

**POLITECNICO DI MILANO**

Scuola di Ingegneria dell'Informazione

**POLO TERRITORIALE DI COMO**

Master of Science in  
Computer Engineering



**Automating Processes in Clarity PPM by Gel Scripting**

Supervisor: Prof. Emanuele Della Valle

M.Sc Thesis of:

Mona Nasery

10339333

Academic Year: 2012- 2013



# **Automating Processes in Clarity PPM by Gel scripting**

A Thesis

Submitted to the Department of  
Computer Engineering  
at Politecnico di Milano  
in Partial Fulfillment of  
the Requirements  
for the Degree of  
Master of Science in Computer Engineering

Supervisor: Emanuele Della Valle

Mona Nasery  
OCTOBER 2013

---

---

## ABSTRACT IN ENGLISH

---

Business Process Automation (BPA) has attracted public's attention over the past decades due to its significant benefits for organizations in today's competitive market. Nowadays, improving business processes is one of the main concerns of companies since it helps them to stay alive and reach their objectives. Although business process improvement could be addressed through several ways, BPA is one of the best solutions which span boundaries of an organization and provide significant business benefits for enterprises by containing costs, improving performance and efficiency.

The main purpose of this thesis is to provide a comprehensive view on how process automation is implemented in a real industrial setting in Mediobanca along with evaluating its impacts on performance of the work activities and on the performance of the business process management system. Some best practices to avoid negative impacts and to maintain processes are introduced by some professionals who have enough experience in this field.

In order to achieve the purpose of the thesis, an appropriate methodological approach is needed which is described in Chapter 4. This study uses existing literature on the topic combined with number of interviews with experts in this field to respond thesis questions. The thesis is useful not only for use in the practical environment where the intention is Process Automation but also in academia for those who are interested to investigate in this topic.

This dissertation was based on a concrete work in which 20 processes were automated using GEL that is the Scripting language in Clarity. The results show the benefits of automating business processes and their impacts on performance. Processes which were automated during this study are presented and two of them as a sample of the work are discussed in detail in Chapter 5. For all of the studied processes, GEL script was used which provides more flexible solutions for process automation. Finally, the Appendix provides a complete hand-manual for system analysts or business analysts in order to use as a multiple-purpose reference.

## ABSTRACT IN ITALIAN

---

Nel corso degli ultimi decenni il Business Process Automation (BPA) ha attirato l'attenzione del pubblico grazie ai significativi benefici garantiti alle aziende nel competitivo mercato attuale. Al giorno d'oggi, il miglioramento dei processi di business è una delle principali attività delle aziende in quanto consente loro di rimanere in vita e raggiungere meglio i propri obiettivi. Anche se non è l'unica soluzione per migliorare il business aziendale, BPA è una delle migliori soluzioni possibili in quanto si estende oltre i confini di impresa fornendo significativi vantaggi in termini di contenimento dei costi, miglioramento delle prestazioni e dell'efficienza.

Lo scopo principale di questa tesi è quello di fornire una visione completa di come l'automazione dei processi è stata implementata in un contesto industriale reale, in Mediobanca, insieme alla valutazione del suo impatto sulle prestazioni delle attività lavorative e sulle prestazioni del sistema di gestione dei processi di business. Alcune linee guida sono inoltre introdotte al fine di evitare impatti negativi e di mantenimento dei processi grazie all'aiuto di alcuni professionisti che hanno esperienza in questo campo.

Al fine di raggiungere questo scopo, è stato necessario un approccio metodologico appropriato descritto nel Capitolo 4. Questo studio utilizza la letteratura esistente sul tema combinato alle interviste ad esperti del settore. Questo progetto di tesi è utile non solo ad un uso pratico per l'automazione di processo, ma anche nel mondo accademico per coloro che sono interessati all'approfondimento di questo argomento.

Questa tesi si basa su un lavoro concreto svolto nel mio periodo di stage durante il quale 20 processi sono stati automatizzati utilizzando GEL, un linguaggio di Clarity scripting che fornisce le soluzioni più flessibili per l'automazione di processo. I risultati mostrano i benefici di automatizzare i processi di business e il loro impatto sulle prestazioni. I processi che sono stati automatizzati nel corso di questo studio sono presentati in questo lavoro ed, in particolare, due di loro discussi in dettaglio nel Capitolo 5. Infine, l'appendice fornisce un manuale utile agli analisti di sistema o gli analisti di business, al fine di utilizzare come riferimento multiplo-scopo completo.

## ACKNOWLEDGMENTS

---

This Master of Science thesis has been carried out at the Department of Electronics and Computer at Politecnico di Milano University. The work has been performed in IT Governance of Mediobanca Group besides spending some more months doing further research on this study under supervision of professors Emanuele Della Valle who provided limitless support during this project.

The team work was truly fun and challenging at the same time. I learned a lot and met engineers who shared their knowledge and experience which I am very grateful and would like to thank.

First I would very much like to thank my supervisor at Politecnico di Milano, Prof. Della Valle who always had the answer to all my questions and guided me to the right way. Their constant encouragement and support throughout the project made it possible for me to complete the work.

I would also like to thank Dott. Gaetano Di Luca, Functional Analyst and system administrator and also Mrs. Laura Wegher, Manager in IT Governance of Mediobanca Group. It was a pleasure having their advices and excellent experiences in the field.

Last but not least, I would like to appreciate lifetime support of my perfect family whom always been backbone for me during the hard-times and good-times. Thank you for keep giving me the positive energy to carry-on and thanks for urging me to choose this path for my life.

Finally, I would like to thank everyone in Politecnico Di Milano University circle, from my colleagues, secretaries to the professors, whom got involved in such a way to let this checkpoint of my life happens.

Thank you all,

Mona Nasery

---

## Table of Contents

---

ABSTRACT IN ENGLISH .....	II
ABSTRACT IN ITALIAN .....	III
ACKNOWLEDGMENTS .....	IV
Table of Contents .....	V
Table of Figures .....	VIII
List of tables.....	VIII
Chapter 1 .....	12
Introduction.....	12
1-1 Dissertation Contribution .....	13
1-2 Dissertation Organization .....	13
1-3 Quality of literature sources.....	14
1-4 Target audience.....	14
1-5 Report Outline .....	14
1-6 Abbreviations.....	11
Chapter 2.....	16
Frame of References .....	16
2-1 Project Portfolio Management (PPM) .....	16
2-2 Process Definition.....	16
2-2-1 Processes in Clarity .....	17
2-3 Business processes.....	17
2-4 Business Process Management (BPM).....	18
2-5 Business Process Automation (BPA) .....	20
2-6 Workflow Management .....	20
Chapter 3.....	22

Main Background.....	22
3-1 Project Portfolio Management Background .....	22
3-2 Process Management background .....	22
3-2-1 IT Tradition .....	23
3-2-2 Six Sigma .....	24
3-2-3 Business Process Reengineering (BPR) .....	24
3-2-4 Business Process Management (BPM).....	25
3-3 State of the Art.....	27
3-3-1 Main Competitors of Project Portfolio Management (tools).....	27
3-3-2 Business Process Modeling Standards .....	35
3-3-3 GEL Script (Generic Execution Language) .....	41
Chapter 4.....	43
Proposed Methodology .....	43
4-1 Research Methods: Deductive vs. Inductive.....	43
4-2 Qualitative vs. Quantitative analysis .....	44
4-3 Data Collection .....	45
4-4 Problem Description .....	46
4-5 Methodological Approach .....	46
Chapter 5.....	49
Implementation Experience & Empirical Observations .....	49
5-1 How to Automate Processes in Clarity .....	49
5-2 Automated Processes .....	51
5-3 Sample Process 1: “Notify Manager Missing Timesheet” .....	55
5-2-1 Process Flow Diagram.....	57
5-2-2 Process Gel Script .....	57
5-3 Sample Process 2: “MB Workflow Request” .....	61
5-3-1 Process Steps .....	62
5-3-2 Process Flow Diagram.....	65
5-3-3 Process Gel Script .....	66
5-4 Impacts of automating processes on performance of the work .....	66
5-4-1 KPI Analysis of automated processes in Clarity .....	66
5-4-2 Qualitative analysis: Interviews contents .....	70
Chapter 6.....	73



Conclusion and Future works .....	73
6-1 Targeted Problem .....	74
6-2 Approach Review .....	74
6-3 Analysis Result Conclusion.....	75
6-4 Future Works .....	76
Appendix.....	80

## Table of Figures

Figure 1: General model of business processes. Source (Ko R., 2009) .....	18
Figure 2: BPM lifecycle, Source: Van der Aalst 2003.....	19
Figure 3: Relationship between basic terminologies [4].....	21
Figure 4: Process Management Development : .....	25
Figure 5: Architecture aligning IT and Business (Berneskog, J, 2011) .....	26
Figure 6: Deductive Approach vs. Inductive .....	43
Figure 7: Thesis methodological approach .....	48
Figure 8: Hierarchical structure of IT department in Mediobanca .....	55
Figure 9: Start step page of Process “Notify Manager Missing timesheet” .....	56
Figure 10: Flow diagram of process “MB Notify Manager Missing Timesheet” .....	57
Figure 11: Action portlet of the step action “Send notification to the organization” .....	63
Figure 12: Action portlet of the step action “Add Team” .....	63
Figure 13 : Flow diagram of process “MB Workflow Request” .....	65

## List of tables

---

Table 1: Summary of thesis chapters .....	15
Table 2: List of Abbreviations .....	11
Table 3: Evaluated Vendors: Product Information and Selection Criteria.....	28
Table 4: Vendor Selection criteria .....	28
Table 5: Forrester Above-The-Line Project Portfolio Management, Q4’ 12 (Cont.) .....	31
Table 6: Forrester Wave™: Below-The-Line Project/Program Portfolio Management, Q4 ’12 (Cont.) ...	34
Table 7: Various status of the steps within the active process .....	40
Table 8: List of automated processes in Clarity with a brief description about them.....	54
Table 9: Summary of the sample process 1: “Notify Manager Missing Timesheet” .....	57
Table 10: Summary of the sample process 2: “MB workflow request” .....	62
Table 11: Preconditions of the start step of the process “MB Workflow Request” .....	63
Table 12: Post-conditions of the step “Request for Approval” .....	64
Table 13: Impacts of automating processes on performance of work activities .....	69

## Abbreviations

BPM	Business Process Management
BPMS	Business Process Management Systems
BPA	Business Process Automation
BPR	Business Process Reengineering
KPI	Key Performance Indicator
PPM	Project Portfolio Management
SaaS	Software as a service
SOA	Service-Oriented Architecture
XOG	XML Open Gateway
GEL	Generic Execution Language
RE	Reengineering
MB	Mediobanca
CP	Compass
ST	Seteci

**Table 1: List of Abbreviations**

# Chapter 1

## Introduction

In the past few decades the role of information technology (IT) in organizations has been increased considerably. Business Processes have gained increased attention since the first years of 1990s. Business Process concept has been always with information technology which is considered as one of the most important enabler of process change. Many argue that Information technology should be seen as an enabler of organizational change rather than as a tool to implement business processes [1].

The ability to implement business processes as efficient as possible has become one of the key factors of the success of an organization. As a result of increasing demand for using business processes on one hand and growing trend in IT development and their importance in organizations on the other hand, research on Business Processes Automation (BPA) is a momentum. Automated business processes can help deliver huge gains in organizational and cost productivity [2].

Today Business conditions are changing continuously, so in order to stay ahead of the competition, companies should change the business processes quickly. To cope with the unstable changes, Business Process Management (BPM) can help companies to follow changes by changing their processes continuously [3]. Many authors such as Wang and Wang (2005) believe that BPM is the solution of enterprises against the “unpredictability” of the business processes. In addition, organizations use BPM for business processes modeling and get the most benefits from these processes<sup>1</sup>.

This master thesis could be an example to show the importance of the role that IT plays in organizations by implementing process automation through gel Script in Clarity PPM. We will see how running these processes can then be used for better analysis of some performance indicators. This could be a proof to show how automating processes facilitate activities, increase users’ satisfaction and save time and efforts in organizations.

---

<sup>1</sup> Neubauer & Stummer, 2007

## 1-1 Dissertation Contribution

The purpose of this thesis is implementing Process Automation in Clarity Project Portfolio Management used by IT Governance in Mediobanca or in general by IT department of any organization and evaluating its impacts on system performance in order to clarify the benefits to be gained from implementing Process Automation. This identifies what makes a process suitable for automation with BPMS and provides a guidance in selecting the solution for automation. In order to introduce the methodology, first it starts by explaining the status-quo and the background work already presented about Business Process Automation and possible tools and techniques. Consequently, the tool-chain details will be introduced. In addition, the final methodology which has designed to test the effects of this automation on system performance will be clarified. Furthermore, the experimental results will be introduced, followed by conclusion and future works and the complete hand-manual appendix.

The methodology used in this thesis is a combination of descriptive and explorative study. The data collection is based on existing literature on BPM and BPA besides talking with BPM professionals and experts. In order to find out if the process is suitable for automation, it is necessary to understand its requirements and preconditions. In general, there are some steps that should be considered when automating a process which are: evaluating process automation potential, determining automation requirements and finding the appropriate BPMS solution. After the process evaluation, automation requirements are taken into consideration. The feasibility and automation requirements are then combined and for each unique combination a suitable solution is provided.

This dissertation focused on exploration of research field not yet well faced with as a methodology analysis, it describes the implementation of processes automation by introducing Gel Script within Clarity PPM and evaluating its impacts on system performance. The methodology proposed in Chapter 3, has main target to provide best and complete information regarding the IT Process Automation and its effects on the application performance. The experimental\_tool we used in this research is Clarity PPM under proprietary of Mediobanca Group, helps driving us towards near-optimal solutions.

## 1-2 Dissertation Organization

The structure of this dissertation is as follows; first, background and state of the art of the topic is going to be illustrated in Chapter 2. In Chapter 3, the proposed methodology is described and selected automation

technique is going to be bolded. In Chapter 4, by introducing the Gel Script, the methodology is going to be illustrated. Finally in Chapter 5, the experimental results will be shown and will have the conclusion and future works on Chapter 5 and 6. At the end of this dissertation, there will be an Appendix representing the overall results in classified mode.

### 1-3 Quality of literature sources

One of the important factors in evaluating a research is quality and reliability of resources used for it. This master thesis used academic resources on Internet and literature available in specific scientific libraries and databases. The first step was reading some related papers in parallel with search in the internet in order to understand the concept of Business Processes, BPM, BPA, PPM and all other required terms used in this study. The next step was selecting the most cited references that analyzed the role of Information Technology in implementing Business Process Management in organizations. All the journals and papers used as a reference had to be published. In case of finding resources from Internet sources, it was considered if those resources are appropriate in order to be included in the dissertation. The judgment was based on logical criteria such as if the material was related to the objectives of the research project, or if other articles that had been proved reliable had referenced the author.

### 1-4 Target audience

The primary target groups are employees and managers in the IT industry where the intention is Process Automation or evaluating the impact of automation on system performance.

The second target group is students and academia seeking information about automation of processes and when to use BPM in managing business processes. It could be used by students to increase the understanding of processes, BPA and to learn implementing process automation in Clarity PPM.

### 1-5 Report Outline

Here is the summary of each chapter provided in this thesis:

Chapter 1: Introduction	This chapter provides an introduction to the research area. Thesis goal, organization and target audiences are also presented in this chapter.
-------------------------	--

Chapter 2: Frame of references	The most necessary terms to understand the thesis are presented in this chapter. It provides definition of the key terms that are used in this thesis.
Chapter 2: Main Background	This chapter provides information related to problem background and state of the art of this research topic.
Chapter 3: Proposed Methodology	This chapter describes the methodological approach of this thesis.
Chapter 4: Implementation Experience	In this chapter, it can be seen how processes are implemented through the methodological approach described in the previous chapter.
Chapter 5: Experimental Results	In this chapter, Impacts of running these processes on application performance and possible solutions to optimize them are evaluated in this chapter.
Chapter 6: Conclusion and Future works	This research study is concluded and future works in this field is discussed.

**Table 2: Summary of thesis chapters**

# Chapter 2

## Frame of References

### 2-1 Project Portfolio Management (PPM)

This is an undeniable fact that nowadays Information technology plays an important role in organization and their outcomes. Beside the growing trend of using IT in organizations (business), a challenge remains how to better manage IT projects while maximizing their economical benefits. Part of that challenge can be tackled by “doing projects right” and part by “doing the right projects” [4]. Project Management concentrates more on the former and Project Portfolio Management hereafter referred to as PPM, is focused on the latter. While Project Management focuses on a single project and Program Management concerns the management of set of projects that are related by sharing a common objective or client or through interdependencies, PPM considers the entire portfolio of projects a company is engaged in, in order to make decisions in terms of which projects are to be given priority, and which projects are to be added to or removed from the portfolio (see also Lycett et al. [5]).

### 2-2 Process Definition

A process is a set of activities undertaken in response to an event in order to generate an output. The term process was first described in 1776 when Adam Smith published his thoughts about work division. Today there are many different definitions of business processes to be found in the literature. One definition is used by Davenport who says that a business process is “a set of logically related tasks performed to achieve a defined business outcome for a particular customer or market” [6]. Another similar definition of a business process that is used by Ljungberg is “A process is a repetitively used network of orderly linked activities using information and resources for transforming inputs to outputs, extending from the point of identification to that of the satisfaction of the customer needs.” [6]



### 2-2-1 Processes in Clarity

Processes automate repetitive steps that you would otherwise perform manually through the user interface. To reproduce accurately a user action, the process impersonates the process initiator to perform the process steps. A process includes a series of steps that result in an end. Each step performs a single or multiple actions that move the process toward its completion. All processes have a start and a finish step. Processes use preconditions and post conditions to connect the steps.

Processes comprise objects, steps, actions, groups of steps, and joins. You can create processes to act on any object type.

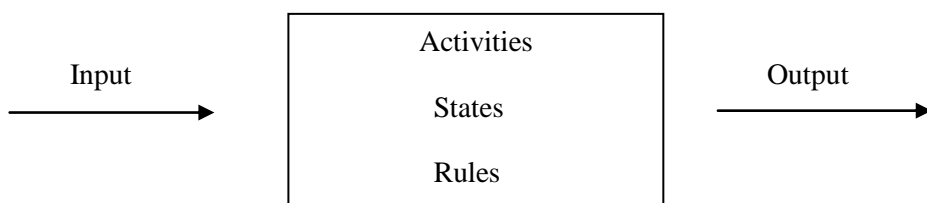
**Activity Definition:** A description of a piece of work that forms one logical step within a process. An activity may be a manual activity, which does not support computer automation, or a workflow (automated) activity. A workflow activity requires human and/or machine resources(s) to support process execution; where human resource is required an activity is allocated to a workflow participant. A process definition generally consists of many process activities which are logically related in terms of their contribution to the overall realization of the business process. An activity is typically the smallest unit of work which is scheduled by a workflow engine during process enactment (e.g. using transition and pre/post-conditions), although one activity may result in several work items being assigned (to a workflow participant) · Wholly manual activities may form part of a business process and be included within its associated process definition, but do not form part of the automated workflow resulting from the computer supported execution of the process. An activity may therefore be categorized as “manual”, or “automated”. Within this document, which is written principally in the context of workflow management, the term is normally used to refer to an automated activity. [7]

### 2-3 Business processes

Business Process is a set of one or more linked procedures or activities executed following a predefined order which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships. A business process is typically associated with operational objectives and business relationships (i.e. insurance claims process, engineering development process). A process can be wholly contained within a single organizational unit as well as can span several different organizations, such as in a customer-supplier relationship. A business process may consist of automated activities, capable of workflow management, and/or manual activities, which lie outside the scope of workflow management [7]. Usually the term “business process” is used to

reference to high-level description of an enterprise. Owen and Raj (2003), define business process as: *“The business process comprises all activities carried out in an enterprise, including e.g. staffing, financing, production, marketing, etc.”*

Business processes (see figure 1), can be seen as processes that have inputs and outputs which usually are the desired outputs for the customer and add value to the enterprise. One of the most interesting definitions of “business process” is by Lusk (2007) *“A business process is uniquely definable; has a defined set of activities that break down into increasingly more granular steps and an agreed to set of sequences; is enabled and/or constrained by culture, skills, policies and rules, laws, measurement/reward systems, workflow design, information systems and facilities; often crosses organizational boundaries; and follows a formal life cycle”* [8]. In other words, a business process is *“an organized group of related activities that together create customer value”* [9]. It is simply the way an organization does its work [10].



**Figure 1: General model of business processes. Source (Ko R., 2009)**

## 2-4 Business Process Management (BPM)

There are a lot of definitions for Business Process Management, but here are some of the most explanatory definitions:

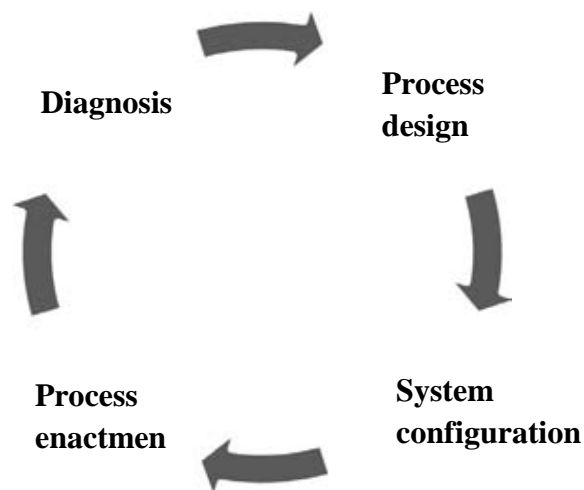
Ko, Lee and Lee (2009) define business process management as a way of *“...supporting business processes using methods, techniques and software to design, enact, control and analyze operational processes involving humans, organizations, applications, documents and other sources of information”*.

Antuanes and Mourão (2010) draw more attention on IT role. More specifically BPM *“integrates a collection of technologies capable to translate business process models into computer-supported activities, relinquishing routine management and control tasks from the organizational agents”*.

Verner (2004) completely agree with Antuanes and Mourão by connecting BPM with IT with the following definition *“BPM technology provides not only the tools and infrastructure to define, simulate, and analyze business process models, but also the tools to implement business processes in such a way*

that the execution of the resulting software artifacts can be managed from a business process perspective.” (Verner, 2004). Finally another author, Ramachandran (2004) points out a more analyzed definition of BPM by stating that “*Business Process Management (BPM) refers to the closed loop, iterative management of business processes over their entire lifecycle. It includes designing, optimizing, documenting, communicating, deploying, evaluating, updating, and retiring processes. Well-managed companies have always had robust, but time consuming, mechanisms for performing all of these functions.*”

To better understand the BPM concept and its features, one should start from BPM life cycle. In the literature, there are many views of BPM life cycle; however, the most popular one is by Van der Aalst (2003). Many authors considered it as the most succinctness and relevant one. [11] It consists of four phases: The process design stage, the system configuration stage, the process enactment stage and the diagnosis stage.



**Figure 2: BPM lifecycle, Source: Van der Aalst 2003**

**Process design phase:** In this phase, business processes are modeled into BPMS.

**Configuration phase:** during this stage, designs are implemented by configuring a process aware information system (e.g., a WFMS). This stage configures the BPMS and the underlying system infrastructure.

**Enactment phase:** Electronically modeled business processes are deployed in BPMS engines. Execution standards dominate this stage.

*Diagnosis phase*, BPM analysts could analyze the operational processes to identify problems and to find things that can be improved. The monitoring tools to do this are in diagnosis standards.

## 2-5 Business Process Automation (BPA)

Process Automation has attracted public's attention in recent years due to its significant effects on companies. Business process automation, or BPA, is the strategy a business uses to automate processes in order to contain costs<sup>2</sup>. It increases business productivity, reduce manual errors and accelerate in the IT service delivery, improve service quality and enforce compliance policies. With process automation, it is possible to manage various, complex environment through reducing business expenses and speeding up business processes. Many companies have automated their business processes and have shared their information in distributed information systems.

## 2-6 Workflow Management

Workflow Management, compared to BPM has a longer history and can be seen as a subset of BPM. Workflow is regarded as "the computerize facilitation or automation of a business processes, in whole or in part". The Workflow Management Coalition (WFMC) define workflow as: "The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules. A workflow management supports the routing of activities (i.e. the flow of work) in an organization in order to do the work at the right time by the right person with the right software tool. [12]

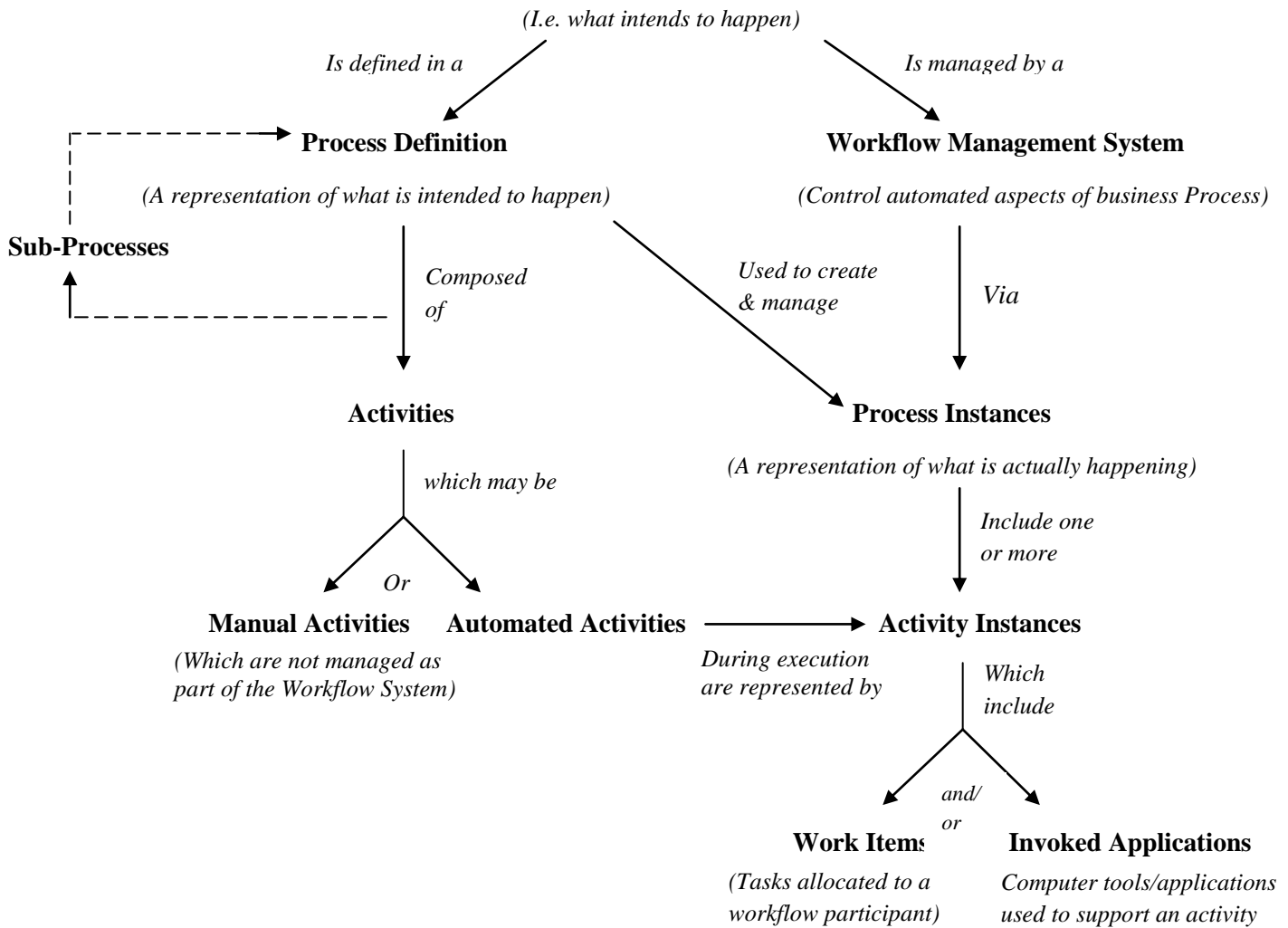
A Workflow Management System (WFMS) is defined as: "A system that define, creates and manages the execution of workflow through the use of software, running on one or more workflow engines, which is able to interpret the process definition, interact with workflow participants and, where required, invoke the use of IT tools and applications." [13]

In the figure below, relationship between basic terminologies is shown:

### **Business Process**

---

<sup>2</sup> [http://en.wikipedia.org/wiki/Business\\_process\\_automation](http://en.wikipedia.org/wiki/Business_process_automation)



**Figure 3: Relationship between basic terminologies [7]**

# Chapter 3

## Main Background

To better understand this research and the proposed methodology, some theoretical definitions regarding the topic of the dissertation are provided in this chapter. First, the main background of the topic such as Project Portfolio Management, Business Process Management, Business Process Automation and Workflow Management are going to be represented at a glance. Afterwards, state of the art of the topic will be reviewed.

### 3-1 Project Portfolio Management Background

PPM was first examined by Lorie and Savage (1955) in the context of a capital budgeting problem in which firms choose from a selection of projects with different costs and returns, subject to a budget constraint. They acknowledged the complexity of the problem and used a trial-and-error method to obtain solutions to the problem. A decade later, Petersen (1967) used a knapsack formulation to solve the PPM problem and implemented an algorithm to find an optimal solution. However, the complexity of the model and the computational limitations of that time restricted him to smaller problems. Since these early days of PPM, many additional features have been incorporated into formal PPM models, including uncertainty with respect to project costs and returns, multiple resource constraints and the resulting threats of bottlenecks, correlations between projects in terms of both costs and returns, short-term versus long-term objectives and the degree to which projects fit the overarching strategy of the firm. [14]

### 3-2 Process Management background

This section provides brief information related to process management origin. Process Management refers to when an organization works with focus on its processes in a structured way. Historically, process management can be viewed on coming from three different branches: Business Management, Quality

control and IT-Management<sup>3</sup>. Process Management has its root in the early publications by Adam Smith and Frederick Taylor. In 1911 Frederick Taylor published “Principles of Scientific Management”, where he argued for work simplification, time studies and systems that measured output. In 1985, Michael Porter published his famous best seller “Competitive Advantage: Creating and Sustaining Superior Performance”, where he introduced The Value Chain. The Value Chain categorizes the generic activities of an organization and is a good tool for strategic planning. The Value Chain divides the activities of an organization into primary activities horizontally and support activities vertically. Defining activities is the foundation of process thinking and help understanding the processes in an organization<sup>4</sup>. At 1993 Michael Hammer and James Champy made a publication with the title “Reengineering the Corporation”.<sup>5</sup> With this publication many experts started to think that managing an enterprise means managing processes.<sup>6</sup> But the spirit began few years before by Frederick Winslow Taylor (1856-1915). This management thinking had an assumption: every time a process changes, the environment should remain stable so that the improved processes will have the time to return the required investment. Of course nowadays nobody should expect from the enterprises, the market and the environment to remain stable because the environment changes quickly and enterprises should be adaptive and react vigorously to those changes. Only with this behavior the enterprises would remain competitive and stakeholders will meet their expectations.<sup>7</sup>

### 3-2-1 IT Tradition

The IT Tradition of Process Management is Process Automation using computers and software applications. The IT tradition started with structured software methodologies and IT architecture. Today the information technology part of BPM involves Business Process Modeling tools and Enterprise architecture<sup>8</sup>. Enterprise architecture defines how information and technology supports business operations through a descriptive framework including goals, business processes, roles and software applications. Good enterprise architecture enables companies to respond to changing conditions in the environment<sup>9</sup>. Before the 1990's, process automation was known as software automation<sup>10</sup>.

---

<sup>3</sup> Harmon, P (2010)

<sup>4</sup> Porter, M (1996)

<sup>5</sup> Harper Business, (1993)

<sup>6</sup> Ramachandran, (2004)

<sup>7</sup> Ramachandran, (2004)

<sup>8</sup> Harmon, P (2010)

<sup>9</sup> Enterprise Architecture

<sup>10</sup> Harmon, P (2010)

### 3-2-2 Six Sigma

In 1986, Six Sigma was introduced by Motorola. Six Sigma started as a quality management strategy, using a set of quality management methods to decrease errors in manufacturing processes, but has come to be applicable to other business processes than manufacturing. Six Sigma is an important branch of quality control and process orientation that later led to BPM<sup>11</sup>. Six Sigma suffered from two limitations, the definition of a business process as any sequence of work activities, which results in an enormous amount of processes within a company. All of these will not have a strategic significance for the company as a whole and will just lead to an unmanageable range of small-scale projects. The second limitation within Six Sigma is the objective to eliminate the variation and the achievement of consistent performance and as consistent not is to be seen as a synonym for good, a process can be defined consistent even though it might not achieve the required level of performance.<sup>12</sup>

### 3-2-3 Business Process Reengineering (BPR)

The trend in process management during 90's was business Process Reengineering (BPR)<sup>13</sup>. BPR refers to when an organization completely remodel its processes. The founders of the term BPR, management consultants Hammer and Champy founded the process thinking that companies like SAP and Oracle use today. The BPR trend decreased when companies realized that they were facing the risk of accidentally cutting value-creating parts of the organization<sup>14</sup>. During the 1990s the term "Process" had a boost when productivity increased in organizations. Instead of looking at functions and procedures, companies started at looking at processes. Figure 4 shows what the process trend had looked like in the past 20 years and the tools used in different stages. As can be seen in the figure, Porter's Value Chain is the precursor in process management followed by six sigma and Process Reengineering of which Six Sigma has hold on and still have influences on today's BPM<sup>15</sup>. Business Process Reengineering neither has the dimension of continuous improvement nor a much disciplined approach, which BPM managed to achieve. However, BPR introduced the refined business process definition: "*end-to-end work across an enterprise that creates customer value*" which enables tracking of delays, non value-adding overhead, errors and complexity. The other contribution from BPR is the focus on process design instead of process execution,

---

<sup>11</sup> Ibid

<sup>12</sup> Harmon, P (2010)

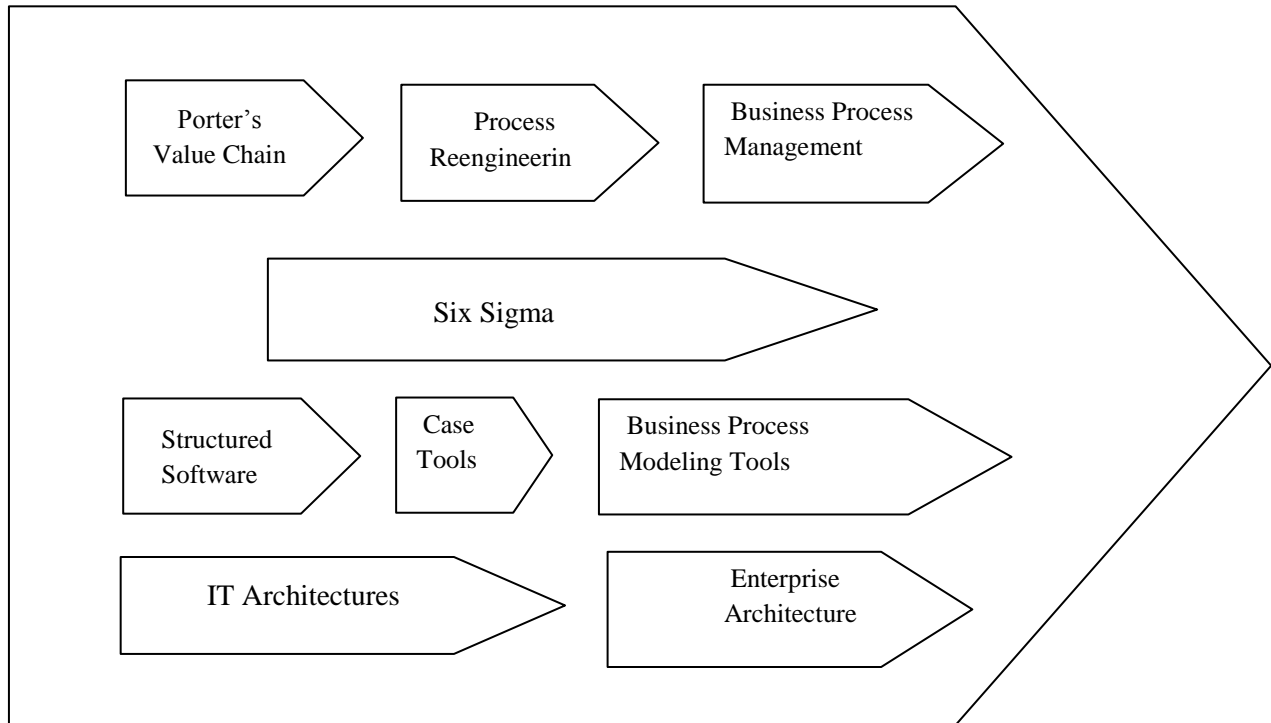
<sup>13</sup> Wallström, M (2010)

<sup>14</sup> Harmon, P (2010)

<sup>15</sup> Ibid



which means that if the performance requirements outdo the design capability, the design is to be replaced with a new one.<sup>16</sup>



**Figure 4: Process Management Development**

### 3-2-4 Business Process Management (BPM)

Business Process Reengineering (BPR) and Six Sigma are the two primary ancestors to BPM, which with their different strengths and weaknesses put up the foundation for the new process management, BPM.

BPM is seen as the third wave within business processes with its aim to emerge the work towards more effective processes, with a descent level of standardization without cutting the value creating parts of the processes. The aim of BPM is to make the processes support the business and in order to manage this, a BPM project must be on two legs, IT and Business.<sup>17</sup> The IT leg is Business Process Management System (BPMS), which refers to the software enabling BPM through facilitation of process design, workflow, applications, and integrations and monitoring.<sup>18</sup>

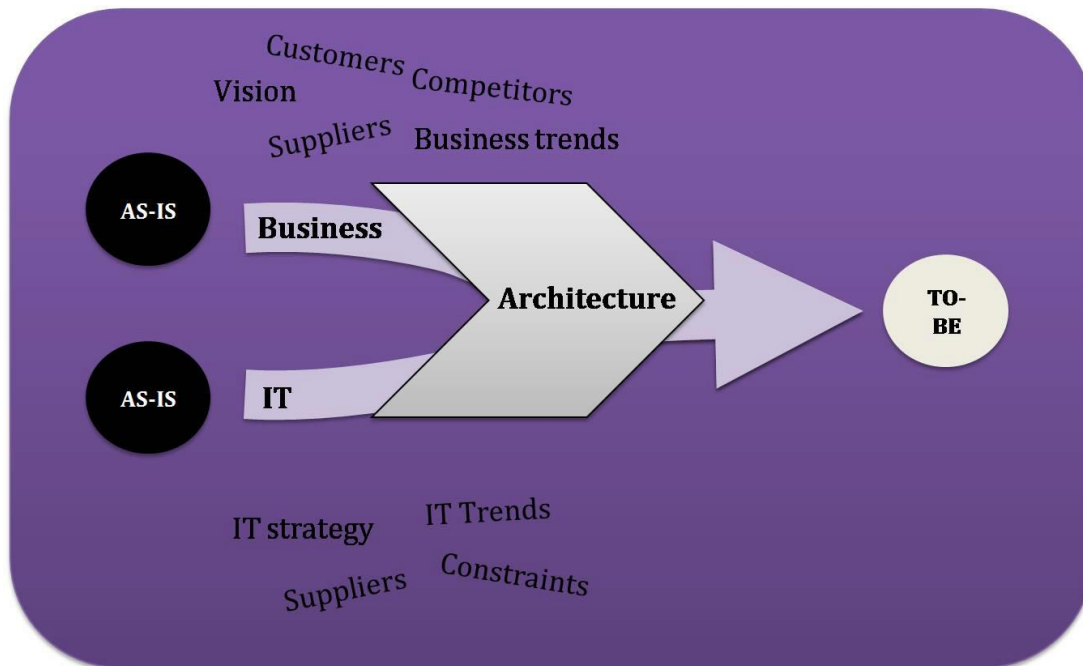
<sup>16</sup> Hammer, M (2010)

<sup>17</sup> Wallström, M (2010)

<sup>18</sup> Brännström, K-O (2011)

BPM technology started in 1980's and has developed significantly, providing enterprises with a series of advantages. BPM is nowadays the 'leader' of managing processes. Businesses and governments are paying attention to business processes in order to make a fully description, automation and management of them. Consequently, BPM is used with combination of computing technology in order to model and automate business processes.<sup>19</sup> Thus, BPM is an approach of “managing change” and specifically managing business processes.<sup>20</sup>

But what was the real situation before BPM years? How the enterprises were acting towards processes? The trend was to produce “data-aware” information systems instead of “process aware” as happens today with the help of BPM.<sup>21</sup> Before BPM, analysts and IT engineers should bring together different processes, methods, isolated systems and manual methodologies in order to work. With BPM no matter how many different systems and processes an enterprise has, all of them are treated as an aggregate “architected system”.<sup>22</sup> Junior learners of BPM should always keep in mind that BPM is not a product. It must not be considered as another IT approach, but as an environment where business processes are created and communicated to the organization.<sup>23</sup> Through architecture the business goals and the IT-goals can be aligned in order to reach a common goal, the To-Be state.



**Figure 5: Architecture aligning IT and Business (Berneskog, J, 2011)**

<sup>19</sup> Verner, (2004)

<sup>20</sup> Owen & Raj, (2003)

<sup>21</sup> Van der Aalst, (2009)

<sup>22</sup> Garimella, Lees, & Williams, (2008)

<sup>23</sup> Cho & Lee, (2010)

### 3-3 State of the Art

This section reviews the most relevant existing works that is state of the art related to the thesis. Their analysis has been necessary to determine the current state of research and practice and to adopt existing principles and mechanisms in the proposed solution. This section is presented combinatorial like as for the most important works could be viewed chronologically. First, the main leading PPM vendors are presented followed by a selected one and the reason to choose it. Next, different fields, disciplines and techniques that are related to business process modeling, process automation and workflow management are embraced.

#### 3-3-1 Main Competitors of Project Portfolio Management (tools)

PPM tools are increasing rapidly and today it is not possible to collect a complete and up-to-date list of PPM vendors and their products. There are now more than 100 tools for PPM! However, here in this study we mention only 10 of the most leading PPM vendors according to Forrester's evaluation. (See figure 5). The measurement is based on their support of above-the-line and below-the-line functionality. Above-the-line vendors support enterprises more focused in portfolio planning targeting analysis that considers benefits, value, and fit while below-the-line vendors are more interested in planning, execution and work management and emphasize on transactional project health such as scope, cost, and time. This evaluation of the PPM market is only a starting point and it is highly recommended to investigate on product details and evaluate according to the needs. Each of the 10 vendors that will be presented in this section has:

- **A formal product dedicated to IT and business-driven PPM.** The focus was to those vendors and products addressing the disruption that Lean and Agile brings to software development and delivery organizations today.
- **An active client base of more than 100 current paying customers.** Our evaluation focuses on vendors with a minimum active client count to address enterprises' risk in working with too small a technology provider.
- **More than 60% of its revenue is coming from IT PPM.** While organizations use PPM much more broadly than just in software and IT, Agile's impact hits here hardest. With business leaders increasingly relying on technology and software to achieve business outcomes, the need for PPM at the crossroads of business and technology planning drove this evaluation.

<b>Vendor</b>	<b>Product</b>	<b>Release date</b>
AtTask	Anaconda 2012	December 2012
CA Technologies	CA Clarity Project & Portfolio Management (CA Clarity PPM), v13.0 CA Clarity Ideation CA Clarity Requirements CA Clarity Agile	December 2011 May 2012 October 2010 May 2012
Clarizen	Clarizen 5.4	June 2012
Daptive	Daptive PPM	July 2012
GenSight	GenSight 8.4	April 2012
HP	HP PPM 9.14	March 2012
Microsoft	Microsoft EPM	May 2010
Planisware	Planisware 5.3	Q1 2012
Planview	Planview enterprise 10.4	February 2012
Rally	Rally 2012.07.07	June 2012

**Table 3: Evaluated Vendors: Product Information and Selection Criteria**

The vendor has a formal product dedicated to IT and business-driven PPM
The vendor has an active client base of more than 100 current paying customers
More than 60% of the vendor’s revenue comes from IT PPM

**Table 4: Vendor Selection criteria**

## ABOVE THE LINE VENDORS COMPARISON

As can be seen in Table 5, the above-the-line market is extremely competitive. Below, these 10 vendors are discussed in more detail with focus on their support of portfolio planning.

According to Forrester's evaluation, *CA Technologies*, *Planview*, *HP* and *Daptive* lead the pack. These vendors have strong top-down reporting capabilities while supporting strategic, portfolio management, application and project management as well. As can be seen in the Table 5, CA Technologies are the first two leaders in strategy with 4.15 and 4.92 (out of 5) respectively. They both have special strength in resource management. *Planview* is really a remarkable vendor in most of the selection criteria from demand management, project and portfolio management till service and business intelligence with score of 5 out of 5. It consistently delivers strong planning capabilities. *Daptive* investments in analytics, integration, and mobile have pushed the vendor into the Leaders category. In addition, Daptive makes good use of HTML5 to provide solid and consistent mobile and web experiences.

*GenSight* has one of the most robust portfolio management capabilities beside its strength in business intelligence, but its weak strategy (1.13/5) keeps it a bit far from strong performers. *Microsoft*, *Planisware*, *Rally*, *AtTask*, and *Clarizen* offer competitive options. Microsoft and Planisware have robust project management and portfolio analytics as well as considerable strength in resource management and business intelligence. Microsoft EPM offers a solid project oriented view on portfolios. Another strong feature of Planisware is its out-of-the-box portfolio scenarios and key performance indicators to monitor portfolios. However, they are not well focused on their strategic alignment capabilities, for example as can be seen in Table 5, Planisware's score for strategy is only 2.53 with product strategy of only 1.5 which is not a strong point. *Rally* and *AtTask*, with more focus on project health and resource performance, have strong project management and provide competent resource management, but they have the lowest overall scores in current offering because they are lighter on portfolio management and business intelligence capabilities than their competitors. Like Rally and AtTask, *Clarizen* is significant in project management and has better resource management capabilities. Solid demand management and collaboration support helps Clarizen stand out.

	Forrester's Weighting	AtTask	CA Technologies	Clarizen	Daptive	GenSight	HP	Microsoft	Planisware	Planview	Rally
CURRENT OFFERING	50%	2.62	4.25	3.05	3.58	4.24	3.62	3.55	4.09	4.73	2.59
Product fundamentals											
Deployment options	2%	3.60	4.60	2.60	3.20	2.00	3.80	3.80	4.60	3.40	3.80
Global support	2%	2.70	4.20	2.50	3.80	2.55	4.50	4.60	3.30	3.50	1.85
Administration	1%	3.00	5.00	5.00	5.00	5.00	2.50	3.00	5.00	2.50	3.00
Integration and customization	5%	3.00	5.00	3.00	4.00	3.00	3.00	2.00	3.00	5.00	3.00
Demand management	14%	2.60	4.20	4.20	4.20	4.20	4.20	4.20	3.80	5.00	3.00
Portfolio management	40%	2.20	4.00	2.10	3.30	4.80	3.00	3.00	4.10	5.00	2.30
Project management	2%	4.40	4.40	4.60	4.00	2.00	4.00	5.00	4.60	5.00	4.60
Resource management	15%	3.80	5.00	5.00	3.20	4.40	5.00	4.40	4.80	4.40	3.30
Business intelligence	10%	2.30	3.90	1.80	4.30	5.00	3.90	4.60	4.60	5.00	1.90
IT-specific functionality											
Application Lifecycle Management (ALM)	2%	3.50	4.50	4.00	4.00	2.50	3.50	4.00	4.50	4.50	4.50

Service management	5%	1.75	4.50	4.00	4.00	3.50	3.00	3.00	3.50	5.00	1.50
Application portfolio Management (APM)	2%	1.00	2.25	0.75	0.75	0.75	2.50	0.75	1.00	0.75	0.75
STRATEGY	50%	3.38	4.15	2.48	3.62	1.13	4.92	3.20	2.53	3.50	3.70
Support for implementations	25%	5.00	5.00	3.50	5.00	1.50	5.00	4.50	5.00	5.00	4.50
Product strategy	65%	3.00	4.00	2.00	3.00	1.00	5.00	3.00	1.50	3.00	3.50
Price	10%	1.80	3.00	3.00	4.20	1.00	4.20	1.20	3.00	3.00	3.00
MARKET PRESENCE	0%	3.63	3.15	3.30	4.00	3.00	2.40	2.31	2.30	2.99	3.40
Installed base	25%	4.50	5.00	4.50	5.00	4.50	5.00	1.25	5.00	0.75	4.00
Financial strength	50%	5.00	3.00	3.75	4.50	3.75	1.50	3.00	1.50	5.00	4.50
Support services	25%	0.00	1.60	1.20	2.00	0.00	1.60	2.00	1.20	1.20	0.60

**Table 5: Forrester Above-The-Line Project Portfolio Management, Q4'12 (Cont.)**

## BELOW THE LINE VENDORS COMPARISON

The evaluation uncovered a market in which core below-the-line planning capabilities helped established vendors score well (see Table 6). Newer vendors to the PPM market tackle planning and tracking functionality surprisingly well.

*HP, Planview, and CA Technologies* lead the pack. They have been prominent vendors in the past several years and still they offer competitive options. They help organizations to expand traditional planning features besides providing them with Agile planning. CA Technologies, in particular by providing both traditional and Agile approaches allows users to manage a wide variety of projects and work activities. Planview and HP offer robust capabilities in resource management with 5 and 3.80 respectively. Particular strength in demand and pipeline management besides versatile resource planning and significant reporting capabilities make these 3 vendors the leaders.

*Microsoft, Rally and Daptiv* competes closely. Rally, a new vendor to Forrester Wave incorporates ALM offerings to PPM solutions to deliver a high level of health in project management and application development. Microsoft has a close competition with Daptiv which is one of the first players in SaaS PPM market. They both offer a robust functionality in project and work management. *AtTask, Clarizen, and Planisware* offer significant flexibility with various environments. *Clarizen and AtTask* are strong in resource management and project management, but they have the lowest score in portfolio management regarding their competitors (2.02 and 2.36 respectively). They work well in team-focused environments, while *Planisware* provides organizations with more mature planning capabilities. It now offers SaaS implementations as an option. The product provides the ability to manage traditional projects and report on Agile projects. Clarizen's integrated email workflows allow users to work on projects both inside and outside of the application. *GenSight* offers strong portfolio management functionalities and comprehensive business intelligence. Despite its good rank in above-the-line market, here it has the lowest overall rank in Table 6 with the overall score of 2.99 in current offering and 1.13 in strategy capabilities. Its business analytic strength is overkill for small work teams.



	Forrester's weighting	AtTask	CA Technologies	Clarizen	Daptive	GenSight	HP	Microsoft	Planisware	Planview	Rally
CURRENT OFFERING	50%	3.56	4.48	4.05	3.67	2.99	3.90	3.93	4.12	4.44	3.78
Product fundamentals											
Deployment options	5%	3.60	4.60	2.60	3.20	2.00	3.80	3.80	4.60	3.40	3.80
Global support	5%	4.10	3.50	3.10	3.00	2.25	4.70	4.50	2.90	3.40	3.95
Administration	1%	3.00	5.00	5.00	5.00	5.00	2.50	3.00	5.00	2.50	3.00
Integration and customization	5%	3.00	5.00	3.00	4.00	3.00	3.00	2.00	3.00	5.00	3.00
Demand management	17%	2.40	4.00	4.60	4.00	4.00	4.00	4.00	3.40	5.00	3.00
Portfolio management	5%	2.02	4.40	2.36	3.80	4.70	3.00	3.00	3.42	5.00	2.80
Project management	20%	4.60	4.60	4.60	3.80	1.80	3.80	5.00	4.60	5.00	4.60
Resource management	20%	4.60	5.00	5.00	3.40	3.80	5.00	3.80	4.60	3.80	3.60
Business intelligence	2%	2.20	4.00	1.60	4.20	5.00	4.00	4.80	4.80	5.00	2.00
IT-specific functionality											
Application Lifecycle Management (ALM)	18%	3.20	4.50	3.70	3.70	2.20	3.20	3.70	4.50	4.50	4.80
Service management	1%	1.75	4.50	4.00	4.00	3.50	3.00	3.00	3.50	5.00	1.50

Application portfolio Management (APM)	1%	1.00	2.25	0.75	0.75	0.75	2.50	0.75	1.00	0.75	0.75
STRATEGY	50%	3.38	4.02	2.61	3.62	1.13	4.92	3.72	2.46	4.28	3.64
Support for implementations	25%	5.00	5.00	3.50	5.00	1.50	5.00	4.50	5.00	5.00	4.50
Product strategy	65%	3.00	3.80	2.20	3.00	1.00	5.00	3.80	1.40	4.20	3.40
Price	10%	1.80	3.00	3.00	4.20	1.00	4.20	1.20	3.00	3.00	3.00
MARKET PRESENCE	0%	3.63	3.15	3.30	4.00	3.00	2.40	2.31	2.30	2.99	3.40
Installed base	25%	4.50	5.00	4.50	5.00	4.50	5.00	1.25	5.00	0.75	4.00
Financial strength	50%	5.00	3.00	3.75	4.50	3.75	1.50	3.00	1.50	5.00	4.50
Support services	25%	0.00	1.60	1.20	2.00	0.00	1.60	2.00	1.20	1.20	0.60

**Table 6: Forrester Wave™: Below-The-Line Project/Program Portfolio Management, Q4 '12 (Cont.)**

## Why Clarity PPM?

According to evaluations of the above tables, here we provide some of the reasons why we chose Clarity PPM. However, it can be seen how competitive the PPM market is, and it is really hard to select one of these vendors as the best choice and selection should be performed with respect to companies needs.

As can be viewed in Table 5, CA Technologies is the only vendor that has strong features in both transactional and strategic levels beside its good score at its presence in the market. It provides robust capabilities in all the evaluated criteria such as demand management, project and portfolio management, resource management, etc. It has no score lower than 3 except in Application Portfolio Management (APM) and Product Service, that concerning its competitors, none of them could reach a score greater than 2.5 and it offers one of the most solid solutions among them. It supports global deployment which let users to either start quickly via software-as-a-service (SaaS) or proceed on a traditional on-premises route that supports a number of configurations. CA's adaptable integration strategy permits companies to bring in critical data from third-party applications such as ALM and service portfolio management to analyze performance and health to determine portfolio alignment. Recent investments in reporting and application portfolio management enable CA to provide greater transparency and insight into BT planning strategies. Regarding below-the-line evaluations, CA Technologies is again an accomplished Leader. With score of 4.60 out of 5 in project management, it provides one of the best project and work management. From demand management till resource planning, it offers strong functionalities among its competitors. CA Technologies offers both traditional and Agile approaches which let users manage a wide range of projects and work activities. Agile organizations are able to collaboratively plan and execute projects regardless of approach. It can be seen from Table 5 and Table 6 that among all other evaluated vendors, CA Technologies is the one that offers robust functionalities in almost all studied criteria in both above-the-line and below-the-line evaluations.

### 3-3-2 Business Process Modeling Standards

Process modeling is widely used within organizations as a method to increase awareness and knowledge of business processes, and to deconstruct organizational complexity.<sup>24</sup> It is an approach for describing how businesses conduct their operations and typically includes graphical depictions of at least the activities, events/states, and control flow logic that constitute a business<sup>25</sup>. It plays an important role in many fields both in industry and in academia and help people to communicate and understand the running

---

<sup>24</sup> Bandara et al., 2005

<sup>25</sup> process Curtis et al., 1992; Davenport, 2005

of an organization. They are used for multiple initiatives, such as business process reengineering, business process management or software development. In summary, business process models are used within organizations for learning, for decision support about process development and design, for control and decision support during process execution, and for analysis of information technology support.<sup>26</sup>

A sign of the relevance of business process modeling is the high number of existing notations for this purpose<sup>27</sup>, each one of them with their specific purposes, strong points and weaknesses. Another characteristic of current work on notations for business process modeling (and on languages for business process execution) is the existence of standardization efforts.<sup>28</sup> However, since each notation has its specific goal, there is not a “universal” notation to model business processes. Some notations defined for industrial purposes and later they have been used for academic research while some others first appeared in academia and later have been adopted in industry. Moreover, such a unique notation could become too complex to understand. Since review of all the existing notations in this thesis is not possible, here we present the notations that can currently be considered the most relevant ones in academia and industry.

Numerous notations compete in the business process modeling space, including UML Activity Diagrams, the Business Process Modeling Notation (BPMN), Event driven Process Chains (EPCs), Workflow nets, and the Business Process Execution Language (BPEL) – the latter one being intended for executable specification rather than modeling.

### *3-3-2-1 BPMN (Business Process Modeling Notation)*

In May 2004, the Business Process Modeling Notation (BPMN) was released by Business Process Management Initiative (BPMI) under supervision of Steven White. It is graphical representation of business processes which was the result of more than two years of effort by the BPMI Working Group and is now maintained by OMG (Object Management Group). Moreover, BPMN is a bridge to fill the gap between business processes design and their implementation. Although BPMN can be used for both business modeling and system modeling, it is more business-oriented. The main goal of BPMN was to deliver a graphical notation which is easily understandable by all users of business process from business analysts to technical developers. The notation offers a Business Process Diagram (BPD) to build graphical models of business process operations which consists of a set of graphical elements. These elements are categorized to Flow Objects, Connecting Objects, Swimlanes and Artifacts. Flow Objects are Activity, Event and Gateway. Events usually have the triggers or the impact and affect flow of the

---

<sup>26</sup> Aguilar-Savén, 2004

<sup>27</sup> Dumas, van der Aalst, ter Hofstede, 2005; Weske, 2007

<sup>28</sup> Ko, Lee, Lee, 2009

process. Based on when they affect the process, they could be Start, Intermediate and End. Activity is a generic term for the work that company performs and is in two types: sub-processes and tasks. Gateways are to make decisions as well as forking, merging and joining paths (exclusive, inclusive, complex and parallel gateways). Connecting Objects make connections between other Flow Objects. They are sequence flows (normal and default), message flows and associations. Swimlanes represent participants of a business process. They can be pools and lanes. Artifacts provide additional information about a business process model. Three types of BPD Artifacts are: data objects, groups and annotations. Moreover, BPMN allows more graphical objects based on users needs. Finally, the notation is supported by many tools and vendors.

### *3-3-2-2 EPC (Event-driven Process Chain)*

Event-driven process chains are an intuitive graphical business process description language introduced by Keller, Nuttgens and Scheer in 1992 at University of Saarland in Germany. EPC is a graph-based language that aims to define processes that are easily understandable by business people and non technical users. It is highly adopted in Germany, Austria and Switzerland both in industry and in academia.<sup>29</sup> Many German companies and researchers have widely used EPC maybe because it was invented in Germany. It consists of Functions, Events and Connectors and supports parallel execution of processes. Functions represent activities (tasks, process steps) within an organization. Events may correspond to the pre-conditions (condition before a function can be performed) or post-conditions (what occurs after performing function). Functions are linked by events. Connectors are used to connect activities and events. They are logical decision points and could be split or join. Split is when the connector has one incoming arc and multiple outgoing arcs while Join is when there are multiple incoming arcs and one outgoing arc. An important feature of EPC is its logical operators that are in three types: AND (denoted  $\wedge$ ), OR (denoted  $\vee$ ) and XOR (denoted  $\times$ ). However, the semantics and syntax of the EPC are apparently not well defined.<sup>30</sup>

### *3-3-2-3 UML (Unified Modeling Language)*

UML (OMG, 2005), is a standard Object Oriented (OO) modeling language for specifying, visualizing, constructing and documenting the artifacts of software systems, as well as for business modeling and other non-software systems [15]. It provides a key foundation for OMG's Model-Driven Architecture,

---

<sup>29</sup> Becker, et al., 2010

<sup>30</sup> van der Aalst, 1999; Kindler, 2004

which unifies every step of development and integration from business modeling, through architectural and application modeling, to development, deployment, maintenance, and evolution [16]. UML 2.0 has 13 diagrams divided into three categories which are Structure diagrams, Behavior diagrams and Interaction diagrams each type consists of some diagrams. Among these 13 diagrams, activity diagram is targeted for business process modeling or describing flow of works through series of actions in a software system. In general, Activity diagrams could be used for:

- Describing a business process or a flow of work
- Describing the steps performed in a use case.
- Describing a method, function or operation in software.

Model elements of Activity diagrams are:

**Activities:** describe the highest-level of behavior and contain several activity nodes and activity edges to show sequence of tasks in a workflow.

**Actions:** represent an executable unit of functionality in an activity. Actions have incoming and outgoing activity edges that specify the flow of control and data to and from other activity nodes.

**Control nodes:** a control node is to coordinates the flow of control in an activity. Decision, Merge and Fork are some sample types of control nodes.

**Object nodes:** an object node helps to define the object flow in an activity. It indicates that an instance of a classifier might be available at a particular point in the activity.

**Activity edges:** an activity edge makes a direct connection between two activity nodes. Activity edges could be in two types: Control Flow edges and Object Flow edges.

- Control flow edges model the movement of control from one node to another by passing a control token
- Object flow edges model the flow of objects or flow of data from one node to another by passing objects or by passing data tokens

#### ***3-3-2-4 YAWL (Yet Another Workflow Language)***

YAWL (Yet Another Workflow Language; (van der Aalst et al., 2004; van der Aalst and ter Hofstede, 2005)) is a workflow language based on a workflow patterns. It is the result of joint effort between

Eindhoven University of Technology and Queensland University of Technology to overcome some limitations of Petri nets and contemporary management systems. In order to overcome the limitations of Petri nets, YAWL has been extended with features to support more workflow patterns involving multiple instances, advanced synchronization and cancellation patterns. Moreover, YAWL allows for hierarchical decomposition and handles arbitrarily complex data [17]. YAWL is inspired by Petri nets but is not just a macro package built on top of high-level Petri nets; It is a completely new language with independent semantics. [18] YAWL has been evolved considerably in the past several years. It is taught in many universities and is used also for industrial projects. The language is XML based and is supported by a software system that includes an execution engine, a graphical editor and a work-list handler. The environment is open source and is structured as a service oriented architecture. YAWL is a state-based language with a formal semantics. Main elements of YAWL notation are: Conditions, Tasks, Joins, Splits, Cancellation regions.

**Conditions:** used to represent the state of a process and can be interpreted as places. They are in two types: Input Conditions and Output Conditions. The first one shows where a process starts and the latter one is used to show where process finishes.

**Tasks:** used to define actions. Composite task, Atomic task and Multiple instance of (Composite or Atomic) task are various types of tasks.

**Joins:** AND join, OR join and XOR join are three types of joins used to represent the union of several paths.

**Splits:** represent the division of a path; like joins they can be AND, OR and XOR splits.

**Cancellation regions:** represent element deactivation upon activation of a task.

### *3-3-2-5 Clarity Process Flow Diagrams*

CA Clarity Provides a comprehensive way of automating business processes. It makes processes more coherent, automated and adoptable which increase the quality of processes, contain costs and lead to more business agility of an organization. CA offers structured flow diagrams which enable business analyst to take action immediately by allowing him to check where each item is in the process. So, it saves your time, increases productivity and degree of satisfaction in customers. Users who have to take actions also receive a notification alert automatically and so there will be no need for endless repetitive emails or calls. Workflow notifications can include:

- Events, such as approvals and rejections

- To-dos, such as new assignments and action items
- Notes to fully explain follow-up actions
- Time submission and approval
- New documents added to the system [19]

Main elements of flow diagrams in Clarity are splits and joins. A split branches processing into multiple directions, while a join consolidate the process flow.

**Splits:** There are four types of splits in Clarity flow diagrams that are Serial split, Parallel split, Decision Point split, Multi-choice split

- **Serial Split:** A serial split is a step that activates only when another step in the same process completes.
- **Parallel Split:** A parallel split is a point in a workflow where a single thread splits into several steps. The multiple steps can execute in either parallel or simultaneously.
- **Decision Point Split:** An exclusive choice (XOR-split) is a point in the workflow, where one of several branches executes based on either decision or control.
- **Multi-choice Splits:** A multi choice split (OR-split) is a point in the workflow. The system can select from a number of branches based on decision or control data. With the type of split, more than one condition can be true.

**Joins:** Joins are where the process flow of two or more steps is merged into a single process flow. The following types of joins are available: Rendezvous (AND), Merge (XOR), Wait and Merge, Multi-thread, First-in-line. [20]

It is possible to check the status of each step in a process. When it is in active mode, they are shown by below colors:

Green	The step is completed.
Yellow	The step is running (it is being processes)
Red	There is a problem in the step.
Blue	The step is ready to start and just is waiting for a proceeding action to be completed.
White	The step is not started yet.

Table 7: Various status of the steps within the active process



### 3-3-3 GEL Script (Generic Execution Language)

GEL (Generic Execution Language) is a scripting language based on Jelly libraries a jakarta.apache.org Commons project and is used in Clarity software. It is a tool that can be used to turn XML into executable code which has been extended and embedded into CA Clarity PPM to enable custom logic to solve business problems. GEL is the basis for the enterprise application integration framework within CA Clarity PPM. [21] The GEL run-time is packaged with XOG in the XOG client. Once the client is installed, it is possible to use the GEL command in the *bin* directory of the XOG client to validate and execute GEL script. GEL has a validator which read scripts in order to confirm if all tags are valid and the scripts are well-formed XML. Here is the sample GEL script that is first validated, and then the hello.xml script, located in the XOG client home directory, is executed:

```
E:\XOG>bin\gel -script hello.xml -validate
File geltest.xml validated.
```

```
E:\XOG>bin\gel -script hello.xml
Hello World 1!
Hello World 2!
Hello World 3! [22]
```

The following figure shows the basic structure for a GEL script.

```
<?xml version="1.0" encoding="UTF-8"?>
<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
           xmlns:core="jelly:core"
           xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
           xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
           xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
           xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
           xmlns:sql="jelly:sql"
           xmlns:xog="http://www.niku.com/xog"
           xmlns:xsd="http://www.w3.org/2001/XMLSchema"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <!-- Code goes here -->
</gel:script>
```

A GEL script is built from qualified elements bound to java code called tags. Using namespace declarations, tags are organized into tag libraries available in a script. Below is the Hello World example in which two tag libraries are defined for the script: Core and GELTagLibrary as seen in tag pairs such as: `<core:??></core:??>` and `<gel:??></gel:??>`. [23]

```
<gel:script xmlns:core="jelly:core"
           xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary">
```

```
<core:forEach indexVar='i' begin='1' end='3'>
  <gel:out>Hello World ${i}!</gel:out>
</core:forEach>
</gel:script>
```

Variables are used a lot through GEL scripts. Many tags can set variables. For instance, core:set is one of the tags which can set variables. The syntax \${variable\_name} is used to refer to variables. In the Hello World example, 'i' is a variable which is set by the forEach tag and is incremented with each loop. Core is a built-in Jelly library that contains general-purpose tags. GELTagLibrary is the primary GEL library.

### **XOG (XML Open Gateway)**

CA Clarity Process Manager supports Business Process Management Language (BPML), an industry-standard, comprehensive means of specifying the processes of an enterprise. CA uses a web service interface called XOG (XML Open Gateway) which includes published, open-standards-based interfaces to import and export data between Clarity PPM and other external systems. XOG read and write data objects to Clarity, execute queries and other server-side actions. It uses Simple Object Access Protocol (SOAP), an open-standard, human-readable, XML-based protocol for communication. [23]

## Chapter 4

### Proposed Methodology

This chapter presents the methodology used in this thesis. First various types of research methodologies and data collection are described briefly followed by the selected methods for this study. Next, description of the problems that this thesis must address is presented followed by the methodological approach used in this study to reach the thesis targets.

#### 4-1. Research Methods: Deductive vs. Inductive

There are two types of reasoning in research analysis approach: Deductive and Inductive methods. Deductive approach (sometimes called “top-down” approach) works from more general to more specific while inductive type (sometimes called “bottom-up” approach) works from specific observations to more general theories.<sup>31</sup>

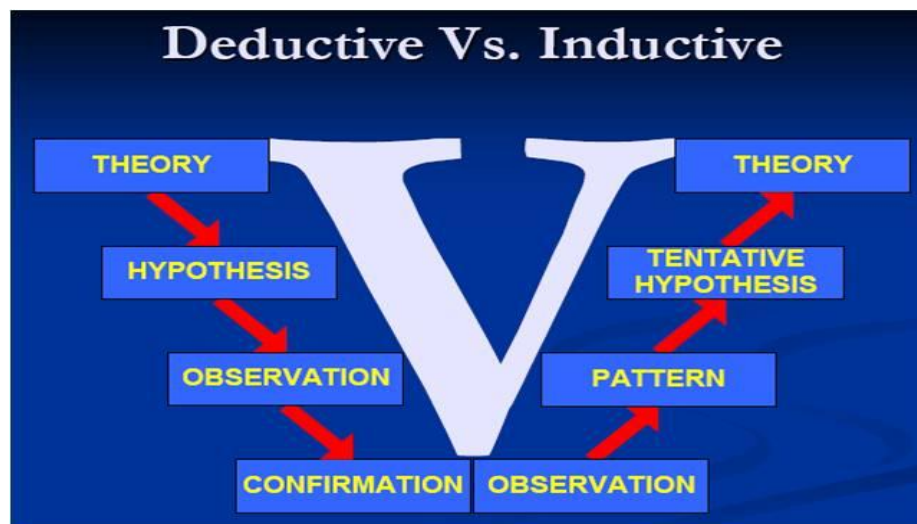


Figure 6: Deductive Approach vs. Inductive<sup>32</sup>

<sup>31</sup> Burney, 2008

<sup>32</sup> Burney, 2008

In deductive reasoning, conclusion follows logically from premises (available facts) whereas inductive approach has a conclusion likely based on premises. Deductive reasoning uses arguments based on laws, rules and accepted principles. Observations are generally used for Inductive arguments.<sup>33</sup> This study used deductive approach.

## 4-2 Qualitative vs. Quantitative analysis

The kind of data that we collect and use in our research will dictate the type of methodologies that we use in our study. In general, there are two main types of data to be collected: Qualitative and Quantitative.

Quantitative method refers to when the collected data for the study are measurable and involves those methodologies, such as closed surveys structured interviews and sociograms (diagrammatic representations of interactions between individuals) which enable data to be collected, measured and compared with a standard.<sup>34</sup> When using quantitative analysis, existing theories and hypothesis are verified and data are collected in accordance with underlying research questions. Based on the collected data, the results are used to generate a new hypothesis. Quantitative-oriented researchers are encouraged to exploit "the potentialities of social observation". Among other assets, field methods can contribute to survey analysis with respect to the validation of results, the interpretation of statistical relationships, and the clarification of puzzling findings. Thus, informants can be utilized during the course of quantitative research and "holistic interpretation" (i.e., context variables) can be used to shed light on quantitative data. More implicitly, the very selection of a research site is typically a function of qualitative data as is the process of building a survey instrument. [24]

However, since everything cannot be measured numerically, quantitative data collection cannot find the answer to understand meanings, experiences, ideas and beliefs. They are better described by qualitative data. Qualitative research is a method of inquiry employed in many different academic disciplines, traditionally in the social sciences, but also in market research and further contexts. Qualitative research involves the studied use and collection of a variety of empirical materials-case study; personal experience; introspection; life story; interview; artifacts; cultural texts and productions; observational, historical, interactional and visual texts- that describe routine and problematic moments and meanings in individual's lives. [25]

---

<sup>33</sup> Burney, 2008

<sup>34</sup> [http://scansw.com.au/sc\\_pages/sc\\_soccultwld/sc\\_meth\\_glossary.html](http://scansw.com.au/sc_pages/sc_soccultwld/sc_meth_glossary.html)

*“Qualitative analysis transforms data into findings. No formula exists for that transformation. There is Guidance; but no recipe. Direction can and will be offered, but the final destination remains unique for each inquirer, known only when—and if—arrived at”, (Michael Quinn Patton, 2002).*

Qualitative studies are used to have a deeper understanding of a specific subject or situation. They cannot be presented in numerical forms. The most important feature of qualitative analysis is its focus on text rather than on numbers. The analyzed “text” is usually the notes from interviews or participant observations. In Qualitative method, the focus is on the way that participants interpret their experiences. An unstructured interview, focus group, open-ended questionnaire and participant observation are some examples of Qualitative research.

In this thesis, both methods were used to collect data. Since the purpose of this study is to understand how processes work and to evaluate their impacts on system performance, so some data came from BPM professionals besides experiences and perceptions of experts in this field. As a result it is Qualitative. Moreover, it can be considered as Quantitative research because we used existing hypothesis and theories.

### **4-3 Data Collection**

There are two types of data collection techniques: Primary and Secondary data collection. Primary data collection uses survey, experiments or direct observations. On the other hand, Secondary data collection may be conducted by collecting information from a diverse source of documents or electronically stored information such as journals or literature.<sup>35</sup> In this thesis, both techniques are used. Primary data obtained from a few number of interviews and secondary data are collected from literature.

Research on business processes started with studying literature. Although Business Process Management and Business Process Automation are relatively young field, however with considerable interests on this topic in the past years, some useful information are available in literature on the subject. Indeed, studying literature could help to understand the studied topic. Studying business process and process automation background could help to identify the most famous researchers in the field with their ideas and experiences. For example, Hammer’s thoughts about which processes are most suitable for BPM were really interesting.

In order to obtain additional information and collect different views of professionals about business processes automation and the way to implement them in organizations, several interviews were held. Although they are not many in numbers, they were necessary for this study. The reason is that lots of

---

<sup>35</sup> <http://www.slideshare.net/hezel3210/primary-secondary-data>

valuable information has been gained through interviews with professionals and experts in this field who have different experiences, knowledge and positions. Interviews could give the opportunity to know how this approach works in practice and to be familiar with the problems and issues that exist in real environment which could not be discovered by studying literature.

#### **4-4 Problem Description**

In the past few decades, process orientation has been spread considerably in organizations. One common way of process orientation is BPM through process automation and Business Process Management Systems. Support for business process enactment, automation or execution based on business process models is among the top 10 business process modeling challenges in industry and in academia. [26]

This thesis was done in Mediobanca which chose Clarity PPM as project portfolio management software in which they automate their business processes. The reason for why they selected Clarity is described in section 3-3-1 after comparing the most famous PPM softwares. As mentioned before, Clarity uses Data Flow Diagrams to model its business processes. It supports different types of action steps such as system action, sub processes, custom script, etc. to automate processes. However, this study focuses more on those processes with custom script (GEL).

The problems that this thesis addresses are described through the following questions which are answered in chapter 5:

1. How automating business processes in Clarity PPM are implemented?
2. What are the advantages of using GEL script for process automation in Clarity?
3. What are the impacts of running these processes on application performance?
4. How these processes could be optimized to avoid performance issues?

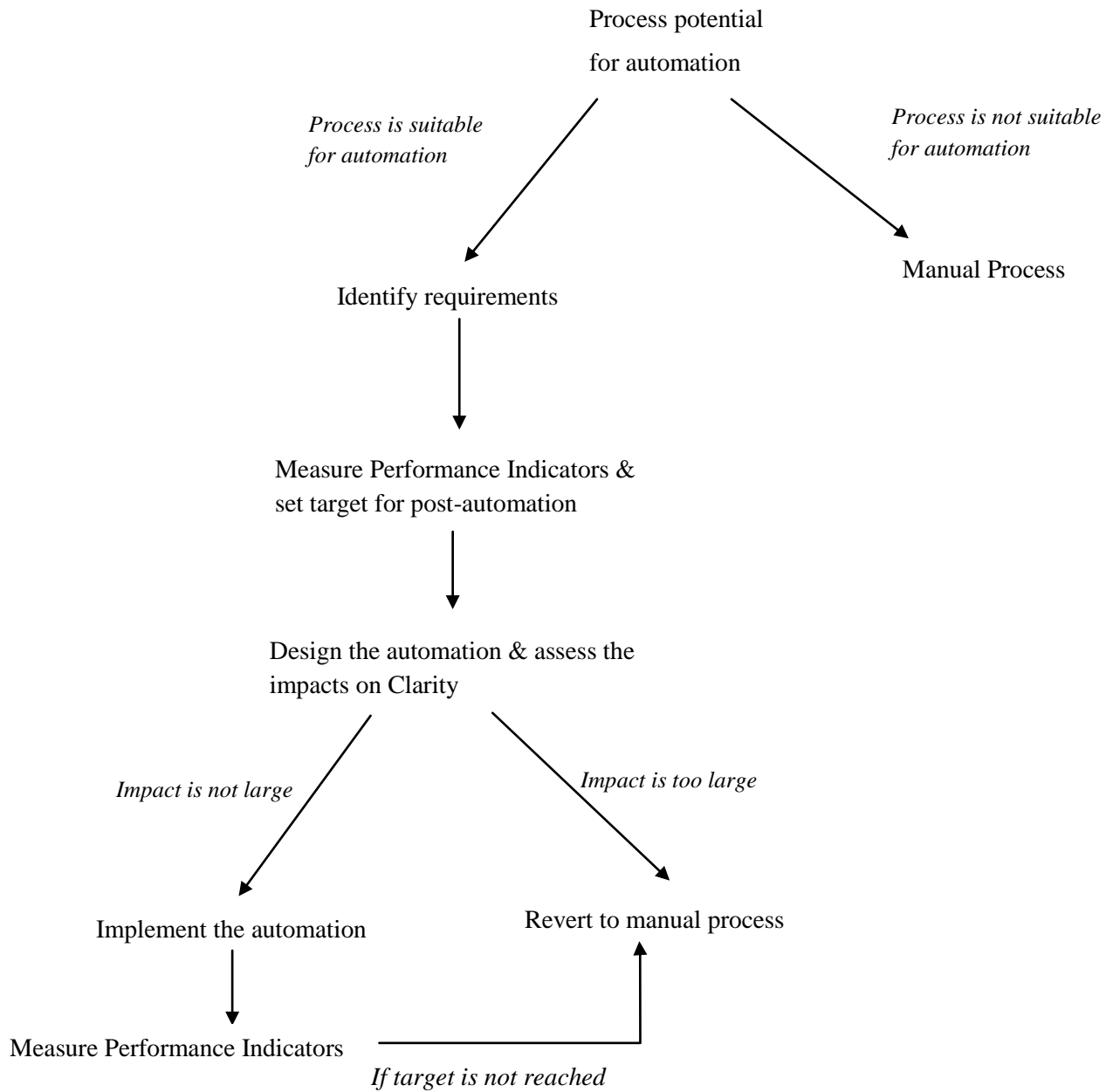
#### **4-5 Methodological Approach**

The objectives of this study will be achieved through an appropriate methodological approach for modeling and implementing business processes automation. The approach consists of the following stages:

- A. Collect requirements for automating parts of a business process using clarity
- B. Identify one or more performance indicators for the part of the business process to automate

- C. Measure the performance indicator before automation and set a target for the post-automation
- D. Design the automation and assess the impact on clarity (this is a decision point, if the impact on clarity is too large the automation may not be implemented)
- E. Implement the automation, e.g., using GEL script, following the best practice to minimize the impact on Clarity
- F. Assess if the automation is the expected impact on Clarity, if not optimize.
- G. Measure the performance indicator(s) and check if the target(s) set in step C has (have) been reached. If the targets have been not reached within a period of time revert to the manual process.

The first step is to identify the processes which have the potential for automation. When suitable processes are defined, the next step would be stating process automation requirements. Then, we have to identify and measure one or more performance indicators in order to evaluate impacts of running these processes on system performance. After that, we have to design the automation including process modeling by flow diagrams and assess the impacts of automation on Clarity PPM. This assessment would help us to decide whether the process automation should be implemented or not. If the impacts are not too much or important, implementation will be done. Otherwise, it should be considered if there are some best practices to minimize those impacts. After implementation, the automation's targets should be checked. If the identified targets are not reached, it should be reverted to the manual processes. The methodological approach used for this study is shown in the figure below:



**Figure 7: Thesis methodological approach**



# Chapter 5

## Implementation Experience & Empirical Observations

This chapter describes the implementation experience in Mediobanca and empirical observations. The achieved experimental results are also going to be illustrated at the end of this chapter. At First, in section 5-1, some necessary information on how to automate processes in Clarity is provided. During this study in Mediobanca, about 20 processes were automated using Gel Script within Clarity PPM which is described briefly in table 8, but here implementation of only two processes are discussed in detail as a sample of this work. The first process is a sample of a process without object which has a very simple flow diagram and the second one is a process with more complicated flow diagram and more number of steps.

### 5-1 How to Automate Processes in Clarity

This section contains necessary information about Process automation in Clarity in order to better understand the implementation experience.

As mentioned before, Processes in clarity are automated repetitive steps to facilitate actions which you could otherwise perform them manually. They contain objects, steps, actions, groups of steps, and joins. All processes have a start and a finish step and use conditions to connect these steps. Processes are run in order to save time and cost in most of the cases.

The first step to create a process is to consider what we would like to accomplish, who should do it and how it must be done. So, before creating a process, we should plan it. Then process properties should be defined. Objects (Primary or linked objects) could be added to a process only if we would like to define manual actions, system actions or sub processes. It is possible to modify, to copy or delete a process. The

sample process that will be described in this chapter run through custom script and it does not need objects.

Processes have two start options: On-demand, Auto-start

- **Auto-start:** means we cannot define when the process starts. It will start automatically when the start event occurs.
- **On-demand:** means we can define when the process should start. In this case there is an start bottom in available processes page and has a check box beside it which we can select and start the process.

The next step would be defining step actions. Step actions are used to send notifications and receive approvals. For example, system actions can be used to call Application Program Interface (API) functions, set attributes or lock attributes. To use a system action in a process, add a step to the process and then supply parameters to the system action for that step. For example, to copy documents from a project template, you can use a system action. A step can have no actions or multiple actions. An action in a step may or may not be dependent on the results of previous actions in the same step. [27]

After creating step actions, process should be validated. It is possible to validate the entire process or validate each step in the process. For example, a certain step can be validated right after creating or modifying the step definition. It must be noticed that when modifying a validated process, the process will become invalid and should be validated again. After validating processes and before the process instance is initiated, it should be activated.

**Process Instance:** When you start a process, a process instance is created. All process instances, those to which you have access and those that you have created, are displayed.

Processes could be started manually or automatically (event driven). Before manually starting a process, the below aspects should be true:

- *Process-Manage* or *Process-Start* access right to the specific process.
- Explicit access rights to the specific object.
- All actions, post-conditions, and pre-conditions are correctly specified.
- All of the steps are connected and the status is “Validated and Active”.

## 5-2 Automated Processes

The table below shows the 20 processes automated in Clarity during this study. It provides a brief description of each process along with start options, conditions and list of their action steps.

	Process Name	Description	Start Options and Conditions	Action Steps
1	MB Check date	Checks if the request start date and finish dates are set correctly. If not, set them to start date = 1900 and finish date = 2079 which are the system default dates/	No primary object is selected, this will be an On Demand process	Check date
2	CP Aggiorna stato	Automatically apply any changes in the status of the requests in real time	Auto-start , Start event: Update, Start Condition: ( ( ( ( ( ( Idea Nuovo Stato = 'Autorizzata' ) or ( Idea Nuovo Stato = 'In Lavorazione' ) ) or ( Idea Nuovo Stato = 'Preso In Carico' ) ) or ( Idea Nuovo Stato = 'Rifiutata' ) ) or ( Idea Nuovo Stato = 'Non Autorizzata' ) ) or ( Idea Nuovo Stato = 'Chiusa' ) ) and ( Idea Nuovo Stato != Idea Nuovo Stato [Previous Value] ) )	Aggiorna date, Aggiorna stato, Stato rifiutata, Stato non autorizzata, Stato chiusa, Stato autorizzata, Stato preso in carico
3	CP Creazione RFC v2.0	Workflow attivato in fase di Creazione della RFC	Auto-start, Start event: Create	Update status by created-by, Update LOG RFC, Set OBS assignment, Call CP Creazione Folder KS, Update log RFC APP, Call CP RFC Approvata, Set Area Inf Aut, Set IdeaView Pers. Intermedio

	Process Name	Description	Start Options and Conditions	Action Steps
4	CP Insert Personale Area Informatica	The process assigns, automatically, the Request to the Manager of IT department	Auto-start, Start event: Update  Start Condition ( ( ( Idea Personale Area Inf. != Idea Personale Area Inf. [Previous Value] ) and ( Idea Personale Area Inf. [Previous Value] = null ) ) and ( Idea Flag Per Area Inf Ins != 1 ) )	Insert access right, Add access folder
5	CP Insert Resource in Group	This process insert resources in groups	Auto-start, Start event: Create	Insert Resource in group
6	CP Modifica Assignment	Workflow attivato a seguito di una modifica dell'attributo assignment e si occupa di modificare l'obs e di fornire all'utente i diritti di edit sulla RFC indicate	Auto-start, Start event: Update, Start Condition ( ( Idea Assignment != Idea Assignment [Previous Value] ) and ( Idea Assignment [Previous Value] != null ) )	Modifica OBS RFC, Set Personale Area Inf
7	MB Inattiva Request old 15 days	Il processo inattiva le request e disattiva il flag open for time entry per le request chiuse da più di 15 giorni	No primary object is selected, this will be an On Demand process	Inactive le request
8	MB - Notifica per request chiusa	When Organization units' requests which are type Long (Manutentiva lunga, Evolutive, Strategiche) or the type "Attività /organizzativa" is closed, the creator of the request will be informed by a notification email.	Auto-start , Start event: Update, Start Condition: ( ( ( ( ( Idea Type = 'Richiesta Manutentiva Lunga' ) or ( Idea Type = 'Richieste Evolutive' ) ) or ( Idea Type = 'Attività Organizzativa' ) ) or ( Idea Type = 'Richieste Strategiche' ) ) and ( Idea Status = 'Closed' ) ) and ( Idea Status != Idea Status [Previous Value] ) )	Notifica creatore request chiusa

	Process Name	Description	Start Options and Conditions	Action Steps
9	MB Workflow Richiesta	This process shows the workflow of various types of requests in Mediobanca.	Auto-start , Start event: Create	Approvazione, Notifica approvazione Manager Admin Creator e customer, Setta initiated by process, Aggiorna team, Verifica cambio manager, Notify reject administrator, Send notification Boss Organization
10	MB Notifica mail per inserimento Due Date	This process sends a notification email to insert due date for the requests of type “Long”	Auto-start, Start event: Update, Start Condition ( ( ( Idea Due Date [Previous Value] = null ) and ( Idea Due Date != null ) ) and ( ( ( Idea Type = 'Richiesta Manutentiva Lunga' ) or ( Idea Type = 'Richieste Evolutive' ) ) or ( Idea Type = 'Richieste Strategiche' ) ) ) )	Notifica Inserimento Due Date
11	Notifica mail per modifica Due Date	This process sends a notification email to modify due date of requests type “Long”	Auto-start , Start event: Update, Start Condition: ( ( ( ( Idea Due Date [Previous Value] != null ) and ( Idea Due Date != Idea Due Date [Previous Value] ) ) and ( Idea Due Date != null ) ) and ( ( ( Idea Type = 'Richiesta Manutentiva Lunga' ) or ( Idea Type = 'Richieste Evolutive' ) ) or ( Idea Type = 'Richieste Strategiche' ) ) ) )	Notifica Cambio Due Date
12	MB Notify Manager Missing Timesheet	This Process sends an email to both resources who did not compile their timesheet and their Managers.	No primary object is selected, this will be an On Demand process	Send email
13	MB Processo Creazione Progetto	Copy data from requests	Auto-start, Start event: Create	Import data of request
14	Ricalcolo Effort Consuntivato	Calculate ETC and efforts at project level	No primary object is selected, this will be an On Demand process	Controllo pesi /ETC, Calcolo ETC consuntivato, Invia Mail

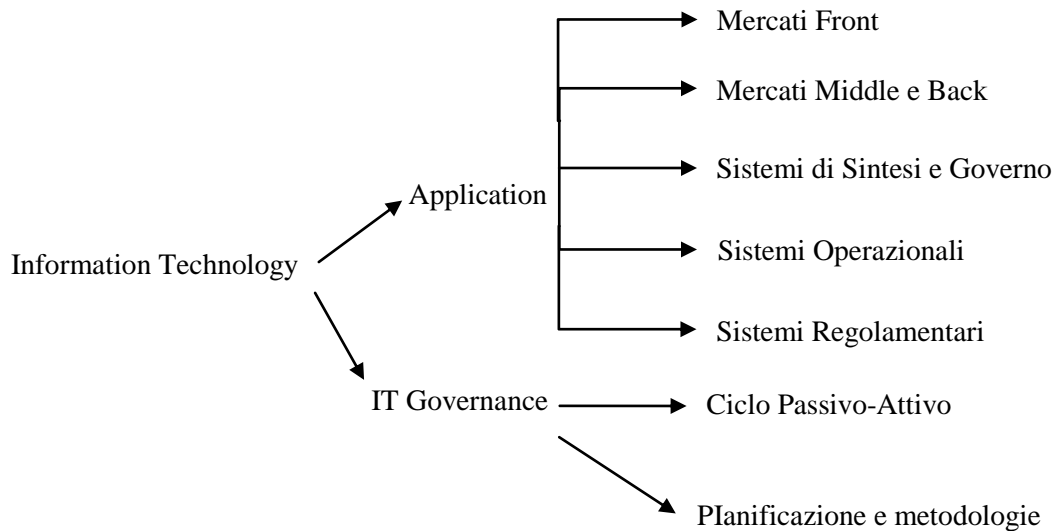
	Process Name	Description	Start Options and Conditions	Action Steps
15	Send Mail Add Resource to Team	Resource will be informed by email if he is allocated to a team.	No primary object is selected, this will be an On Demand process	Send mail to add res to team
16	Set OBS Assignment in Request V.2	This process inserts OBS of a request if it is changed (for requests with status "Approved")	No primary object is selected, this will be an On Demand process	Set OBS Assignment in Request V.2
17	ST - Notifica per request suspended	Send an email to the Operation group when a request is suspended	Auto-start , Start event: Update, Start Condition: ( ( ( ( ( Idea Type = 'Richiesta Manutentiva Lunga' ) or ( Idea Type = 'Richieste Evolutive' ) ) or ( Idea Type = 'Richieste Strategiche' ) ) and ( Idea Status = 'Suspended' )	Notifica suspended
18	MB Open_closed TP	The Process closes time periods and open time periods of timesheets from 4 periods (weeks) before to the 2 periods in the future	No primary object is selected, this will be an On Demand process	Open closed time period
19	MB Check Baseline	This process sends an email to PM if the baseline is not saved.	No primary object is selected, this will be an On Demand process	Check Baseline
20	CP Somma ETC/ACT per RFC INSERT	Workflow used to copy the ETC / ACT associated with the role (Update) within the attributes presented on the RFC	Auto-Start, Start event: Update, start condition: ( ( Team CP Effort Preventivo per Ruolo != Team CP Effort Preventivo per Ruolo [Previous Value] ) or ( Team CP Effort Consuntivo per Ruolo != Team CP Effort Consuntivo per Ruolo [Previous Value] ) )	Somma ETC/ACT per RFC

**Table 8: List of automated processes in Clarity with a brief description about them**

### 5-3 Sample Process 1: “Notify Manager Missing Timesheet”

The process name is “Notify Manager Missing Timesheet”. The reason to create this process was that all the resources in Mediobanca have to compile their timesheet which means they should write the total hours they spent on each task they are assigned to. They can find these tasks in their timesheet page in Clarity.

This process finds all the resources in “Information Technology” OBS<sup>36</sup> unit who did not compile their timesheet in specific time period. Then it sends a notification email to those resources and also to their managers reminding them to complete their timesheet. OBS units’ hierarchy in IT (Information Technology) is shown in figure below:



**Figure 8: Hierarchical structure of IT department in Mediobanca**

Each OBS unit has its own manager (only IT Governance has the same manager for both units). So, since the process must send email to the resources and their own manager, in total the query must retrieve list of the resources who did not compile their timesheet and 6 managers related to each part depending on the OBS unit in which resources work.

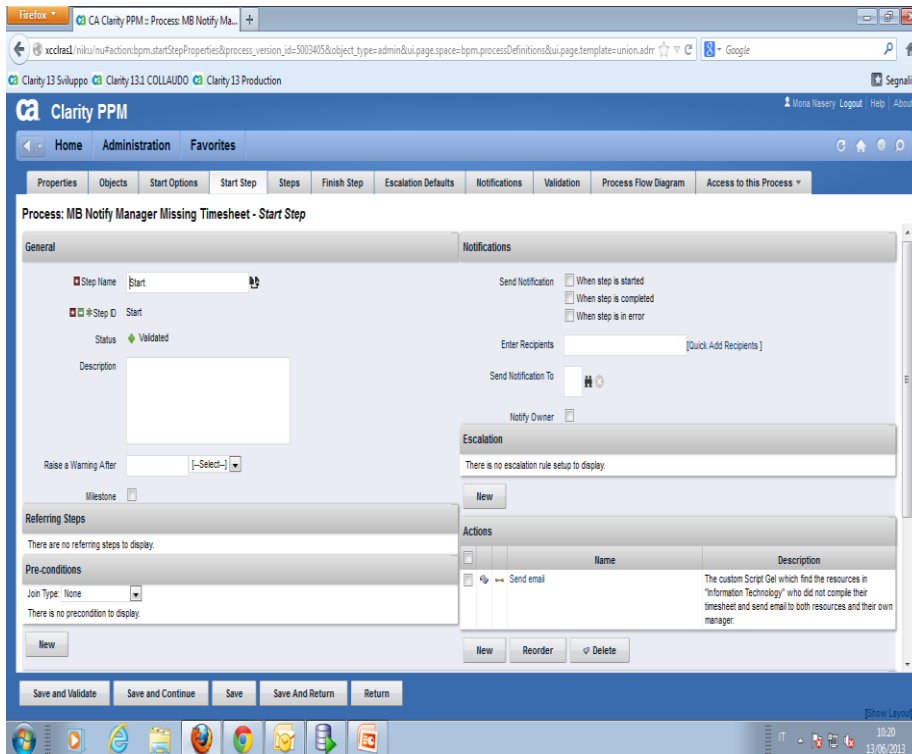
<sup>36</sup> Organizational Breakdown Structure (OBS) is a hierarchical representation of the company structure

Finally, in order to execute the process in specific time, for example every Friday at 16:00, the job<sup>37</sup> must run. In this way, if some resources forget to compile their timesheet they receive a notification email every week and therefore the maximum delay to compile timesheets would be 1 week.

In Action portlet, it is possible to define how we would like to create the process. In general, there are five action types:

1. **Manual Action:** to send an action item to a resource, group, role, etc.
2. **System Action:** to perform an operation on Clarity object
3. **Run Job:** to run a Clarity job
4. **Custom Script:** execute a custom script (Through Gel Scripting)
5. **Sub process:** invoke another process within the context of the current process

The following screenshot shows the start step of this process. If the process would have any preconditions must be defined in the portlet “Pre-condition” in this page. In action portlet by clicking “New”, the list of action step types will be shown. For instance, in this example custom script was selected.



**Figure 9: Start step page of Process “Notify Manager Missing timesheet”**

<sup>37</sup> CA Clarity PPM includes a list of stock jobs. With programming experience, it is possible to create new jobs, such as writing queries, statements, procedures, and Java classes.



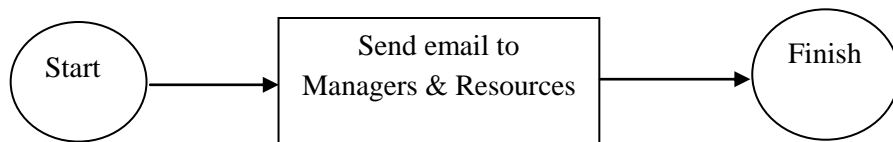
Table below include the summary of information related to this process.

Process Name	MB Notify Manager Missing Timesheet
Process Description	This Process sends an email to both resources who did not compile their timesheet and their Managers.
Object	No Primary object is selected
Start Options and Conditions	This is an On Demand process (it will not run automatically)
Steps	Start, Send email, Finish
KPI before running the process	Average delay: 2 weeks Maximum delay: 4 weeks
KPI after running the process	Average delay: 3 days Maximum delay: 1 week

**Table 9: Summary of the sample process 1: “Notify Manager Missing Timesheet”**

### 5-2-1 Process Flow Diagram

This process has only one step action, so it has a very simple flow diagram.



**Figure 10: Flow diagram of process “MB Notify Manager Missing Timesheet”**

### 5-2-2 Process Gel Script

This section contains the gel script of the first sample process. Gel Script structure is describe in Chapter 3 to better understand the script of the sample. As described earlier, Gel is a scripting language which Clarity uses to enable custom logic to solve business problems.

Here is the script of the process “Notify Manager Missing Timesheet” which is broke down in some parts which are explained separately to make the process actions more clear:

The first part is the header of all GEL scripts which is described in chapter 3.

```
<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql"
  xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<core:invokeStatic var="config" className="com.niku.union.config.ConfigurationManager"
method="getInstance"/>
```

First, the Sql statement is run to retrieve the name of managers of various OBS units in “IT department”. As described earlier, there are 6 managers in IT department that should be retrieved. The result is stored in the variable “result3”.

```
<gel:setDataSource dbId="Niku" var="clarityDS"/>

<core:set var="EntryURL"
value="\${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}/>

<sql:query dataSource="\${clarityDS}" var="result3">
  select distinct SR.ID, SR.LAST_NAME, SR.FIRST_NAME, CMN.EMAIL_ADDRESS, CMN.ID
  from SRM_RESOURCES SR, CMN_SEC_USERS CMN
  where SR.USER_ID = CMN.ID
  and CMN.ID in ('5002073', '5001022', '5001009', '5008076', '5001002', '5000000')
  and SR.is_active = '1'
</sql:query>
```

Then, for each item in result3 (that are managers), the required information (such as id, name, last name, etc) must be set by the “core:set” tag for later uses. This tag is used to set basic variables which do not need to be extracted from an XML document. It is possible to refer to the variable by using the \${variablename} syntax. The tag “gel:log” is used for what should be shown in the log, e.g. Since, we would like to see the managers’ last names in the log, mng[1] which is the value of Manager Last Name is written in the “gel:log” tag.

```
<core:forEach items="\${result3.rowsByIndex}" trim="true" var="mng">
  <core:set value="\${mng[0]}" var="userid"/>
  <core:set value="\${mng[1]}" var="ManagerLastName"/>
  <core:set value="\${mng[2]}" var="ManagerFirstName"/>
  <core:set value="\${mng[3]}" var="ManagerEmail"/>
  <core:set value="\${mng[4]}" var="idManager"/>

  <gel:log> manager ${mng[1]} </gel:log>
```

The tag “gel:email” sends a notification email to managers to inform them which resources did not compile their timesheets. “\${mng[3]}” shows the manager email which was set in the previous part.

*Note: the reason why we opened the tag “gel:email here before extracting the resources who did not compile their timesheet is to let managers see the list of these resources in their email.*

```
<gel:email from="clarity@mediobanca.it"
           subject="MISSING TIMESHEET}"
           to="${mng[3]}">
Si segnala che questi risorse non hanno compilato il timesheet:
```

Next, we have to extract all the resources in “IT department” that have to compile their timesheet. To do this, again we need to run Sql query to find those resources. The result is stored in a variable “result”. The last two lines guarantee that resource is “active” and the timesheet for him/her is open.

```
<!-- Estraggo gli utenti che devono compilare un timesheet -->
<sql:query dataSource="${clarityDS}" var="result">
select distinct SR.ID, SR.LAST_NAME, SR.FIRST_NAME, SR.EMAIL,SR.MANAGER_ID,CMN.EMAIL_ADDRESS,
CMN.ID
from SRM_RESOURCES SR, CMN_SEC_USERS CMN, PRJ_RESOURCES PRJ
where SR.USER_ID = CMN.ID
and SR.ID = PRJ.PRID
and SR.MANAGER_ID in ('5002073', '5001022', '5001009', '5008076', '5001002', '5000000')
and SR.MANAGER_ID = ${idManager}
and SR.is_active = '1'
and PRJ.PRISOPEN = '1'
</sql:query>
```

In order to find the resources who did not compile their timesheet, we need a loop so that for all the resources extracted from previous part that are stored in variable “result”, find the ones that did not complete their timesheets in a specific time period (for the past two months). This will be done through the following Sql statement, the result is stored in the variable “result2”:

```
<core:forEach items="${result.rowsByIndex}" trim="true" var="row">
<core:set value="${row[0]}" var="idUser"/>
<sql:query dataSource="${clarityDS}" var="result2">
SELECT SR.ID AS RES_ID,
SR.FULL_NAME,
SR.EMAIL,
TO_CHAR(TP.PRSTART, 'DD-MON-YYYY'),
TP.PRID,
SR.MANAGER_ID,
CMN.EMAIL_ADDRESS
FROM
PRJ_RESOURCES PR, SRM_RESOURCES SR, CMN_SEC_USERS CMN,
PRTIMEPERIOD TP,
(SELECT DISTINCT PR.PRID RESOURCE_ID, TP.PRID PRTIMEPERIODID
FROM PRTIMEPERIOD TP, PRJ_RESOURCES PR
WHERE PR.PRTRACKMODE != 0
AND TP.PRSTART between sysdate - 60 and trunc(sysdate,'D')-1
AND TP.PRISOPEN != 0) RESTP
LEFT OUTER JOIN PRTIMESHEET TS ON
```

```

        (TS.PRRESOURCEID = RESTP.RESOURCE_ID AND RESTP.PRTIMEPERIODID =
TS.PRTIMEPERIODID)
LEFT OUTER JOIN PRTIMEENTRY TE ON TE.PRTIMESHEETID = TS.PRID
LEFT OUTER JOIN PRTIMESHEET ADJ_TS ON ADJ_TS.PRADJUSTEDID=TS.PRID
        WHERE SR.ID = ${idUser}
        AND SR.ID = RESTP.RESOURCE_ID
        AND SR.USER_ID = CMN.ID
        AND TP.PRID = RESTP.PRTIMEPERIODID
        AND PR.PRID = RESTP.RESOURCE_ID
        AND (PR.PRISOPEN != 0 OR TS.PRID IS
NOT NULL)
        AND SR.IS_ACTIVE = 1 AND ( (TS.PRSTATUS IN (0,2) OR
TS.PRID IS NULL)) AND SR.RESOURCE_TYPE=0
        GROUP BY SR.ID,
                SR.FULL_NAME,
                SR.EMAIL,
                TP.PRSTART,
                TP.PRID,
                SR.MANAGER_ID,
                CMN.EMAIL_ADDRESS
        ORDER BY TP.PRSTART asc

</sql:query>

<![CDATA[

<br> </br>

]]>

```

**Note:** the scripts in pink color, are used to make a line break. The reason is that we would like the manager to see the list of resources in separate lines in his/her email.

Finally, for each item is “result2” (that are resources who did not compile their timesheets), we would like the name of those resources and the time period when they have to complete their timesheet to be shown in the log. As can be seen, “\${tms[1]}” and “\${tms [3]}” are written in “gel:log” which show the resource full name and the time period respectively. It can be seen that these variables are set before (in the tag “core:set”). Moreover, there is a message in the email which contains the link to access timesheet in Clarity.

**Note:** the tag “gel:email” is closed in this part after the message link. The reason not to close it before is that we wanted the results of all previous parts to be shown in the email to ease the approval procedure for managers. (e.g. list of resources and time periods)

```

<core:forEach items="${result2.rowsByIndex}" trim="true" var="tms">
    <core:set value="${tms[0]}" var="id_risorse"/>
    <core:set value="${tms[1]}" var="Full Name"/>
    <core:set value="${tms[2]}" var="Email"/>
    <core:set value="${tms[3]}" var="Time_Period"/>
    <core:set value="${tms[4]}" var="Prid"/>
    <core:set value="${tms[5]}" var="MANAGER_ID"/>

    <gel:log> Risorsa: ${tms[1]} periodo: "${tms[3]}"</gel:log>

<![CDATA[
<br> Risorsa: ${tms[1]},    periodo: ${tms[3]} </br>
]]>

```

```

</core:forEach>
  </core:forEach>
  <![CDATA[
<br> </br>
]]>

<![CDATA[
<br> To access the timesheet in Clarity: </br>
]]>

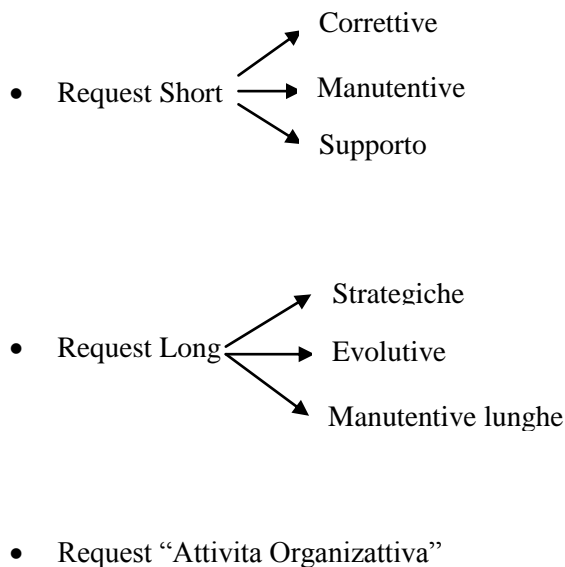
${EntryURL}/niku/nu#action:timeadmin.timesheetBrowserReturn

  </gel:email>
  </core:forEach>
</gel:script>

```

### 5-3. Sample Process 2: “MB Workflow Request”

The second sample automated process is “MB Workflow Request” which shows the workflow of requests in Mediobanca. This process starts when a new request (Idea) is created. There are 3 types of requests in Mediobanca.



**Request short:** only resources in “Information Technology” have rights to create this type of request which is approved automatically.

**Request long:** resources in “Organization unit” have rights to create this type of request which has to be approved by the head of the Organization and be assigned to IT employees. (Because this type of request is related to IT)

**Request Attivita Organizattiva:** resources in “Organization unit” can also create this type of request which has to be approved by the head of organization. Clarity assigns the request to its creator automatically. Every week, resources should compile their timesheet related to the activity. Head of the organization approve the timesheets. If he approves the timesheet, data will be elaborated and sent, but if he rejects it, the timesheet is returned to ask the resource to revise his timesheet. When the activity is finished it will be closed by each resource. The table below shows the summary of information related to this process:

Process Name	MB Workflow Request
Process Description	This process shows the workflow of various types of requests in Mediobanca.
Object	thisIdea
Start Options and Conditions	Auto-Start, Start event: Create
Steps	Approval, Request for approval, Notify approval Manager Admin Creator and customer, Set imitated by process, Add team, Verify if manager is changed, Notify reject administrator, Send notification Boss Organization
KPI before running the process	There are a lot of steps such as request approval, notify Manager, or insert the team which without this process it takes a long time to perform them manually.
KPI after running the process	Based on the status of the request, the process automatically takes care of approval of requests and waits for this outcome or notifies someone an action to take.

**Table 10: Summary of the sample process 2: “MB workflow request”**

### 5-3-1 Process Steps

1. **Start:** The process start is a decision point which means it has some preconditions as stated in table below:

If...	Then Go to
-------	------------

((Idea.thisIdea type = 'Richiesta Correttiva') or (Idea.thisIdea type = 'Richiesta Supporto')) or (Idea.thisIdea type = 'Richiesta Manutentiva'))	Add Team
((Idea.thisIdea type = 'Richiesta Evolutive') or (Idea.thisIdea type = 'Richiesta Strategiche')) or (Idea.thisIdea type = 'Richiesta Manutentiva Lunghe'))	Set Initiated by process
(Idea.thisIdea type = 'Attivita Organizzativa')	Send Notification to Organization Boss

**Table 11: Preconditions of the start step of the process “MB Workflow Request”**

It means that if the request is type “Short” (Correttiva, Manutentive, Supporto) then go to step “Add team”, if the request is type “Long” (Strategiche, Evolutive, Manutentiva lunga) then go to step “ Set initiated by process” and if the request is type “Attivita Organizzativa” then go to step “Send notification to the organization boss”.

2. **Send notification to the organization boss:** locks the field (System action) and send a notification email to the manager of an organization unit (Manual action). Figure below shows the action portlet of the request. It can be seen how system actions and manual actions are represented in Clarity action portlet.

Actions		
	Name	Description
	locked fields	locked i campi non utilizzati dall'ufficio Organ
	Send Notification Responsabieli Org	Invia una notifica al manager dell'organizzazione

**Figure 11: Action portlet of the step action “Send notification to the organization”**

3. **Add Team:** executes a custom script by Gel which inserts resources in team and locks the field “type” in the request page. The graphical representation of system action and custom script can be seen in the figure below:

Actions		
	Name	Description
	Aggiuge Team	- Aggiorna Team
	locked type	locked the type 's request

**Figure 12: Action portlet of the step action “Add Team”**

4. **Set Initiated by process:** runs a SQL statement (action type: Custom script)

5. **Request for Approval:** Send a request for approval and wait for the answer (action type: Manual action). Then there are some post conditions as below:

If...	Then Go To
(Idea.thisIdea Status Idea MB = 'Rejected')	Notify reject administrator
((Idea.thisIdea Status Idea MB = 'Approved') and ((Idea.thisIdea Assignment != 'Trabucchi, Lorenzo') or (Idea.thisIdea Manager != 'Campanini, Alessandro'))))	Notify approval Manager, Admin, creator and Customer

**Table 12: Post-conditions of the step “Request for Approval”**

It means if the request is rejected then goes to the step “Notify reject administrator” which sends a notification to creator and admin to inform them that the request has been rejected and if the request is approved then go to the step “Send notification to manager, admin and creator” in order to inform them that the request has been approved. Then go to the step “Verify when manager is changed” in order to change the request manager and assign the correct access right.

Finally, it goes to the “Finish” step which close the web service call. All of the above steps are shown in the following figure which is the flow diagram of this process.



### 5-3-2 Process Flow Diagram

The figure below shows the flow diagram related to various types of requests in Mediobanca. (For type “Attivita Organizzativa”, the manager of activity and the person who insert the request are the same, so it is possible to have a workflow like request type “Short” which is approved automatically)

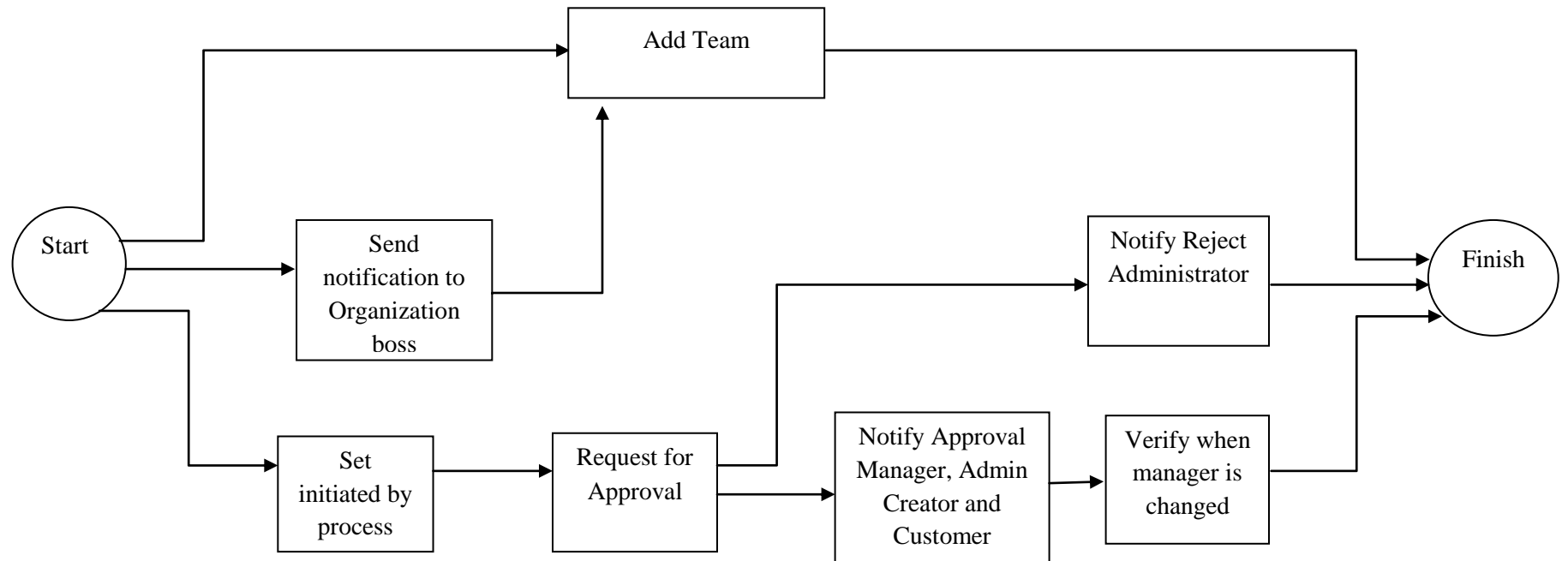


Figure 13 : Flow diagram of process “MB Workflow Request”

### 5-3-3 Process Gel Script

Since in this process more than one step has custom script and the scripts are quite long, this section provides just a very short and simple one which is related to the step “Set initiated by” of this process. As mentioned before, if the request is type “long” then this process will run. It can be seen the header part is quite the same in all scripts. The main part of this script is the <sql:update> statement which set the field “initiated-by” equal to one.

```
<gel:script      xmlns:core="jelly:core"      xmlns:f="jelly:com.niku.union.gel.FileTagLibrary"
xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"      xmlns:obj="http://www.niku.com/xog/Object"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sql="jelly:sql"      xmlns:u="jelly:util"      xmlns:util="jelly:util"      xmlns:x="jelly:xml"
xmlns:xog="http://www.niku.com/xog"      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <gel:parameter default="admin" var="XOGUsername"/>
  <gel:parameter default="mediobanca" secure="true" var="XOGPassword"/>
  <gel:parameter default="http://localhost" var="XOGURL"/>
  <gel:setDataSource dbId="Niku"/>
  <core:set value="{gel_objectInstanceId}" var="IdeaID"/>
  <core:set value="{gel_processInstanceId}" var="InstanceProccesID"/>

  <gel:log>idea ...${IdeaID}</gel:log>
  <gel:log>InstanceProccesID ...${InstanceProccesID}</gel:log>

  <sql:update>
    update bpm_run_processes
    set INITIATED_BY = 1
    where id = ${InstanceProccesID}
  </sql:update>
</gel:script>
```

## 5-4 Impacts of automating processes on performance of the work

The target of this section is to investigate the impacts of automating processes in Clarity on the work activity and performance of the Business Process Management System.

### 5-4-1 KPI Analysis of automated processes in Clarity

Table 11 shows the information about Key Performance Indicator (KPI) analysis of processes which implemented during this study. By KPI for processes we mean any indicator which could cause some improvements in the work activity. So, in this study KPI can be time, cost, number of resources involved in the

activity, work precision, etc. If automating a process lead to improve any of these indicators (for example by saving time or reducing manual errors), it means that the automation was successful and improved performance of the work activities.

As can be seen, this table could be a proof for benefits of automating processes in Clarity.

	Process Name	KPI before analysis	KPI after analysis
1	MB Check date	Possibility to have the wrong data. (e.g. in one week the number of mistakes could be 10/15 )	All dates will be correct, error rate will be zero.
2	CP Aggiorna stato	The workflow identifies the various changes in status of the Demand Management Client. The user, due to the change status of the request had to report any actions to be taken to that status through sending emails manually.	The process has the advantage of communicating in real time, to the various actors present in the Demand Management process of client, the change in status of Request and it automatically communicates any actions to be taken in that status.(( Save time , since Users had to change the status manually (e.g. from Approved to WIP and send email for each change that he has done))
3	CP Creazione RFC v2.0	Some attributes had to be populated manually by the user. (There are many steps which doing them manually could takes a long time, with possibility of wrong data insertion)	The attributes are inserted automatically through the process. (Save time, effort of users and minimize the errors by automatically doing these actions through Gel Scripting process)
4	CP Insert Personale Area Informatica	The user would have to enter, according to the IT Area of expertise of the request, forward the activities' of the latter to the respective Department Manager	The process assigns, automatically, the Request, in a well defined state of the same, the latter to the Department Manager of IT area of expertise. (Save time, effort of users
5	CP Insert Resource in Group	Users, not the Administrator application, cannot enter a resource in the respective group	Through the selection of the Group and Resource (read Only), the insert of the latter in the group is assigned to the process, in this way, it is not necessary to give users the right to edit but just to read about the resources and the various Groups. (Save time, effort of users)

	Process Name	KPI before analysis	KPI after analysis
6	CP Modifica Assignment	The user, in addition to changing the assignee, should combine to hand the Department of inquiry, in order to make it visible to all users associated with the Department in question.	The process is concerned, to automatically set the department's new assignment of the request, so user does not have to remember the Department of the new assignee. When manager is changed, this process modifies the OBS and set the new manager for the resource and also sends notification to inform resources that manager has been changed.
7	MB Inattiva Request old 15 days	The system was very slow to load the requests because of the huge number of the requests in all time periods.	Let the portal be more clear e.g. when filter by active show the reasonable number of active requests instead of a huge amount of it
8	MB - Notifica per request chiusa	The user should verify, on a daily basis, if the request within its competence was closed and then resolved.	The process sends automatically an email to the user who created the request soon as it is closed (resolved), and so the user is informed in real time.
9	MB Workflow Richiesta	There are a lot of steps such as request approval, notify Manager, or insert the team which without this process takes a long time to perform them manually.	The process based on the status of the request, automatically takes care of the approval request and waits for this outcome or notifies someone an action to take.
10	MB Notifica mail per inserimento Due Date	The user should connect to the application to check if the PM has inserted the due date (date of issue) of the Request.	The user is sighted, through an e-mail, automatically, when the PM insert the due dates. (save the time, make actions easier so: Delay: approximately zero, since email will be sent in less than a second and it is very fast)
11	Notifica mail per modifica Due Date	The user should connect to the application to see if the due dates that were previously entered by the PM is still valid. Delay: Depends on the user check intervals, so if users do not check application one day the delay could be more than 24 hours!	The user is informed through an e-mail, automatically, when the PM change the due date (the same as previous one)
12	MB Notify Manager Missing Timesheet	Average delay: 2 weeks Maximum delay: 4 weeks	Average delay: 3 days Maximum delay: 1 week
13	MB Processo Creazione Progetto	From the request it is possible to create a project, there are some attributes of the request and shall be provided on the project and this action was carried out manually.	When a request is converted to the Project, the common attributes are automatically reported by Request on the Project created. (again saving time since performing this action for all the request could be very time consuming)

	Process Name	KPI before analysis	KPI after analysis
,14	Ricalcolo Effort Consuntivato	Effort/ETC were calculated manually. (High probability of manual errors)	Starting from the project tasks, recalculates the effort / ETC, based on various weights present on tasks, and returns this value on the phase and on project. It helps the PM to give a complete overview on the project.
15	Send Mail Add Resource to Team	The resource only through the interface of the application could find out if it had been included in the team	The user is informed, in real time, when inserted into the inquiry team
16	Set OBS Assignment in Request V.2	OBS related to all requests had to be inserted manually which could be inserted mistakenly.	This process is a workaround, because sometimes the organizational structure is not inserted correctly or the user mistakenly deletes from the request. (This process help to avoid mistakes in inserting OBS and save time )
17	ST - Notifica per request suspended	The user should connect to the application to see if the Request is in the status "Suspended"	The user is sighted, through an e-mail, automatically, when the PM suspended the Request as further clarifications are needed for the release in production of the same. (Save time)
18	MB Open_closed TP	The user through the application page on the integration of the timesheet could also change the timesheet associated prior to the last month. This did not allow a clear view to the PM of the amount of hours spent in the project. (App interface was not clear with a huge amount of past timesheets)	The process, leaving open the timesheet the last four weeks allows the PM to carry out comprehensive reports on the hours spent on the project. (Make the Interface more clear and user friendly)
19	MB Check Baseline	PM might forget to save baseline.	The process has been implemented to remind the PM to create the project baseline.  A project baseline can be used to compare your initial project plans, or the state of a project at certain key points as it progresses, to the current state of a project
20	CP Somma ETC/ACT per RFC INSERT	Total ETC/ACT calculated manually with possibility of manual mistakes.	Increase precision by automatically calculating ETC/ACT.

**Table 13: Impacts of automating processes on performance of work activities**

#### 5-4-2 Qualitative analysis: Interviews contents

As mentioned in chapter 4, the results of this research study are extracted from both quantitative and qualitative research methodologies. This section provides information obtained from interviews with experienced professionals working in IT and more specifically on IT Governance, BPM and BPA to better understand the impacts of automating processes on the work activities and performance of the Business Process Management System. Some best practices to reduce these impacts and to maintain processes are also covered in this section.

Part of the dissertation was done through qualitative method where interviews with specialists in this field were conducted. Number of interviews that was done to achieve the thesis target is covered in this section. Many experts were conducted but only some of them had the required knowledge and time to answer the questions. The interviewers are managers, developers or Clarity consultants, with different levels of experiences in BPM. The main goal of interviews is to provide the opportunity to understand how process automation works in practice, and to express the issues that exist and finally to cover all the problems that haven't been discovered by literature review.

In the following, some of the main questions of this study are presented followed by the best answers that are collected from interview by specialists in this field:

***Question 1: What are the advantages of automating processes by GEL? (For instance compared with running jobs)***

- *There are some GEL advantages over creating Clarity jobs (java, sql, xbl ...) for the same tasks. GEL scripts allow for faster processes completion as the Clarity process engine just executes the processes steps through without stopping. If you implement the same thing with a job then the processes engine will stop after reaching and scheduling the job and will wait for it to be completed. That will take some time. Assuming the system has many processes instances, faster processes completion has an impact on background process performance because there are less process step transitions that need to be checked. GEL scripts can also run parallel in two process instance while job usually are incompatible with themselves, meaning there is only one instance of the job running at one point in time. This job incompatibility can be good if you need some cheap concurrency control but this can slow the process instances further. If you have one processes that gets a lot of instances in a short time (like the time approval) then a job will slow down your bg while a GEL will allow for faster processes completion times and less load on the bg. There are other tradeoffs with using GEL script, but not related to performance like complexity and code maintenance. (Sergiu Gavrila, Software engineer, Clarity Technical Consultant)*

- *The advantage is being able to do many things which cannot be done the options in process design in the GUI and in some cases things that otherwise would require unsupported customization of the database. Use Gel if you cannot do it simply in any other manner. (Martti Kinnunen, Clarity Consultant)*
- *Gel script is readable un-encoded script that can be understood by every Clarity administrator that comes to support Clarity in the organization. This stands against usually some compiled code that is harder to understand. With gel script you already have some built in constants and connections that speed up the integration on the Clarity side. Gel scripts processes can be easily moved from one environment to another and its parameters/code quickly changed to support the new environment. Gel scripts processes can be scheduled from within the clarity application and so all your integrations can be maintained from a central location. (Fefer Amir, Consultant at CA Technologies)*

**Question 2: Does automating processes by GEL in Clarity PPM have any impacts on performance of the Business Process Management System? If yes; how?**

- *Just at general level the performance depends on what sort of activities it does, how much database activity does it create. If it is properly designed to do something meaningful should not affect performance more than time slice roll over, datamart extraction and such like do. (Martti Kinnunen, Clarity Consultant)*

*Every process you run creates a small amount of load on the process engine for Clarity application. However, most of the times, the impact would be negligible. Normally, the process would reduce the manual effort of the users. (Hemant Joshi, Senior CA Clarity/Business Consultant for Deutsche Bank, at HCL Technologies)*

- *Everything you do that puts a load on the system theoretically affects performance. I think what you need to analyze is what is the impact of each GEL script you are writing. Something basic like a notification from a query should be low impact to the BG, but there are best practices that should be adhered to when generating GEL scripts to avoid having performance problems. However it really boils down to the number of transactions that are performed by the script. The more transactions, the longer it takes, the more memory it needs, therefore the more impact it will have. (Federico Pena, Clarity Integration Specialist)*
- *One way in which GEL scripts do affect application performance is that the responsibility for making sure they perform adequately switches from CA to you, the client (or whoever wrote the script for you). The developer should therefore prepare a proper test plan that account for both the hardware capabilities of your production platform, the likely level of traffic the script might generate and perhaps the amount of parallel competing activity that might be going on. I would also recommend the approach of keeping GEL scripts as modularized and simple as possible. (Philip Smythe, Clarity technical specialist)*

**Question 3: Are there any ways or best practices to avoid performance issues and to maintain processes in Clarity?**

- *Just as a Best Practice, you would want to regularly cleanup the Old Completed/Failed process instances, unless you might need them for Audit purpose. Also, if any process is failing regularly, kindly check and rectifies the process steps, since all the processes in failed state un-necessary creates an additional load over the process engine, whenever the background services are re-started. After all, processes only help us to automate the redundant and repetitive steps in Clarity. (Hemant Joshi, Senior CA Clarity/Business Consultant for Deutsche Bank, at HCL Technologies)*
  
- *The following Best Practices might help you on your way to have well performing processes:*
  1. *Try to keep start condition, pre-condition and post condition expressions as simple as possible.*
  2. *For a process with an on-update start condition, ensure that the actions the process takes result in the start condition evaluating to FALSE by the time the process finishes executing. This will prevent the process from firing over and over again.*
  3. *The number of active processes tied to an object definition does have an effect on performance, due to the number of start condition evaluations that must be conducted upon UPDATE or CREATE. Keep the total number of active processes for a given object to a minimum. For instance, have a single "create" process with appropriate branching logic instead of having multiple "create" processes.*
  4. *Identify processes that do not need to be kept around and schedule the Delete Process Instance job to delete them when they have completed (nightly or weekly basis). Frequently run processes are good candidates to delete as they typically carry almost no auditing benefit.*
  5. *Break up big processes into a series of smaller manageable sub processes. Processes are then easier to manage and perform better.*
  6. *Reduce the number of steps used by a process by performing more than one action within a step. This will create a more efficient and better performing process.*
  7. *Processes can span multiple objects. Instead of creating two or more separate processes, it might be more efficient to create one multi-object process.*
  8. *During BG services startup, the process engine loads, compiles, and caches all active processes. This can cause an initial spike in database CPU usage. (Ravindra Verma, Associate operations manager at Cognizant Technology Solutions)*
  
- *As long as you make sure that the following are covered, it should be fine:*
  1. *Excessive logging in GEL scripts, particularly logging within a loop, is worth avoiding.*
  2. *Use of tags that can cause stalling in Clarity (util:sleep). The <util:sleep> tag is designed to put the thread to sleep and should be avoided.*
  3. *One of the most commonly missed SQL issues seen in GEL scripts is the lack of SQL bind parameters. Not using SQL bind parameters can have a couple of very bad unintended results: Security: Lack of bind parameters allows for the possibility of SQL injection attacks.; Stability: When a GEL variable is directly interpolated in a SQL statement, even the presence of an apostrophe in the variable value is all that is necessary to cause the SQL statement to fail to parse. This causes the GEL script to throw a SQL Exception and stop. (Navdeep Joshi, Application developer at CA Technologies)*



- *In the development we try to estimate some metrics (or KPI for processes) which in our opinion have an impact on global processes performance or customer acceptance, like:*
  1. *Average counts of new processes instances per week.*
  2. *Peak count of new processes instances in a day. A time approval process could generate a lot of new instances in a short period at the end of the time reporting period.*
  3. *Average time it takes for a process to complete.*
  4. *Average count of extra notification for user type (like project manager, line manager, team member) because of the new process. A user should be able to process all the notifications (action items, emails) he gets from clarity.*

***Simple example:** a customer wants to have a project "implementation" process. The amount of new processes instances is small, about 10 per week (one for each new project). But the average time the process would take to complete was initially huge, one to two years. This could be big problem if you build new process versions, because you have instances which cannot be migrated to the new process version.*

***Solution:** break the one process in small one which reacts to changes in project status. GEL is one technical way of implementing parts of processes, a good one (I even have a GEL tag for running XBL scripts) but not the only one. GEL scripts, like jobs, must not take too long to process (one hour GEL script will block one execution pipe for one hour, not well). Long lasting processing task can be better implemented as (java) jobs. (Sergiu Gavrilă, Software engineer, Clarity Technical Consultant)*

## Chapter 6

### Conclusion and Future works

This chapter concludes the study based on the methodological approach, implementation experience and empirical observations that are described in previous chapters. Limitations of the dissertation and ethical issues are also provided in this chapter. At the end of this chapter, future evaluation of the thesis work is described.

## 6-1 Targeted Problem

The main contribution of this dissertation was focused on implementing process automation in Clarity PPM, impacts of automating them on performance of the system and work activities and introducing some best practices to maintain processes and decrease those impacts.

This research study had some objectives, each one tackled in the literature review or by interviews during research. The effort was to address all the thesis questions and defined targets in an appropriate and comprehensive way. The first chapter was an introduction to the topic. Contribution of the dissertation and the main structure of the thesis are described in this chapter. Frame of references which defines the most fundamental concepts related to topic such as PPM, BPM, BPA, workflows and etc., is presented in second chapter. They are necessary to understand the methodology and implementation experience. Chapter 3 provides the main background and state of the art of the topic which reviews the most relevant existing works. The reason why Mediobanca chose Clarity as PPM software among the most famous PPM vendors, the most relevant process modeling standards followed by Clarity process flow diagrams, and finally an introduction to GEL as a custom script used in Clarity to automate processes are all presented in this chapter. In order to fulfill objectives of the thesis a methodological approach is presented in chapter 4. Finally, best answers to the most important questions of this thesis which are *“how to automate processes in Clarity by GEL”* and *“what are the impacts of automating processes on performance of the system or work activity”* and set of best practices to maintain processes in Clarity are addressed in chapter 5.

## 6-2 Approach Review

The methodological approach of the dissertation is described in Chapter 4. First, different types of research methodologies and data collection were compared and selected methods in this study were mentioned. Next, questions that this study must answer were presented followed by the proposed methodology for this thesis.

As described earlier in methodology, first suitable processes for automation are selected. Then process requirements are defined, KPI metrics are measured and a target is set. Then process is automated through number of action steps and finally performance indicators are measured again and the target is checked to decide if the process automation was useful or the manual process should be substituted.

The approach used various types of action steps to automate processes in Clarity such as System action, Custom script, etc. However the focus of the study was on automating processes through GEL (custom script). Using GEL resulted in performing actions that cannot be done through other types of step actions and also as **Martti Kinnunen, CA Consultant** said *“The advantage is being able to do many things which cannot be done the options in process design in the GUI in some cases things that otherwise would require unsupported customization of the database. Use Gel if you cannot do it simply in any other manner.”*

### 6-3 Analysis Result Conclusion

It can be observed from the Processes KPI analysis (referred to table 9 in chapter 5), that almost all of the studied processes that were automated in Clarity had some positive effects on our work activities. In general they improve performance by:

- Saving time
- Reducing the number of human resources involved in activities
- Increasing precision by decreasing the common mistakes that could exist if those tasks were done manually
- Increasing the level of users' satisfaction (Making users happier)

Based on many factors such as the complexity of the workflow diagram of a process, number of steps within the process and complexity of the GEL script that is used for a process, the time to execute the process could vary. However, for 20 processes which are automated for this study the difference in their execution time is really negligible. All of them are run in less than seconds while it is obvious that if they were done manually, it could takes much more time.

Another important advantage of automating processes is saving the effort of resources who are involved in related activities. As a very simple example, a workflow process which is described in chapter 5 has many steps and if the processes were not automated many resources that are involved in each steps had to put efforts and spend time to do the activities.

Moreover, process automation decreases the mistakes that are unavoidable by human calculations. For instance, the process that calculates the effort / ETC could provide more precise results which help PM to have a complete view on the projects.

Apart from the above advantages of automating processes in Clarity, they have also some impacts on performance of the application and work which are discussed in Chapter 5. In order to know what are those impacts and what are the best practices to maintain processes number of interviews was done with professionals and experts who work in this field to benefit from their valuable experiences in a practical environment. As mentioned before, all processes that are executed put a small amount of load on the system (Process engine in Clarity) but these impacts are usually negligible. Types and number of the database activities and number of performed transactions are the important factors that must be considered. If there are many transactions to be done, it will take more time and also uses more memory.

However, some best practices were described to decrease these impacts such as deleting the Old Completed/Failed process instances in a regular basis which reduce the additional load over process engine. In general, automating processes in Clarity reduce the manual efforts of resources and speed up actions and repetitive steps.

#### **6-4 Future Works**

Process automation area is dynamic and many challenges exists which need further research. Processes are changing continuously and according to their dynamic nature, it is not advisable to create a fixed model for processes. Therefore, business processes must be supported in rapidly changing environment. Process mining techniques help to monitor and improve business processes by extracting knowledge from event logs. It aims to create a “live process models” which are created on a daily basis and instead of “Static models” which end without continuous change.

Process mining technique includes the automated discovery of processes from event logs [28]. It creates a process model based on events that are shown in the log such as executed activities. Process mining technique automatically improves the process model based on the observed behavior in the event logs [29] . For instance they can help for process automation requirements elicitation. Till now process mining techniques construct a specific process model based on a given event log and there exists some techniques to merge a single process model to a configurable process model. Further research can be done to create a configurable process model which is based on the collection of real event logs. [30]

## Bibliography

- [1] T. H. Davenport, *Process Innovation: Reengineering Work Through Information Technology*, Boston: Harvard Business School Press, 1993.
- [2] J. Noel, "BPM and SOA: Better Together," *IBM Corporation*, 2006.
- [3] P. M. Antunes, "Resilient Business Process Management: Framework and services. Expert Systems with Applications," *Elsevier*, 2010.
- [4] S. Z. A. S. Linenberg Y, "Optimising organisational performance by managing project benefits," in *PMI Global Congress*, Europe, 2003.
- [5] R. A. D. J. Lycett M, "Programme management: a critical review," in *Int J Proj Manag*, 2004, pp. 22(4),289-99.
- [6] M. Weske, *Business process management*, Springer, 2012.
- [7] WFMC, *Workflow Management Coalition (WfMC) Terminology & Glossary*, 1999.
- [8] S. Lusk, *BPM Institute.org*, April 2007. [Online]. Available: <http://www.bpminstitute.org/articles/article/article/the--process--life--cycle--and--project--life--cycle-->. [Accessed 2013].
- [9] M. Hammer, "The Process Enterprise: An Executive Perspective," *Hammer and Company*, 2001.
- [1 T. Davenport, "The Coming Commoditization of Processes," June 2006.  
0]
- [1 a. s. Business process management (BPM) standards, "Ryan K.L. Ko, Stephen S.G. Lee, Eng Wah Lee,"  
1] *Emerals*, 2009.
- [1 W. V. d. A. A. T. H. M Dumas, *Process Aware Information Systems*, John Wiley, 2005.  
2]
- [1 I. John Wiley & Sons, *Workflow handbook 1997*, Workflow Management Coalition, New York, NY,  
3] USA, 1997.
- [1 B. D. Reyck and M. L. ., S. R. C. ., M. M. ., A. S. Yael Grushka-Cockayne, *The impact of project portfolio  
4] management on information technology projects*, ELSEVIER, 2005.

- [1 R. S. Aguilar-Saven, "Business Process Modeling: Review ad Framework," *ELSEVIER*, 2003.  
5]
- [1 U. R. Page, "UML: Object Management Group," [Online]. Available: [www.uml.org](http://www.uml.org). [Accessed 2013].  
6]
- [1 L. A. M. D. a. A. t. H. W.M.P. van der Aal, "Design and implementation of the YAWL system," *Springer*,  
7] 2004.
- [1 A. t. H. W.M.P. van der Aalst, "YAWL: Yet Another Workflow Language," *ELSEVIER*, 2005.  
8]
- [1 C. Technologies, "CA Clarity PPM for New Product Development," 2008. [Online]. Available:  
9] [http://www.itdesign.de/fileadmin/user\\_upload/Broschueren/PPM/CA\\_Clarify\\_for\\_NPD\\_Functional\\_Overview\\_Brochure.pdf](http://www.itdesign.de/fileadmin/user_upload/Broschueren/PPM/CA_Clarify_for_NPD_Functional_Overview_Brochure.pdf). [Accessed 2013].
- [2 C. Technologies, "CA Clarity administration guide," CA Technologies, 2012.  
0]
- [2 C. Technologies, "XML Open Gateway Developer Guide," 2012. [Online]. Available:  
1] [https://supportcontent.ca.com/cadocs/0/%20%20-Bookshelf\\_Files/PDF/CAClarityPPM\\_XOG\\_DeveloperGuide\\_ENU.pdf](https://supportcontent.ca.com/cadocs/0/%20%20-Bookshelf_Files/PDF/CAClarityPPM_XOG_DeveloperGuide_ENU.pdf). [Accessed 2013].
- [2 C. Technologies, "GEL Scripting," CA Technologies, 2013.  
2]
- [2 C. Technologies, "XML Open Gateway developer Guide," 2012. [Online]. Available:  
3] [https://supportcontent.ca.com/cadocs/0/%20%20-Bookshelf\\_Files/PDF/CAClarityPPM\\_XOG\\_DeveloperGuide\\_ENU.pdf](https://supportcontent.ca.com/cadocs/0/%20%20-Bookshelf_Files/PDF/CAClarityPPM_XOG_DeveloperGuide_ENU.pdf). [Accessed 2013].
- [2 T. D. Jick, "Mixing Qualitative and Quantitative Methods: Triangulation in Action," *JSTORE*, 1979.  
4]
- [2 N. K. & L. Y. S. Denzin, *The Sage Handbook of Qualitative Research*, Sage , 2005.  
5]
- [2 J. R. M. R. P. G. Marta Indulska, "Business Process Modeling: Current issues and future challenges,"  
6] Amsterdam, Netherlands, 2009.
- [2 C. Technologies, *Clarity System Administrator Guide*, 2012.  
7]
- [2 V. R. H. M. W. V. B. F. v. D. E. K. C. W. G. W. M. P. van der Aalst, "Process mining: a two-step  
8] approach to balance between underfitting and overfitting," *Springer*, 2008.

- [2] M. L. R. H. R. B. v. D. W. v. d. A. J.C.A.M. Buijs, "Improving Business Process Models using observed behavior," *International Federation for Information Processing*, 2013.
- [3] J. C. A. M. Buijs, B. F. v. Dongen and W. M. P. v. d. Aalst, "Mining Configurable Process Models from Collections of Event Logs," *Springer*, 2013.
- [3] S. Z. A. S. Linenberg Y, "Optimising organizational performance by managing project," in *PMI Global Congress*, Europe, 2003.
- [3] IBM, "IBM Process Engine," October 2011. [Online]. Available:  
 2] <http://pic.dhe.ibm.com/infocenter/p8docs/v5r1m0/index.jsp?topic=%2Fcom.ibm.p8.sysoverview.doc%2Fp8sov109.htm>. [Accessed 2013].
- [3] C. Technologies, "CA Clarity Tuesday Tips, Process Maintenance," CA, 2012. [Online]. Available:  
 3] [https://communities.ca.com/web/ca-clarity-global-user-community/message-board/-/message\\_boards/view\\_message/99009629;jsessionid=83B07F97AEF77D67EFF440126EC3FBEA.usilap723?\\_19\\_threadView=tree&](https://communities.ca.com/web/ca-clarity-global-user-community/message-board/-/message_boards/view_message/99009629;jsessionid=83B07F97AEF77D67EFF440126EC3FBEA.usilap723?_19_threadView=tree&). [Accessed 2013].
- [3] S.-L. Ja'msa"-Jounela, "Future trends in process automation," *ELSEVIER*, 2007.  
 4]
- [3] C. Technologies, "CA Clarity PPM On Demand Technical Overview," [Online]. Available:  
 5] [http://www.ca.com/us/~/\\_media/files/technologybriefs/ca\\_clarity\\_od\\_technical\\_overview\\_brief\\_213501.aspx](http://www.ca.com/us/~/_media/files/technologybriefs/ca_clarity_od_technical_overview_brief_213501.aspx). [Accessed 2013].

# Appendix

## Process 1) MB Check Date

```
<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>
  <sql:query var="checkRequest">
select inv.schedule_start, inv.schedule_finish, inv.code
  from inv_investments inv
  join odf_ca_idea idea on inv.id= idea.id
  where inv.odf_object_code='idea'
  and (inv.schedule_start = to_date ('01/01/1900' , 'DD/MM/YYYY')
  or inv.schedule_finish = to_date ('03/06/2079' , 'DD/MM/YYYY'))
  and idea.mb_status_idea = 7
  and idea.partition_code != 'CP'
</sql:query>
  <core:choose>
    <core:when test="${checkRequest.rowCount == 0}"/>
    <core:otherwise>
      <core:forEach items="${checkRequest.rowsByIndex}" var="Request">
        <core:set value="${Request[0]}" var="schedule_start"/>
        <core:set value="${Request[1]}" var="schedule_finish"/>
        <core:set value="${Request[2]}" var="code"/>
        <gel:log>Start: ${schedule_start}</gel:log>
        <gel:log>Finish: ${schedule_finish}</gel:log>
        <gel:log>Code: ${code}</gel:log>
        <!-- setta la date di start del 1900-->
        <sql:update var="UpdateScheduleStart">
update inv_investments
  set schedule_start = created_date
  where id in (select inv.id
  from inv_investments inv
  join odf_ca_idea idea on inv.id= idea.id
  where inv.odf_object_code='idea'
  and inv.schedule_start = to_date ('01/01/1900' , 'DD/MM/YYYY')
  and idea.mb_status_idea = 7
  and idea.partition_code != 'CP')
</sql:update>
        <!-- setto la date di fine del 2079-->
        <sql:update var="UpdateFinishStart">
update inv_investments
  set schedule_finish = last_updated_date
where id in (select inv.id
  from inv_investments inv
  join odf_ca_idea idea on inv.id= idea.id
  where inv.odf_object_code='idea'
```



```

        and inv.schedule_finish = to_date ('03/06/2079' , 'DD/MM/YYYY')
        and idea.mb_status_idea = 7
        and idea.partition_code != 'CP')
    </sql:update>
</core:forEach>
</core:otherwise>
</core:choose>
</gel:script>

```

## Process 2) CP Aggiorna Stato, Aggiorna date

```

<gel:script xmlns:core="jelly:core"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:obj="http://www.niku.com/xog/Object"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:util="jelly:util" xmlns:x="jelly:xml"
  xmlns:xog="http://www.niku.com/xog" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <core:set value="{gel_objectInstanceId}" var="idea_id"/>
  <gel:setDataSource dbId="Niku"/>
  <sql:query var="date_da_agg">
    SELECT a.Value ,replace (e.full_name,' ',''), To_Char(SYSDATE,'DD-MM-YYYY HH24:MI:SS') date_f,
    c.name
  FROM odf_multi_valued_lookups a, odf_ca_idea b , cmn_lookups_v c , odf_ca_cp_stati_compass_rfc d,
  srm_resources e
  WHERE b.id={idea_id}
  AND c.language_code='en'
  AND c.lookup_type ='CP_STATI_COMP_RFC'
  AND a.pk_id=d.id
  AND a.attribute='cp_l_date_da_agg'
  AND c.lookup_code=b.cp_s_nuovo_stato
  AND d.cp_l_new_stato=c.lookup_code
  AND e.user_id=b.last_updated_by
  </sql:query>

  <core:forEach items="{date_da_agg.rowsByIndex}" var="row">
    <core:set value="{row[0]}" var="attr"/>
    <core:set value="{row[1]}" var="upda"/>
    <core:set value="{row[2]}" var="date_full"/>
    <core:set value="{row[3]}" var="name_st"/>
  </core:forEach>
  <core:choose>
    <core:when test="{attr eq null}">
      <sql:update>
        update odf_ca_idea
        set cp_s_log_rfc= rtrim (cp_s_log_rfc) || chr(10) ||
        '{date_full}' || ' - ' || '(' || '{upda}' || ')' || ' - Stato RFC modificato in ' || '{name_st}'
        where id={idea_id}
      </sql:update>
    </core:when>
    <core:otherwise>
      <sql:update>
        update odf_ca_idea
        set {attr}=sysdate, cp_s_log_rfc= rtrim
        (cp_s_log_rfc) || chr(10) || '{date_full}' || ' - ' || '(' || '{upda}' || ')' || ' - Stato RFC
        modificato in ' || '{name_st}'
        where id={idea_id}
      </sql:update>
    </core:otherwise>
  </core:choose>
</gel:script>

```

## Process 2) Aggiorna Stato, Aggiorna stato

```

<gel:script xmlns:core="jelly:core"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:obj="http://www.niku.com/xog/Object"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"

```

```

xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sql="jelly:sql" xmlns:util="jelly:util" xmlns:x="jelly:xml"
xmlns:xog="http://www.niku.com/xog" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<core:set value="\${gel_objectInstanceId}" var="idea_id"/>
<gel:setDataSource dbId="Niku"/>
<sql:query var="date_da_agg">
SELECT a.Value ,replace (e.full_name,' ',''), To_Char(SYSDATE,'DD-MM-YYYY HH24:MI:SS') date_f,
c.name
FROM odf_multi_valued_lookups a, odf_ca_idea b , cmn_lookups_v c , odf_ca_cp_stati_compass_rfc d,
srm_resources e
WHERE b.id=\${idea_id}
AND c.language_code='en'
AND c.lookup_type ='CP_STATI_COMP_RFC'
AND a.pk_id=d.id
AND a.attribute='cp_l_date_da_agg'
AND c.lookup_code=b.cp_s_nuovo_stato
AND d.cp_l_new_stato=c.lookup_code
AND e.user_id=b.last_updated_by
</sql:query>
<core:forEach items="\${date_da_agg.rowsByIndex}" var="row">
<core:set value="\${row[0]}" var="attr"/>
<core:set value="\${row[1]}" var="upda"/>
<core:set value="\${row[2]}" var="date_full"/>
<core:set value="\${row[3]}" var="name_st"/>
</core:forEach>
<core:choose>
<core:when test="\${attr eq null}">
<sql:update>
update odf_ca_idea
set cp_s_log_rfc= rtrim (cp_s_log_rfc) || chr(10) ||
'\${date_full}' || ' - ' || '(' || '\${upda}' || ')' || ' - Stato RFC modificato in ' || '\${name_st}'
where id=\${idea_id}
</sql:update>
</core:when>
<core:otherwise>
<sql:update>
update odf_ca_idea
set \${attr}=sysdate, cp_s_log_rfc= rtrim (cp_s_log_rfc) ||
chr(10) || '\${date_full}' || ' - ' || '(' || '\${upda}' || ')' || ' - Stato RFC modificato in ' ||
'\${name_st}'
where id=\${idea_id}
</sql:update>
</core:otherwise>
</core:choose>
</gel:script>

```

### Process 5) CP insert resources in groups

```

<gel:script xmlns:core="jelly:core"
xmlns:f="jelly:com.niku.union.gel.FileTagLibrary"
xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
xmlns:obj="http://www.niku.com/xog/Object"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sql="jelly:sql" xmlns:u="jelly:util" xmlns:util="jelly:util"
xmlns:x="jelly:xml" xmlns:xog="http://www.niku.com/xog"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<gel:parameter default="admin" var="XOGUsername"/>
<gel:parameter default="mediobanca" secure="true" var="XOGPassword"/>
<gel:parameter default="http://localhost" var="XOGURL"/>

```

```

<!-- Valorizza variabile col Progressivo dell'Idea fornito dal Processo Commentare in test-->
<core:set value="{gel_objectInstanceId}" var="grup_res_id"/>
<!-- Solo per il test: progressivo in chiaro Commentare in Produzione
<core:set var="IdeaID" value="5000001"/> -->
<gel:log>ObjectID ...${grup_res_id}</gel:log>
<gel:setDataSource dbId="Niku"/>
<!-- <u:sleep millis ="3000"/> -->

    <soap:invoke endpoint="{XOGURL}/niku/xog" var="auth">
        <soap:message>
            <soapenv:Envelope
                xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xog="http://www.niku.com/xog">
                <soapenv:Header/>
                <soapenv:Body>
                    <xog:Login>
                        <xog:Username>${XOGUsername}</xog:Username>
                        <xog:Password>${XOGPassword}</xog:Password>
                    </xog:Login>
                </soapenv:Body>
            </soapenv:Envelope>
        </soap:message>
    </soap:invoke>

    <gel:set asString="true" select="$auth/soapenv:Envelope/soapenv:Body/xog:SessionID/text()"
var="sessionId"/>

    <sql:query var="select_res_gru">
        SELECT a.cp_l_gruppo, r.last_name,    r.FIRST_NAME ,
            r.EMAIL ,
            r.UNIQUE_NAME,
            nvl(r.EXTERNAL_ID, '') FROM ODF_CA_CP_INSERT_RES_IN_GR  a,    srm_resources r
WHERE a.cp_l_risorsa=r.id
AND a.id=${grup_res_id}
</sql:query>

    <gel:log>Record trovati: ${select_res_gru.rowCount}</gel:log>
    <core:choose>
        <core:when test="{select_res_gru.rowCount eq 1}">

    <core:forEach items="{select_res_gru.rowsByIndex}" var="res_gr_row">
        <core:set value="{res_gr_row[0]}" var="group_id"/>
        <core:set value="{res_gr_row[1]}" var="LastName"/>
        <core:set value="{res_gr_row[2]}" var="FirstName"/>
        <core:set value="{res_gr_row[3]}" var="Email"/>
        <core:set value="{res_gr_row[4]}" var="Name"/>
        <core:set value="{res_gr_row[6]}" var="ExternalId"/>

    </core:forEach>

    <gel:parse var="userupdate">
        <NikuDataBus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../xsd/nikuxog_user.xsd">
            <Header action="write" externalSource="NIKU" objectType="user" version="12.0.3.5164"/>
            <Users>
                <User externalId=" " userLanguage="Italian" userLocale="it"
                    userName="{Name}" userStatus="ACTIVE"
                    userTimezone="Europe/Amsterdam" userType="INTERNAL">
                    <PersonalInformation emailAddress="{Email}" firstName="{FirstName}"
lastName="{LastName}"/>
                    <Groups >
                        <Group id="{group_id}"> </Group>
                    </Groups>
                </User>
            </Users>
        </NikuDataBus>
    </gel:parse>

```

```

<soap:invoke endpoint="${XOGURL}/niku/xog" var="userResponse">
  <soap:message>
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xog="http://www.niku.com/xog">
      <soapenv:Header>
        <xog:Auth>
          <xog:SessionID>${sessionID}</xog:SessionID>
        </xog:Auth>
      </soapenv:Header>
      <soapenv:Body>
        <gel:include select="$userupdate"/>
      </soapenv:Body>
    </soapenv:Envelope>
  </soap:message>
</soap:invoke>

  <gel:set asString="true" var="loadStatus"
select="$userResponse/soapenv:Envelope/soapenv:Body/XOGOutput/Status/@state" />
  <gel:expr
select="$userResponse/soapenv:Envelope/soapenv:Body/XOGOutput/Statistics/@failureRecords/text()" />
<gel:log>
  <gel:expr select="$userResponse"/>
  </gel:log>
  <core:switch on="${loadStatus}">
    <core:case value="SUCCESS">
      <gel:log level="INFO"> Risorsa ${Name} inserita nel Gruppo </gel:log>
    </core:case>
    <core:case value="FAILURE">
      <gel:log level="INFO">Errore Inserimento Risorsa </gel:log>
    </core:case>
    <core:default>
      <gel:log level="INFO">Errore generico XOG. </gel:log>
    </core:default>
  </core:switch>
</core:when>
<core:otherwise>
<gel:log level="INFO">Risorsa Non Trovata </gel:log>
</core:otherwise>
</core:choose>
</gel:script>

```

### Process 7) MB inattiva request old 15 days

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:core="jelly:core"
xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:invokeStatic
  className="com.niku.union.config.ConfigurationManager"
  method="getInstance" var="config"/>
  <core:set
  value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>
  <sql:query var="checkRequest">
select i.id, i.schedule_finish

```

```

from inv_investments i
join odf_ca_idea o on i.id= o.id
where i.odf_object_code = 'idea' and o.MB_STATUS_IDEA= 7
and i.IS_ACTIVE=1
and i.SCHEDULE_FINISH between to_date( '01/07/2012', 'dd/mm/yyyy' ) and (sysdate - 15)
and o.partition_code != 'CP'
</sql:query>

<core:choose>
  <core:when test="\${checkRequest.rowCount == 0}"/>
  <core:otherwise>
    <core:forEach items="\${checkRequest.rowsByIndex}" var="Request">
      <core:set value="\${Request[0]}" var="id"/>
      <core:set value="\${Request[1]}" var="schedule"/>

      <gel:log>id_Request: \${id}</gel:log>
      <gel:log>Schedule: \${schedule}</gel:log>

      <!-- toglia il time entry di tutte le risorse dell'investments-->
      <sql:update var="updateTeam">
        update prteam
        set prisopen = 0
        where prprojectid =\${id}
      </sql:update>

      <!-- toglia il time entry dell'investments-->

      <sql:update var="updateTeam">
        update inv_investments
        set is_open_for_te=0
        where id = \${id}
      </sql:update>
    </core:forEach>
  </core:otherwise>
</core:choose>
</gel:script>

```

### Process 9) MB Notifica per request chiusa

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:set value="\${gel_objectInstanceId}" var="ProjID"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="\${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
    var="EntryURL"/>

  <!--mi estrapolo le info sulla request di chi ha aperta la request -->
  <sql:query escapeText="false" var="res">
    select email_address,
           first_name||' '||last_name nome,
           inv.name progetto,
           i.general_notes,
           inv.description
    from cmn_sec_users u, inv_ideas i, inv_investments inv
    where i.initiator_id=u.id
    and i.id=\${ProjID}
    and i.id=inv.id
  </sql:query>

```

```

<core:forEach items="{res.rowsByIndex}" trim="true" var="riga">
  <core:set value="{riga[0]}" var="email"/>
  <core:set value="{riga[1]}" var="nome"/>
  <core:set value="{riga[2]}" var="progetto"/>
  <core:set value="{riga[3]}" var="general_notes"/>
  <core:set value="{riga[4]}" var="descrizione"/>
</core:forEach>

<gel:email from="clarity@mediobanca.it"
  fromName="Amministratore Clarity"
  subject="La richiesta {progetto} e' stata chiusa"
to="{email} UFFICIOORGANIZZAZIONE@mediobanca.it">
La richiesta {progetto} - {descrizione}
e' stata chiusa

Eventuali Note:
{general_notes}

```

---

```

link per accedere in Clarity:
{EntryURL}/niku/app?action=pma.ideaProperties&id={ProjID}
</gel:email>

<!--controllo che nel campo customer sia stato specificato un indirizzo email-->
<sql:query escapeText="false" var="customer">
  select MB_CLIENTE_STRING, MB_RESOLUTION
  from odf_ca_idea
  where id={ProjID} and MB_CLIENTE_STRING like '%@%'
</sql:query>
<core:forEach items="{customer.rowsByIndex}" trim="true" var="customer_riga">
  <core:set value="{customer_riga[0]}" var="customer_adress"/>
  <core:set value="{customer_riga[1]}" var="resolution"/>
</core:forEach>

  <!--se si notifica al customer che la richiesta è stata chiusa -->
<core:choose>
<core:when test="{customer_adress ne NULL}">
  <gel:log>NOTIFICO AL CUSTOMER {customer_adress}</gel:log>
  <gel:email from="clarity@mediobanca.it"
    fromName="Amministratore Clarity"
    subject="La richiesta {progetto} e' stata chiusa"
to="{customer address}">
La richiesta {progetto} - {descrizione}
e' stata chiusa
  Eventuali Note:
  {general_notes}
  Risoluzione:
  {resolution}
</gel:email>
</core:when>
  <core:otherwise>
    <gel:log>Nessuna email specificata nel campo customer</gel:log>
  </core:otherwise>
</core:choose>

  <!--controllo la data di fine della request-->
<sql:query escapeText="false" var="finequery">
  select schedule_finish
  from INV_INVESTMENTS
  where id={ProjID}
</sql:query>
<core:forEach items="{finequery.rowsByIndex}" trim="true" var="data_fine">
  <core:set value="{data_fine[0]}" var="date_end"/>
</core:forEach>

<gel:log>Id Request :{ProjID} </gel:log>
<gel:log>Data:{date_end}</gel:log>
  <!--Se non è presente la data di fine inserisci la data di oggi-->
<core:choose>

```

```
<core:when test="{date_end eq NULL}">
  <gel:log>data non presente. Aggiorno con la today</gel:log>

  <sql:update var="DateEndDate">
    update inv_investments
    set schedule_finish = sysdate
    where id = ${ProjID}
  </sql:update>
</core:when>

  <core:otherwise>
    <gel:log>data presente</gel:log>
  </core:otherwise>
</core:choose>

  <gel:log> Chiusa ${EntryURL}/niku/app?action=pma.ideaProperties&amp;id=${ProjID}</gel:log>
</gel:script>
```

## Process 10) Step: "Notify Approval Manager, Admin Creator and Customer"

```
<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:core="jelly:core"
xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:sql="jelly:sql"
xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
<gel:setDataSource dbId="niku"/>
<core:set value="{gel_objectInstanceId}" var="ProjID"/>
<core:invokeStatic className="com.niku.union.config.ConfigurationManager" method="getInstance"
var="config"/>
<core:set value="{config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>
<sql:query escapeText="false" var="res">
  select email_address,
         first_name||' '||last_name nome,
         inv.name progetto
  from cmn_sec_users u, inv_ideas i, inv_investments inv
  where i.initiator_id=u.id
  and i.id={ProjID}
  and i.id=inv.id
  UNION
  select res.email,
         res.first_name||' '||res.last_name nome,
         inv.name progetto
  from srm_resources res, odf_multi_valued_lookups mvl, inv_investments inv
  where mvl.object='idea'
  and mvl.attribute='mb_ass_cc'
  and inv.id={ProjID}
  and inv.id=mvl.PK_ID
  and res.unique_name= mvl.value
</sql:query>
<core:forEach items="{res.rowsByIndex}" trim="true" var="riga">
  <core:set value="{riga[0]}" var="email"/>
  <core:set value="{riga[1]}" var="nome"/>
  <core:set value="{riga[2]}" var="progetto"/>
  <gel:email from="clarity@mediobanca.it" fromName="Amministratore Clarity" subject="La richiesta
${progetto} e' stata approvata" to="{email}">
  La richiesta ${progetto} e' stata approvata

  Per accedere clicca qui:
  ${EntryURL}/niku/app?action=pma.ideaProperties&id={ProjID}
</gel:email>
  <gel:log> to="{email}"</gel:log>
</core:forEach>
<gel:log> Approvata ${EntryURL}/niku/app?action=pma.ideaProperties&id={ProjID}</gel:log>
<!--controllo che nel campo customer sia stato specificato un indirizzo email-->
<sql:query escapeText="false" var="customer">
  select MB_CLIENTE_STRING, MB_RESOLUTION
         from odf_ca_idea
  where id={ProjID} and MB_CLIENTE_STRING like '%@%'
</sql:query>
<core:forEach items="{customer.rowsByIndex}" trim="true" var="customer_riga">
  <core:set value="{customer_riga[0]}" var="customer_adress"/>
  <core:set value="{customer_riga[1]}" var="resolution"/>
</core:forEach>
<!--se si notifica al customer che la richiesta è stata chiusa -->
<core:choose>
  <core:when test="{customer_adress ne NULL}">
    <gel:log>NOTIFICO AL CUSTOMER ${customer_adress}</gel:log>
```



```

    <gel:email from="clarity@mediobanca.it" fromName="Amministratore Clarity" subject="La richiesta
    ${progetto} e' stata approvata" to="${customer_adress}">
    La richiesta ${progetto} e' stata approvata

</gel:email>
    </core:when>
    <core:otherwise>
        <gel:log>Nessuna email specificata nel campo customer</gel:log>
    </core:otherwise>
</core:choose>
<!-- fine controllo customer -->
<sql:query var="request">
    select last_name||','||first_name appr_name from cmn_sec_users a, odf_ca_idea b
    where a.id=b.last_updated_by
    and b.id=${gel_objectInstanceId}
</sql:query>
<core:forEach items="${request.rowsByIndex}" var="requestRow">
    <core:set value="${requestRow[0]}" var="ApprID"/>
</core:forEach>
<gel:log>Approved by ${ApprID}</gel:log>
<sql:update>
    UPDATE odf_ca_idea
    SET approved_by='${ApprID}' , data_di_approvazione = sysdate
    WHERE id=${gel_objectInstanceId}
</sql:update>
</gel:script>

```

#### Process 10) Step: "Notify reject administrator"

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:core="jelly:core"
xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
    xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
    xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:sql="jelly:sql"
xmlns:xog="http://www.niku.com/xog"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <gel:setDataSource dbId="niku"/>
    <core:set value="${gel_objectInstanceId}" var="ProjID"/>
    <core:invokeStatic className="com.niku.union.config.ConfigurationManager" method="getInstance"
var="config"/>
    <core:set value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>
    <sql:query escapeText="false" var="res">
        select email_address,
            first_name||' '||last_name nome,
            inv.name progetto
        from cmn_sec_users u, inv_ideas i, inv_investments inv
        where i.initiator_id=u.id
        and i.id=${ProjID}
        and i.id=inv.id
    </sql:query>
    <core:forEach items="${res.rowsByIndex}" trim="true" var="riga">
        <core:set value="${riga[0]}" var="email"/>
        <core:set value="${riga[1]}" var="nome"/>
        <core:set value="${riga[2]}" var="progetto"/>
    </core:forEach>
    <gel:email from="clarity@mediobanca.it" fromName="Amministratore Clarity" subject="La richiesta
    ${Progetto} e' stata rifiutata" to="${email}">
    La richiesta ${Progetto} e' stata rifiutata

    Clicca qui per accedere:
    ${EntryURL}/niku/app?action=pma.ideaProperties&amp;id=${ProjID}
</gel:email>

```

```

<!--controllo che nel campo customer sia stato specificato un indirizzo email-->
<sql:query escapeText="false" var="customer">
  select MB_CLIENTE_STRING, MB_RESOLUTION
                                     from odf_ca_idea
  where id=${ProjID} and MB_CLIENTE_STRING like '%@%'
</sql:query>
<core:forEach items="${customer.rowsByIndex}" trim="true" var="customer_riga">
  <core:set value="${customer_riga[0]}" var="customer_adress"/>
  <core:set value="${customer_riga[1]}" var="resolution"/>
</core:forEach>
<!--se si notifica al customer che la richiesta è stata chiusa -->
<core:choose>
  <core:when test="${customer_adress ne NULL}">
    <gel:log>NOTIFICO AL CUSTOMER ${customer_adress}</gel:log>
    <gel:email from="clarity@mediobanca.it" fromName="Amministratore Clarity" subject="La richiesta
${progetto} e' stata rifiutata" to="${customer_adress}">
La richiesta ${Progetto} e' stata rifiutata
</gel:email>
  </core:when>
  <core:otherwise>
    <gel:log>Nessuna email specificata nel campo customer</gel:log>
  </core:otherwise>
</core:choose>
<!-- fine controllo customer -->
</gel:script>

```

#### Process 8) step "Add Team" (Insert resources in the team)

```

<gel:script xmlns:core="jelly:core" xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:sql="jelly:sql" xmlns:u="jelly:util"
xmlns:util="jelly:util" xmlns:x="jelly:xml"
  xmlns:xog="http://www.niku.com/xog" xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <gel:parameter default="http://localhost" var="XOGURL"/>
  <gel:parameter default="admin" var="XOGUsername"/>
  <gel:parameter default="mediobanca" secure="true" var="XOGPassword"/>
  <gel:setDataSource dbId="Niku"/>
  <core:set value="${gel_objectInstanceId}" var="richiesta_id"/>
  <!-- Mi prendo le date -->
  <sql:query var="selectIdea">
    select
      to_char (i.schedule_start,'YYYY-MM-DD'),
      to_char (i.schedule_finish , 'YYYY-MM-DD'),
      to_char (sysdate,'YYYY-MM-DD'),
      '/' || nbi_get_obs_flat_path_fct(child.id) OBS
    from
      srm_resources r,
      PRJ_OBS_ASSOCIATIONS poa,
      odf_ca_idea oci,
      INV_INVESTMENTS i,
      PRJ_OBS_UNITS child
    where i.id = ${gel_objectInstanceId}
      and i.id = oci.id
      and oci.mb_managerid = r.id
      and r.id = poa.record_id
      and child.id = poa.unit_id
      and poa.table_name = 'SRM_RESOURCES'

  </sql:query>
  <core:choose>
    <core:when test="${selectIdea.rowCount == 0}"/>
    <core:otherwise>
      <core:forEach items="${selectIdea.rowsByIndex}" var="obsUnitRow">
        <core:choose>
          <core:when test="${obsUnitRow[0] == null}">
            <core:set value="${obsUnitRow[2]}T00:00:00" var="dateStart"/>
          </core:when>
          <core:otherwise>
            <core:set value="${obsUnitRow[0]}T00:00:00" var="dateStart"/>
          </core:otherwise>
        </core:choose>
      </core:forEach>
    </core:otherwise>
  </core:choose>

```

```

        </core:otherwise>
    </core:choose>
    <core:choose>
        <core:when test="{obsUnitRow[1] == null}">
            <core:set value="{dateStart}" var="dateFinish"/>
        </core:when>
        <core:otherwise>
            <core:set value="{obsUnitRow[1]}T00:00:00" var="dateFinish"/>
        </core:otherwise>
    </core:choose>
    <core:set value="{obsUnitRow[3]}" var="obs"/>
    <util:replace new="/" old="/" value="{obs}" var="obs"/>
    <gel:log>obs:{obs}</gel:log>
</core:forEach>
</core:otherwise>
<gel:log>Data start:{dateStart}</gel:log>
<gel:log>Data fine:{dateFinish}</gel:log>
</core:choose>
<sql:query var="res">
                                select
                                res.unique_name uni_name,
                                inv.name nam_ide,
                                inv.code cod_ide
                                from odf_ca_idea o, inv_investments inv , SRM_RESOURCES
res
                                WHERE inv.id = o.id and inv.id= ${richiesta_id} and
res.user_id = inv.created_by
</sql:query>
    <core:choose>
        <!-- Check Assignment found-->
        <core:when test="{res.rowCount eq 0}">
            <gel:log>Assignment not found </gel:log>
        </core:when>
        <core:otherwise>
            <core:set value="{res.rows[0].uni_name}" var="resourceID"/>
            <core:set value="{res.rows[0].nam_ide}" var="ideaname"/>
            <core:set value="{res.rows[0].cod_ide}" var="ideacode"/>
            <gel:log>ResourceID: ${resourceID} ,IDEANAME : ${ideaname} ,IDEACODE : ${ideacode}
</gel:log>
            <gel:parse var="updateidea">
                <NikuDataBus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../xsd/nikuxog_idea.xsd">
                    <Header action="write" externalSource="NIKU" objectType="idea" version="8.1"/>
                    <Ideas>
                        <Idea finish="{dateFinish}" name="{ideaname}" objectID="{ideacode}"
openForTimeEntry="true" start="{dateStart}">
                            <InvestmentResources>
                                <Resource defaultAllocation="0" openForTimeEntry="1" resourceID="{resourceID}">
                                    <AllocCurve/>
                                </Resource>
                            </InvestmentResources>
                            <InvestmentTasks>
                                <Task name="{ideaname}" orderID="1" outlineLevel="1" percComp="0" status="0"
summary="false" taskID="~rmw" topDownPercent="0">
                                    <Assignments>
                                        <TaskLabor actualWork="0" baselineWork="0" estPattern="1" lastUpdatedBy="1"
remainingWork="0" resourceID="{resourceID}" unpostedActuals="0"/>
                                    </Assignments>
                                    <estimateRules/>
                                    <CustomInformation/>
                                </Task>
                            </InvestmentTasks>
                            <OBSAssocs>
                                <OBSAssoc id="mborg" name="Struttura Organizzativa" unitPath="{obs}"/>
                            </OBSAssocs>
                        </Idea>
                    </Ideas>
                </NikuDataBus>
            </gel:parse>
        <!-- login xog -->
        <soap:invoke endpoint="{XOGURL}/niku/xog" var="auth">

```

```

    <soap:message>
      <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xog="http://www.niku.com/xog">
        <soapenv:Header/>
        <soapenv:Body>
          <xog:Login>
            <xog:Username>${XOGUsername}</xog:Username>
            <xog>Password>${XOGPassword}</xog>Password>
          </xog:Login>
        </soapenv:Body>
      </soapenv:Envelope>
    </soap:message>
  </soap:invoke>
  <gel:set asString="true" select="$auth/soapenv:Envelope/soapenv:Body/xog:SessionID/text()"
var=sessionID"/>
  <gel:log>${sessionID}</gel:log>
  <gel:log>
    <gel:expr select="$updateidea"/>
  </gel:log>
  <soap:invoke endpoint="${XOGURL}/niku/xog" var="idearesult">
    <soap:message>
      <soapenv:Envelope>
        <soapenv:Header>
          <xog:Auth>
            <xog:SessionID>${sessionID}</xog:SessionID>
          </xog:Auth>
        </soapenv:Header>
        <soapenv:Body>
          <gel:include select="$updateidea"/>
        </soapenv:Body>
      </soapenv:Envelope>
    </soap:message>
  </soap:invoke>
  <gel:set asString="true"
select="$idearesult/soapenv:Envelope/soapenv:Body/XOGOutput/Status/@state" var="stato"/>
  <gel:set asString="true"
select="$idearesult/soapenv:Envelope/soapenv:Body/XOGOutput/Records/Record/ErrorInformation/Severity/text()"
var="errorLevel"/>
  <core:switch on="${stato}">
    <core:case value="SUCCESS">
      <core:choose>
        <core:when test="${errorLevel == 'FATAL'}">
          <gel:log level="ERROR">Errore Inserimento Team e Inserimento OBS</gel:log>
        </core:when>
        <core:otherwise>
          <gel:log>Resource Inserita nel Team </gel:log>
        </core:otherwise>
      </core:choose>
    </core:case>
  </core:switch>
  </core:otherwise>
</core:choose>
<core:choose>
  <gel:log> entreo nel choose</gel:log>
  <gel:log> start ${obsUnitRow[0]} è nulla</gel:log>
  <core:when test="${obsUnitRow[0] == null}">
    <sql:update>
      update INV_INVESTMENTS
      set schedule_start=null
      where id = ${richiesta_id}
    </sql:update>
  <gel:log> ho trovato nulla start</gel:log>
  </core:when>
</core:choose>
<core:choose>
  <core:when test="${obsUnitRow[1] == null}">
    <sql:update>
      update INV_INVESTMENTS
      set schedule_finish=null
      where id = ${richiesta_id}
    </sql:update>
  </core:when>
</core:choose>

```

```

    <gel:log> ho trovato nulla finish</gel:log>
  </core:when>
</core:choose>
<gel:log> start ${obsUnitRow[0]}</gel:log>
<gel:log> finsih ${obsUnitRow[1]}</gel:log>
<gel:log>until ce</gel:log>
<util:sleep millis="60"/>
</gel:script>

```

### Process 11) MB Notifica mail per inserimento due date

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:set value="${gel_objectInstanceId}" var="ProjID"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
    var="EntryURL"/>

  <sql:query escapeText="false" var="res">
    select email_address,
           first_name||' '|last_name nome,
           inv.name progetto,
           i.general_notes,
           inv.description,
           to_char(idea.scadenza, 'DD/MM/YYYY')
    from cmn_sec_users u, inv_ideas i, inv_investments inv, odf_ca_idea idea
    where i.initiator_id=u.id
    and i.id=${ProjID}
    and i.id=inv.id
    and idea.id=inv.id
  </sql:query>
  <core:forEach items="${res.rowsByIndex}" trim="true" var="riga">
    <core:set value="${riga[0]}" var="email"/>
    <core:set value="${riga[1]}" var="nome"/>
    <core:set value="${riga[2]}" var="progetto"/>
    <core:set value="${riga[3]}" var="general_notes"/>
    <core:set value="${riga[4]}" var="description"/>
    <core:set value="${riga[5]}" var="due_date"/>
  </core:forEach>
  <gel:email from="clarity@mediobanca.it"
    fromName="Amministratore Clarity"
    subject="La Due Date della richiesta ${progetto} e' stata inserita" to="${email}
UFFICIOORGANIZZAZIONE@mediobanca.it">
  La Due Date della richiesta ${progetto} e' stata inserita: ${due_date}

  Descrizione:
  ${description}

  Eventuali Notes:
  ${general_notes}

  <hr/>
  Link per accedere a clarity:
  ${EntryURL}/niku/app?action=pma.ideaProperties&id=${ProjID}
</gel:email>
  <!--controllo che nel campo customer sia stato specificato un indirizzo email-->

```

```

<sql:query escapeText="false" var="customer">
select MB_CLIENTE_STRING, MB_RESOLUTION
  from odf_ca_idea
 where id=${ProjID} and MB_CLIENTE_STRING like '%@%'
</sql:query>
<core:forEach items="${customer.rowsByIndex}" trim="true" var="customer_riga">
  <core:set value="${customer_riga[0]}" var="customer_adress"/>
  <core:set value="${customer_riga[1]}" var="resolution"/>
</core:forEach>
<!--se si notifica al customer inserimento della due date -->
<core:choose>
  <core:when test="${customer_adress ne NULL}">
    <gel:log>NOTIFICO AL CUSTOMER ${customer_adress}</gel:log>
    <gel:email from="clarity@mediobanca.it"
      fromName="Amministratore Clarity"
      subject="La Due Date della richiesta ${progetto} e' stata inserita"
to="${customer_adress}">
La Due Date della richiesta ${progetto} e' stata inserita: ${due_date}

Descrizione:
${description}

Eventuali Notes:
${general_notes}
</gel:email>
  </core:when>
  <core:otherwise>
    <gel:log>Nessuna email specificata nel campo customer</gel:log>
  </core:otherwise>
</core:choose>
  <gel:log> Due Date Inserita in ${due_date} -
${EntryURL}/niku/app?action=pma.ideaProperties&id=${ProjID}</gel:log>
</gel:script>

```

## Process 12) MB Notifica mail per modifica Due date

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:set value="${gel_objectInstanceId}" var="ProjID"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>

  <sql:query escapeText="false" var="res">
select email_address,
  first_name||' '||last_name nome,
  inv.name progetto,
  i.general_notes,
  inv.description,
  to_char(idea.scadenza, 'DD/MM/YYYY')
from cmn_sec_users u, inv_ideas i, inv_investments inv, odf_ca_idea idea
where i.initiator_id=u.id
and i.id=${ProjID}
and i.id=inv.id
and idea.id=inv.id
</sql:query>
  <core:forEach items="${res.rowsByIndex}" trim="true" var="riga">
    <core:set value="${riga[0]}" var="email"/>
    <core:set value="${riga[1]}" var="nome"/>
    <core:set value="${riga[2]}" var="progetto"/>

```

```

    <core:set value="${riga[3]}" var="general_notes"/>
    <core:set value="${riga[4]}" var="description"/>
    <core:set value="${riga[5]}" var="due_date"/>
  </core:forEach>
  <gel:email from="clarity@mediobanca.it"
    fromName="Amministratore Clarity"
    subject="La Due Date della richiesta ${progetto} e' stata inserita" to="${email}
UFFICIOORGANIZZAZIONE@mediobanca.it">
La Due Date della richiesta ${progetto} e' stata inserita: ${due_date}

Descrizione:
${description}

Eventuali Notes:
${general_notes}

```

```

Link per accedere a clarity:
${EntryURL}/niku/app?action=pma.ideaProperties&id=${ProjID}
</gel:email>
  <!--controllo che nel campo customer sia stato specificato un indirizzo email-->
  <sql:query escapeText="false" var="customer">
    select MB_CLIENTE_STRING, MB_RESOLUTION
      from odf_ca_idea
     where id=${ProjID} and MB_CLIENTE_STRING like '%@%'
  </sql:query>
  <core:forEach items="${customer.rowsByIndex}" trim="true" var="customer_riga">
    <core:set value="${customer_riga[0]}" var="customer_adress"/>
    <core:set value="${customer_riga[1]}" var="resolution"/>
  </core:forEach>
  <!--se si notifica al customer inserimento della due date -->
  <core:choose>
    <core:when test="${customer_adress ne NULL}">
      <gel:log>NOTIFICO AL CUSTOMER ${customer_adress}</gel:log>
      <gel:email from="clarity@mediobanca.it"
        fromName="Amministratore Clarity"
        subject="La Due Date della richiesta ${progetto} e' stata inserita"
to="${customer_adress}">
La Due Date della richiesta ${progetto} e' stata inserita: ${due_date}

Descrizione:
${description}

Eventuali Notes:
${general_notes}
</gel:email>
  </core:when>
  <core:otherwise>
    <gel:log>Nessuna email specificata nel campo customer</gel:log>
  </core:otherwise>
</core:choose>
  <gel:log> Due Date Inserita in ${due_date} -
${EntryURL}/niku/app?action=pma.ideaProperties&id=${ProjID}</gel:log>
</gel:script>

```

#### Process 14) MB processo creazione progetto

```

<!-- Lettura progetti PRODUZIONE -->
<!-- definizione librerie da usare -->
<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql"
  xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"

```

```

    xmlns:util="jelly:util">
<!-- parametri -->
<gel:parameter var="XOGURL"      default="http://localhost"/>
<gel:parameter var="clarityUser" default="admin"/>
<gel:parameter var="claritypwd"  default="mediobanca" secure="true"/>

<!-- Log into XOG and get a session ID -->
<soap:invoke endpoint="\${XOGURL}/niku/xog" var="auth">
  <soap:message>
    <soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
      xmlns:xog="http://www.niku.com/xog">
      <soapenv:Header/>
      <soapenv:Body>
        <xog:Login>
          <xog:Username>\${clarityUser}</xog:Username>
          <xog:Password>\${claritypwd}</xog:Password>
        </xog:Login>
      </soapenv:Body>
    </soapenv:Envelope>
  </soap:message>
</soap:invoke>

<!-- Checking wheter a sessionID is returned. If not, it means that Login was unsuccessful -->
<gel:set asString="true" select="\$auth/SOAP-ENV:Envelope/SOAP-ENV:Body/xog:SessionID/text()"
var="sessionID"/>
  <core:choose>
    <core:when test="\${sessionID == null}">
      <gel:log>Couldn't Log in. Check the username/password</gel:log>
    </core:when>
  </core:choose>

<!-- Valorizza variabile col Progressivo del Progetto fornito dal Processo -->
<core:set var="ProjectID" value="\${gel_objectInstanceId}"/>

<!-- Solo per il test: progressivo in chiaro -->

<core:set var="ProjectID" value="5004041"/>
-->

<!-- Valorizza variabile col nome del DB -->

```

```

<gel:setDataSource dbId="Niku"/>

<!-- Query con l'ID dell'idea -->
<sql:query var="projectCursor">
select mb_assreq from ODF_CA_PROJECT
where id = \${ProjectID}
</sql:query>

<gel:log>Ho letto: ... \${ProjectID} ...</gel:log>

<!-- Estrae il campo dal cursore -->
<core:forEach items="\${projectCursor.rowsByIndex}" var="projectrow">
<core:set var="IDIDEA" value="\${projectrow[0]}"/>
</core:forEach>

<gel:log>ProjectID: ... \${IDIDEA} ...</gel:log>

<!-- UNIQUE NAME-->
<sql:query var="progettoCursor">
select unique_name from SRM_PROJECTS
where id = \${ProjectID}
</sql:query>

<!-- Estrae il campo dal cursore -->
<core:forEach items="\${progettoCursor.rowsByIndex}" var="progettorow">
<core:set var="uniqueName" value="\${progettorow[0]}"/>
</core:forEach>

```



```

<gel:log>Unique Name: ... ${uniqueName} ...</gel:log>

<!-- Query con l'ID dell'idea dal Progetto-->
<sql:query var="CursorIdea">
select mb_idea_id, mb_cliente, mb_uffreq, mb_applicazione, mb_requesttype
from ODF_CA_IDEA
where id = ${IDIDEA}
</sql:query>

<gel:log>Ho letto IDEA: ... ${IDIDEA} ...</gel:log>
<!-- Mi prendo anche l'unique code della request-->
<!-- Estrae il campo dal cusore -->
<core:forEach items="${CursorIdea.rowsByIndex}" var="rowidea">
<core:set var="mb_idea_id" value="${rowidea[0]}"/>
<core:set var="mb_cliente" value="${rowidea[1]}"/>
<core:set var="mb_uffreq" value="${rowidea[2]}"/>
<core:set var="mb_applicazione" value="${rowidea[3]}"/>
<core:set var="mb_requesttype" value="${rowidea[4]}"/>
</core:forEach>

<gel:log>mb_idea_id da IDEA: ... ${mb_idea_id} ...</gel:log>
<gel:log>mb_cliente da IDEA: ... ${mb_cliente} ...</gel:log>
<gel:log>mb_uffreq da IDEA: ... ${mb_uffreq} ...</gel:log>
<gel:log>mb_applicazione da IDEA: ... ${mb_applicazione} ...</gel:log>
<gel:log>mb_requesttype da IDEA: ... ${mb_requesttype} ...</gel:log>

<!-- Gaetano mi prendo l'unique code della request-->
<sql:query var="CursorIdeaCode">
select code from INV_INVESTMENTS
where id = ${IDIDEA}
</sql:query>
<core:forEach items="${CursorIdeaCode.rowsByIndex}" var="querycodiceuniquo">
<core:set var="uniquecode" value="${querycodiceuniquo[0]}"/>
</core:forEach>
<gel:log>Unique code: ${uniquecode}...</gel:log>

<!-- Gaetano mi prendo dall'ufficio richiedente il nome della lookup-->
<sql:query var="InternalCustomer">
SELECT a.name from prj_obs_units a
where id= ${mb_uffreq}
</sql:query>
<core:forEach items="${InternalCustomer.rowsByIndex}" var="queryclienteinterno">
<core:set var="internocliente" value="${queryclienteinterno[0]}"/>
</core:forEach>

<gel:log>Ufficio richiedente ${internocliente}...</gel:log>

<!-- ho salvato il nome dell'ufficio richiedente in interno cliente-->
<core:forEach items="${CursorIdeaCode.rowsByIndex}" var="rowideadesc">
<core:set var="unique_code" value="${rowideadesc[0]}"/>
</core:forEach>

<core:if test="${mb_uffreq == NULL}">
<gel:log>Ufficiò NULLO...</gel:log>
<core:set var="mb_uffreq" value="NULL"/>
</core:if>

<core:if test="${mb_idea_id == NULL}">
<gel:log>ID NULLO...</gel:log>
<core:set var="mb_idea_id" value="NULL"/>
</core:if>

<core:if test="${mb_applicazione == NULL}">
<gel:log>Applicazione NULLA...</gel:log>
<core:set var="mb_applicazione" value="NULL"/>
</core:if>

<!-- Aggiorno il progetto coi campi dell'Idea -->

```

```

<sql:update>
update ODF_CA_PROJECT set
mb_cliente_string = '${internocliente}',
mb_uffreq = ${mb_uffreq},
mb_applicazione = ${mb_applicazione},
code_request = '${uniquecode}',
mb_requesttype = '${mb_requesttype}'
where id = ${ProjectID}
</sql:update>

<sql:update>
update ODF_CA_IDEA
set mb_status_idea=8,
mb_progetto=${ProjectID}
where id = ${IDIDEA}
</sql:update>

<sql:query var="selectProgetto">
select code,name
from inv_investments
where id = ${ProjectID}
</sql:query>

<!-- Estrae il campo dal cusore -->
<core:forEach items="${selectProgetto.rowsByIndex}" var="rowprogettonome">
<core:set var="progetto_codice" value="${rowprogettonome[0]}/">
<core:set var="progetto_nome" value="${rowprogettonome[1]}/">
<gel:log>codice progetto: ${progetto_codice} Nome ${progetto_nome} ...</gel:log>
</core:forEach>

<sql:query var="selectIdeaObs">
select '/' || nbi_get_obs_flat_path_fct(child.id) OBS,
to_char (inv.schedule_start,'YYYY-MM-DD'),
to_char (inv.schedule_finish , 'YYYY-MM-DD')
from srm_resources r,
PRJ_OBS_ASSOCIATIONS poa,
PRJ_OBS_UNITS child,
inv_investments inv,
cmn_sec_users us
where r.user_id=us.Id
and r.id = poa.record_id
and child.id = poa.unit_id
and poa.table_name = 'SRM_RESOURCES'
and inv.id=${ProjectID}
and us.id=inv.MANAGER_ID
</sql:query>

<gel:log>Record trovati: ${selectIdeaObs.rowCount}</gel:log>

<!-- Estrae il campo dal cusore -->
<core:forEach items="${selectIdeaObs.rowsByIndex}" var="rowideaobs">
<core:set var="obs" value="${rowideaobs[0]}/">
<!--core:set value="${rowideaobs[1]}T00:00:00" var="dateStart"/>
<core:set value="${rowideaobs[2]}T00:00:00" var="dateFinish"/-->
<gel:log>OBS ${obs} ...</gel:log>
</core:forEach>
<core:choose>
<core:when test="${selectIdeaObs.rowCount == 0}"/>
<core:otherwise>
<core:forEach items="${selectIdeaObs.rowsByIndex}" var="obsUnitRow">
<util:replace var="obs" value="${obs}" old="/" new="/" />
<gel:log>obs:${obs}</gel:log>
<!-- Accertata, e ricavata, la presenza dell'OBS in Clarity, si predispono la
struttura dell'XOG per l'aggiornamento dell'idea -->
<gel:parse var="ideaWrite">
<NikuDataBus xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="../xsd/nikuxog_project.xsd">
<Header action="write" externalSource="NIKU"
objectType="project" version="12.0.3.5164"/>
</Projects>

```

```

name="{progetto_nome}" >
    <Project projectID="{progetto_codice}"
        <Allocations/>
        <scenarioDependencies/>
        <InvestmentAssociations>
            <Allocations/>
            <Hierarchies/>
        </InvestmentAssociations>
        <OBSAssocs>
            <OBSAssoc id="mborg" name="Struttura
Organizzativa" unitPath="{obs}"/>
        </OBSAssocs>
        <BurdeningAssocs/>
    </Project>
</Projects>
</NikuDataBus>
</gel:parse>
<!-- <gel:serialize fileName="c:\XOG\Output\obs.xml" var="{ideaWrite}"/> -->
<!-- Si apre la connessione verso i WS di XOG -->
<gel:log>Connessione in corso verso i WS di XOG.</gel:log>
<soap:invoke endpoint="{XOGURL}/niku/xog" var="auth">
    <soap:message>
        <soapenv:Envelope
            xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
            xmlns:xog="http://www.niku.com/xog">
            <soapenv:Header/>
            <soapenv:Body>
                <xog:Login>
                    <xog:Username>{clarityUser}</xog:Username>
                    <xog>Password>{claritypwd}</xog>Password>
                </xog:Login>
            </soapenv:Body>
        </soapenv:Envelope>
    </soap:message>
</soap:invoke>
<gel:set asString="true"
    select="{auth/soapenv:Envelope/soapenv:Body/xog:SessionID/text()}"
var="sessionId"/>
    <gel:log>{sessionId}</gel:log>
    <core:choose>
        <core:when test="{sessionId == null}">
            <gel:log level="INFO">Errore connessione verso i WS di XOG.</gel:log>
            <gel:log>Errore connessione verso i WS di XOG.</gel:log>
        </core:when>
        <core:otherwise>
            <gel:log level="INFO">Connessione verso i WS di XOG stabilita. Session ID:
{sessionId}</gel:log>
            <gel:log>Connessione verso i WS di XOG stabilita. Session ID:
{sessionId}</gel:log>
            <!-- Aggiornamento Idea -->
            <soap:invoke endpoint="{XOGURL}/niku/xog" var="ideaResponse">
                <soap:message>
                    <soapenv:Envelope
                        xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
                        xmlns:xog="{XOGURL}">
                        <soapenv:Header>
                            <xog:Auth>
                                <xog:SessionID>{sessionId}</xog:SessionID>
                            </xog:Auth>
                        </soapenv:Header>
                        <soapenv:Body>
                            <gel:include select="{ideaWrite}"/>
                        </soapenv:Body>
                    </soapenv:Envelope>
                </soap:message>
            </soap:invoke>
        </core:otherwise>
    </core:choose>
</core:forEach>
</core:otherwise>
</core:choose>

```

```

<!-- <core:otherwise>
    <gel:log level="ERROR">Impossibile determinare lo UUID del gruppo DM Main.</gel:log>
    <gel:log>Impossibile determinare lo UUID del gruppo DM Main.</gel:log>
  </core:otherwise> -->
<util:sleep millis="60"/>
</gel:script>

```

### Process 16) MB send mail add resource to team

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
var="EntryURL"/>
  <!-- elenca tutte le Risorse non notificate presenti nel team di tutte le idee -->
  <sql:query var="checkTeam">
select s.email email, i.name ideaname, i.id rich_id, T.prid id team
from inv_investments i, odf_ca_idea o, PRTEAM T, SRM_RESOURCES S, odf_ca_team t1
where i.odf_object_code = 'idea'
and i.id = o.id
and T.PRPROJECTID = i.id
and T.PPRESOURCEID = S.ID
and t.prid = t1.id
and o.mb_managerid != t.ppresourceid
and t1.team_notified= 0
and s.email is not null and s.is_active = 1 and s.PERSON_TYPE != 0
</sql:query>

  <core:choose>
    <core:when test="${checkTeam.rowCount == 0}"/>
    <core:otherwise>
      <core:forEach items="${checkTeam.rowsByIndex}" var="team">
        <core:set value="${team[0]}" var="email"/>
        <core:set value="${team[1]}" var="ideaname"/>
        <core:set value="${team[2]}" var="richiesta_id"/>
        <core:set value="${team[3]}" var="team_id"/>

        <gel:log>ideaName:${ideaname}</gel:log>
        <gel:log>Email:${email}</gel:log>
        <gel:log>Richiesta ID :${richiesta_id}</gel:log>
        <gel:log>Team ID :${team_id}</gel:log>

        <gel:email from="clarity@mediobanca.it"
          fromName="Amministratore Clarity"
          subject="Sei stato aggiunto al team della request ${ideaname} "
          to="${email}"/>
Sei stato aggiunto al team della request ${ideaname}

        ${EntryURL}/niku/app?action=pma.ideaProperties&id=${richiesta_id}
</gel:email>
        <gel:log> Notify ${EntryURL}/niku/app?action=pma.ideaProperties&id=${richiesta_id}</gel:log>

        <sql:update var="updateTeam">
          update odf_ca_team
            set team_notified = 1
            where id = ${team_id}
        </sql:update>
        <gel:log>

```

```

        update odf_ca_team
        set team_notified = 1
        where id = ${team_id}
    </gel:log>
</core:forEach>
</core:otherwise>
</core:choose>
</gel:script>

```

### Process 18) ST notifica per request suspended

```

<gel:script xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:core="jelly:core"
  xmlns:file="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:sql="jelly:sql" xmlns:xog="http://www.niku.com/xog"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance">
  <gel:setDataSource dbId="niku"/>
  <core:set value="${gel_objectInstanceId}" var="ProjID"/>
  <core:invokeStatic
    className="com.niku.union.config.ConfigurationManager"
    method="getInstance" var="config"/>
  <core:set
    value="${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}"
    var="EntryURL"/>
  <core:set var="EntryURLMB" value="http://xpclarity.xpiamrp.mbdom.mbgroun.ad:88"/>

  <sql:query escapeText="false" var="res">
    select email_address,
           first_name||' '||last_name nome,
           inv.name progetto,
           i.general_notes
    from cmn_sec_users u, inv_ideas i, inv_investments inv
    where i.initiator_id=u.id
    and i.id=${ProjID}
    and i.id=inv.id
  </sql:query>
  <core:forEach items="${res.rowsByIndex}" trim="true" var="riga">
    <core:set value="${riga[0]}" var="email"/>
    <core:set value="${riga[1]}" var="nome"/>
    <core:set value="${riga[2]}" var="progetto"/>
    <core:set value="${riga[3]}" var="general_notes"/>
  </core:forEach>
  <gel:email from="clarity@mediobanca.it"
    fromName="Amministratore Clarity"
    subject="La richiesta ${progetto} e' stata sospesa" to="${email} roberto.pesci@seteci.it
    Mauro.Restelli@seteci.it">
    La richiesta ${progetto} e' stata sospesa

    ${EntryURLMB}/niku/app?action=pma.ideaProperties&id=${ProjID}
    Eventuali Note:
    ${general_notes}

  </gel:email>
  <gel:log> Suspended ${EntryURLMB}/niku/app?action=pma.ideaProperties&id=${ProjID} </gel:log>
  <gel:log> email ${email}:roberto.pesci@seteci.it Mauro.Restelli@seteci.it </gel:log>
  <gel:log> notes ${general_notes} </gel:log>
</gel:script>

```

### Process 20) MB Check Baseline

```

<gel:script xmlns:core="jelly:core"
  xmlns:f="jelly:com.niku.union.gel.FileTagLibrary"
  xmlns:gel="jelly:com.niku.union.gel.GELTagLibrary"
  xmlns:obj="http://www.niku.com/xog/Object"
  xmlns:soap="jelly:com.niku.union.gel.SOAPTagLibrary"
  xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"

```

```

xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:sql="jelly:sql" xmlns:u="jelly:util" xmlns:util="jelly:util"
xmlns:x="jelly:xml" xmlns:xog="http://www.niku.com/xog"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">

  <gel:setDataSource dbId="Niku"/>
  <core:set var="EntryURL"
value="\${config.getProperties().getWebServer().getWebServerInstance(0).getEntryUrl()}" />

  <sql:query var="ProjectWithoutBaseline">
    select
      odf_project_v2.odf_pk,
      odf_project_v2.name,
      odf_project_v2.project_code,
      RES.email
    from ODF_PROJECT_V2, srm_resources res
    WHERE ODF_PROJECT_V2.IS_ACTIVE=1 and ODF_PROJECT_V2.baseline_start is null
    and ODF_PROJECT_V2.IS_PROGRAM=0 and ISOPEN=1 and ODF_PROJECT_V2.actuals != 0
    and res.user_id = ODF_PROJECT_V2.manager_id
  </sql:query>
  <gel:log>How many project haven't got baseline? : \${ProjectWithoutBaseline.rowCount}</gel:log>
  <!-- for each project without baseline, find the Project Manager and send him/her an email-->
  <core:choose>
    <core:when test="\${ProjectWithoutBaseline.rowCount == 0}" />
    <core:otherwise>
      <core:forEach items="\${ProjectWithoutBaseline.rowsByIndex}" var="project">
        <core:set value="\${project[0]}" var="odf_pk" />
        <core:set value="\${project[1]}" var="name" />
        <core:set value="\${project[2]}" var="project_code" />
        <core:set value="\${project[3]}" var="email" />
        <gel:log>Project id:\${odf_pk}</gel:log>
        <gel:log>Project Name:\${name}</gel:log>
        <gel:log>Project Caode:\${project_code}</gel:log>
        <gel:log>Email:\${email}</gel:log>
        <gel:email from="clarity@mediobanca.it"
          fromName="Amministratore Clarity"
          subject="missing baseline " to="\${email}">

Have to save the baseline of the project \${name}

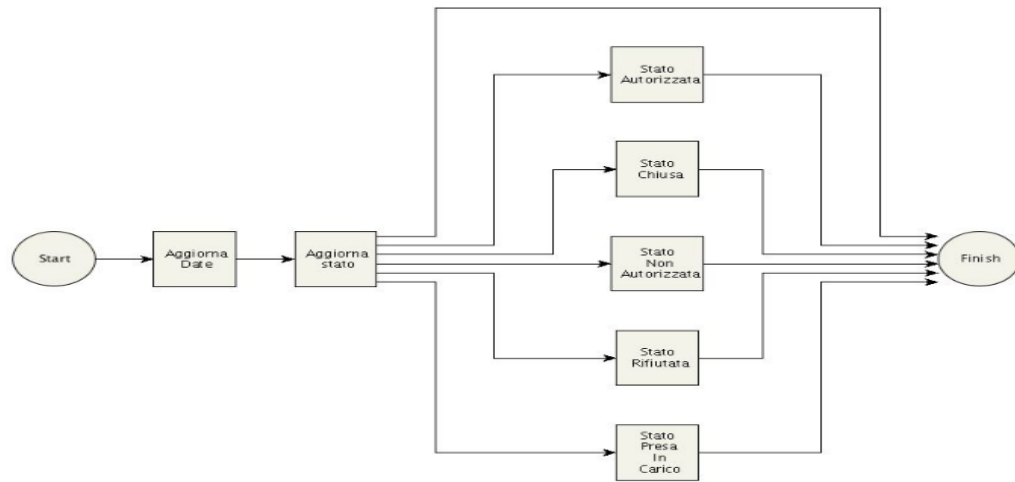
Contact System Administrator

\${EntryURL}/niku/nu#action:projmgr.projectProperties&amp;id=\${odf_pk}

</gel:email>
      <gel:log> Notify
\${XOGURL}/niku/app?action=projmgr.projectProperties&amp;id=\${odf_pk}</gel:log>
    </core:forEach>
  </core:otherwise>
</core:choose>
</gel:script>

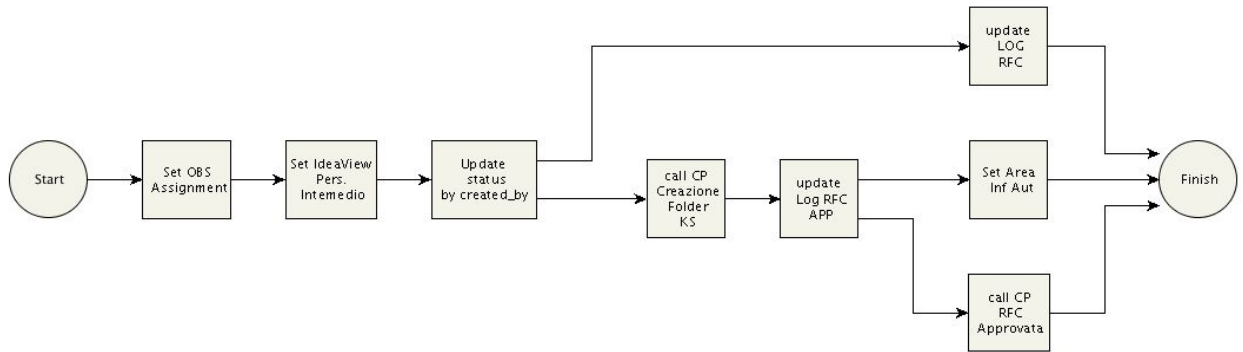
```

Process: CP Aggiorna Stato - Process Flow Diagram



Continue Return

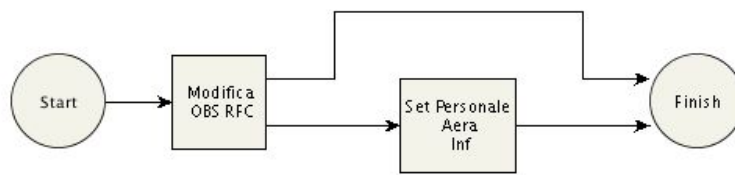
CP Creazione RFC v2.0 - Process Flow Diagram



Process: CP Insert Personale Area Informatica - Process Flow Diagram



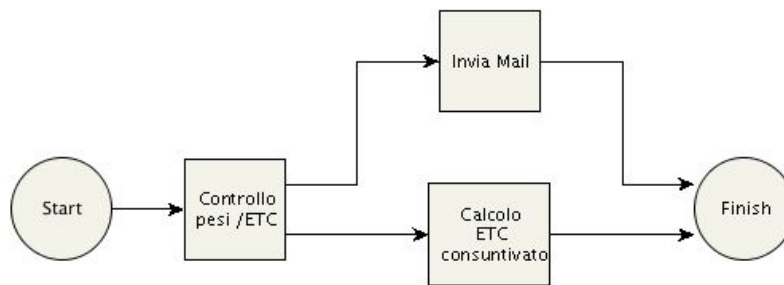
**Process: CP Modifica Assignment - Process Flow Diagram**



**Process: MB Notifica mail per inserimento Due Date - Process Flow Diagram**



**Process: Ricalcolo Effort Consuntivato - Process Flow Diagram**



**Process: MB Send Mail Add Resource to Team - Process Flow Diagram**



**Process: MB Check baseline - Process Flow Diagram**





**Process: CP Somma ETC/ACT per RFC INSERT - Process Flow Diagram**

