active stakeholders of the change (DIF_ACST). If the people, the actors, who stand the most benefits from the project are not the users, you have lots of passive resistance (DIF_RESI) and it only through a huge expenditure of efforts that the project will go through (AFT_FAIL).

For example let's say you have a software work-flow which is motivated by regulatory purpose, in this industry it would be like lots of logging software, lots of compliance stuff, users would be highly resistant to that. So it will never be fully accepted (DIF_RESI). It would be accepted only if the management is highly persistent. On the other hand if the change has actual benefits for the users it stands better chances of being accepted (DIF_ACCE), the thing being there

is a significant part of ground work to be done, some marketing if you want (DIF_ADVE). If the changes are obvious then you do not have to do the marketing but usually it is better for the proponents of the technology to do some kind of workshop demonstrations to show the added value in the day to day work-flow (DIF_ADVE, DIF_ACCE). Otherwise people will say: “sorry, I used this software for twenty years, I do not change!” (DIF_RESI). So they need to be kind of taken by hand and shown how wonderful it would be to work with it. Sometimes it's even true (laugh). Then, the second factor is not only having an added value but also how to sell the added values to the guys (DIF_ADVE). There are lots of cautious on these projects, lots of persons that had great ideas but they could not market it, they could not advertise it enough for it, and then they failed (DIF_ADVE, AFT_FAIL). Then after that, there is a third factor but I do not rank factor by importance, I just tell them as they come in my mind, the third factor would be bureaucratic entrust. You can have projects which make perfect sense for the business, which give us lots of added value to the end user but if the project does not please one department which is fairly powerful (ENV_POLI, ENV_ORGB), and which does not see its interest in it they might try to kill it (AFT_FAIL).

So similarly to try to market the idea to the end users, potentially all the stakeholders through the project of the software innovation project, you have to sell it to them as well (DIF_ADVE), which might be difficult, more difficult than for an end user, because you are going to have one department who hates the software, because it mines the current work-flow (INN_COMP, ENV_TECH), because it is very

manpower intensive, but if in the current you have lots of employees, then they have power and if you suddenly decide they will have a new work-flow which will allow us to save ninety percent of the man power, you have to think about what is their interest, that is needed for them, so that they do not actively oppose it (DIF_RESI, INN_EMPO).

Either that or you have to plan for their resistance, and you have to bring some tips for the management so they can force the department to give up, or give them an other carrot elsewhere (ENV_POLI). I have seen that kind of thing several times in my carrier, and there was several projects which made lots of sense, which would add lots of value, but the I.T. department, supposed to do the testing, were not yet ready for that, they had to suffer for one or two more years of failures of operational and production problems with their current solution, before they were ready to accept they needed something better (INN_NEED, DIF_ACCE). So the first iteration of the project failed (AFT_FAIL), it was cured by a coalition of departments and after they were sure enough to accept that they needed it (INN_NEED). They no longer opposed the project (DIF_ACCE). So it is a kind of an extension derivative of making the user department to accept the innovation.

And then the last, and actually the most common factor, that would be the risk reward matrix, you can have a great software innovation but typically it is not always fully ready, fully developed. So you can do a great demo and then the management will ask the dread

question: “Who are the other clients who are using it?” and then for usually for innovation would be: “sorry we do not have any” or “we do not have a big one” (ENV_SOCI, DIF_RESI). If you can say one of your competitors is using it and is loving it, then OK, you have a point (DIF_ACCE) (laugh). If the industry is very competitive, then software innovation would tend to be better accepted (DIF_ACCE, ENV_INDU). If you have major blocks or fairly static, then there will not be a preference to take the risk (DIF_UNCE). This is the main core things that I would push forward as a risk factor.

Yannick: thank you, let's move on to the next question then if you agree.

Steven: Sure!

Regarding the next section, regarding how to deal with unplanned, unforeseen events, we can have many stories, good stories about them. I will just take a few samples about the surprises we had during projects, the things you did not plan for, they can happen at any stage (DIF_UNCE). I will give you an example that happened to one of my bosses, who was leading a very large project of software innovation, and he though he had plans for everything, so he had consulted, contacted every stakeholders (DIF_ACST), but at the end he discovered he forgot one stakeholders. It was a fairly big software innovation project. At the end of the project, that would have caused a merger between two operation departments (DIF_UNCE, AFT_FAIL), and in that case he forgot a small legal requirement, there was a workers union who said: “sorry but if the project can

affect the jobs, then we have to be consulted and we can veto the project" (INN_EMPO). And they did.

Yannick: Wow! (laugh)

Steven: that was at the end of the project, then after he put them within the decision loop, he restarted the project and took into account their wishes. That took about one year and half, and after that, unfortunately, the company's organisation has already changed (DIF_UNCE), and the project was no longer relevant to the environment so double failure (laugh). In that case it was a failure of planning (AFT_FAIL), and in this case he tried to deal with it, with the crisis management teams but the lesson from that story was simply that you should consult and advise everyone that could be affected by the project (DIF_ACST). So obviously there will be many people who say they do not care and more people saying that they do not agree and that the project is a waste, but only after you get their opinions, you can decide if their opinion matters to the project, if they can lead the project to fail (DIF_RESI).

So that was one unplanned situation. There are many accident where you can say it took me by surprise, to activate a crisis management response, but in my experience, half of these cases are simply the planning project management was not done carefully enough (DIF_UNCE, AFT_FAIL), it might have been planned, sometimes the incident, the risk factor you cannot mitigate it (DIF_UNCE, AFT_FAIL), but in that case if you have plans for it, you can say:

"here is a warning sign, we are going to have problems" (AFT_SUCC) and you can either try to reduce the risk or you can pull the plug, stop the project and waste less money (DIF_RESO). The other path of a real unplanned situation would be mostly single points of failures. So the most common problematic situation, crisis situation should I say, is for example, you have one critical guy, critical to a project which suddenly becomes unavailable (DIF_UNCE). The most frequent situation is an accident, like a car accident or anything similar, and he cannot walk any-more, so he is unavailable for the project.

An other common situation is the guy got fed up or has family problems or he got hired by the competition (DIF_UNCE). So in that case let's hope that you had a good project manager and this critical guy, you already have someone else that could stand in for him, as a replacement. He will not be as good as the primer guy but it will be that. If it is a risk that you planned for, it is easier to deal with. On the other hand, I witnessed an other example, we had a critical resource, a database expert, he had a ski accident, and broke his leg. We though it was OK, that he could still come but we found out there was no elevator in the building (laugh). In that case, we installed a web-cam and we put him on the ground floor with a video-conferencing solution. So in that case, it we used common sense to solve it (AFT_INTU).

Steven (reading the next question): Problematic situations: sometimes there are things that should have worked and there is no apparent reason it would fail for, and in the middle of the project, you

discover the rub, that is going to block your project (DIF_UNCE). I have an example: we had a big database migration, that was required for a new data-mining process, in order to do that we have done lots of software preparation and we asked the IT guys our requirements, from an Oracle database to the next version, they thought it would be very easy. We thought we would keep the same structure, same everything so it should work out of the box. And at one point when we started doing the tests, we found out that in some cases it did not work, and it looked quite random (DIF_UNCE, ENV_TECH). We tried the usual check-list, to identify what works, what does not, trying to isolate the source of the issue, and in the end, after three weeks of searching we still could not find an explanation.

So we got a bit creative, and we used our home computers to simulate the same migration process, and we tried to corrupt the data. In the end it was one in a million issue, we discovered that the database ODBC driver had changed behaviour between the versions. If there was a leading space in the field on the new version, it was removed, whereas it was not on the previous version (laugh).

So that was an unplanned event, that was a migration risk that was completely under-estimated and from now on, I really pay close attention to "easy and painless" migrations (laugh) (AFT_INTU).

Yannick: in that case you would say the issue was due to a lack of testing, for the migration process?

Steven: we thought that since we have been sticking to the same software vendor, the migration would be easy and would be guaranteed 100%, and that the behaviour would be documented. We were wrong, the behaviour was not guaranteed, and it was not documented. Regarding the most common causes of failure of software innovation diffusion projects, I already covered some of that ground in my initial answers, I will just try to do a quick recapitulation. So there is a big divide regarding software diffusion projects initiated from the bottom up, for these the failure rate is much lower, expecting the cases where the software innovation is unsuited to the needs (INN_NEED, AFT_FAIL), the most common failure cause for bottom up projects would be that the project is not compliant with the overall IT guidelines, network guidelines, the unification of the IT systems, etc (ENV_TECH, AFT_FAIL).

Regarding the top-down projects, the projects pushed by Management, the failures are much more common, ranking them down, the main cause of failure would be the lack of planning for risk factors (DIF_RISK), then it would be the cases where the Management does not react quickly enough to the warning signs, to initiate corrective measures (AFT_WARN). The second most common cause for failure would be if the software innovation project does not improve significantly the user experience and productivity (INN_ADVA), regardless if this criticism is real or not. You can have a project very costly, but if it is too complex to be used by users, it will fail (INN_COPL, AFT_FAIL), on the other hand, if the software is fairly easy to use but is not marketed enough (DIF_ADVE, AFT_FAIL), or does not plan for the end user training, the users will

perceive the software as being difficult to use (INN_COPL, AFT_FAIL), and then it will fail. It has to be marketed to the users for ease of use (DIF_ADVE), to gain users trusts, etc. If they do not feel they own the project, it will fail (INN_EMPO, DIF_ACST).

The next failure factor would be regarding factors of corporate interest. If you have a department which has a big interest for the software innovation, and the others would stand to lose for the project, they will tend trying to sabotage it (ENV_POLI, ENV_ORGB, AFT_FAIL), and usually they will succeed. They can also delay the deployment (DIF_RESO). The last factor would be risk reward, if the project is highly risky, it can fail. Management has to plan for time to treat incidents related to the risk, if the project is planned optimistically, with negative margins, at the first incident the top management would say: "Sorry, no more money, so we pull the plug!" (DIF_RESO, AFT_FAIL). These are the biggest reasons in my experience. Usually in all these cases you have signs that could warn of the possibility of failure (AFT_WARN), but in many of these cases, if the project management methodology is not fully applied (AFT_THEO), you have most of the risk monitoring functions called meetings which are not done.

So if there was signs in any way what could warn about failure, the project management did not do enough steering committees to pass the warnings (AFT_WARN).

Yannick: Do you think something could have been done to prevent these failures?

Steven: In many cases, yes. It often depends on how professional the project management is conducted, in many cases the project management does not fit all the standards for industry, so PMP, prince and the like (AFT_THEO). To be fair, if you apply fully the PMP methodology, it is a lot of paper work and you need a full time project manager for that. Even if you have a professional management team, even if they succeed to detect all the risks factors (DIF_RISK), there is honestly half of cases where the failure is independent from the management (DIF_UNCE).

Yannick: thank you!

B. Interview B

So for the first question: “what are the main themes and questions that you identify as being core to the success or failure of Software Innovation Diffusion projects?”. So I identified six questions, so the first would be how the platforms where you want to install the software where there is the innovation just adequate, or just the software itself (ENV_TECH). What I mean, to clarify, for instance you want to install a software that actually requires a 64 bits Operating System and most of the workstations in your company are 32 bits, just a few of them 64, in all the department you target, then you have an issue. That's the main idea, so basically, the fact that you have to forecast the fact that your software is adapted to your IT environment, even the hardware. Then, the other point I thought about was the procedure of the diffusion of your software (AFT_THEO).

So basically from my point of view, if you do not have any procedure to deploy, you will have a lot problems and a high probability of failure (AFT_FAIL). Like for instant, related to the first point I stated, could be part of the procedure. First thing to identify is if the IT environment is OK for the deployment of what I want to do. So that is the thing, so you have to have a procedure on various point from my point of view to make sure that the deployment will be done in a safe way, even the scheduling of the meetings with the heads of the department where you want to implement the thing so you really have to have steps (AFT_THEO), maybe even before implementing the software, so this procedure could be adapted for many different

deployments of the new software, but you have to have a framework. That was the second point.

The third one is one that maybe could sound different than the technical aspect but there is the cost aspect (DIF_RESO). How much will it cost to deploy the software. So basically there will be a trade off between what the innovation will bring to you, to the company in term of added value and the cost it will have to be deployed, and what I mean by cost is, the human resources you have, what is the cost of the human resources, but as well the training (DIF_ADVE), because when you have a software innovation, it often comes with training of the staff, and you do not train them, of even if you do not advertise the innovation, from my point of view, you will have high chances of failure as well (AFT_FAIL). And this should be included in the cost, so it has to be taken into account (DIF_RESO).

The fifth point, the way I understood it was, the way the software innovation diffusion could start when the software already exists. But in the case we take this software innovation diffusion project including the pre-software innovation, there should be a framework to estimate the real innovation and the advertising of the innovation (INN_NEED). What I mean is that sometime you could say this software is an innovation and will bring this to the company and I am pretty sure it will work and there will be a high success out of it. Yes, but do you have a framework, do you have a scale, do you have anything to estimate the actual impact of the actual innovation (INN_NEED)? Because one could say: "oh, that is my innovation,

everybody will click on that and you will gain ten minutes each day of work-time, etc. and you will gain this and you will gain blah blah blah". And then at the end of the day you will say: is the real innovation a gain of time, well you have to have a framework to estimate the innovation (INN_NEED, INN_ADVA). And this could be a failure or a success of this diffusion project (AFT_FAIL, AFT_SUCC).

The sixth and last point I highlighted was the accessibility, because obviously if your innovation is aiming to a department, even just the department or at the company level, if your innovation is so complex that at the end of the day even the training you will actually have a heat ratio so the success rate of reaching the people that could be actually using the innovation is just twenty percent, you will fail (INN_COPL). So basically you have to figure it out and you could come in a procedure as well to estimate if the average people is able to understand the innovation, your average target people are able to understand your innovation, and use it. So that is the first question.

The second question was: "What is the nature of the problematic situations induced by uncertainty you have encountered in these projects?". OK, it's bit redundant, but again, the system, the incompatibility of the system with the software (ENV_TECH). Compared to my first point, at this stage it is not really part of the process of estimating if the hardware is OK, is the uncertainty at a certain level of the compatibility within a company where there are many different softwares and interactions that you cannot forecast

(ENV_TECH, DIF_UNCE). And that should be taken into account because, by experience, it happened to me many times, there is an innovation, there is a new software but just to name it, Excel and Word make it crash, or something like that. And there are interactions that you cannot even forecast before. That is what I call the incompatibility of the systems, or softwares if you want. The second point is more a subjective one, is the uncertainty perceived by the end user? And again it is a bit related to the things I talked about in the previous question somewhat. The thing is, when you install a software, you will train them, you will advertise the innovation and the end users will start thinking about what it will bring to them, in their daily routine (INN_EMPO). And so the uncertainty will not be just the uncertainty of the project, but also the uncertainty that your end user will feel (DIF_UNCE), and this has a great impact because obviously you will have, in general I would say, at least at the beginning, a lot of resistance (DIF_RESI).

A lot of resistance by the end user to change. Just because of his fear of the uncertainty. Even if he sees and if he understands the innovation and the added value of what you bring to him, the first idea will be: "stop, I have to think about it, this is going too fast. And what will it mess in my daily routine?" (INN_EMPO). So this will have an impact in term of timing because you will have a little time to let the users to integrate that. And this you do not really control. But maybe you can state a procedure, or you have to think about a way for the project manager to take into account a way to give some space, or even have the feedback, in some sort of loop (DIF_ACST), going back and forth with the end user, to make sure that the

integration is ongoing. Because if you have a training which is "everyone understood? OK", it's not just the technical, it's as well to understand so even the second step, if you need a training that will take one week (DIF_ADVE), each day, into the procedure from my point of view, you have not just to understand if the user gets the way it works, but also to measure his resistance to it (DIF_RESI). This could be a failure if you do not take this into account, and you do not try to correct that resistance (DIF_RESI, AFT_FAIL).

You have to have this feeling and this subjectivity (INN_ADVA), and to correct this subjectivity. That was the second point. The third one, you do not know if your company will cut your budget (DIF_RESO). That is an uncertainty you have to live with (DIF_UNCE), and from my point of view with these budgets you always have to plan a cushion, a cushion of budget, of money, or even in resources, because if you are at the limit, and the company comes back to you and says: "oh, now there is a merger", which was my first point, a restructuring of your company (DIF_UNCE). But it's more complex than the first point, so I stay on the third but the restructuring part is merger, restructuring or just about cutting your budget of twenty percent. If you are at one hundred ten, you will have a lot of problems, and this could lead to a failure (DIF_RESO, AFT_FAIL). But this is uncertainty, because this you cannot plan in advance (DIF_UNCE). So it arises just like that. First point, the restructuring itself, is that you plan a software, that aims to help a department. But the restructuring makes this department disappear (laugh) (DIF_UNCE). So this could be a big issue, obviously, that is the extreme, but they could potentially change the bosses, the way people

work, change the procedures that will change your plan (DIF_UNCE). So these could be some uncertainty situations, and problematic situations, that is the second point.

The third question: "How did you deal with them?", so it happened indeed to me, so the first one for the incompatibility of the systems and this kind of thing (ENV_TECH, DIF_UNCE), is obviously to try as much as possible to map, to map in each department things the head, you can involve the heads of the departments, to ask them, to ask to their staff, to map all the softwares and things they have on their workstations. Because the thing is, in general, you cannot really install any software on you workstation, but it happens that you have a range of flexibility, so you can install a software, and the IT department is not aware about that, but it has an impact, so you have to map it. So that is the first thing because the problem was the uncertainty of the compatibility of the systems, this could help a little bit, at least (DIF_UNCE, AFT_SUCC). For us, it helped. And the second point was related to the uncertainty perceived by the end users. So obviously, the way we dealt with that was to explain better the advantages, but not just the advantages of the software itself, but the advantages that will be his (the end user) advantages and most importantly that these advantages will not erode his (own) added value (INN_ADVA, INN_EMPO). Because obviously what is he worries about is: "will I loose my added value? will this software help me to sell more?", because most of the time, the end users in a company, most of the time they do not care about the company (laugh), apart from the executive directors, directors, most of them do not care about the company.

So thing is, they will really think in term of: "will it help me to sell me better? to sell myself better" (INN_EMPO). So in a sense, these things should be highlighted. In the process of innovation, you should highlight the fact that it will not just be heritage from the innovation, he will also be able to sell himself better thanks to it. It will help, and ease a little bit, his worries about the uncertainty of his position in the future, because of or thanks to that innovation. The third one (laugh) was the budget problem, so if you have a cut. Unfortunately there is no magical trick at this stage, we will talk about negotiations with the bosses (laugh), we have to negotiate, so it is really a cornerstone step that you have to discuss with your bosses, you have to negotiate which means try to take back a little bit of money, from the cut, obviously, and if they do not, try to negotiate the things you can give up, in the project, and things you have to keep (DIF_RESO, ENV_POLI). But giving up on a part of the project could sound like: "that is my business, I am the project manager", but in fact, not really. Because you have the process, if you want to win, you have to make it clear to your management, because if they cut twenty percent, you first have to, in a negotiation process, to establish the risks they take with respect to what they expected the project to give, of the cut (DIF_RESO, DIF_RISK). And in this case you say, I can cut this and this, but you have to be aware I will cut that.

Are you OK? do we agree? it's an important step. If they say yes, if they have to validate it by e-mail, etc. And then you get into politics (laugh) but at this stage it is extremely important that they get aware about the impact they will have on the project jus saving this money

(DIF_RESO, ENV_POLI), and what they will lose, because it will give you a leverage to keep this on-board, and to keep what is really important for the project, and not just, let's say, the satellites projects of the project. For the first point, which is the restructuring itself, from my experience, the things I could have done in the past, again there is no magical trick, it is a regular business the fact that we have to live up with restructuring (DIF_UNCE). But it is obviously a huge point of uncertainty. But the only thing you could is to accumulate as much and as often as possible, information about where they are going (laugh) (DIF_UNCE, AFT_SUCC). And that is the only thing to do, and try to modify the project and the schedule of the project in line with this, so it will give extra work and extra meeting, and extra brainstorming.

Obviously, one of the difficult points with this is to be able to estimate what is actually a real piece of information. What I mean is that, it is not because they are updating a piece of information about the restructuring every day or every week that these pieces are actually important for the project. So it is not because there is a new thing that: "oh, ok, let's organise a meeting" (straight away) because you would create chaos (DIF_UNCE), and this will lead to failure instead of success (AFT_FAIL). Because people will lose path, and there will be a lower pace, they will lose their directions, and the project will fail, not even because of restructuring but because you modified too much the things, or made thinking to people they had to modify their plans, beside the problem of estimating the probability of the reliability of the piece of information. So it is complicated, but it has to be done, and you really have to be very close to your management

(ENV_SOCI), as much as possible to bring the info, estimate the probabilities, and do the trade off, and decide if it's worth to be discussed. So that is the difficult part but money has to be discussed (DIF_RESO), in order to get things back to the right direction.

OK, next question: "to what extent, did they appeal to the use of intuition?". Ah. From my point of view there is a loop between intuition and experience (AFT_INTU). Experience could lead to intuition, but intuition could lead to experience as well (AFT_INTU). It is like the chicken and egg, you do not necessarily know which comes before the other, but it is definitely something important, and probably even before any plan. So it is important, it is cornerstone, because even a process that looks perfect, for instance if you are at your third of forth software innovation project, you probably already have a framework, you probably already have something that looks like rational, but this comes from experiences, from previous experiences. That comes probably from intuition (AFT_INTU), and vice versa, so there is always a feedback loop, but something that looks rational, it actually comes from this, so there is a huge part that comes from that. And if you want to improve the process, it would be probably a route form the intuition and the experience. So it is extremely important. Maybe after a while, it can become less and less important, at least when you start to have a lot of experience, and you had the intuition to build your framework and for the way you managed that. But at the very beginning, for sure, this is more important than just the rational (AFT_INTU).

Next question: "according to your experience, what are the most common causes of failure for this type of project?". I have got five points. The first point is simple, innovation idea that wasn't (laugh). So it is related to estimate a framework or a way to estimate if an innovation is a real innovation (INN_ADVA, AFT_FAIL). And I saw that many times, people trying to sell, they sell themselves to they are trying to sell anything they do is an innovation but it is not. And then at the end of the day they have the capability of the political situation to push the project even if it is not an innovation (ENV_POLI). Selling it is an innovation at the end of the day, there will be a lot of expenses, and everybody will finally realise it does not bring any value and that it was not a real innovation and brought nothing to the company (INN_ADVA).

Second point, it happened to me, over-estimating the capacity of the company to undertake such a task (AFT_FAIL). So basically, you had a crystallisation of the company in the way it was working, and you thought that innovation was just OK, flowing through these pipelines. But the thing is that you were wrong, and since you were wrong, actually you start seeing that you have far more resistance everywhere, that you did not expect (DIF_RESI). And I am not just talking about the end user at this stage, but also the IT department that do not have the tools you thought they had to help you for the migration (DIF_UNCE, ENV_TECH). Again because you didn't do you mapping correctly or whatever, you realise that everybody was not using the Operating System you expected them to and eighty percent of your company is actually using an other Operating System (DIF_UNCE, ENV_TECH). So you will have an issue, and your

company is not ready, and not able to help you. It happened to me with a database, on a big project, with a brand new database and normally it should have taken about six months, and in this one I realised it was more resistant, not just for technical reasons but also for political reasons that I was not aware of (DIF_UNCE, ENV_POLI), because the needs were difficult, it was too difficult at this stage, it was too early at this stage to realise that the political issues would lead me to a failure, because it was a failure, and the project took, instead of six months, it took five years (DIF_RESO, ENV_POLI, AFT_FAIL). And it was not really complete, and there was a lot of fight, and political fight around it.

So basically the company was not ready for that, even if it was an innovation, and so, you realise that because the schedule started to get longer, because then everybody started to get tired of it, because they were just fighting for nothing and they lost direction (DIF_RESO). Yes, it is related to that point, about too important resistance in certain companies across the hierarchy (DIF_RESI), so basically not just the end user, because the first thing is OK, the end user is a pain in the ass (laugh). He is the one that does not want your innovation (DIF_RESI), but sometimes you realise that even if the head of the department is not the end user, he has his own view about that it will bring to himself (DIF_RESI, INN_ADVA), or not, or take off from him. So he will resist, and maybe his boss, etc. So there is really a hierarchy of resistance to think of, and not just the technical part, and it could really lead to failure (AFT_FAIL), by experience. Again, restructuring could cause a failure, or the fact that the project would disappear, so you cannot do much about what I said before

regarding the restructuring, mergers or acquisitions (DIF_UNCE).

OK there is a fifth point which is a bit related to the innovation that wasn't, but that is not the point, it is not just the innovation that wasn't. The problem here is the persons that are involved in that project, realise that the innovation isn't a real innovation after a while (INN_ADVA). But the costs involved in the company was so high that they are ashamed to admit that the innovation isn't one (ENV_SOCI, ENV_POLI, AFT_FAIL), they are afraid to lose their premium, they are afraid to lose maybe their job, so they keep ongoing and it leads to a failure because you do not have any proper feedback loop, on the actual reality of the project. So it has to be thought that you have to have a procedure, or a way to make sure that you control your staff do not retain any corner stone information that is covered by the project, even if it is to say the project does not work. It is better to stop before, even if it is difficult (AFT_SUCC), so it means at this stage, to make people talk, to make clear from the beginning, that nothing will be done against them if, at a certain stage, they realise there was a mistake. They should even have a premium to discover a mistake (AFT_SUCC). There should be, from my point of view, an incentive to discover a mistake. Not an incentive so high that they will try to forcibly find a mistake to get a premium, no that is not the point (laugh). So it has to be rational, but there should have an incentive and not the other way around, otherwise they will just do not say anything.

OK, next question: "What are the signs that could have warned

management about the possibility of failure?". Could have the management seen that before? Yes, they could have, and they knew, and they heard about political issues (ENV_POLI, ENV_ORGB), because I talked about resistance of the IT department to that project (DIF_RESI), etc. But the problem is that it is not always easy to break the taboos and to deal with the politic interlinks (ENV_POLI, ENV_ORGB, AFT_FAIL). There is some kind of willing-full blindness. Second point, they clearly saw the time of the project and the schedules getting longer and longer, so they should have been warned that there was an issue, because if things are getting longer, could get more costly (DIF_RESO). So there are the three easiest points that could have been spotted that something was going wrong. And I am not talking about the quality of the innovation here, there is an issue, at least trigger a question like is the innovation still here, or the innovation is still here but we will not ever see it, but still these three points could have made the things clear. So it was visible from my point of view but nothing was done about it.

Last question is: "do you think something could have been done to prevent this failure?", actually I made just one point, yes but it would have required a lot of political skills and management skills (ENV_SOCI, ENV_POLI, AFT_SUCC) to do that. So at this stage from my point of view it is not really technical as you could understand. It is really political and management. Because the technical part, let's say it this way, you could always find a solution (ENV_TECH). We know that, for instance, in my experience, each time I had to deal with a project which was involving just a pure technical innovation, I have never been really worried, because in a

sense you feel that you will arrive to a solution anyway (ENV_TECH). But the human problem, the social one, political one, brings a lot more of uncertainty and pressure to a project (DIF_UNCE).

Make part of your company, not just on the technical scale, but also on the management scale, you have to be really involved in the management (ENV_SOCI, ENV_POLI, AFT_SUCC), and get to know the company, not only from a technical point of view, but also regarding the relationships, because that will make the difference between two project managers, one with a good project, the other one with a good project, and actually facing the same budget cutting, one with the best knowledge of the company, the knowledge of the way the hierarchy works in the company, this one has chances to bring the project to success, the other one will fail (ENV_POLI, AFT_SUCC).

Yannick: thank you.

C. Interview C

What are the main themes and questions that you identify as being core to the success or failure of software innovation diffusion projects?

In all projects, I believe communication is key. It's also the area that usually causes most problems as it is difficult to gauge how much communication is required, or how to best use it for the project. Too little communication does not generate enough interest around the subject (DIF_ADVE), and too much causes people to stop being curious about it. Targeting the right user community (if it does not impact everyone) is also quite a challenge. On projects where innovation is the main driver, employee engagement is as key as communication (DIF_ACST, DIF_ADVE). It will help spread the word on the innovation, foster positive communication around it and help increase the interest and user engagement (DIF_ADVE).

What is the nature of the problematic situations induced by uncertainty you have encountered in these projects?

Uncertainty makes it difficult to stick to the project plan and therefore adds a layer of complexity to the project. Where you can rely on previous experience for standard migration projects, projects driven by innovation by essence bear a lot of unknowns (DIF_UNCE). Users who would not know the technology, might be sceptical about it and could potentially bring your project plan to a

halt if the innovation is not welcomed as well as it was planned (DIF_RESI, INN_OBSE, INN_ADVA). In such projects, one must also plan hyper-care activities and specific activities to help users engage with the project and the innovation, otherwise user frustration may lead to a total disinterest in the product which can lead to a total lack of use of the product is perceived as making the users' life more complicated than initially advertised (INN_COPL, INN_ADVA).

How did you deal with them?

As mentioned in point 1, communication and employee engagement is key (DIF_ACST, DIF_CHAN). Where a project involves a major innovation for a large community of users, it is important to engage with top management first, so that the news and positive communication comes from the bottom down (INN_ADVA, ENV_ORGB).

To what extend do they appeal to the use of intuition?

Where innovation is involved, as the project manager you have to keep your eyes and ears open for comments from the user community and act fast in response to the various comments (DIF_ACCE, ENV_SOCI). A couple of users to whom the solution was not communicated properly or who do not engage with the innovation can drag other users along (ENV_SOCI, DIF_CHAN)

which can then cause a major issue in the overall project roll-out. You also do have to take some distance and put yourself in the shoes of the basic user and how the innovation could potentially change their work habits (INN_ADVA).

According to your experience, what are the most common causes of failures for this type of project?

Again poor communication (DIF_CHAN), or wrong type of communication as well as the lack of early on engagement by the user community (DIF_ACST) and management will most certainly make the project fail. The innovation must also bring value to the daily work (INN_ADVA) and a high return on investment (DIF_RESO) for it to be considered necessary within a work environment.

Were there signs that could have warned Management about the possibility of failure?

The lack of interest from the user community (INN_ADVA, AFT_FAIL). If the innovation is not being talked about, or talked about in a negative way, these are clear signs that the solution will not be adopted (DIF_CHAN, DIF_RESI, AFT_FAIL).

Do you think something could have been done to prevent these failures?

Good practices include bringing a team of champions from across the organisation together and demo'ing the product or creating workshop sessions around it (DIF_ADVE). Champions will help spread the news on the innovation and foster interest (DIF_CHAN).

D. Interview D

Brian: I was trying to figure that on the drive in, unfortunately the traffic kinda sucks so I was preoccupied about how to get to work but the fusion projects, the one I was thinking of and that was kind of in line with what you were expecting was IPS. So we created the IPS, the new packaging system and obviously when you create that for an Operating System that already use a packaging system has to basically use that new technology so ... is that kind of in line with what you were expecting?

Yannick: Yes, it is perfectly in line with it So I guess the first question was the main themes and questions that are core to the success.

Brian: I was just going to say for IPS we made design choices, one of them that was controversial and actually cause a whole bunch of consternation, was the lack of post-installation scripting (ENV_TECH, DIF_RESI). So I think, going back to the question that the core of the success of IPS was making good line choices and then we had been hearing to those, because there was an enormous pressure to enable post-install scripting or enable some jail free ways you could just install a script from package install and we consciously avoided and defied doing that because if we had done that, that really opens you up to do whatever a developer wants on a system, and for us, one of the primary tense for IPS was making sure it's a correct OS installation so ... in also being able to do a self-assembly. So when you put a software on the system, you are able to install all the software and get the guest up and running. And if you have self-assembly, you get out of correctness because that script could do anything it

wanted, and that was a fundamental flaw with the previous packaging system, and that is why it was super important that we actually had here to no scripting.

Yannick: Did end users and customers ended up thinking that, yes, that was the right choice, or did they keep complaining?

Brian: No and that was kind of an interesting thing and it went on for a few years. Part of the problem when you do innovative projects or releases, on my experience, of doing innovative projects, you know, FMA being the first experience and then SMF and IPS, there is a change in direction, that they say, paradigm shift and how people think, so when you shift, everybody's mind is not shift with you as much as much as you hope it would, and in fact there is a lot of resistance (DIF_RESI). Any-time you do anything innovative and new, people are generally going to be resistant initially to it (DIF_RESI), regardless of if it's right or wrong. If something is wrong, you will have consistent resistance (DIF_RESI, AFT_FAIL) and at some point you have to do something, find some alternative, but in the case of IPS, when you do things right, there is an initial resistance, and then people see the benefits (INN_OBSE, INN_ADVA), and the benefits are, you know, a correct install, your system is always updated, and it just works. And after you experience that, you know, developers experience that, customers experience that, then people think: "oh! this is actually very cool! and this is actually a win" (DIF_ACCE) and resistance think: "oh, now I understand". And then for some customers, and for Project A for

example, now that they understand and appreciate it they went and told their suppliers that, if they want to deliver to them they need IPS packages, because they saw the design principles and they saw the benefits and it was very useful for them (DIF_ACCE, DIF_CHAN). So it is a tough one because you have to make, you know, choices, that may cause a lot of drawback initially and you may think these are the right choices but time will tell, eventually, one way or an other, either everybody will come on-board (AFT_SUCC), or you will have consistent resistance and you will need to change. Anyway in the IPS case, we made the right choice and every-time I talk to new hires or anything like that, I will talk about how IPS or packaging in Solaris 10 would be the pin number one for customers. Now it is in Solaris 11 and update 2 specifically, it is now one of the number one selling points. I either package management is the number one point and was one of the major highlights for the update 2 launch. It's gone a full circle for the IPS case.

Yannick: Was the communication good around IPS at the beginning?

Brian: You know, any-time in my experience was for things like this there was never enough communication (DIF_ADVE, DIF_CHAN), part of that is probably where engineers are introverted and there is also the fact that everybody has all these questions at once, and I do not know if it's infeasible or just really really hard to get to everybody the information they need (DIF_ADVE, DIF_CHAN). Part of that is worsen by the fact that you try to answer to questions and you also try do development. With IPS, we had the basic components and we

were actually releasing an Operating System but we knew there was a whole bunch of years of work still to do, and so it is really hard to prioritise communication when the implementation is still, well, down the road (DIF_ADVE, ENV_SOCI). It is a tough one.

Yannick: and I think communication is not really your role, you have a more technical role

Brian: Yes, I have to say one of the things I believe is I look at our role (his team and himself) is to get the job done and to do whatever is necessary and usually that is going to the management, you know, if we need to get a product in and that requires going on to the CEO, we will go through the process to see the CEO and get his/her buy-in. so communication, yes our role is technical role, doing development. At the same time, we are here to making a change in the part of our job, to communicate that out to people. And it is tough, more role is technical, communication is general. Communication is generally not our strength and that you know when you are going to hire somebody you don't ask him "How good as a communicator are you?"

Yannick: Yes, I understand

Brian: So, it's one of the thing, I think that every project that we've done and most likely, every project coming forward, we need to do a better job to communicate. It can always be improved. I saw one coming here, regarding the use of intuition, and I am not sure it will

directly apply to that question but, one of the things that really helped IPS was the concept of boot environment snapshots and that was one of the things customers got on-board quick and I think it is one of the big keys to rule out innovation, there has to be something people can latch onto as being a: "oh, this is cool, and it is very intuitive" (INN_ADVA, INN_COPL, INN_TRIA). And having snapshots of your boot environment, people without having any sort of communication quickly understand (DIF_CHAN), like almost instantly how that is implemented, how that work, they will probably do not care, but how it is intuitive (AFT_SUCC) and that allows onto, "I do not know about this scripting feature I lost but this boot environment capability is super cool!" so that kind of blocks out some of the other changes, and at the end of the road they will appreciate this.

Yannick: so when a technology is intuitive it can be convincing pretty quickly to any user.

Brian: Well, a part of it, you know. You try to get every part of it very intuitive, but there will be aspects that will be less intuitive, and having sub-components of that being intuitive, I think, it really helps. About the question on the intuition, I think that one of the things that caught us by surprise, we were preparing IPS for customers and as a result, one of the things we missed out on was internal developers and how they are going to use it (DIF_ACST), and it took us by surprise and we were resistant to making changes because it did not impact the customers (ENV_ORGB), it was an internal only problem,

so I think one of the things we could have done better was first of all think about how developers are going to use this and not just customers. Because developers are actually customers as well, to a certain extent, they are not paying ones but they are really important, and I think we should have spent more time educating them (DIF_ADVE) and also creating more tooling around managing change between different builds (INN_COMP, INN_OBSE), and being able to be clear about what was done in each build and, you know, being able to communicate clear error messages (DIF_CHAN). Part of this is it is a very hard problem and error messaging, we continue to make improvements but the process of figuring out dependency analysis is very hard and we are trying to work through that.

And I think, you know, in how to deal with these problems, just first of all acknowledging they are there, and communicating, and this is what we are doing, and not to hide behind it but also educate people (DIF_ADVE) that this is a really hard problem, we are not making enough progress on this, not because we are not working on it, but because we are trying to figure out the right way to do it. You want to get things communicated, but at the same time you have to be clear that, hey, it is not an easy problem to solve and I think it is an area where we did not do a very good job, we could have done better. The way IPS works for developers was not intuitive (INN_COPL) and that turned out to be problematic because part of the problem, specifically to developers is we engaging them or they are getting involved early in the development process, so getting back to the communication concept, first of all, we are not trying to do implementation yet, there

is a whole bunch of questions, so we need there is implementation that needs to be done and we are trying to do that but as a result we are not communicating very well, it is a really tough choice.

I think we needed to step back a little bit and, we really need to get it to the customers but, in the short term we need to get the developers on-board and we should have made this more intuitive (INN_OBSE, INN_COPL), and I think that would have actually helped. The trade off would have been delaying the delivery to customers, that is always the push and pull of gaining out as possible time to market (DIF_RESO, ENV_INDU), and, minding the fact that you have a bit release coming, Solaris 11.

BJ (moving onto the next questions): The most common causes of failure, this is, project by project, but generally what I have seen around here in Operating System development is a couple of things, one initially is, first of all it is hard to find people that are good at creating new, really big innovative projects, and most people would want to start with a white page of paper and start and just go create software (INN_DESI), and you actually find that people do not actually enjoy that, they need a basic framework and they like to play in that framework, but starting from a white sheet of paper is actually overwhelming to some people, to a lot of people. So first of all building from the ground can be a challenge, and then there is an other problem, big projects like this, can sometimes you have to restrict what you are going to do in phase one, and you have to select what to push off to later phases, and I think this is really important to

get something out there instead of just build and build and build and build (DIF_RESO, ENV_INDU).

As an example, we got the same reproach when we worked on the documentation. The old school way to work on documentation was, you start with a design document, you go write it up, you do eventual research and you spend with the time you needed to come up with a coherent design and once you had that, you could start to go into the development. Well, in high technology this is just not feasible where time to market is key (ENV_INDU), it is more important to get something out there that the customers can play with, a year earlier than waiting and coming a year later maybe with a slightly better solution. So you know it is really important to get something done and out, at least from my perspective, this is what we are trying to do in my team. You know, put the framework in place, design the overall framework, high level, so everybody can know what is going to happen and once you have that framework then you can pull people in and start populating different parts of it, handling security, and authorisations, and logging. Implementing all the different sub-components of that framework, it works out very well. Trying to get everything great before pushing it out.

Next question: "Where there signs that could have warned management of possibilities of failures?", I think in this one, all the management was worried of the possibility of failure, and management was overly concerned and did not have confidence in us, because designing systems which are fundamental to Operating

Systems (DIF_RESI), and if it fails, your Operating System fails, so I think that everybody in the case of IPS was overly concerned about failures, so once we actually delivered, and things were actually working, people were, a bit surprised I guess (INN_OBSE, INN_TRIA, DIF_ACCE).

I know that in other projects like SMF, we have been off way too much and from the schedule standpoint needed a lot management alerts sooner so they can plan and get better ideas about how long things could take (DIF_RESO). What can be done to prevent failures, I think one of the fundamental things is to have a clear line of communication (DIF_CHAN), and having the courage to tell management: "hey, were are not going to have that feature in this release", we did that with SMF, "we should have told you earlier but at this point in time we know that we are not going to be able to get this done in time for the release".

I think any time you have failures, step one is clear, direct communication. When you get into the situation, you can feel that pressure from management or whatever, and it is really tough and it is really important to, what a lot of people end up doing is cutting back in quality, one way to save time (DIF_RESO) is to just make up some much smaller test suites or much smaller test execution plans, and that can save you weeks. The problem is that then you have a poor product because it does not work as expected and that can cause fatal failures, because customers are like they are not going to lose time to understand this (DIF_RESI, INN_ADVA, AFT_FAIL).

Yannick: thank you for your time.

E. Interview E

Yannick: Does it feel clear to you what I ask for?

Gerry: Reasonably, yes

Yannick: Please let me know if you need any more clarification as we go on the subject

Gerry: Sure. So I think it is probably useful to think about a couple of real projects because it is really easy to discuss the hypothetical but the real projects are some more insightful. So there are two projects I am thinking of here, one happened a few years ago, it is how we changed, how we built Solaris 10 updates, the other one is one we are working on now which is around improving the efficiency of the integration process, the put-back process, and I think around change management, the important thing is all about people (ENV_SOCI), and it is very much about human interactions and a behavioural interaction (ENV_SOCI). It is not about logic or logical argument (ENV_TECH), because as engineers, by default, we tend to discuss things as logical arguments, that is plan A is better than plan B, groovy ruby on rails better than python jungle, you know, ruby better than PHP, and we tend to, by default, as engineers, to discuss in logical arguments, and, you know, if you go on soft skill courses around collaboration or around team building or influencing, one of the things they teach you is that different types of people have very different ways to be influenced (ENV_SOCI). So if you are talking to a sales guy, or a marketing guy, it is much more about painting pictures, so the vision, here is what we can be in two years time, and imagine when you have got your commission, you have your jaguar parked in the car park, and that is how you get their attention.

Whereas if you try to talk to them and say well here is my super-cluster, it is better than exadata because we got blah blah, you will literally see their eyes glaze over and they literally be there trying to keep awake, so how you influence depends on the type of persons you are trying to influence, and it is very important as an engineer to understand the way we work by default, is not the way everyone else works.

And that is a very important thing. So with change management, one of the first things is to identify your stakeholders, so who are all of those stakeholders that you need to influence (DIF_ACST). In Sun, it was probably more key engineers, than management. If you have them on-board, the management will follow, and you may also need to get the management on-board, but in Sun. So first you need to identify your stakeholders, and rank them, first by importance, and you can rank them by influence. So you might have someone that is in an important position, but they do not tend to ask for opinions, they are probably going to be easy to influence. You have people who are less important but they are loud and they are listened to, and therefore they have a far bigger span of influence than you might think. And also you need to be thinking from their point of view, what are the barriers to adoption from their perspective, maybe it is not invented here, which is typical engineering reaction, I did not call it therefore it cannot be any good, it may be fear of change (DIF_RESI) or insuring their shoes (INN_ADVA), you know, we have always been making thing this way, it might be better if we do it that way but we do not know the down sides, there could be risks (DIF_RISK).

So you need to understand what are the potentials there, and maybe, honest and positive reasons of fear about whatever else, or there may be something in for them for you not to succeed (ENV_POLI, AFT_FAIL), and you need to understand what are the stakeholders, what is their influence (ENV_SOCI), and what is their likely position. And there is a good book on this, which I cannot remember. How is it called? (thinking for a while). I read two books back to back, one is called the Drunkard's Walk, by Leonard Mlodinow, and that is all about random theory, and it is basically saying that success is random. So you study the rest of it and really at the end they picture your intelligence as being basically locked in time. And the other one basically takes exactly the opposite view. What was his name? Bruce Bueno. This guy actually worked for foreign affairs policy for quite a while and you can see it, it is all about prediction.

The name of the book is game theory, it is how to see and check the future with game theory, it is basically a game theory. So this is basically who is behind game theory, and you know this whole thing about identifying the stakeholders, their influencing position, what are their moves, what influences them, and then, you basically work on it. Like a chess game, and I think his view of the world is extremely negative, and is based I think on Ian Round type, you know that greed is good and that everyone works from a greed principle, and yes, there is fairly some truth in that but it is a fairly negative view of the world. They are two very interesting books to read back to back, especially if you read Mlodinow first. So basically once you have identified your stakeholders, it is really about influencing them and as I said it is a purely human task (ENV_SOCI), and if you are trying

to influence someone in sales or marketing, the way you are going to try to convince they is going to be very different than if it is an engineer.

If you are trying to influence a VP, you need to try from a business perspective (ENV_SOCI, AFT_SUCC) so what is the VP's goal for this year, what did Larry (Ellison, CEO of Oracle) tell the VP to do, what is his strategic view, and if you can fit what you are doing into that line, then you are fine. But if what you are trying to do goes against his goals, good luck! you are going to have fun, so you need to understand what is happening here, and you need to look at the people thing. Then you got to have a communications plan, and this communication plan needs to be as detailed as any code and implementation plan you are going to do and it needs to be: "communicate, communicate, communicate!" (DIF_CHAN). It really is get the word out, hear the concerns, address those concerns (DIF_RESI, DIF_ACST), not just bat them down, but trying to figure out, well, ok, mitigate that by this, trying to make people important (DIF_ACST).

If you can get people feel that they have ownership with the solution, so let's say you want to make a process more efficient (DIF_ACST, AFT_SUCC). Well instead of going on your own and to a blackboard to figure out on your own how to make the process more efficient, and then trying to impose it on people, if you can get their input and be able to show how their input has shaped the output or if they can actually come up with the output (DIF_ACST, AFT_SUCC), that

takes quite a lot of people management to make sure people do not actually go off in some completely different direction (ENV_SOCI). Let's say the more happen in here, you know, the better. So for example, in department A, a number of years ago, we set a very rough goal that every engineer must do thirty six integrations a year. Why thirty six? why not? is thirty seven a better answer? Thirty five? it does not matter. The point was to focus people's minds that our job is to fix bugs, and that is your job, fix bugs.

And what we had at the time is that we had some people exceeding seventy, ninety bugs a year, and some people fixing two, three, four, five a year. And obviously the goal was not to bring the seventy one back down to thirty six, but to show the others there is a minimum bar here and there is a minimum threshold that you need to reach. Now we did not tell them how to reach a goal of thirty six but the goal was there, that is the minimum that was considered. And believe me, there was a lot of yelping and moaning, whining and still is! but the idea there was showing there is a minimum acceptable level. Now if you are going off doing something else, that is considered of value to the business (INN_NEED, INN_ADVA), which is not integration, you can get equivalent credits, but do not come with a team goal of equivalent credits because people game the system, that is a five credits, that is half a credit. So we do not do that, the only thing you get a real credit for is an integration and everything else is between you and your manager to evaluate it as an equivalent effort.

Now we got back away from that, because now we have the integration level back to a reasonable level. They were hardly motivating factors. What happened is the early Solaris 10 updates,

the process with the system we used to use for building updates was that the size of features we put in the S10 updates was way way bigger than what we put in Solaris 8 or 9, and it was destabilising the sustaining gate, we were getting bad kernel patches, we were blowing production customers up because the features were poor quality and so on. And then we had many new technologies, and the old process was simply not fit for purpose (INN_NEED), it could just not handle that amount of change, it was not designed for that, it was designed for new drivers and bug fixes, it was all it was designed for. So it was very clear that the old process was not fit for purpose, and for me, in the middle of it, trying to firefight all of the issues, it was very hard to step back what was the solution here (INN_DESI). I know exactly what the problem was, but it was very hard to see a way out, and I actually wrote a document, about 35 pages long, basically just listing all of the issues. And there you go, sixty issues, and maybe twenty of them were major issues, you know, why the process is completely broken (INN_NEED). And I actually gave that document out to a number of people, saying here is where I see the problems, can you help me find the solutions (DIF_ACST, DIF_CHAN).

And then one of the reasons that was very important is if I had come up with a concept on my own, it would not have been accepted (DIF_RESI, AFT_FAIL), and it could have been the greatest idea, it would simply not have been accepted. To get that sold, you know in the old “Sun” days you could talk to everyone that had plans and discuss, well I do not agree with that, and very hard to get any problematic widespread change through in that Sun culture. At that time the first question at Q&A was: "What is the exception process?",

how do I get out of it, which is very much different from the Oracle culture which is "I told you what to do, now do it". With that Sun bottom up culture, it was very hard to do change management, the proposal had to come from the right people. If it was coming from the right people, it did not matter what solution my guys came with, it was not going to succeed, it was as simple as that. Because even if I was able to convince the VPs, the senior engineers would have made sure it did not work (ENV_ORGB, ENV_POLI). As simple as that. By getting some of the very respected engineers on-board, it was up to my team to implement the solution, by getting the right political dynamics (ENV_POLI), even if it was very much a joint solution.

Having these engineers on-board, you get rid of so many arguments (ENV_POLI), so finding the right stakeholders, the right influencers (DIF_CHAN) and then doing it. Similarly on the management side, at that stage one of the managers was quite a character, she was extremely decisive, you only never really knew which way she was going to fall. You were in the meeting and you knew that it was going to be a 100% back decision or a 100% white decision, and it was going to be very passionately one of the other. It was certainly not going to be something in the middle, for sure, so I was called to a meeting with her, with all of her direct reports, and I briefed her previously I think, and when I asked her about her opinion on the project, she said: "I think it is a good idea, I am going around the room and I am going to ask you one by one if you are on-board or not".

Yannick: Wow!

Gerry: Yannick, are you on-board?

Yannick: laugh

Gerry: and she got explicitly a yes from each of them. Right, now you are not anonymous so, make it so! No discussion, no touchy feely, no painting a vision or anything else, that was direct leadership. So, completely other side of change management, but equally effective, and that is where you influence the key stakeholders, in that case, her, it takes care of everything else in the organisation. So you need to know what sort of person you are working with, so that is where you need to understand the people, the structure, you know. So when you are going to a meeting, like a political meeting, you want to know what the outcome is going to be before you go into that meeting (ENV_POLI). So if you go to a meeting with ten stakeholders, you do not want to go in with with your fingers crossed and hoping that you arguments will get the change management. What you want in is to have at least the key stakeholders, and have a meeting with them one on one, understand their concerns, get out, asking them if they have any concerns about the project (DIF_RESI), do you see any difficulties with this, how do you feel this is going to work out, what more can I do to help you or help your organisation, get all that done, so they are fully on-board before they go to that meeting. Because if you have ten people in the room, one of them will probably go off in a tangent, if you got four or five people including yourself prepared, you can bring them back. If you have not done that preparation, then it is like trying to herd sheep. You have one sheep going over that way, one sheep going over that way and you are trying to get them back together and good luck! You are going to have fun!

So it is all about people, it is not about how good the technology is (ENV_TECH), it is all about people and influencing (ENV_SOCI), and really think like a politician unfortunately. What about the use of intuition? everything is about people, and behaviours, and influencing, it all about people. What are the most common causes of failure? Failing to engage, failing to listen to the feedbacks (DIF_RESI, AFT_FAIL), not acting on the feedback, you know if people perceive that you are ignoring that feedback, well, A you are not going to have any more feedback and B, you gain up resistance (DIF_ACST). So you need to engage and you need to show, you know we heard this, and this is what we have done about it. Sometimes some people say something, and other people say that, so it is probably going to be in the middle so here is what we are going to do about it. If you do not communicate, people will fear the worst (DIF_CHAN, DIF_RESI, AFT_FAIL).

Yannick: thank you