# Business Process Modeling

## White Paper

April 2008

# 1 Introduction

## 1.1 Abstract

**Business Process** thinking is at the heart of the management of modern Enterprises. The configuration of business has evolved considerably from the **hierarchical**, integrated and comprehensive structure of major corporations in the twentieth century to today's more globalized and **virtual network** of partnering companies which create value for the end Customer.

Nowadays, many managers think and talk about their business in terms of Business Processes. Extensive literature is available on Business Process Management, very often influenced by software vendors. We describe the **Business Process lifecycle** and provide a global set of definitions to position the various terms and acronyms that are frequently used, such as: Business Process Management (BPM), Business Process Modeling (also BPM!), Analysis, Design, Simulation, Automation, Business Activity Monitoring…

This white paper also explains how Business Processes can be described and what **benefits** can be derived from **Business Process Modeling,** such as:
- **Understanding** how the Enterprise Operates
- **Measuring** how the Enterprise performs
- **Designing** the future Operations of the Enterprise (Transforming the Enterprise)

One of our key objectives is to provide a standard way of **discovering** and **describing** existing Business Processes, which should reconcile and serve the objectives of **all parties** in the Enterprise (Business Analysts, QA experts, Risk managers, IT developers…). That is why we have developed a minimal set of terms to define Business Processes such as **Enterprise**, **Action**, **Actor**, **end to end Process**, **organized Process**, **Function** and **Activity**. For instance, we will define a **Business Process** as "a coordinated suite of Actions triggered by an **independent** Event to bring **Value** to a **Process Client".**

We advise separating the **core business** (what has to be done), from the **Organization** (who does what). The Core business is mostly made up of stable and invariant Activities while the Organization can frequently vary.

Then, we explain how to **define a good Process Model**. The scope of a Business Process Modeling project can vary:
- If the scope is the **whole Enterprise**, the objective is to list and map all the Processes of the Enterprise. This approach usually aims at getting a broad picture of all the Processes, but not necessarily a detailed description of each Process.
- If the scope is **one specific Business Process**, the aim is usually to document or Transform the Process, which requires a detailed analysis and design of the Process.
- If the scope is intermediary, like **a whole Business Domain**, the Process Modeling project will require a mix of the two approaches described above with the following deliverables:
  - A map of the Business Processes in this Business Domain and
  - A detailed description of one or several Processes in this Business Domain

We first describe how to **discover a list of Processes** across a broad scope (like a Business Domain or an Enterprise). We recommend using an **Entity-based approach** and we provide the example of the "Arpege" method developed by Air France KLM.

Then we give some tips on how to **design a single new Process** including the following key steps:
- Start from a clean baseline, scope definition and business strategy understanding
- Use properly **skilled** resources
- Identify the relevant **Business Entities** and their basic Functions
- Break down the Process into Functions at the right level of granularity
- **Reuse Process templates** and develop variants rather than new Processes

- Isolate Business Rules
- Assign Actions to Actors **at the end only**
  - Split end to end Processes into Organized Processes
  - Group Actions into Activities to optimize Resource utilization
- Check the Process design and iterate

We suggest some simple criteria to **assess the Process design**.

After applying this approach to a **simple example**, we explain how Process design can be linked to **software design**, and especially how reuse is possible using Process design patterns.

Then, we describe briefly some **existing modeling standards** (like UML or BPMN) and map our concepts against these approaches.

Finally, we go beyond Process modeling and introduce Process **automation and monitoring** as the next steps in the management of the full life cycle of a Business Process, which would deserve a full white paper in its own right.

## 1.2 Objectives of this white paper

This white paper aims to:
- Provide a set of **definitions** of terms which are used when describing Business Processes and mapping common terms used in Business Process Management today
- Explain why Business Process **modeling** is important and what benefits you can derive from it
- Provide a standard way of **discovering** and **describing** existing Business Processes, which should reconcile and serve the objectives of **all parties** in the Enterprise (Business Analysts, QA experts, Risk managers, IT developers…)
- Propose a standard way of **designing** new or improved Business Processes (especially defining the level of detail for Process description)
- Link Business Process modeling to other business disciplines like Organizational Theory and Organizational Design

In doing so, some key aspects of our white paper include:
- Splitting what is **generic** for all competitors in an industry, and what is considered as **specific** for each Enterprise (like a **competitive advantage**)
- Separating what is **specific to organization** (who does what) and which resources are used and how, from **what** has to be done
- Developing reusable Process templates to reduce complexity and increase flexibility (for instance by defining Process templates at Group level and specializing them at company or country level)
- Making design decisions based on alignment to business objectives

The guiding principle of this white paper is to **keep it simple** (develop few concepts, use a concrete vocabulary, and provide examples).

## 1.3 How has Business Process thinking emerged?

Understanding and representing the activities of a firm as a set of processes instead of just as a hierarchy of teams has evolved gradually since its first introduction at the beginning of the 20th century.

### 1.3.1 Brief History of Organizational Design and Organization Theory

Although all civilizations have some experience in structuring groups of people, modern Organization Theory dates back to the end of the 18th century in the writings of **Adam Smith**. Smith was the first author to introduce the **division of labor** as a new way to structure organization. At the end of the nineteenth century, **Max Weber** described (positively) how a large scale and complex organization evolves naturally as a **rational bureaucracy** with the following characteristics:
1. Specification of jobs with detailed rights, obligations, responsibilities, scope of authority
2. System of supervision and subordination
3. Unity of command

4. Extensive use of written documents
5. Training in job requirements and skills
6. Application of consistent and complete rules (company manual)
7. Assigning work and hiring personnel based on competence and experience

**Scientific Management**, developed by Frederick **Taylor**, pushed the rational approach to its full extent. By simply observing movements of physical labor and advising workers how to become more efficient, productivity was significantly improved. In parallel, attention was given to effective systems of variable pay, so that workers were motivated to apply more efficient work methods. Managers dealt with organizational design and planning, workers took care of execution. Henri Fayol, of the same school of thought, focused on organizational structure, looking at things like the most ideal team size and optimal "span of control". This school assumes that there is one optimal way to perform and that the people involved in the organization are rational actors in a closed system. The **hierarchical model** of the organization dates back to these thinkers. But human beings do not behave like mechanical systems. The **Human Relations School** developed in the twentieth century led by the likes of Mayo, McGregor and Maslow. Actors in the organization were to be considered as fully fledged social individuals with specific **motivations** like getting attention, being part of an elite… and not only having a job and a good pay.
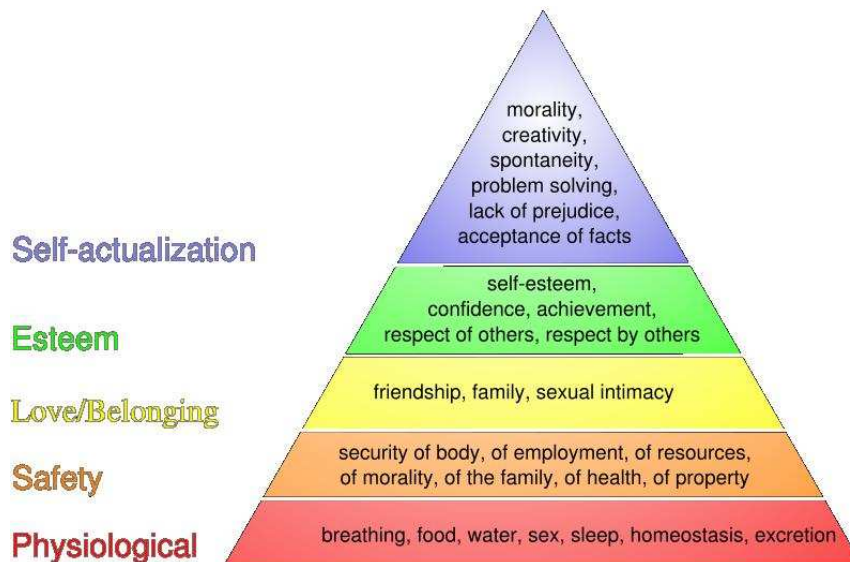


**Figure 1 - Pyramid of Maslow**

But this did not fundamentally change the way companies are structured (hierarchically).
Later on, the "bureaucratic" type of organization resulted in several forms such as:
- The **functional** structure: Functional structure groups specialize in similar skills in separate units.
- The **matrix** structure: A matrix structure overlays two organizational forms in order to leverage the benefits of both. The two dimensions can be product and geography or a functional structure and project teams.
- The **divisional** structure: is formed when an organization is split up into a number of self-managed units, each of which Operates as a profit center.

It is important to note the major influence the Taylor-based approaches have had on Organizational Design. Nevertheless, they have been challenged because they focus on structure and provide a static view of the organization. This creates tall structures with a heavy middle management and divides the company into **vertical silos** where actors lose the overall view and objective of the company.

Several parallel trends in the twentieth century tried to develop a more dynamic view of the enterprise. Starting in the 30's, but really developing after World War II, Toyota's **Just in Time** theory was developed to cut down inventory costs and unnecessary tasks with no added value. Continual improvement, called **Kaizen** in Japan, also developed at the same time and became part of the **Toyota Production system**. **Total Quality Management** formalized the efforts of analyzing the activities of a company in all its aspects to get the best out of the organization and its resources. This was later

formalized in **ISO 9000**, which clearly developed a focus on Processes to describe the activities of the company and optimize them, independently of the organizational structure of the firm. **Six SIGMA** and **Lean** are also quality approaches seeking to optimize activities in a company, deriving from industrial companies.

Key characteristics of these approaches are:
- A horizontal and dynamic focus across the value chain, linking all actors together
- The effort to remove any non value adding activity

### 1.3.2 Emergence of Business Process Management

This was further emphasized by Activity Based Costing/Activity Based Management (**ABC/ABM**) in the 70's and 80's, a bottom-up approach listing and measuring the cost of all activities in the company. The focus was on the measurement of the value creation for the end Customer in each activity, independently of the organizational structure.

In the 90's, the business process reengineering (**BPR**) school of Champy & Hammer developed the previous approaches in a radical movement to rethink major corporations. The Business Process was defined as "a collection of activities that takes one or more kinds of input and creates an output that is of value to the Customer." And thus, in times when American companies faced major competition from the rest of the world, BPR was introduced as "... the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical contemporary measures of performance, such as cost, quality, service, and speed". Hammer's claim was simple: Most of the work being done does not add any value for Customers, and this work should be removed, not accelerated through automation. Instead, companies should reconsider their processes in order to maximize Customer value, while minimizing the consumption of resources required for delivering their product or service. Hence BPR was a lot more ambitious and radical than the continuous improvement approach. Moreover, BPR really put the Customer back at the heart of the company.

The **globalization** of the economy, generalized at the turn of the twenty-first century, pushed the boundaries of business processes beyond the scope of the company to its suppliers and partners world-wide. As Information Technology and transportation optimization drove down transaction costs between corporations, it became possible to rethink the core mission of the company and decide which activities should be managed internally and which should be outsourced. But this approach emphasized even more the need for a clear definition of business processes spanning several companies in order to achieve successful coordination of all parties. The **extended enterprise** was born.
Through EDI, the T0.5 concept (Third party level 0.5) emerged and allowed big industries to outsource more than production; they pushed outsourcing to the design of whole parts of cars, as in the automotive industry. Strong and close relationships in production impose the design of strong and close processes with providers 0.5, which have to be very close to the business (an order to delivery approach) and highly automated (reduced delivery times).

Finally, we can mention the impact of an increasing number of **mergers and acquisitions** as of the 90s. Merging organizations impose the redesign of processes, to integrate both companies and achieve the synergies sought for in the merger.

### 1.3.3 Organization Theory and Business Process Modeling

Most of these initiatives have not gone all the way in terms of modeling processes in a standard way but they have transformed the way companies Operate; the move from a hierarchical model to a network model (extended to third parties).
Business process modeling and formalization can be a powerful tool to help organizations optimize their operating model. Nevertheless, no standard has ever emerged for representing business processes in this area. Most business process engineering or reengineering work undertaken by companies with the help of major consulting firms has used proprietary modeling tools and techniques.

BPM is also essential to software development and most of what exists today has been standardized by IT professionals to either:

- analyze a process before automating it  or
- manage the execution of the automated process (most of the work has been done in this area)

### 1.3.4  Business Process in organizations today

In most Enterprises, Business Analysts model new Processes and IT Developers implement them.
Some Enterprises have adopted a Matrix structure with vertical Functions and horizontal Business processes. A new role has appeared, called **Process Owner** or **Process Pilot**. He/she is responsible for managing the Operations and improvement plans of end to end Processes. For example LCL, a French bank and subsidiary of Crédit Agricole SA, has a team of 25 Process Pilots to cover the entire scope of activities of the company (including primary and support activities).

Even without going all the way to a new Business Process based structure, most Enterprises today describe their activity as Business Processes. Hence Business Processes are currently at the heart of most organizations. The next paragraph of this white paper will try to provide a complete picture of the activities included in Business Process Management.

## 1.4  Business Process life cycle

Business Processes evolve a lot from the first idea to a fully optimized and fine-tuned process after years of experience. We propose the following Business Process Lifecycle:
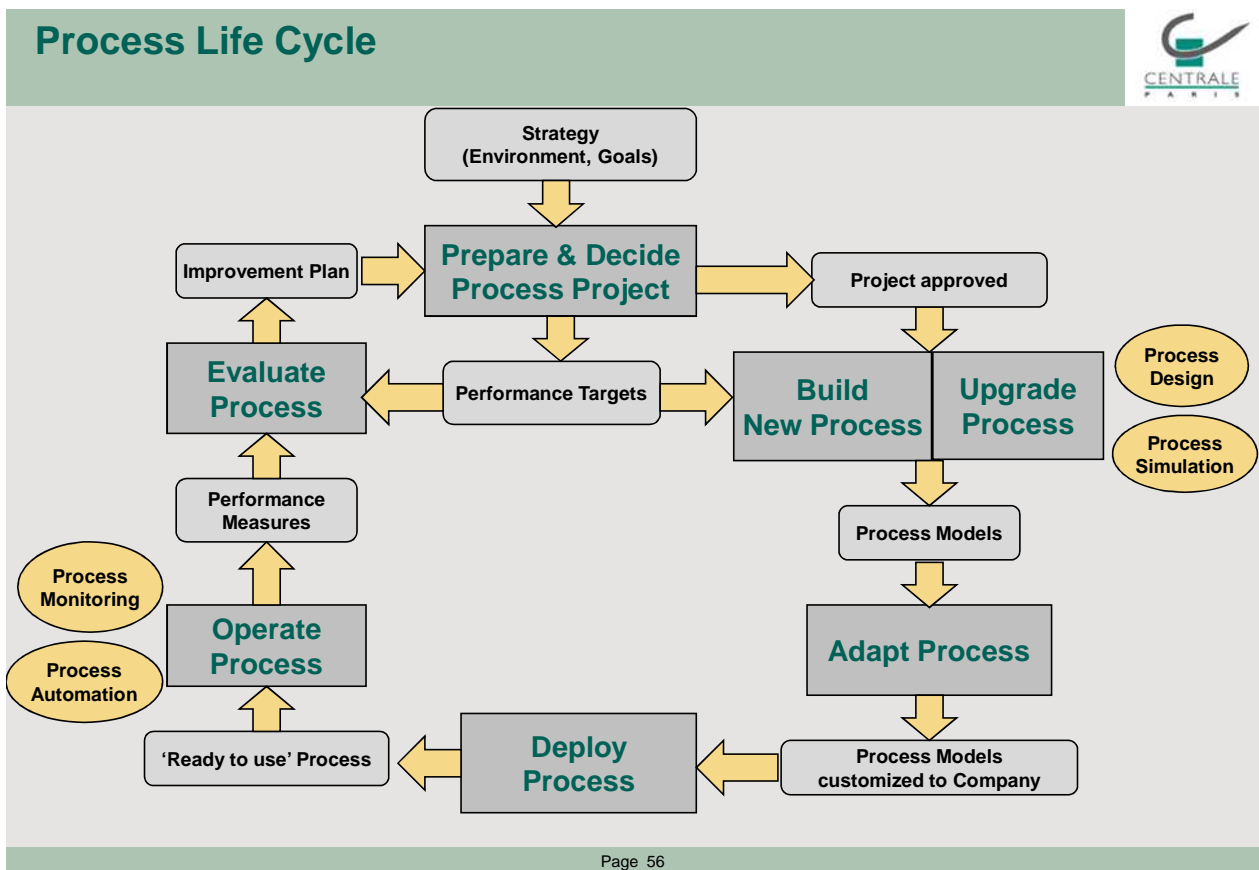


Business Process Life cycle is a loop (continuous evolution). There are six main steps in the loop:
- Preparing and Deciding Process Building/Upgrading Project
- Building/Upgrading the Process
- Adapting the Process
- Deploying the Process
- Operating the Process
- Evaluating the Process Performance

### 1.4.1 Preparing and Deciding

Before investing resources in building or upgrading Process, the Enterprise management team has to decide whether the project is profitable to the organization, and also if this project is better than other projects which are in the opportunities portfolio of the Enterprise. This decision usually requires a feasibility study to **explore the benefits and costs** related to the project (The "Prepare" phase). Then this project will be evaluated alongside other projects in the portfolio. It will be selected if it is among the most profitable projects that can be financed by the Enterprise (the "Decide" phase). This work is fed by information from:

- The goals and **strategy** of the Enterprise
- The analysis of the existing performance of the Enterprise processes (**baseline**)

### 1.4.2 Building or Upgrading

This phase can either be the development from scratch of a new process or the improvement of an existing process. If the new process is partially or totally automated, the software that will be used can be developed in-house or bought as a standard package (make or buy decision).

The following steps will be necessary:

- **Formalizing Requirements**:
    - Initially, the Process definition requires a description of its Client, objectives and environment (attributes and constraints of the Process). The input from the strategy and the description of the existing Processes are required.
- **Designing Process:**
    - Design Target Process (possibly using a Business Process Modeling tool)
    - Process target performance metrics should be defined
    - For subsequent iterations, Process is redesigned using input from the Process Evaluation
- **Implementing Process:**
    - Select who does what, which parts should be automated
    - For manual parts of the Process: document SOPs and develop training package
    - For automated parts of the Process: Develop IT solution (using workflow engine, packaged application or custom development)
- **Testing Process:**
    - For automated Processes, in addition to standard software engineering testing techniques, **Business Process Simulation** can (sometimes) be used
        - Provides an estimate of Business Performance when many instances of the Process are executed at the same time / or compete for resources
        - Process design weaknesses can be pointed out
- **Accepting Process:**
    - Check compliance of testing and/or simulation results with user requirements
    - Approve process or update design and restart loop

### 1.4.3 Adapting

This phase is used when a standard solution can be rolled out in several different Companies of an Enterprise. The roll-out will require some integration in the local context of the Company.

- **Customize:** Some customization and translation into local culture and constraints might be required
- **Interface:** Integration in the local environment requires creating or adapting interfaces to existing solutions in the Company
- **Prepare Data Migration:** Before deploying the new or updated Process, all the information required for the new or updated Process to run must be prepared. If the Process is automated, an automatic data take-on software might be developed.

### 1.4.4 Deploying

This phase consists in the actual roll-out to one or many Organization Units. If the roll-out is planned to impact many Organization Units (as many local branches of a distribution network for instance), a dedicated team might be created to manage the roll-out.

- **Training Actors:** All Operations Actors should be trained to execute the new or upgraded Process. Training is one of the key activities of change management, which should be developed carefully to make sure all Actors understand the new Process and the benefits the Enterprise and themselves can get from it.
- **Preparing Support:** A Support team should be identified, trained and tested. Depending on the complexity of the Process, the support might simply require training an existing support team or the creation of a dedicated new support group.
- **Installing Solution:** If the Process is automated, some hardware and software will have to be installed and configured.
- **Migrating Data:** Information required to feed the Process will have been prepared in the Adapt phase. In the Deploy phase, this data is fed into the new Process (especially if the Process is automated).

### 1.4.5 Operating

In this phase, the Process is executed in the day to day Operations of the Enterprise.
- **Executing Process:**
  - Process Instances are created, delivering value to Customers
  - Operation Actors can get support from a central support and administration team
- **Managing Incidents and Monitoring Performance:**
  - Process execution is monitored. Possible incidents are solved, so the Process can be completed
  - Process metrics are gathered to report on the Process performance and fed into the Evaluation step
  - If some recurring problems arise, the Process will have to be upgraded

### 1.4.6 Evaluating Process

This key objective of this phase is to compare the actual performance measures of the Process with the expected Performance targets to check if the planned **benefits** are actually being delivered. A list of known problems or areas of improvement can be gathered in an **improvement plan**, which can be used to justify an upgrade project.

### 1.4.7 Existing terminology

The following terms and acronyms are frequently used in the literature on Business Processes: Business Process Management (BPM), Business Process Modeling (also BPM!), Business Process Analysis, Business Process Design, Business Process Simulation, Business Process Automation, Business Activity Monitoring.

A degree of confusion about these terms has arisen due to the specific meaning given to these words by software editors of design or automation packages. Here are some clarifications of the definitions:

| Term | Existing definitions | Definition in this document |
|---|---|---|
| Business Process Management | Business Process management is often used to identify software solutions at any phase in the life cycle. Depending of the vendor, a BPM package can be a design tool, a workflow engine to execute the Process, a Process monitoring tool … | Involves the management of the entire life cycle of a Process, from Design to Execution The acronym BPM stands for Business Process Management (not Modeling). |
| Business Process Analysis | Used by software vendors and analysts to describe tools used to draw a graphical representation or map of one or several Processes. This involves describing existing Processes as well as designing improved or new Processes | Analysis requires gathering facts about an existing Process. This can be done to understand and document an existing Process and also possibly to prepare for some re-engineering of this Process. We usually do not use this term and prefer using Business Process Design. |
| Business Process Design | Used alternatively for Business Process Analysis. | The engineering or re-engineering of a Process. |
| Business Process Modeling | Used alternatively for Business Process Analysis and Business Process Design. | The creation of a Business Process description (Model) as a result of Business Process Analysis and Design. |
| Business Process Simulation | This term is usually used by software vendors who provide Simulation functionality for automated | Involves any activity to test a designed Process in a virtual environment to evaluate |

| | Processes. Usually closely linked to a Business process Analysis/Design tool. | its future performance before putting it into Operation. |
|---|---|---|
| Business Process Automation or Execution | This term is usually used by software vendors who provide engines to execute automated Processes | Used as common sense prescribes … |
| Business Activity Monitoring | This term is usually used by software vendors who provide monitoring tools for automated Processes. These tools try and relate technical incidents in the automated system with the actual impact on Business Activity. | Used with the same meaning. |

**The rest of this white paper will focus on Business Process Modeling, which includes Business Process Analysis and Design.**

# 2  Objectives and benefits of Business Process Modeling

In this chapter, we will break down the objectives of Business Process Modeling into three categories. For each of them, we will provide some examples of benefits we found in different Enterprises.

Business Process Modeling helps:

- Describe, **understand** and **communicate** the present Operations of the Enterprise
- **Assess** how well the Enterprise performs in terms of cost and value creation
- **Transform** the Enterprise by:
  - **Designing new processes** to develop new Operations (invest in a new field or create new products and services, or a new distribution channel)
  - **Upgrading existing Processes** to adapt to a new context or higher performance targets
    - The upgrade can be limited to the **optimization** of a small set of activities, as in the continuous quality improvement programs like Lean Six Sigma. The general design of the Process is not impacted
    - The upgrade can be a major **re-engineering** initiative with a bigger scope. Analyzing the current state of the Processes is important but innovating to find really new ways of Operating is even more essential

## 2.1  Understanding how the Enterprise Operates

Standard description of its Processes helps the Enterprise clearly understand how it works.

This formalization is useful for several reasons:

- It provides a **formalized and explicit description** of how your Enterprise Operates
  - It helps you communicate what you do, why and how you do it
  - This also helps create material to **train** new Organization Actors. This proves very useful in high turnover rate organizations, like call centers
  - Modeling Processes is also essential when you want to **replicate a "best practice"** Process in many different Organization Units across your Enterprise, globally
- It is a communication tool between **several parties interacting** with each other in a Business process. It is especially necessary between two partnering companies wishing to integrate their activities in a seamless business process

Moreover, this helps to describe the baseline required for any improvement project.

## 2.2  Measuring how the Enterprise performs

This formalization helps to:

- **Identify** areas of **improvement:** Just describing clearly what you do on a daily basis can demonstrate clearly where discrepancies are and generate a lot of improvement ideas for the current Process. Knowing yourself is usually the first step on the road to excellence. The Competence Center on Business Process Modeling of BNP Paribas, one of the leading banks in Europe, found that most of its internal Clients cite this benefit first when they are asked about the usefulness of Business Process Modeling
- **Compare** efficiency of different Business Units, Operating the same Processes
- Control compliance with **Regulatory** constraints and security rules: New regulations may involve implementing controls throughout the Enterprise to monitor how the business is Operating and demonstrating that the Enterprise abides by the law (Sarbanes-Oxley, Bale II...). These projects include the description of many Processes and the definition of controls on each process. They have been one of the major drivers for Process Modeling in the recent years [The cost of a SOX project was estimated by ARC Morgan (2005) at 3 million per billion of revenue; other sources estimated these costs at up to 0.8% of revenue, depending on enterprise size]
- Build activity measures (**Process performance** metrics): Performance criteria are important attributes of a Process. They must be chosen carefully as optimizing the performance of one Process should not be done at the expense of another Process
- Build **Cost measures** (Activity based costing): As human costs range from 40% to 60% of total enterprise costs, the human part for processing the business is a main source of savings. Process modeling must provide a correct valuation of human costs per process. Thus costs

valuation ought to include direct and indirect costs. Other costs directly related to process running can be added to build a complete and coherent model.

# 2.3 Transforming the Enterprise

Transformation can take two forms:
- **Continuous improvement**: Upgrading existing Processes to adapt to a new context or higher performance targets
- **Breakthrough**: Designing new processes to develop new Operations (investing in a new field or creating new products and services, or a new distribution channel)

## 2.3.1 Continuous improvement

This kind of project usually has a limited scope. One set of Activities or one Process is modified at a time. The benefits might not come from process redesign but simply from **optimization** of interfaces between tasks, as we found in several examples of Lean Six Sigma projects we observed. The aim and core definition of the end to end Process are not modified; the split of Activities and allocation to Organization Actors in time and space can be the only area of concern. Process Modeling might help build a database of Process models, which is useful if there is a will to share best practices between Organization Units.

## 2.3.2 Breakthrough

Another situation is the true **Transformation** of the Enterprise, which requires the ability to design new processes from scratch. This is especially useful when a company wants to invest in a new business and does not want to acquire another company to do so. It is also the case when a company is facing difficulties and has to consider a **complete re-engineering** of its processes to survive. We call this situation "Breakthrough" or "Big Leap Forward". The change is more drastic, given the way the Enterprise Operates is Transformed. Potential gains are higher. Some scenarios include:
- Managing **End to End** Processes: Consolidating smaller partial processes to build truly complete Processes serving the Customer. This reduces the number of Processes and implies a Process Owner (who does not usually fit in the Organization Structure because end to end Processes navigate across different Organization Units)
- Recognition of "**sharable**" Processes and reengineering of Enterprise Processes across global Business Units (shared service centers: back-office, service desks...)
- **Extended Enterprise** : building new relations with partners or Customers (with shared end to end Processes)
- **New Business**
  o New Product
  o New Partner
  o New Customer relationship
- **Mergers & Acquisitions**
- **Business Process Automation**

## 2.3.3 Business Processes and IT Systems

Business Process Automation is seen as a major potential for improvement as it can help reduce lead times and internal costs.

In current Enterprise Architecture practice, Business Processes are described in what is usually called Business Architecture and IS Applications are described in the IS Architecture (see TOGAF). Business Process maps and IS Applications maps are linked and interact both ways.

Describing Business Processes in the Business Architecture serves as an input to develop IS Applications and align the IS Architecture to the Business needs.

But the IS Application can also impact the Business Architecture. If you buy a major software package (such as an ERP or CRM system) to build the backbone of your IS Application landscape, you also import the Business Processes which are implemented in this software. You then need to verify if this package can be configured to suit your needs (especially check if the configuration of the Organization structure can be easily changed in the system).

In general, any Business Process automation project raises the following questions:

- How to align Processes with the IT System?
- How to analyze a Process so that the software that automates it can **quickly adapt to successive organizations**?
- How to take advantage of a **Rule Engine** or a **Workflow Engine** to help Business Professionals be more productive by directly updating Business Rules and Organization themselves?
- How to manage a set of **sharable Software Services** which are related to sharable Process templates?

Technology also has an impact on Business Process modeling. New Business Process Design, automation and monitoring tools have emerged, like workflow engines and rule engines, which modify the way IS applications are designed and built. In a Service Oriented Architecture, there can be a direct link between a basic Function in a Business Process and the software service that will automate it, which is not the case in the monolithic application landscape we usually find today.

An IT solution can impose constraints on business, even if IT opportunities can help to improve and change the Business. The challenge is to design IT solutions as adaptable tools to support Business changes. Process modeling helps to articulate both.

## 2.4 Where is the potential for improvement?

### 2.4.1 What Processes to model?

Global Enterprises Operate several thousands of Business Processes. Companies we have interviewed have between 1500 and 2000 Processes. It seems almost impossible to model or automate all of them. Hence, an Enterprise has to decide which Business Process to optimize first.
Some priorities could be:
- Focusing on where the value is: identifying areas of improvement and development of the business (refer to strategy); for instance, an Enterprise might want to optimize its new product design process to reduce time to market
- Giving priority to processes where **automation** makes sense; for instance, when there are numerous instances of a Process and high volumes
- Avoiding immature processes which change a lot (business in development)
- Focusing on processes which are cross-functional, requiring synchronization of several actors
- Do not forget Transformation Processes. These Processes (like Project Management) are essential for the Enterprise to be able to adapt to its changing environment

The selection of processes to design or redesign should be approved by the senior management team of the Enterprise as impactful changes in business Operations will occur.

Business Process design is an opportunity for an Enterprise to develop new and innovative ways of operating. Many efforts to optimize existing Processes are constrained by the current way of operating. Incremental and continuous improvement of Business processes is virtuous behavior, but Process designers should also be free to completely rethink the way the Enterprise creates value for its Customer.

### 2.4.2 Defining what to improve

Business Process re-engineering gives some hints on what to improve in the organization:
- Focus on the future Processes and not present Operations
- Embrace the Process end to end, from the triggering Business event to the final deliverable, even if different Business Units or different Enterprises are involved. Process description must not be limited to the Enterprise perimeter or to a Business Unit of the Enterprise. Include relations with partners, Customers, providers... (EDI, JIT…)
- Automate where possible
- Reduce the number of Actors
- Favor parallel Functions if possible
- Prefer several variants rather than one complex process
- Reduce and group controls

# 3 Concepts for describing a Business Process

In this chapter, we will introduce a small set of concepts and definitions which are useful for understanding what a Process is and how to design it. We will be referring to the CEISAR Enterprise Model. The three dimensions of this Model are:

- The split between the **real world** and the description we make of the real world, called the **Model**
- The split between the **Operation** and the **Transformation** of the Enterprise. At any given time, an Enterprise Operates using a specific Operations Model. This Model describes how the Enterprise Operates. Any project or activity aiming at improving or changing the way the Enterprises Operates is part of Transformation, not Operations
- The split between what is **specific** to an Organization Unit in the Enterprise, and what is **shared** between several Organization Units of the Enterprise

The concepts defined below are used to describe the current Operations of the Enterprise in an Operations Model.

## 3.1 Enterprise

An **Enterprise** is an economic agent which creates economic "**value**" for **Shareholders** and **Customers** under the **same responsibility:**
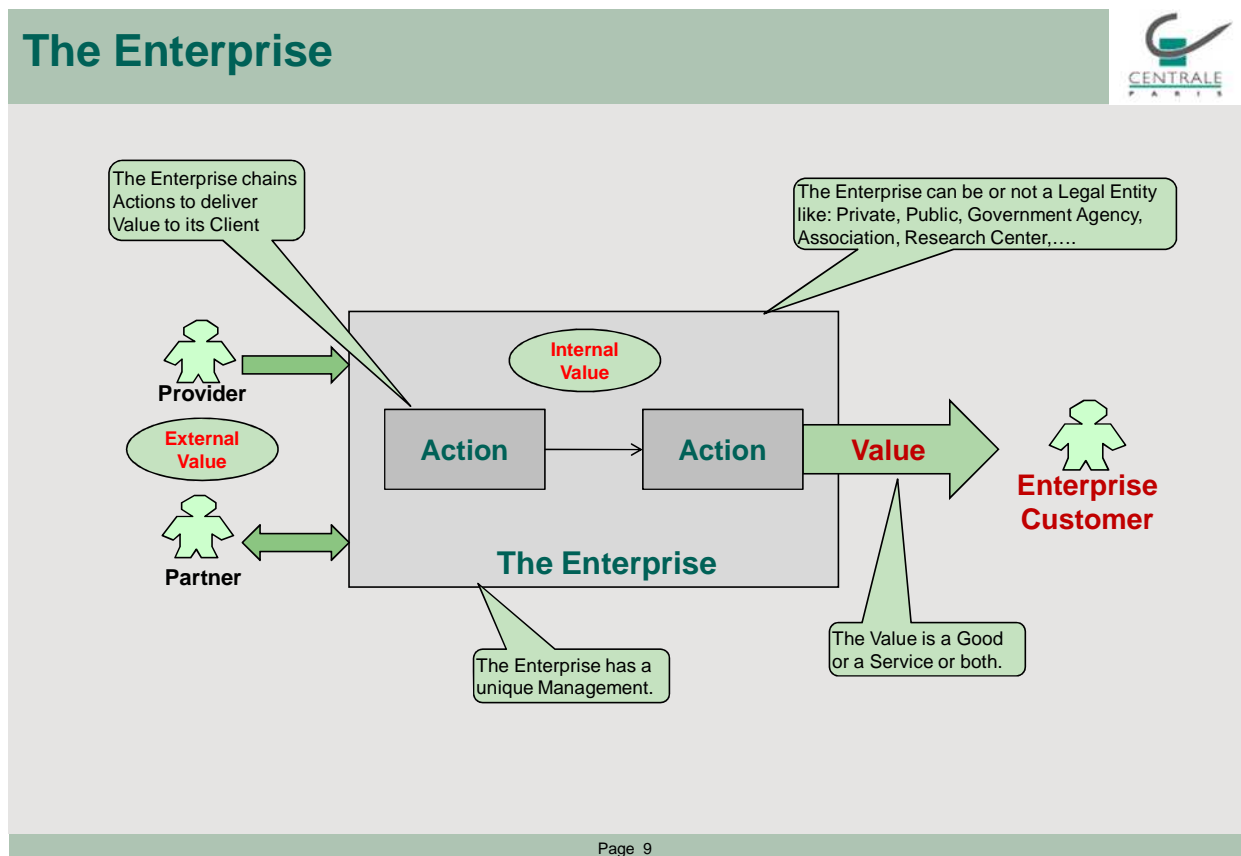


## 3.2 Actors

### 3.2.1 Business Actors

Business Processes involve **Business Actors** like Customers, Providers, Partners, Government Agencies… a **Person** or a **Legal Entity,** external to the Enterprise.
A **Business Actor Role** must be defined for each Business Actor.

### 3.2.2 Organization Actors

The **Organized Actors** execute Actions for the Enterprise. They can be a Person or Computers or an association of both, which makes 3 Organized Actors categories:

- The **Worker** using tools (not programmable) and free information
- The **Automate** using a programmable Server accessing Structured Information and Network facilities
- The **Assisted Worker**, combining a Person, a programmable Work Station accessing Structured Information and Network facilities
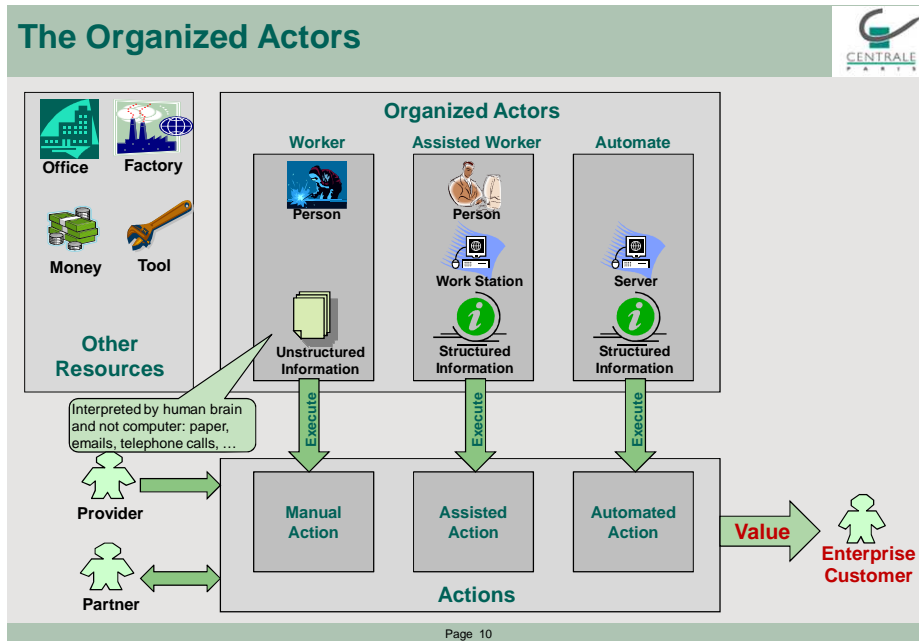


Their attributes are:

- Actor Entity: which worker or which Automate
- **Position**

The **Organization Role** is the role played by the Organization Actor when he is a Person, like an Assistant, Salesman, or Branch Director: generally it is the same as the **Position Role**.

The Organization Actors are organized into **Organization Units** (such as Direction, Department, Division, or Branch).
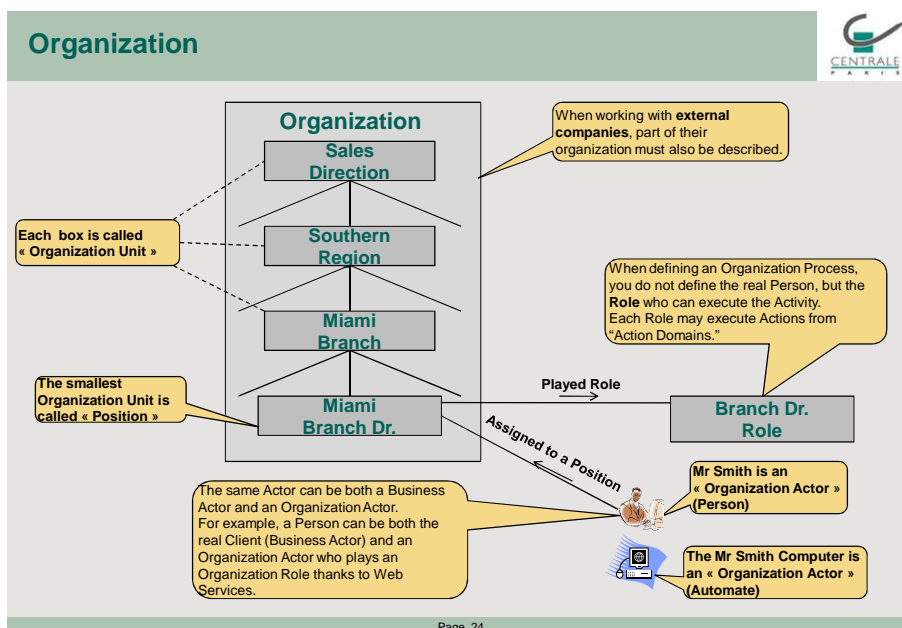
The same Person can be both a Business Actor and an Organization Actor: for example a Customer (Business Actor) which executes Actions (Organization Actor) through Internet.

## 3.3 Action

Describing how an Enterprise works means describing chains of Actions.
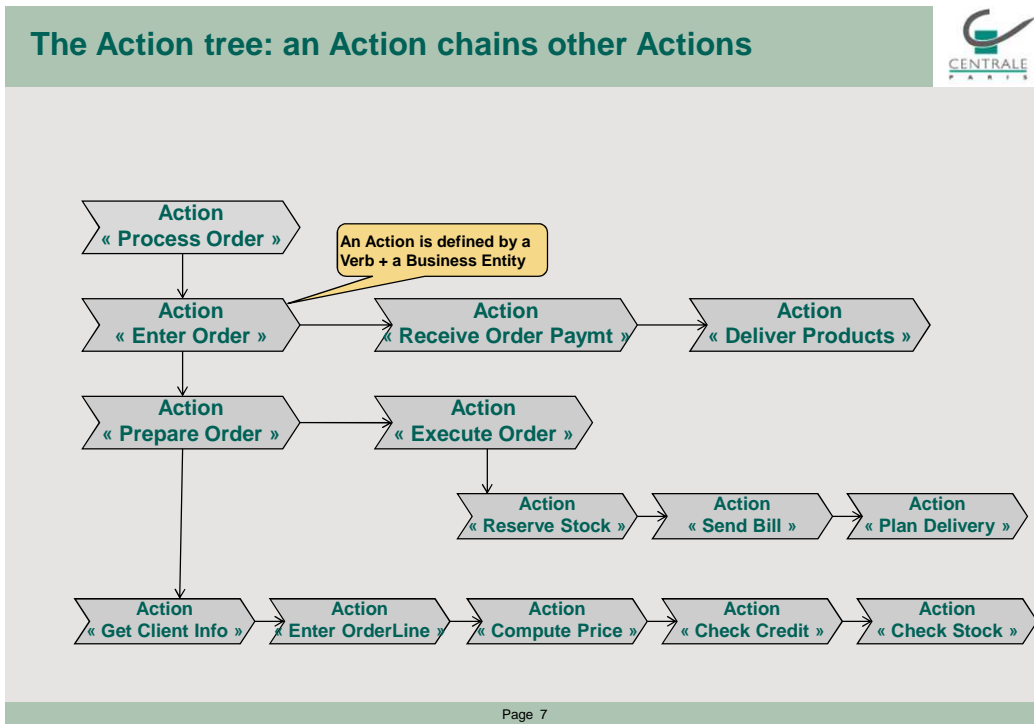


Many different words are used to name Actions. Many different Models are proposed. Our objective is to identify the underlying concepts and to define the smallest number of terms which are required to describe Business Actions and Processes. As a consequence, it should be possible to define a common language which allows different enterprises to communicate and share experiences and solutions.
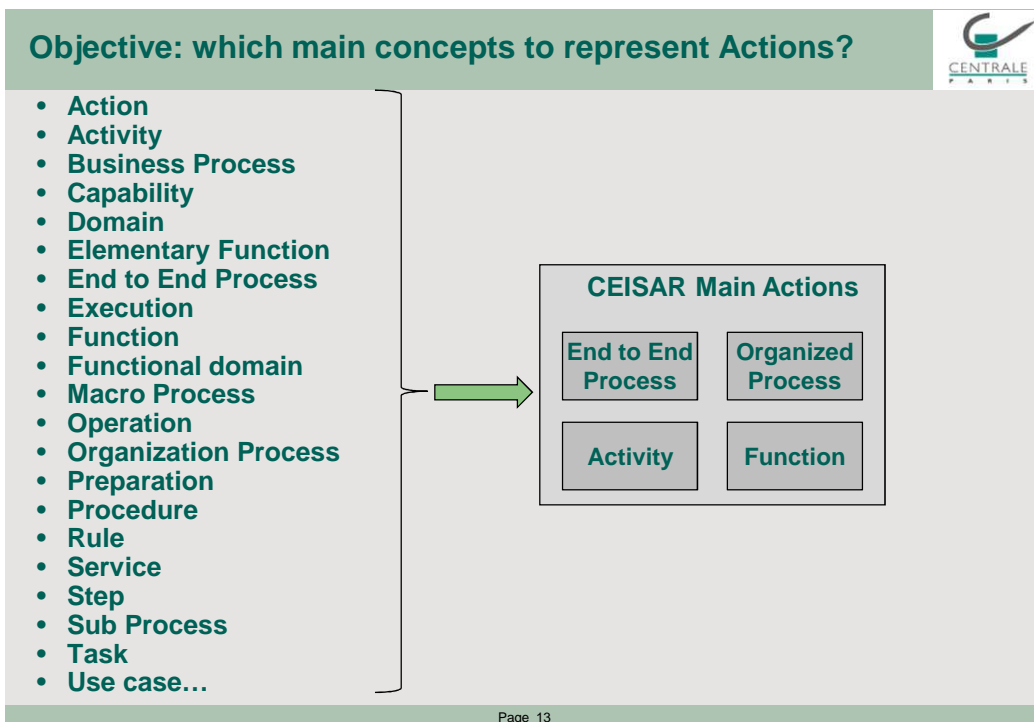


We propose the use of 4 concepts: **End to End Process, Organized Process, Activity, and Function**.

### 3.3.1  Main Enterprise Action = create Value for its Customers and Shareholders

The main objective of an Enterprise is to deliver value to its stakeholders. For example, "Deliver a Service to a Customer", "maintain a Product for a Customer" …

To offer this Value, the Enterprise must manage its Resources: "Hire a new Employee", "Get money from the Bank", "Rent offices", "Buy Computers"…

### 3.3.2  Action = Verb + Business Object

To create Value for its Customer, an Enterprise must act:
Each Action is described by a Verb + a Business Entity.

### 3.3.3  Each Action receives an Input and delivers an Output

From a systemic point of view, what is at the boundary of the action can be classified:
- Event  or Command (the input which starts the action)
- Information as Input, which describes useful data to process the action
- Information as output which describes the result of the action
- Memory or permanent information, which is already inside the action (as a system) and useful for processing it

Example
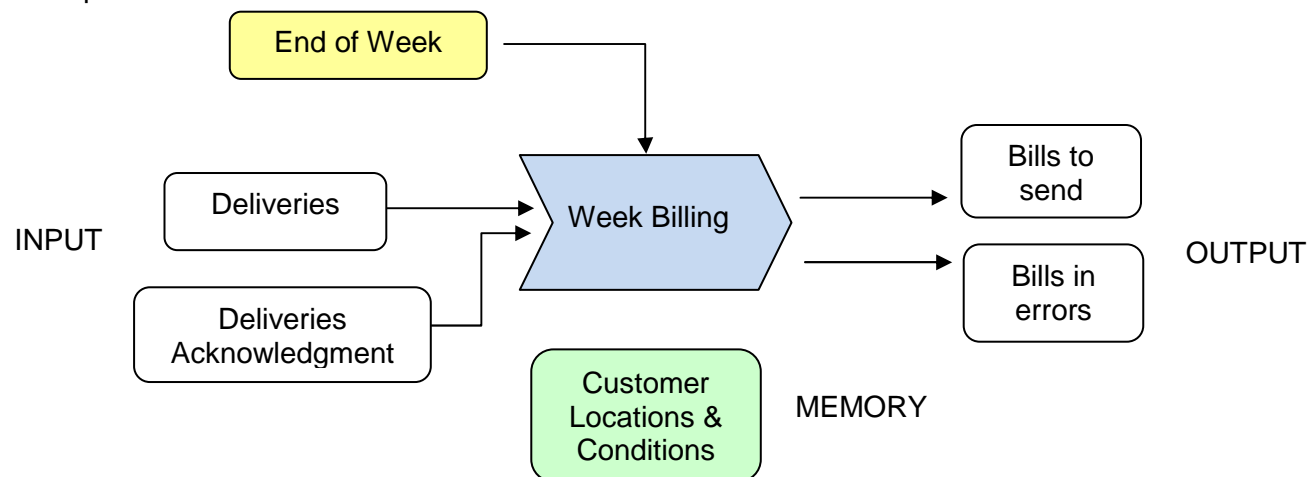


### 3.3.4  Each Action is triggered by an Event

An **Event** is defined as **independent** from the Action it triggers. It might carry no information or come with data. In this case, we call it a **"Message".** Messages are used when an Action calls another Action. Identifying the triggering event is mandatory.

### 3.3.5  Each Action is broken down into Actions

An Action is described as a *chain* of smaller Actions. Each of them is also described as a chain of actions: it represents a cascade of Actions. The Enterprise value chain is composed of a full tree of Actions, which are linked together in a logical manner.

Some Actions are big like "Plan resources", some are small like "Compute Price", "check data entry", "compute price", "send message to Customer"...

To help understand if an Action is big or small, Enterprises give names to levels of the tree: "Business Domain", "Macro Process", "Business Process", "Micro Process", "Activity", "Function", "Elementary Function" …

Example of a hierarchy of Actions:
- Business Domain "*Manage Customer*"
  - Business Process "*Create Customer*"
  - Business Process "*Evaluate Customer Risk*"
- Business Domain "*Manage Resources*"

- Business Domain "*Manage Employees*"
  - Business Process "*Welcome Employee*"
  - Business Process "*Evaluate Employee Performance*"
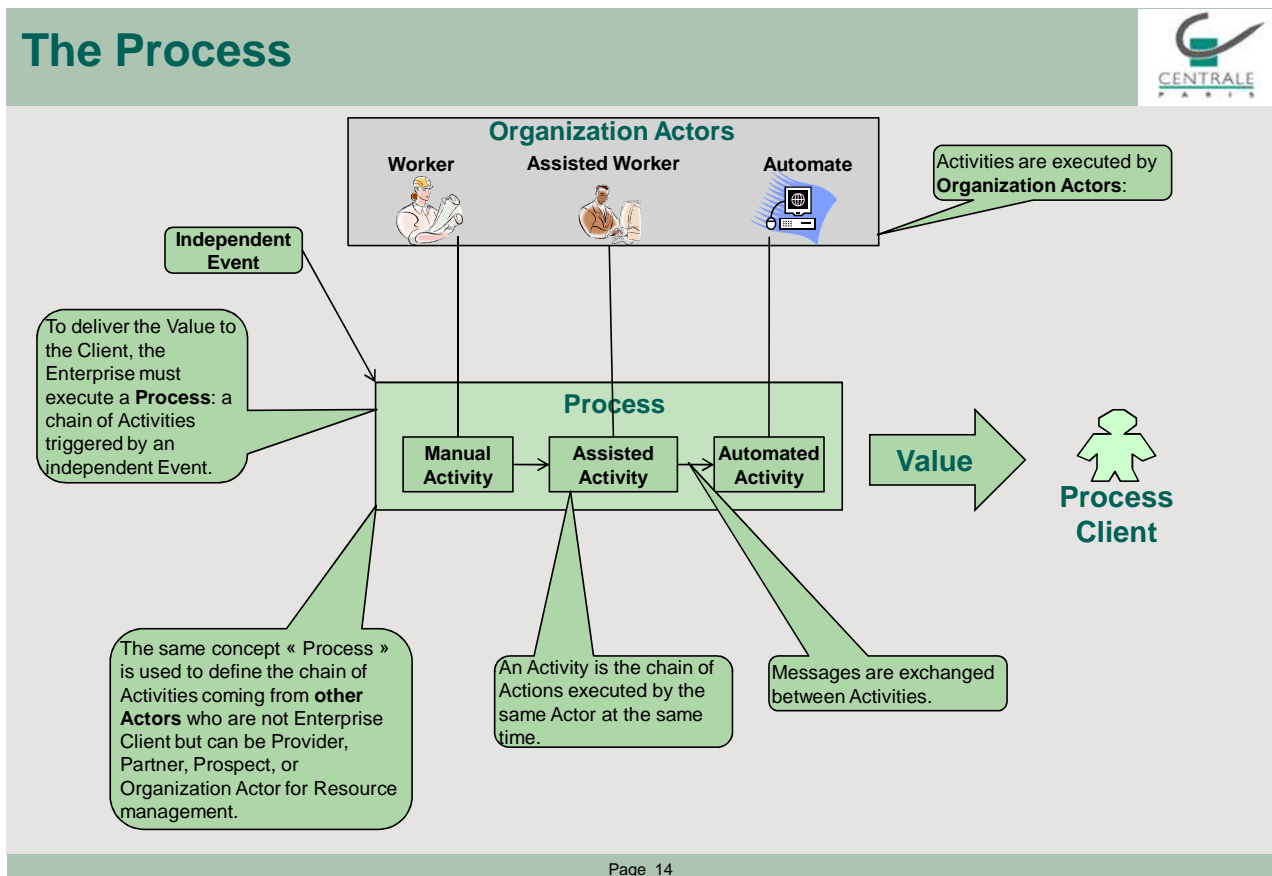- Business Domain "*Manage Hardware*"

# 3.4 Business Process

When we observe people working, we understand that they switch from one Action to another depending on Customer solicitation if they belong to a front office, or depending on work assignment if they belong to a back office. Sometimes, at the end of an Action, they send files and documents to another Actor who must continue the work.

The set of Actions which follow each other from a single **independent** Business event is called a **Business Process**. A Process generally applies to a main **Business Entity** and delivers its Output to a **Process Client**. A Business Process is the main Action of the Action tree.

To summarize: a **Business Process** is a co-coordinated suite of Actions triggered by an **independent** Event to bring **Value** to a **Process Client.**
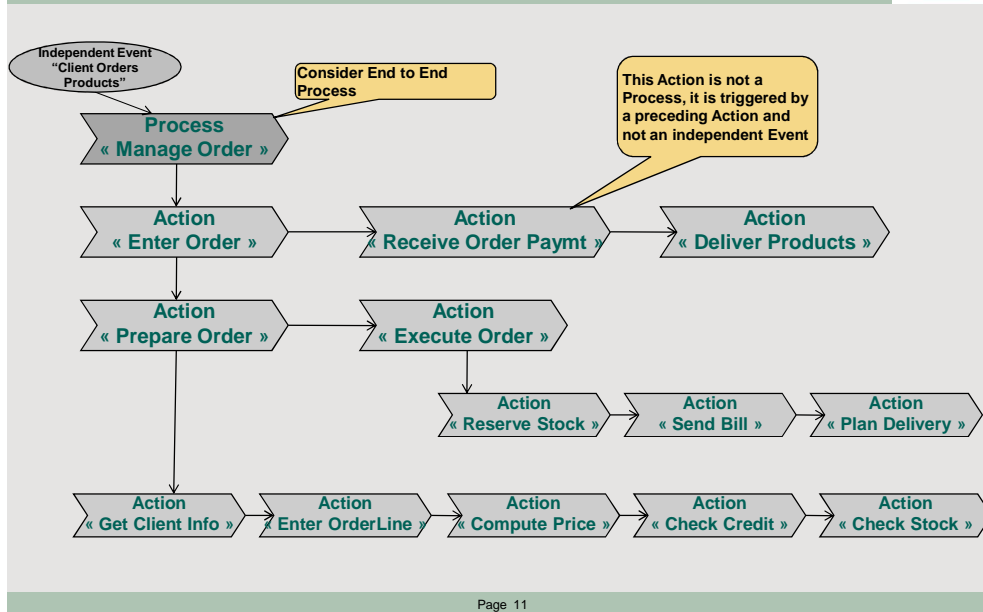
Ex: Subscribe a Contract is a Business Process triggered by an Event "a Customer asks for a Product". This Event is independent because it is generally not generated by a preceding Action.

It can be broken down into Actions like: "define suitable Product", "quote contract", "Enterprise Proposes Contract", "Client Accepts Contract", "Client pays Contract".

## The Process

**Organization Actors**

Worker    Assisted Worker    Automate

Activities are executed by **Organization Actors**:

**Independent Event**

To deliver the Value to the Client, the Enterprise must execute a **Process**: a chain of Activities triggered by an independent Event.

**Process**

| Manual Activity | Assisted Activity | Automated Activity |

**Value**

**Process Client**

The same concept « Process » is used to define the chain of Activities coming from **other Actors** who are not Enterprise Client but can be Provider, Partner, Prospect, or Organization Actor for Resource management.

An Activity is the chain of Actions executed by the same Actor at the same time.

Messages are exchanged between Activities.

Page 14

Here is a Business Process example:

**Business Process example**

Independent Event "Client Orders Products"

Process « Manage Order »

Consider End to End Process

This Action is not a Process, it is triggered by a preceding Action and not an independent Event

Action « Enter Order »

Action « Receive Order Paymt »

Action « Deliver Products »

Action « Prepare Order »

Action « Execute Order »

Action « Reserve Stock »

Action « Send Bill »

Action « Plan Delivery »

Action « Get Client Info »

Action « Enter OrderLine »

Action « Compute Price »

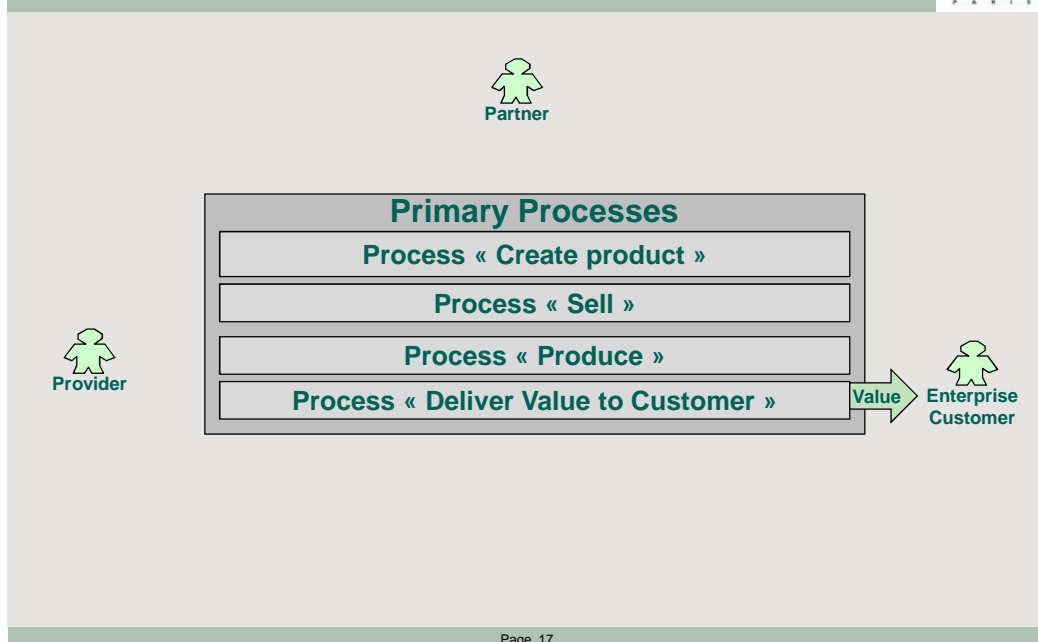Action « Check Credit »

Action « Check Stock »

Page 11

### 3.4.1 How to classify Operation Processes?

The Enterprise requires 3 categories of Processes:
- ➢ The **Primary Processes**. They define what must be done for the Enterprise Client:
  - **Create or modify a Product Model**: generally by strategic marketing
  - **Sell** to Client: done by the Distribution Units (some Distribution channels may be Partners) and the Operational marketing who organizes advertising and marketing campaigns
  - **Produce Product instances**
  - Deliver the Value to the Customer: deliver **Goods** or deliver **Services** or both.



**The Primary Processes and the Business Actors**

Partner

**Primary Processes**

Process « Create product »

Process « Sell »

Process « Produce »

Process « Deliver Value to Customer »

Provider

Value

Enterprise Customer

Page 17

Service Enterprises have a key characteristic: the processes "Produce" and "Deliver Value" are the same because Services cannot be stored in inventories. They represent 70% of all added value in the developed countries, and this proportion is still increasing.



The Service Enterprises (70% of the Activity)

When the Product is a Service, the Processes « Produce » and « Deliver Value » are the same.

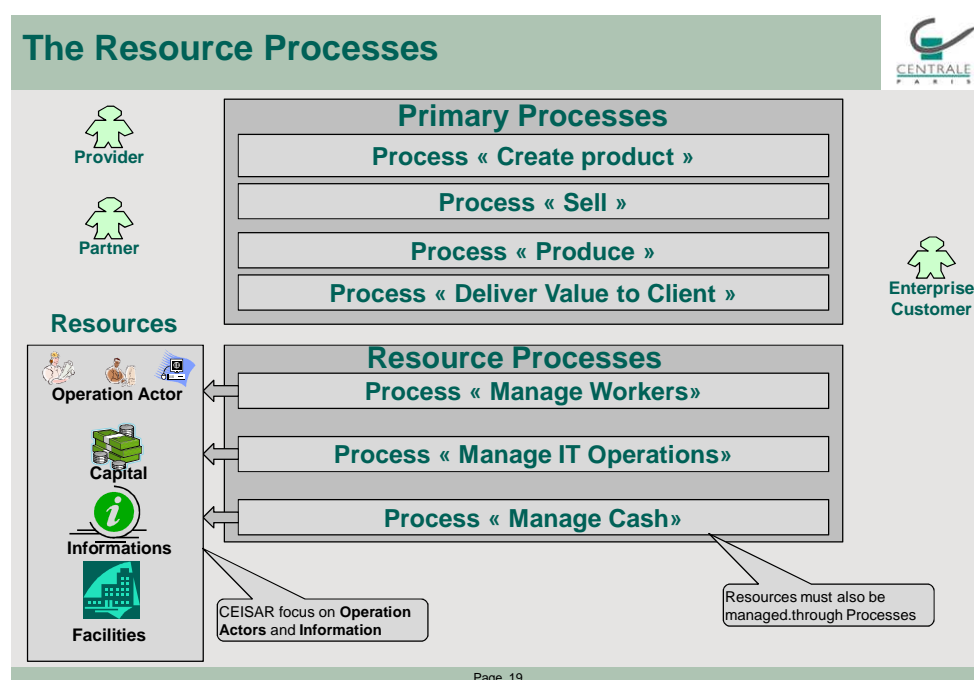| SERVICE Enterprise | Product | Sell | Deliver Product |
|---|---|---|---|
| Bank | Loan Product | Loan Contract | Transfer Money |
| Insurance | Car Insurance Product | Subscribe a Policy | Damage payment if Claim |
| Telecom | Telecom Offer | Open a Line | Transfer voice, data |
| Transport | Flight | Buy a Ticket | Transport the Person |

Page 18

## ➤ The Resources Processes

The Resource Processes, sometimes called "Support Processes", enable management of Resources: Employees, Facilities, Automates, Software, Information, Cash...
**Operate** an IT Operation Center comprises a set of Processes like: manage exceptions, tune, report quality level.
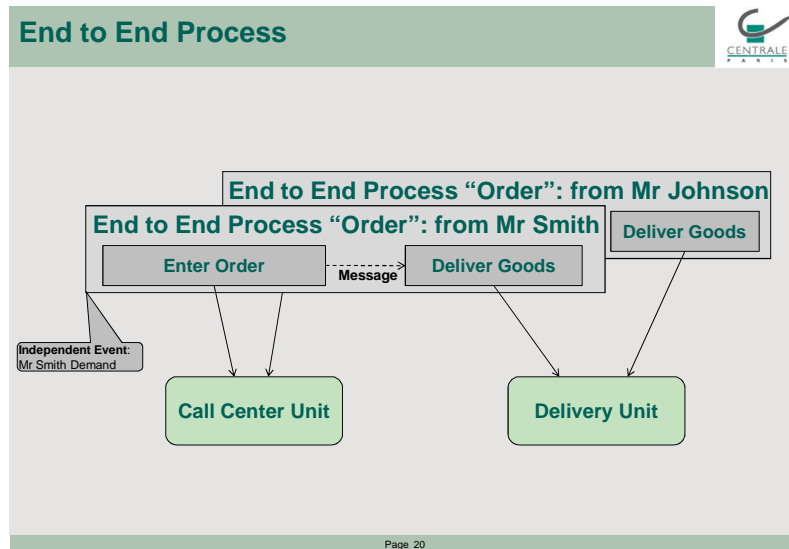
Surprisingly, when Enterprises formalize their Processes, they discover that Resource Processes are more numerous than Primary Processes!



The Resource Processes

Provider

Partner

Resources

**Primary Processes**
Process « Create product »
Process « Sell »
Process « Produce »
Process « Deliver Value to Client »

Enterprise Customer

Operation Actor

Capital

Informations

Facilities

**Resource Processes**
Process « Manage Workers»
Process « Manage IT Operations»
Process « Manage Cash»

CEISAR focus on **Operation Actors** and **Information**

Resources must also be managed.through Processes

Page 19

**Optimization of these Resources** is an important goal of any Enterprise. It often implies that Enterprises must divide their End to End Process into parts called "**Organized Processes**".

Let's take an example:
The "Order "Process is described as a Process where the Client request triggers an Action "Enter order", which is followed by an Action "Deliver goods".



**End to End Process**

**End to End Process "Order": from Mr Johnson**
**End to End Process "Order": from Mr Smith**

| Enter Order | Message | Deliver Goods | Deliver Goods |

**Independent Event**:
Mr Smith Demand

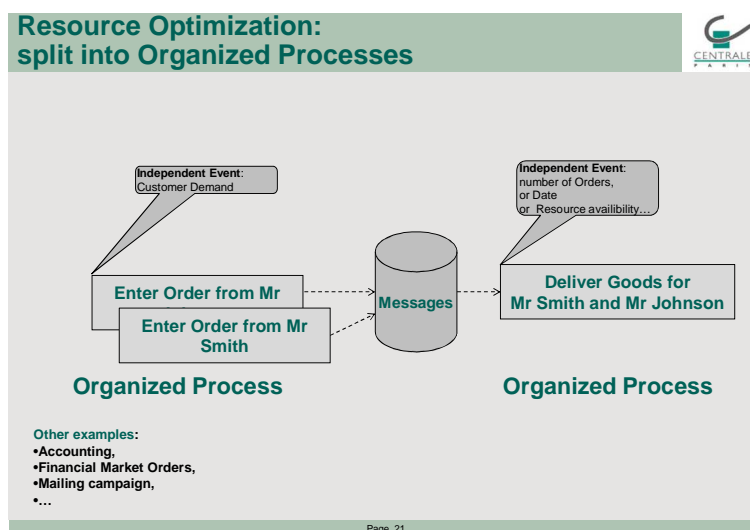**Call Center Unit**

**Delivery Unit**

Page 20

An Enterprise may decide to group deliveries to optimize Delivery costs.
It means that the End to End Process "Order" is broken down into 2 Organized Processes:
- One Organized Process "Enter order" which is triggered by the Client request
- One organization Process "Deliver several Goods" which is triggered by another event depending on the optimization: "one delivery a week" is based on time, "deliver when truck is full" is based on quantities of goods.
  This break-down allows resource optimization.
  Enterprises do it currently:
- Accounting can be done at night by an independent Process
  There is sometimes confusion between end to end Process (that which targets the end Client) and Organized Process: some Enterprises have lost this vision and merely focus on Organized Processes.
  This is why some Enterprises have defined **end to end Process Pilots** which have a cross-Unit responsibility.



**Resource Optimization:
split into Organized Processes**

**Independent Event**:
Customer Demand

**Independent Event**:
number of Orders,
or Date
or Resource availibility…

| Enter Order from Mr | | Messages | Deliver Goods for |
| Enter Order from Mr Smith | | | Mr Smith and Mr Johnson |

**Organized Process**             **Organized Process**

**Other examples**:
•Accounting,
•Financial Market Orders,
•Mailing campaign,
•…

Page 21

Over time, Process Organization is improved by automating more and more Activities.

For the same "**End to End Process**", several "**Organization Processes**" may exist, each one being a different assignment of Activities to Actors.
End to End Process defines **what** to do, while each Organization Process adds **who** does it.

➢ **The Management Processes**

Managers require information.
Aggregates must be computed and compared to budgets.
Aggregation is done according to several dimensions:
- Budget/real (with several successive versions)
- Time: instant (if stock) or period (if flow)
- Organization Units which represents territories (hierarchy)
- Measure: numbers, € or $, % ...
- Product (hierarchy)
- Nature: like accounting nature (with composed aggregates)
  Aggregates can be presented on manager reports according to presentation Models.

Managers may also need to analyze information on a non-regular basis. They require specific analysis tools to do so.
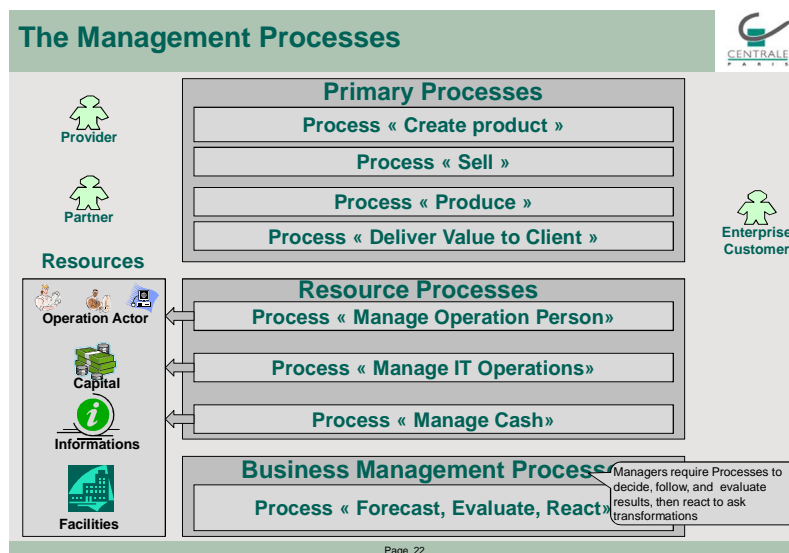


## 3.5 Function

A Process is broken down into **Functions**. Each Function may in turn call other Functions.
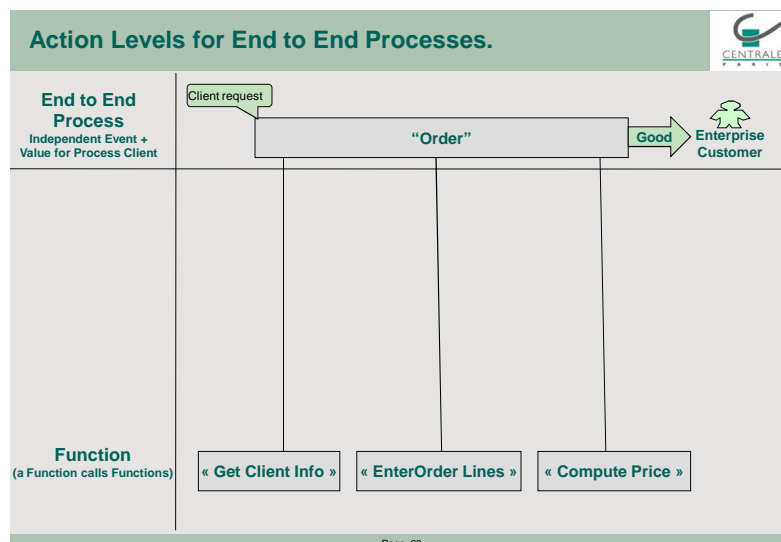
### 3.5.1 Competitive advantage

A Business Process is broken down into 2 parts:

- The Business Process **Invariants** which define the Functions of the Business Process which are the same for all competitors
- The Business Process **Specificities** which define what has been added by the Enterprise to customize the Business Process and to gain a **competitive advantage**

Business Functions can be classified as:

- **Invariant Business Functions**: the same for all Enterprises which use this Business Process. They can be specialized by Country.
- **Advantage Business Functions**: they are added by each Enterprise to offer more Business Value and obtain a competitive advantage
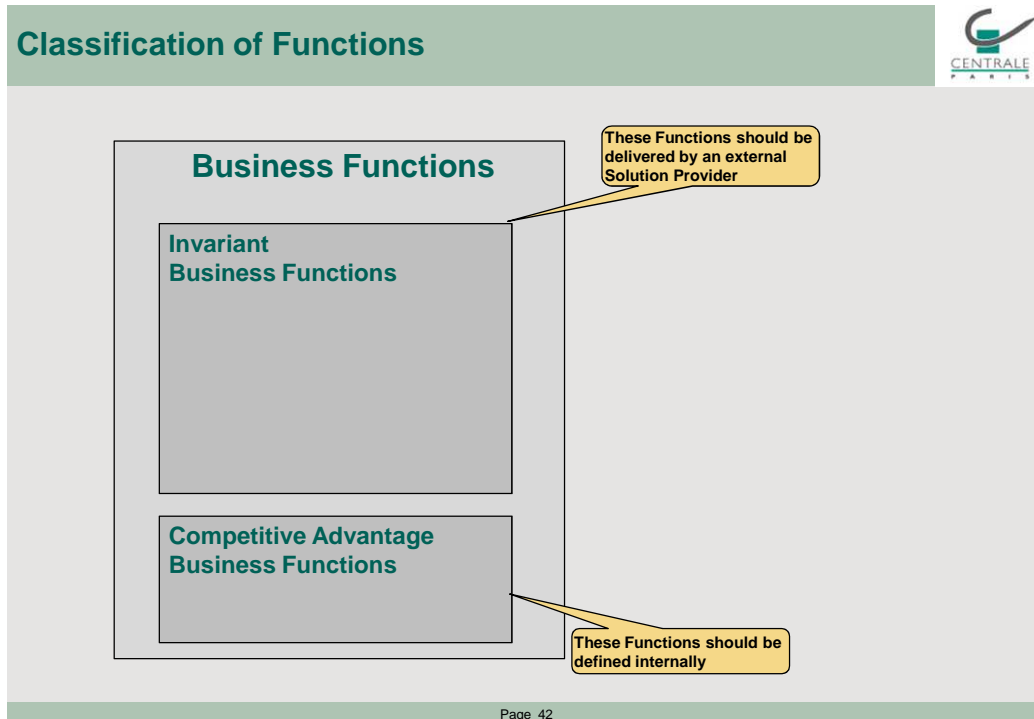


### 3.5.2 Core Business and Organization

Many Company evolutions focus on Organization and not on the core business. We may ask: with same Customers, products and contracts, is it possible to:

- Reduce Operating costs by **optimizing** processes?
- Propose to the **Customer** to do part of the work directly through **Internet?**
- **Outsource** part of the work?
- **Partner** with other companies who can play the role of Distributor or Producer?

The **time** required to implement these changes is mainly linked to **IT changes** and **user training**.
Is it possible to implement all that changes with very light Software modifications?

When we describe an Enterprise we propose describing 2 worlds:

- Business Core: what the Enterprise must do
    - Business Actors like Customers
    - products
    - contracts
    - **End to end Processes**; the suite of Actions independently from who does what
- Organization: how the Enterprise is organized to do it
    - Organization Actors
    - **Organized Process**

**Business Actors** are always present even if organization changes: Customers, prospects, providers, public authorities, business professionals.

**Organization Actors** depend on the chosen organization: employees, external consultants, outsourced people, automates, partners who do part of the work.

The same Person can be **both** a Business Actor and an Organization Actor. For example an eBay user is a Customer (buyer or seller) and an Organization Actor with ID and password because he/she uses the eBay IT System. It represents a new Model where eBay Employees just define game rules, processes and IT systems, and let its Customers become the Organization Actors.

This trend will spread to more classical service industries like: Finance, distribution, health, transport, knowledge…

But it will take time and we still have to analyze Enterprises by splitting Business and Organization.

Here is an example of two alternative Organized Processes for a Business Process:

## Two Organized Processes for one Business Process "Order"

| Organized Process 1 | | Business Process | Organized Process 2 | |
|---|---|---|---|---|
| **Organization Actors** | **Org.Functions** | **Bus.Functions** | **Org.Functions** | **Organization Actors** |
| Customer | | Capture Demand | | Salesman |
| | | | Warn Back Office | Salesman |
| Automate | | Check Client Credit | | BO risk management |
| | | | Warn stock mgt | Automate |
| Automate | | Control product availibility | | Stock Mgt |
| | | | Warn salesman | Salesman |
| Automate | | Compute Price | | Salesman |
| | | | Authorize Order | Sales mgr |
| Automate | | Inform Customer | | Salesman |
| Customer | | Agree on conditions | | Customer |
| Delivery Dept | | Deliver | | Delivery Dept |
| | | | Acknowledge delivery | Customer |
| Automate | | Bill statement | | Bill Dept |
| Automate | | Receive payment | | Acctg Dept |
| Automate | | Generate Bus.Intell.element | | Automate |

Page 44

When an industry is **new**, then the competitive advantage will come from the Business Side: a new product, or a new Business Process (like eBay or Amazon), and organization will be optimized later on.

When an Industry is **mature**, the enterprise prevents the others from copying their Business competitive advantage by legal Protection, or confidentiality, or continuous innovation. But if innovation is slow, competitors have time to import good ideas coming from others: competitors will copy you sooner or later. Competitive advantage is not only concentrated on the Business part, but also on the Organization part: how to define more effective Organized Processes?

For example, the Oil industry: formerly the only genuine Business competitive advantage resided in scientific knowledge to optimize the probability of finding oil in a given spot. Yet most of the competitive advantage today comes from organization and processes: how to explore, produce, distribute.
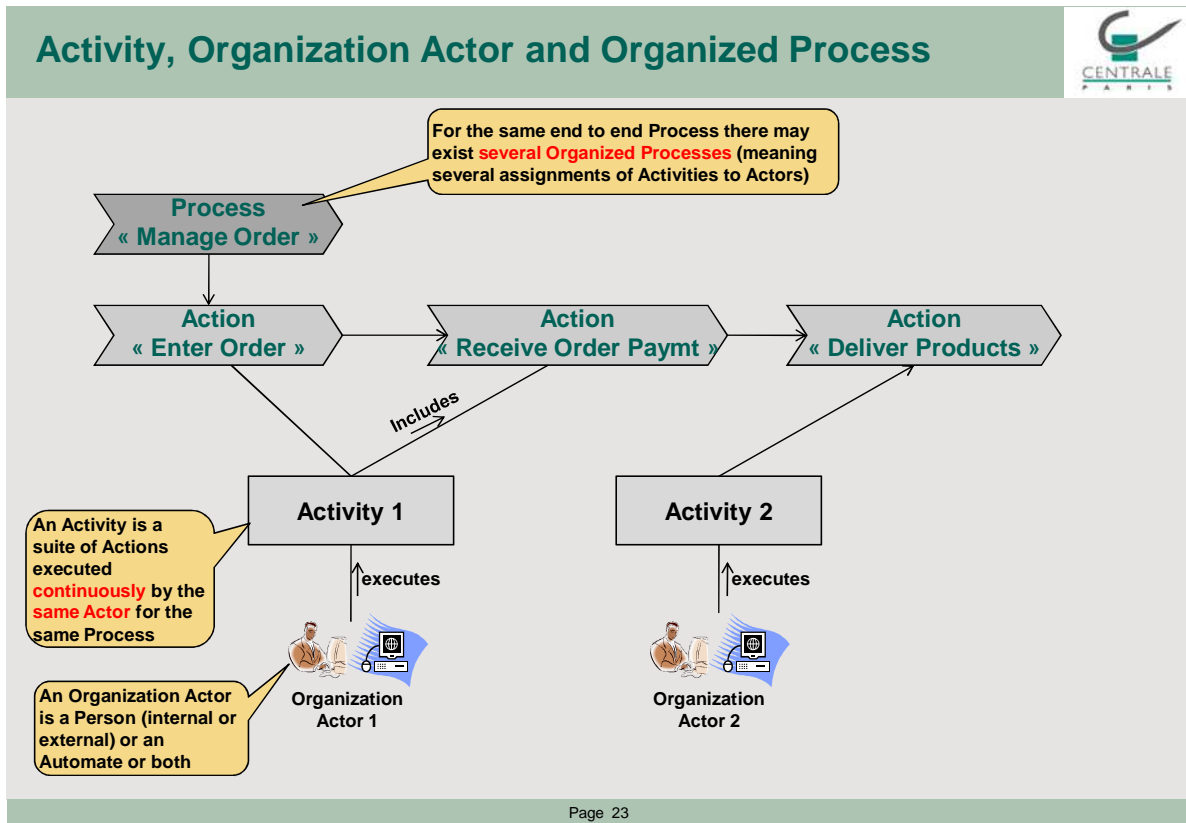
# 3.6 Activity

All Actions cannot always be executed at the same time. All Actions cannot always be executed by the same Organization Actor.

The suite of consecutive Actions belonging to the same Process, executed by the same Organization Actor in a continuous period is called an **Activity.** An Activity is schedulable**.**

Assignment of Activities to organization Actors can be different from one Enterprise to the other, or can evolve by successive versions in the same Enterprise. The same Action tree supports different Organization. But the Action tree does not change: it is independent from Organization.

By extension, the same **end to end Process** may be organized into many **Organized Processes**.
For each Business Process, there may exist different scenarios of assignments of Activities to Actors. Each scenario can generate one or several **Organized Processes**. Each Organized Process is broken down into Activities.



### 3.6.1  Organization Functions

To implement a new Organization, it is not sufficient to assign Activities to Organization Actors, it is also useful to add some Functions like "Authorize current user", "automatically assign next Activity to an Organization Actor", "give me the next priority Action to execute" …
These are called "**Organization Functions**".

### 3.6.2  Independence Business/Organization

Life duration for Software is 10 to 20 years.
Life duration for Organization is getting shorter and shorter: 2 to 5 years.
Is it possible to analyze Processes so that the deduced software adapts to different Organizations?
If Organization changes often, **grouping** of Actions into Activities, and **assignment** of Activities to Organization Actors should not require software modification which is a heavy Process, but should be done directly by users in charge of Organization: it requires **Workflow** tools and definition of who the
**Process Instance Owner** is (important if the end to end Process crosses different Organization units).
*Remark: do not confuse Process Instance Owner who executes the process and Process Model Owner who defines the Process.*

## 3.7 Summary

| Concept | Core Business | Organization |
|---|---|---|
| Business Process | End to end Process (Primary, Resource or Management Process) | Organized Process |
| Activity | Not Applicable | Activity |
| Function | (Invariant or Advantage) Business Function | Organization Function |
| Entity | Business Entity | Organization Entity |
| Actor | Business Actor | Organization Actor |



Action Levels for Organized Processes.
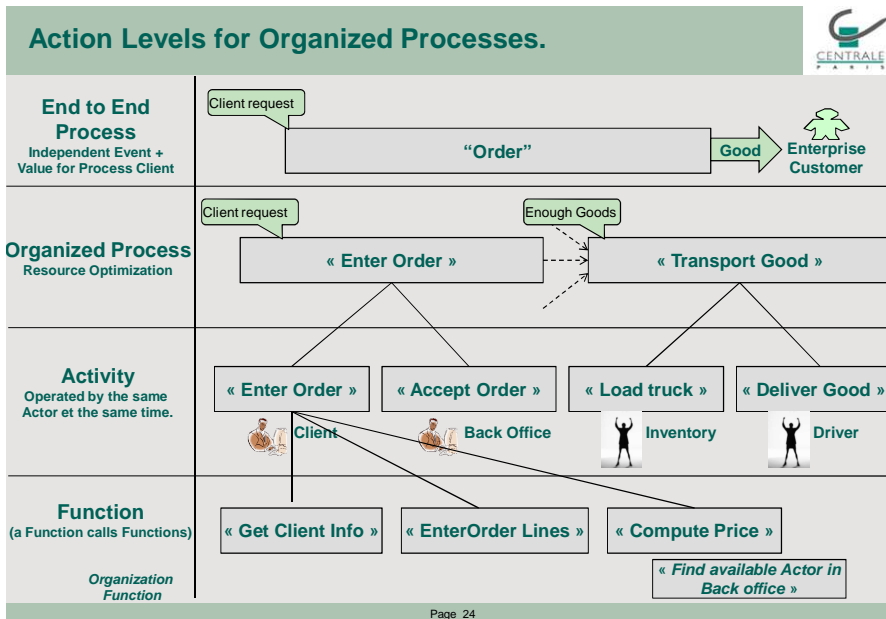
## 3.8 The "maps"

When describing a full Enterprise System, many objects (Processes, Functions, Activities, Entities, Software…) are identified and need to be classified and grouped. Classical Enterprise Architecture frameworks provide taxonomies to classify objects into architectural descriptions, often called "maps". These maps are not absolutely indispensable per se, but can be useful to communicate a clean description of the system.



Main Concepts may be grouped into "Maps"
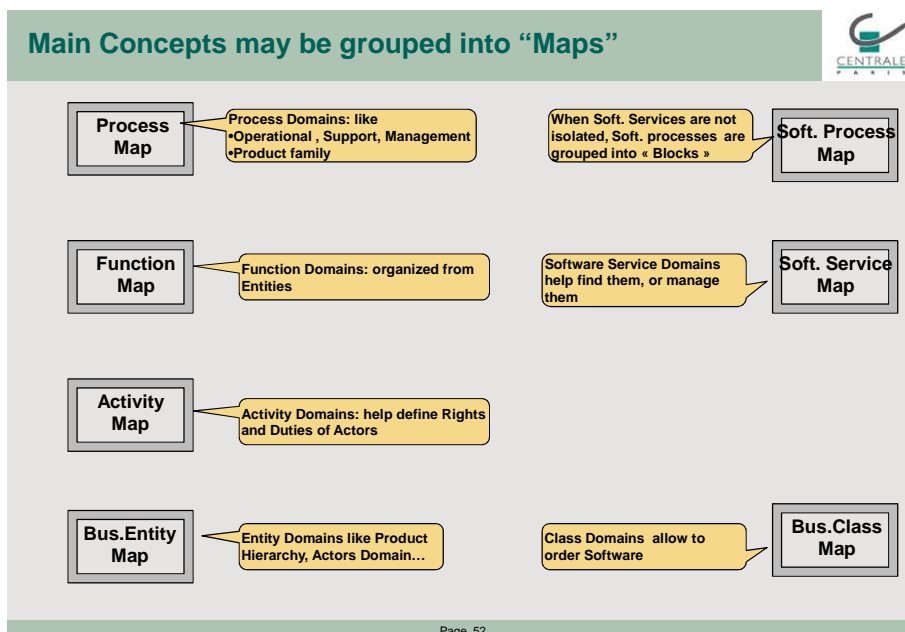
# 4 How to define a new Process Model?

The scope of a Business Process Modeling project can vary:

- If the scope is the **whole Enterprise**, the objective is to list and map all the Processes of the Enterprise. This approach usually aims at obtaining a broad picture of all the Processes, but not necessarily a detailed description of each Process
- If the scope is **one specific Business Process**, the aim is usually to document or Transform the Process, which requires a detailed analysis and design of the Process
- If the scope is intermediary, like **a whole Business Domain**, the Process Modeling project will require a mix of the two approaches described above with the following deliverables:
  - A map of the Business Processes in this Business Domain and
  - A detailed description of one or several Processes in this Business Domain

We will first describe how to identify a list of Processes across a broad scope (like a Business Domain or an Enterprise). Then we will give some tips on how to design a single new Process.

## 4.1 How to identify a comprehensive list of Processes?

### 4.1.1 Drawing a Business Process Map of the Enterprise

The first step is to design a general map of all the processes of the Domain or the Enterprise in one page. This is an essential communication tool to align all the business actors (including senior managers) on a common understanding of the Operations of the Enterprise.

To do so, you can split the Enterprise into its main activities. One available option is to use the three categories of Processes we described in § 3.4 (Primary, Resource and Management Processes).

You might consider using Michael Porter's value chain as a starting point:
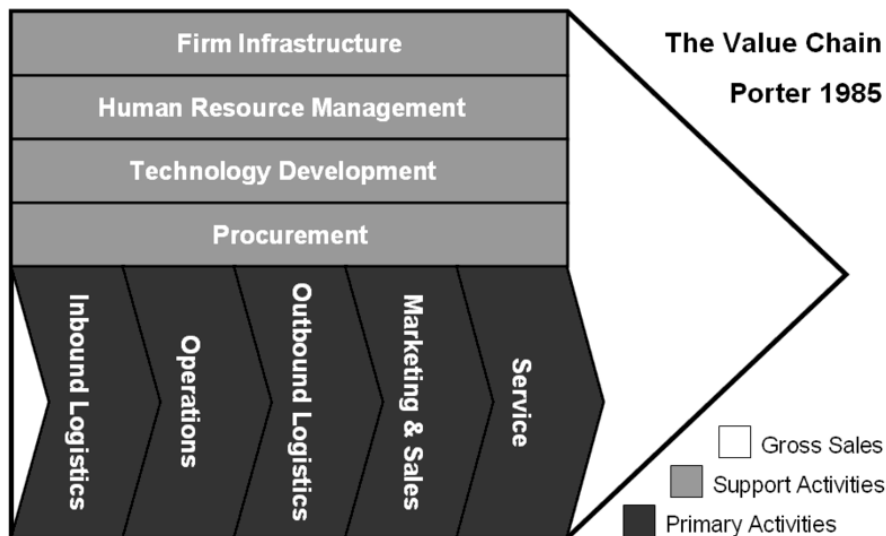


For Primary Processes, a sensible approach is to focus on the Customer segments of the Enterprise and/or the portfolio of products and services. It remains the best way of identifying the external Customer of the primary processes and the value delivered to this Customer through the products and services of the Enterprise.

For Resource and Management Processes, some practitioners use the Organization chart as a starting point to identify the main Business Functions and Activities. However, one must be very careful and make sure that the Business processes are end to end and not linked to the Organization.

This map should exhibit all major Actions and Clients of the Enterprise. Basically, this will include all major Business Process Domains (1 or 2 levels) and the list of Processes in each Domain. It is important to be able to refer to this central map in every subsequent modeling of processes. Every process, whatever level of detail it is at, should be linked to the big picture.

We assume that the Core Business of the Enterprise is stable and well known. We don't need detailed objectives and strategy at this point. Strategy is about Transforming the Enterprise Model to improve performance. Hence it will be necessary to decide where to focus when we start a detailed process modeling exercise.

## 4.1.2 Identifying Business Processes through Business Entities

> **Checking definitions of Business Entities before defining Actions**

When describing the activities of an Enterprise, the natural tendency is to focus on Actions. But a clear definition of Business Entities (like Product, Client, Contract …) can help in defining Business Processes. Business Entities are the object on which Actions are executed. In fact, if you use a proper naming convention for your Actions, you should use a Verb + an Object (= a Business Entity). This requires a clear understanding of what each Business Entity is and a shared definition of each major Business Entity between business people and IT professionals in the Enterprise. Hence, when examining User Requirements, you should focus on the Business Entities which appear and define a simple glossary of terms to avoid any misunderstanding because of synonyms (different words, same meaning) or homonyms (same word, different meanings).

> **Defining Entity life cycle**

Once you have a list of clearly defined Business Entities for your Business Domain or Enterprise, you can start analyzing and developing their life cycle to derive main Business Processes. Inspired by object oriented modeling, this approach focuses on the different states of the Business Entity and the transitions between these states. Actions are the means to realize the transitions we just found. Last, Organization Actors are identified and Actions assigned to them. This approach has been described in the Praxeme Methodology (www.praxeme.org). This last step is not necessary when you are simply drawing a general map of Processes.

## 4.1.3 Arpege: Air France method to identify Processes

The aim of "**Arpege**" is to identify, structure and classify all the Processes of the Enterprise while ensuring:
- Independence with regard to the Organization
- Independence with regard to any specific point of view of subject matter experts (quality, IT, regulatory compliance, risk management …)
- A high level of invariance

**All Processes** should be identified in a top-down and comprehensive approach, starting from the products and services delivered by the main business lines of the Enterprise (Passenger Transport, Cargo Transport, Aeronautical engineering and maintenance). The method focuses on the **resources** which are required to deliver these products and services to the Customer. "Resources" refer to all the Business Entities brought into play in the Process

Four kinds of resources are involved:
- The Client of the Process (internal or external)
- The product or service delivered by the Process
- The other business resources consumed or used by the Process (Human Resources, IT Resources, Facilities …) called the Means
- The financial resources

For each type of resource, a standard and common life cycle is defined, taking into account several time horizons.

You achieve …

… **Independence with regard to the Organization** using two key guiding principles:

- Keep an end to end view of the Processes linked to the resources along the business cycle of the Enterprise; this leads to structuring the Processes according to a standard sequence (long -term, medium-term, short-term):
    - Defining Policy
    - Dimensioning
    - Organizing
    - Assigning
    - Executing
    - Following up/Checking

    These general terms can be customized depending on the Enterprise and the kind of resource.

- Managing the resources consistently over time means engaging in three main activities:
    - Acquiring and disposing of the resources
    - Using the resources
    - Maintaining the resources

    Note that all Processes are dealt with the same way (no difference between Primary Processes and Support Processes).

**… Independence with regard to any specific point of view of subject matter experts**:
Each Process has a predecessor and a successor in a logical sequence. Processes are not described and classified according to a taxonomy based on subjective criteria implied by the specific objective or point of view of subject matter experts (quality, IT, regulatory compliance, risk management…). These points of view can be taken into account in a dedicated view which can be derived from the general framework.

For instance, for the "Crew member" resource:

| Manage the "Crew member" resource | Defining policy | Dimensioning | Organizing | Assigning | Executing | Following-up/ Checking |
|---|---|---|---|---|---|---|
| **Acquire and dispose of the resources = recruitment and resignation** | Define the Recruitment policy (or other forms of acquisition: subcontracting...) | Identify the recruitment needs for the year | Plan the recruitment (mid-term, short-term) | Organize recruitment sessions | Recruit | Crew recruitment follow up |
| **Use the resources = Assign on flights** | Define the Crew management Policy (working conditions...) | Evaluate the number of crew members need to realize the global activity | Build rotations (generic schedules) | Plan flight crew (assignment of crew member on schedules) | Regulate Plan (flight execution, Crew risk management / absence/ ...) | Crew assignment follow up |
| **Maintain the resources = Train** (leave management is another 'Maintain' Process, which is not described here) | Define the Crew training policy | Evaluate the number of crew members to train or re-train | Build crew training plan | Organize training sessions | Train crew member | Crew training follow up |

**… High level of invariance:**
This approach lists all Processes of an Enterprise and gives a generic description of Process templates. The detailed description of a Process is defined later on, based on the Process template, which is specialized in Process variants according to local constraints like geography and technology.

## 4.2 How to design a Process?

When it comes to modeling a single Process in detail, two objectives can be envisaged:

- Describe an existing Process, and optionally improve it (continuous improvement). The objective here is to document your existing Enterprise Model for this Process as a baseline for a potential Transformation. The source Process Model should be detailed enough before Transforming it
- Design a brand new process or completely re-engineer an existing Process (breakthrough). Since most of what will be designed did not exist before or will be fully replaced, documenting the source Process Model (baseline) might not be necessary. It is preferable to focus on the target Process Model

### 4.2.1 Input to the Process Transformation project

The way the Enterprise is Operated today can be described in an Enterprise Model. Any action to improve the way the Enterprise performs requires a Transformation.
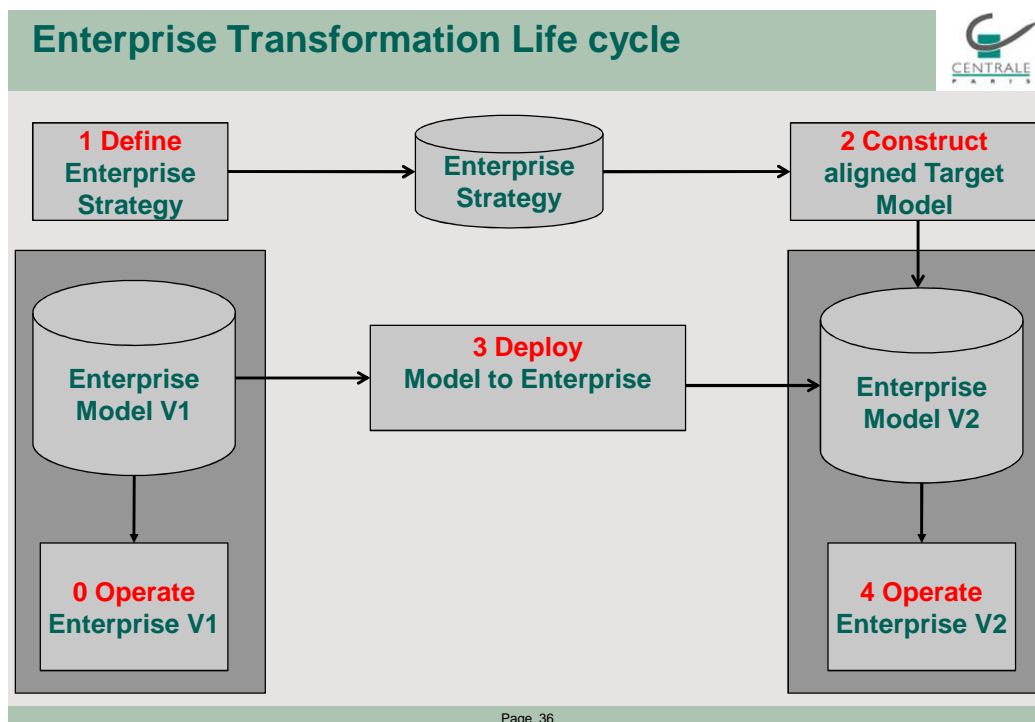


To be able to transform the Enterprise, you have to understand:

- **What the Enterprise strategy is:**
  The strategy of the Enterprise is made up of Business objectives (or functional requirements) and structural objectives (or non functional requirements, like "improve security", "improve ease of use", "scalability" …). The main input for Process modeling is Business objectives.
  In this white paper, we will consider that the Objectives and the Strategy are formalized, i.e. written and approved by the CEO. A strategic plan should be in place. All these elements are well known and understood by the Transformation project team.
- Secondly, if the project is about upgrading an existing Process, you need to know **How it Operates today** (current Enterprise Model)

The Process designers should also understand the stakes:

- Performance lag with regard to the competition
- Productivity issues
- New technologies opportunities for product/service innovation or improved productivity...

### 4.2.2 Build the proper skills in the modeling project team

Before starting a Process modeling project, the proper team has to be brought together. Depending on the objectives of the project (automation of the Process or just Business improvement with no IS impact), you will have to mobilize different skills.
The following competencies are required:

| Activity | Competency |
|---|---|
| To draw a Model | Analytical thinking (abstraction, synthesis) Rigor Ability to stand back and see the bigger picture Ability to identify analogies Team work |
| To describe an existing Process | Same + Business Knowledge Listening Knowledge of the culture of the Enterprise |
| To design a new Process | Same + Creative thinking |

### 4.2.3 Clarify the scope and baseline

The project team can only start to design the new Process once the project scope is well defined. Scoping includes defining what is in scope but also what is out of scope.

A general map showing all the Process Domains (see § 4.1) is very useful for precisely locating the Process to be designed or re-designed, and its interfaces to other Processes (which are not in scope).

The scope has to be clearly defined and approved by the decision makers before moving on to Process design.

The first step in the design is then to identify:
– Who the Process Client is (internal or external)
– What the value delivered by the Process to the Client is
– What Independent Event triggers the Process start
– Who the Business Actors involved are

The process designers must understand what the current situation of the organization is; here are some process issues that can be identified:
- Actors are working in vertical silos and discrepancies or gaps have appeared at the interface between Activities
- Suboptimal Processes have been created over time
  - Data is processed/keyed in many times; duplication of effort and lack of consistency
  - Many unnecessary exceptions or forks have been created; history has built many layers of structure without looking for simplification and consistency
- Poor control: some Activities are not controlled which results in poor overall performance of the Process (bad quality, lack of traceability, goods evasion)

The baseline has to be described with hard facts. It might include calculating costs and timing of Activities in the Process.

### 4.2.4 Break the Process down into its parts

The traditional approach can be summarized as follows:
1. People describe Organized Processes: who does what or who should do what
2. They break down Organized Processes into Activities
3. They break down Activities into Sub-Processes
4. They break down Sub-Processes into Functions

Remark: Business Processes and Business Entities are never formally defined.

## Classical Process Design



Page 58
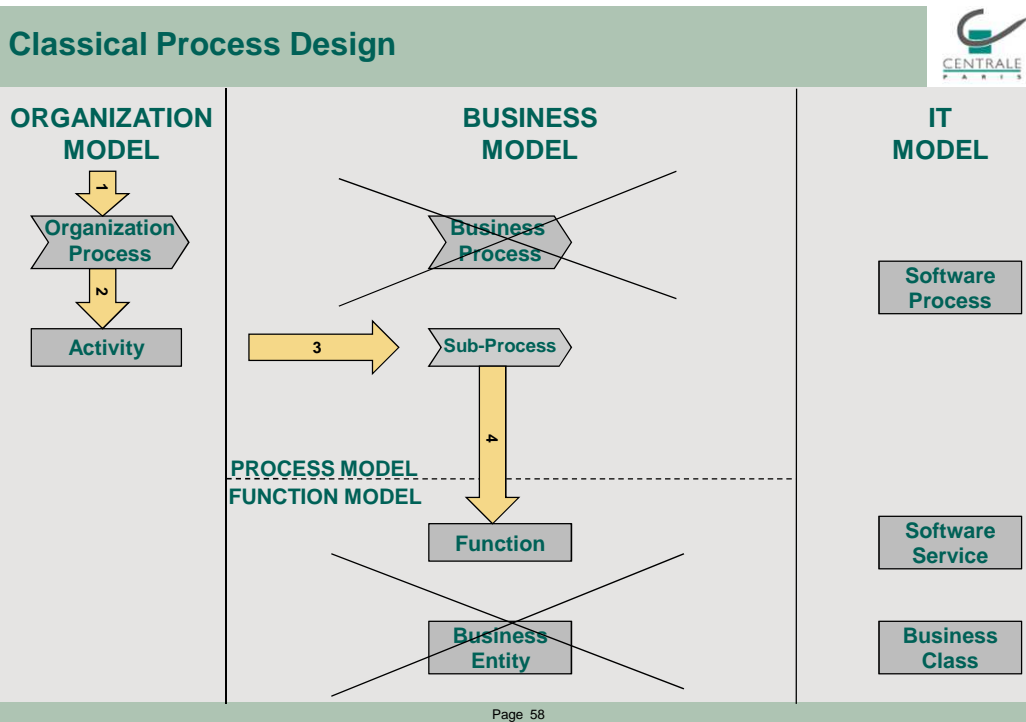
The proposed CEISAR approach is to:
1. Define main Business Entities
2. Assign a first set of essential Functions to the Business Entities (like create, update, delete)
3. Define the Business Processes which are in scope
4. Break them down into Functions (add Functions missing from step 2)
5. Define Activities as groups of Sub Processes executed by the same Actor
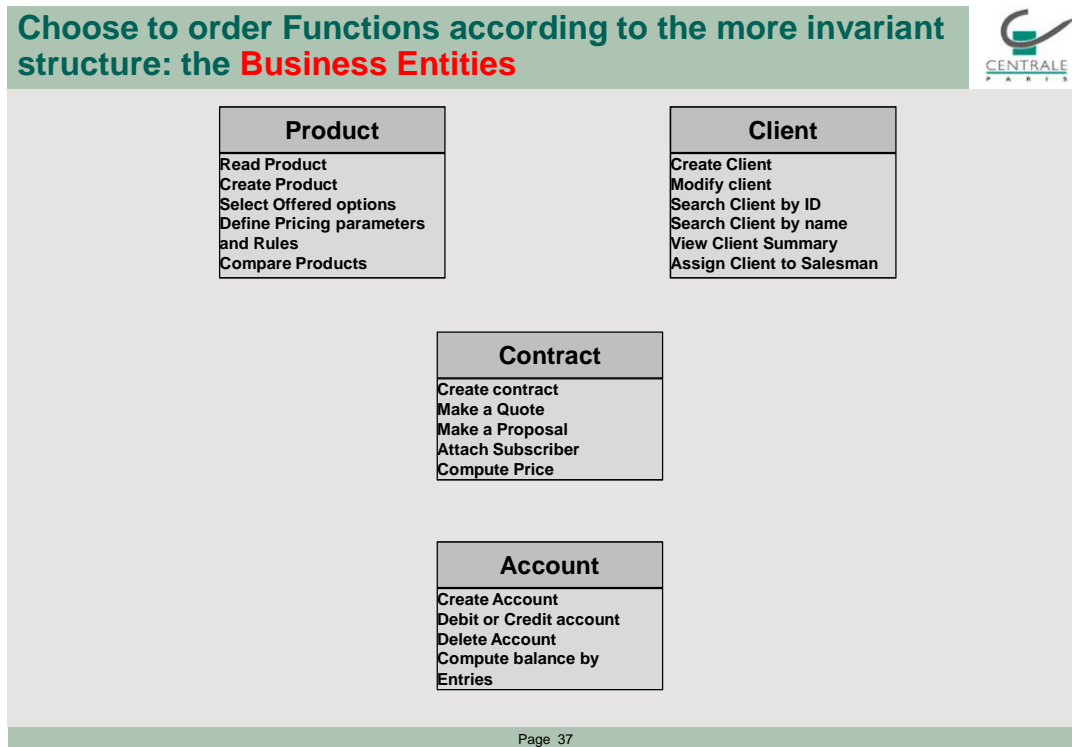6. Configure required Organized Processes

When defining an Action, a Verb + a noun should be used. When choosing a noun, you should try reusing the name of an existing Entity. Thus, you can find out if the Action has already been defined in the life cycle of the Entity. This provides more consistency in the Enterprise Model.

## CEISAR Process Design



Page 60
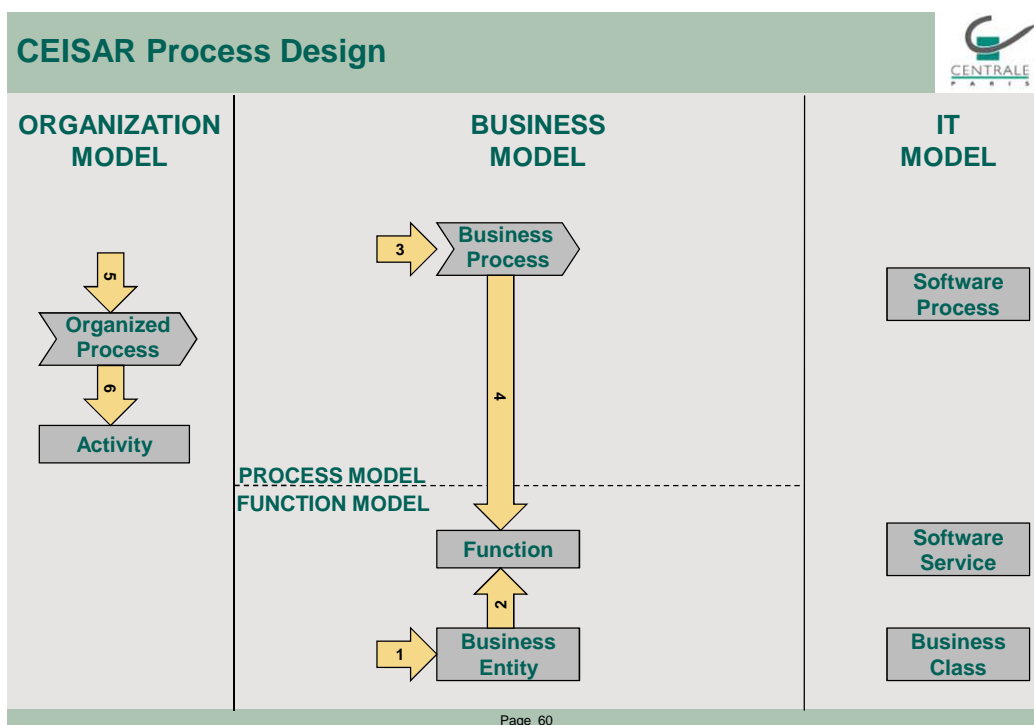
The advantages are:

- Defining Processes **after** Entities makes it easier to define **end to end Processes**. It you define the Business Entity "Order", you will find basic Functions like "Process Order" which includes Activities like "Enter Order", "Pay Order", "Deliver Ordered Products". Otherwise you may forget "Process Order" and just focus on independent visible Activities
- This approach allows you to start **building software** from Entities and Functions **without waiting** for the Process definition
- Business Entities (like Customer, Product, and Contract…) are very stable in time. This approach allows you to define a Business Process **independent from the Organization**
- Sorting Functions by Entities will enable discovering **reusable Functions** more easily.

Here are some examples of classical and generic Business Entities and the associated Processes:



**Choose to order Functions according to the more invariant structure: the Business Entities**

**Product**
Read Product
Create Product
Select Offered options
Define Pricing parameters and Rules
Compare Products

**Client**
Create Client
Modify client
Search Client by ID
Search Client by name
View Client Summary
Assign Client to Salesman

**Contract**
Create contract
Make a Quote
Make a Proposal
Attach Subscriber
Compute Price

**Account**
Create Account
Debit or Credit account
Delete Account
Compute balance by Entries

Page 37

### 4.2.5  Classify Functions into 3 categories

Think on three levels:

- **Invariant Business Functions**: required for all competitors in the same industry (should be delivered by a package provider)
- **Competitive Advantage Business Functions**: specific to each Enterprise to obtain a competitive advantage
- **Organization Functions**: only required for Organization purposes

### 4.2.6  Reuse Process Patterns

Some Processes resemble each other. For example, in an Insurance Company, "create a non-life Insurance Product" or "create a life Insurance Product" are different Processes, but they share common parts.

We suggest **isolating what is common in a Process Pattern** "Create an insurance product" and reusing this Pattern to describe the 2 real Processes. This reuse of Pattern can be used not only for Processes but also for **any Action** in the Action tree. Check that modeling tools may support this feature (called "inheritance" in the **Object Oriented** world). Verify if successive modifications to the Process Pattern are or not automatically applied to the derived Processes.

### 4.2.7  Define Processes with the right granularity

**What level of detail should we aim for?**
Process can be **grouped** into a hierarchy of **Process Domains**.
On the other side, Process can be **broken down** into recursive **Functions**. The difficulty is to define the right level of granularity:
- detailed enough to allow good design and identification of **sharable Functions**
- not too detailed, to save a lot of useless work

A simple rule of thumb is to always limit the level of detail to what is strictly necessary. Process automation will certainly require more detail than simply documenting a Process to communicate on the mission of an Organization Unit.

**How to define Actions homogeneously?**
When breaking down a Process into a tree of Actions, a practical rule is to limit each level to less than 10 Actions (so that it can be described on one single page).

On each level, all Actions should roughly have the same weight. You can estimate the weight of an Action by:
- the time required to execute the Action
- the cost of the Action, or
- the complexity of fully describing what is done (using a function point approach, for instance)

For example, if the Process you are modeling is 'Manage an IT Project', two Actions could be: 'Plan kick-off meeting' and 'Implement software'. They obviously cannot be at the same level as the first one is a lot 'lighter' than the second one.

### 4.2.8  For each Action define Interface: input and output

All information flows between Actions, as Actions are connected through information flow.
The total information flow must be consistent. No new data can be created from scratch inside the Action.  An action is a Transformer of information. It is possible to do simple consistency checks like:
- All the necessary inputs are available to the Action to produce the expected outputs
- The outputs of an Action are consistent with the inputs of the following Action

### 4.2.9  Isolate Business Rules

Business Rules are static constraints which limit the possible outputs of an Action. It might be useful to make some rules explicit as they are essential constraints to the Process.

When a Process is automated, frequently modified Actions (like "Compute Price") may require a Rule Engine (see CEISAR white paper on rule engine) which allows rule modifications without going through the heavy process of Software modification.
We suggest that when defining Processes, these rules are identified so that, at implementation time, a Rule Engine can be considered.

### 4.2.10 Define the Organization

Once all the necessary Actions have been defined, they have to be assigned to Organization Actors. Hence, you have to define the Organization structure through a hierarchy of **Organization Entities**, down to **Positions** (if necessary). Organization Units and Positions play **Roles**, which will have to be defined too. Positions are then filled by **Organization Actors**, which can be Internal Actors (like Employees or consultants) or External Actors (like Partners or Customers).

Organization — Page 24
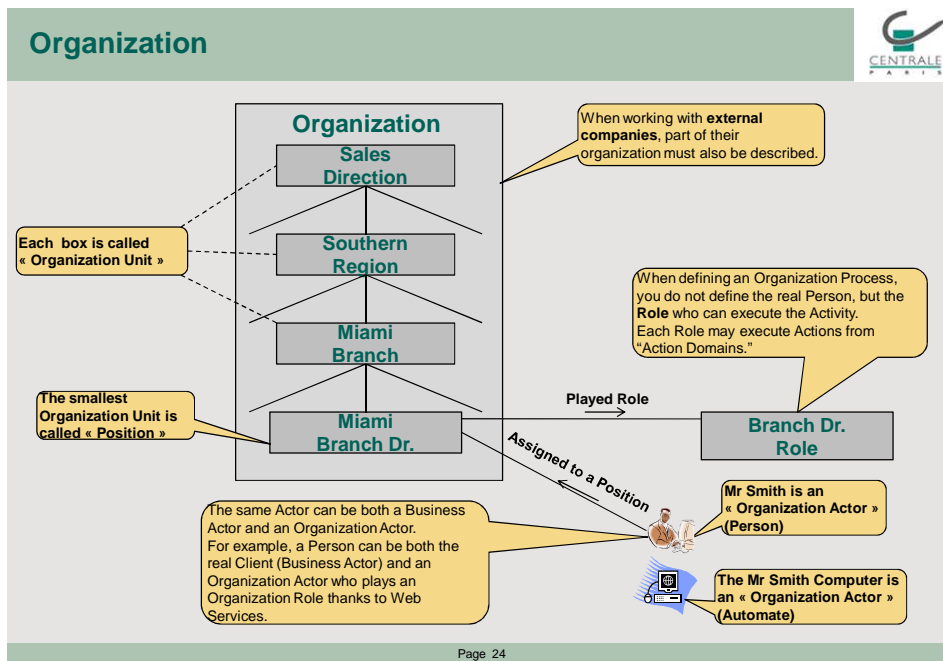
Assignment of Actions to Actors is done through **Activities**.
Group Actions which can be executed by same Actor in the same period and call them Activity.
Add Organization Actions like "Authorize", "Assign to another Actor", "Suspend Task"...

For automated Processes, assignment of Actors to Activities should be done independently from Process design thanks to a **workflow engine** used at run time.

Some principles you can use to design Organized Processes are:
- Keep the **number of Activities** for each Organized Process to a **minimum**. The classical division of labor is less and less used. For efficiency reasons, Actors are trained to be more and more versatile and adaptable.
- Move up as many Activities as possible **close to the requestor** (Customer) to reduce lead time and workload
- Expect to have (potentially) **many external partners** involved. The use of the Internet has enabled virtual teams working across geographies and companies.
- Consider **sourcing** when designing Activities. You might want to relocate some activities in the future (off-shoring).
- Defining **Roles** is essential. People can play different Roles in a Matrix Organization. People can be assigned to varying Roles over time. For instance, they can be assigned to a Project Team for the duration of a project (in parallel with other responsibilities).
- **Synchronize** Activities. Be careful with regard to latency times.

### 4.2.11 Introduce exceptions

During the first steps, the Process designer should focus on a standard and simplified Business scenario, which represents the majority of cases.
Introducing exceptions should be done separately and at a later stage so that discussions on exceptions do not disturb the project team during the design of the main part of the Process.

### 4.2.12 Validate Process definition and iterate

A set of objective criteria should be defined to validate that the Process has been properly designed. We can suggest some basic rules as a checklist:
- Naming standards
  - Actions should be named by Verb + noun
  - The verbs that can be used have to be chosen carefully (possibly in a predefined list). Do not use general terms like 'manage' or 'make'
  - Noun should refer to Business Entities, picked from a predefined list.

- Each level fits on an A4 sheet of paper (not more than 10 boxes)
- A complete Process Map should not have more than 3 to 5 levels
- For alignment to software : all Actions are related to an Entity class or a Process class

From the definition of the Process, some questions could arise, like:
- What is the competitive advantage?
- Did we forget to define patterns: do some Processes resemble each other?
- Did we forget to call already defined Actions instead of redefining them? (check through naming conventions)
- Are there many Organization Actions?
- Are there many Activities by Process?

Designing a Process is an iterative task. Do not be afraid to test your design with several independent colleagues and generate new versions from their feedback. It is also possible to refine the Process later on (for instance when you decide to automate it).

## 4.3 Example: the savings reallocation Process

As an example, we will discuss the Process of 'assignment of savings assets in a life insurance contract'.

The first three key questions are:
- **Who is the Process Client?**
  - The owner of the savings assets
- **What is the Process Value for the Process Client?**
  - Allow a Customer to reallocate his/her savings assets
- **What is the Independent Event?**
  - order sent by the Client

First, we have to identify invariant Actions.
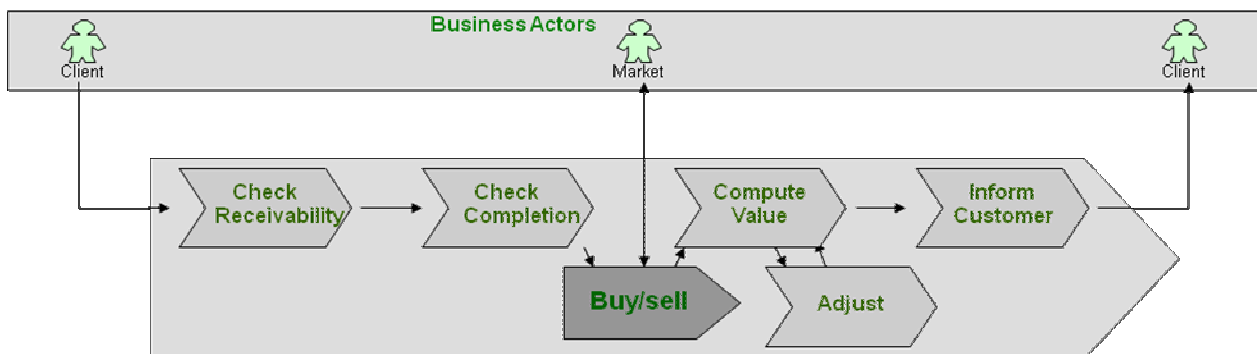


**The Invariant Functions are the same for all competitors:**
- **Check acceptability**: check that the Contract allows reallocation, that chosen movements are allowed
- **Check completion**; check that all required documents are present (decided by legal authorities)
- **Buy/sell**: stock orders on financial markets; this is not a Function like other boxes, but **another Process** which can be launched independently
- **Compute Value:** apply quotation values to the stock lines
- **Adjust**: balance quantities and prices
- **Inform Customer** of the result

Then we have to identify Business Advantage Actions.
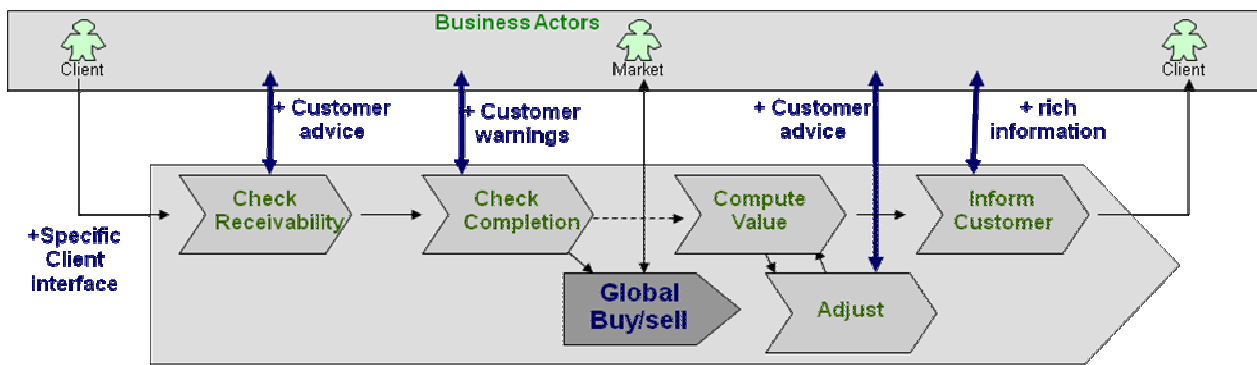
**The Business Advantage Actions: to deliver an added value to the Client:**
- **Specific Client Interface:** the Client may use an Internet application to describe his/her reallocation order
- **Customer Advice**: If receivability is negative, advice should be given to the Client to adapt his order
- **Customer Warnings**: messages can be sent to Client to explain, as soon as possible and several times, what is missing; complementary documents may be required by the Enterprise
- **Global Buys/Sells:** the different orders coming from different Processes are globally traded in another global process. The quotation applied to each order is the exact quotation at noon, even if the Global Process was executed at other times
- **Customer Advice**: as sale and buy do not exactly tally, contact Client to advise what to do
- **Rich Information**: just send Operation notice or new global portfolio or more

Finally, we have to assign resources in an Organized Process:
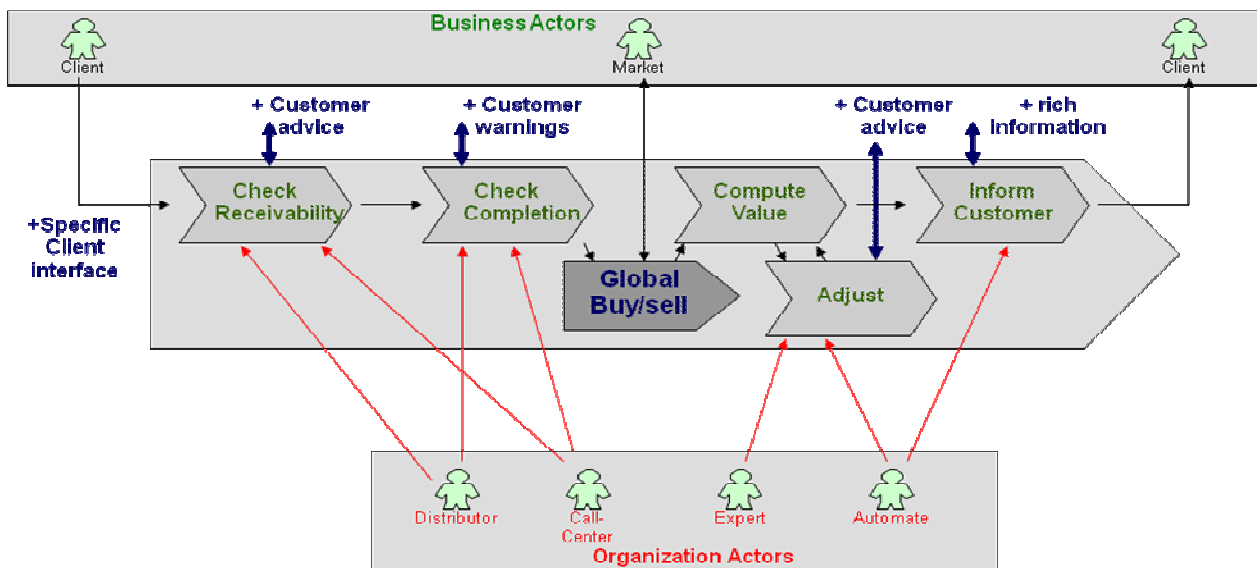


**Define who does what:**
- Define Organization Actors
- Define if Combinations of activities are allowed

Lastly, refine the Process:
- If required, define **several** Organized Processes for the same end to end Process
- Define contents of **Flows** between Actors and Functions
- **Introduce exceptions**: what if Controls are negative, if due date is not respected … and create complementary flows
- **Break down** Functions into smaller Functions

# 4.4 How to align Process design and Software?

In this chapter, we will consider how Process Design is used and adapted when the goal is to automate the Process using Software.

Some approaches make distinctions between:
- Define Processes (Process Model or Business Model)
- Then Define Functions (Function Model)
- Then Define Software (Application Model)

We suggest grouping in the same Action Tree the Processes and smaller Actions.
Action Patterns (reuse at Construction time) and callable Actions (reuse at Execution time) are isolated.
Software Structure is not the same as Action Tree for different reasons:
- Some Actions are implemented into pre-existing Blocks
- Some Actions are reusable Actions (reused at execution time)
- Some Actions are Pattern Actions (reused at development time)

CEISAR suggests the following:
- Obvious rule: for pre-existing Blocks, list Action modifications which belong to them
- For new Blocks
  - Classify basic Functions by **Entity** class
  - Chaining Actions are classified by Process class
  - Entity classes and Process Classes can be grouped into Blocks (often called Functional Blocks)

How to align Software with the Action tree?
Processes are implemented with Software Processes.
Functions are implemented as Software services.
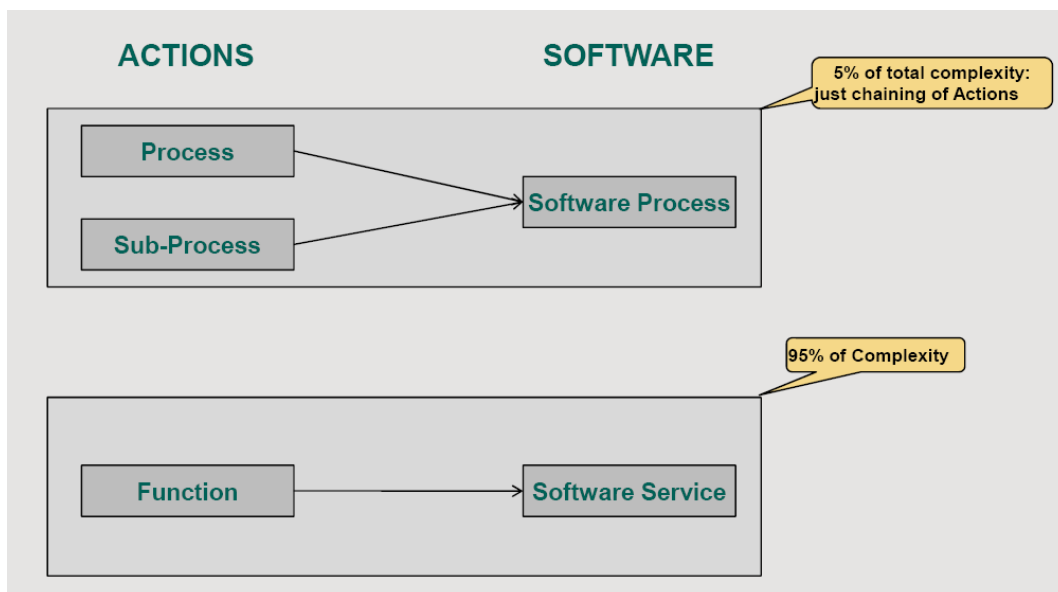


# 4.5 How to reuse a Process or Action?

There are two ways of reusing an Action:
- When executing a Process : Call an Action which already exists (or Black Component or Call)
- When designing a Process : Resembles another existing Action (or White component or Pattern)

## 4.5.1 Call an Action

The same Action may appear in different branches of the Action tree.
For example, "Get info about a Customer" is an Action reused in many places. It can be a big Action like "Plan a Project" or a small Action like "check that Client is already registered". This action must be described and broken down only once.

To recognize that an Action is already described, **naming rules** are useful: standardized Verbs and Business Entities allow automatic retrieval of Actions which have the same wording. Then check that they are really the same:

- If yes, replace description of an Action by a relation to the same Action already defined
- If no, change the wording for one of the homonym Actions

A **Business Process may reuse a Business Process**, just like an Action may reuse an Action.

For example, when welcoming a new employee, you assign a computer to this person. But assigning a computer to a person can also be an independent Process that can be used to renew the computer of this person when it is out of date.
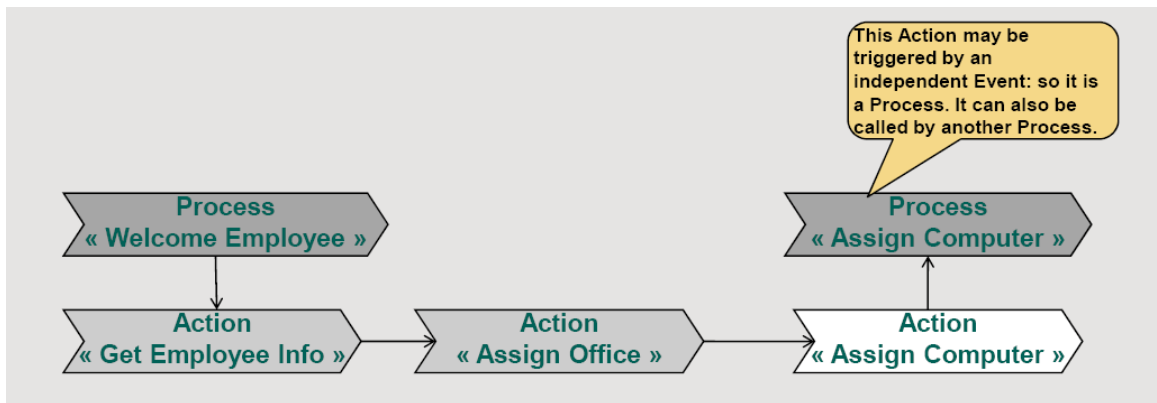


At the end of an Action tree are basic Functions. Some Functions can be highly reusable. A Business Action may call a reusable Function:
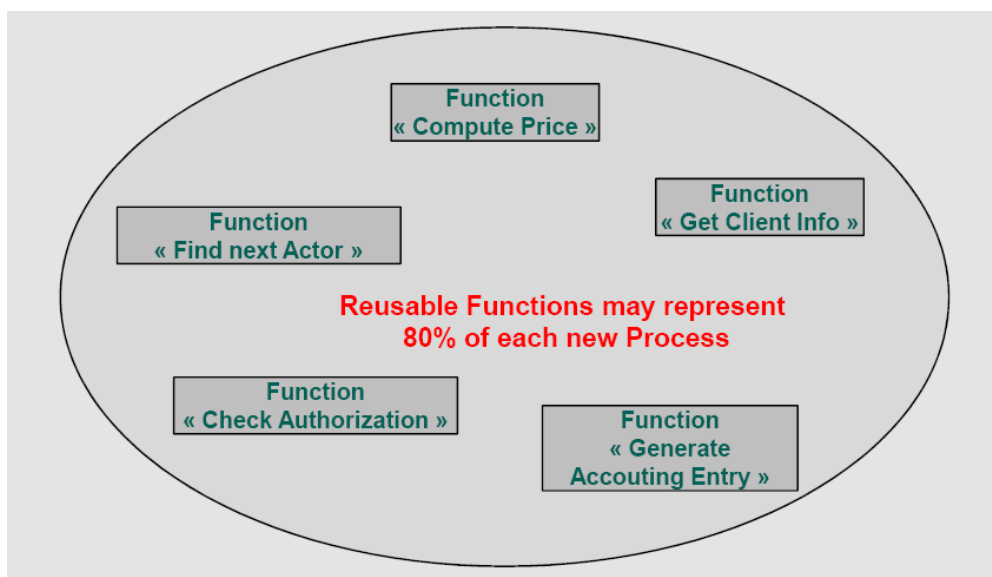


### 4.5.2 Action Pattern

An Action can look like another one: the common part is called an Action Pattern (The Object Oriented expert calls it "inheritance").

For example, an insurance company may define a "Manage Claim" pattern which is reused by "Manage Car Damage Claim" or "Manage Home Damage Claim".
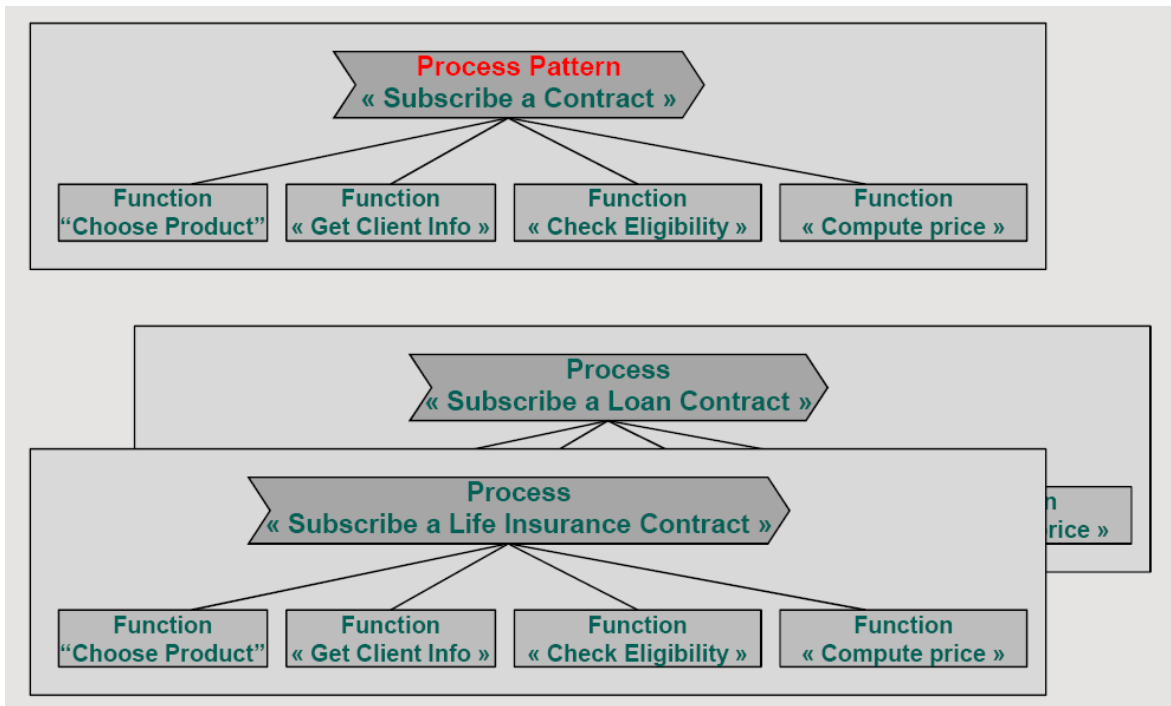
The same approach can be applied to Functions:
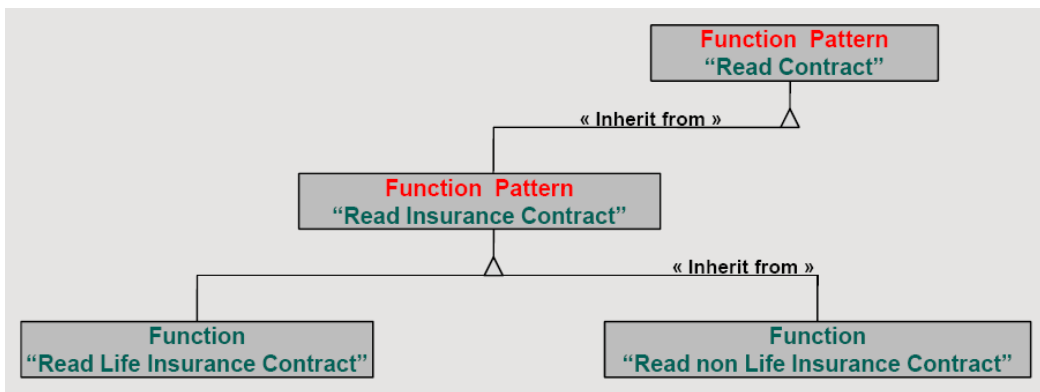


### 4.5.3  Pattern "Operation" and "Descriptor"

Good examples of Process Templates are Operations and Descriptors:

- An **Operation** is an Entity which has a standard life-cycle:
  - an Operation is **PREPARED** in one or several steps by one or successive Actors.
  - When an Operation is Applicable (validity of data and authorized), it can be **APPLIED**, which means that **irreversible** actions like the updates of other Entities can be performed. An Operation can be Applied just once. After being Applied, the Operation can be kept persistent or archived. All Operations must be signed by an Actor so that it is possible to track who is responsible for changes and when these changes happened.

Any Entity which reuses this Pattern will benefit from some predefined Processes. For example define "Reservation of an Airline flight" as an Operation will implicitly define the Actions : "Prepare a reservation", "Modify a reservation", "Abandon a reservation", "Suspend a reservation", "Authorize a Reservation", "Execute a Reservation"...

| Operation Example | Preparation | Execution |
|---|---|---|
| Money transfer | Capture amount, sending and receiving accounts, check provision, authorize | Send Money, update account |
| Enter Order | Get Client info and order lines, compute price, chek credit, check stocks, authorize | Create order, Reserve Stock, Produce Bill, Plan delivery |
| Receive Order Payment | Get Payment info, Get Bill | Update Order, Generate accounting entries |

- A **Descriptor** (like Person or Organization Unit) is an Entity which has the most simple life-cycle. The possible states are:
  - 'created',
  - 'modified' as many times as necessary and finally
  - 'deleted'.

A Descriptor is not Executed as an Operation, but a Descriptor is generally modified by Execution of an Operation. The advantage is that you just define the Entity "Person" as a Descriptor and you do not have to describe the Processes "Create Person", "Modify Person", "Delete Person".

## 4.5.4  Reusable Software

**Reusable Actions** are implemented as **reusable Software Services** (we also call them "Black Components"): input and output of each Action help define the Software Service **Interface**.

**Action Patterns** (we also call them "White Components") are implemented in a different way
- If traditional tools are used, use Pattern Software which is used as a bootstrap by each developer
- If Object-Oriented tools are used, inheritance will be used. Advantage is that Pattern evolutions will be automatically applied to all Actions which inherit from the Pattern

# 5 Mapping with other modeling standards

Standards arise from widespread common usage, when Norms are defined and written through business law or official business organization. This means that nobody can impose standards. They must emerge from the business and the IT analysts.

So, BPM standards derive mainly from the usage of methods & tools like UML, SADT, IDEF3, BPMN…. and associated software, because that's what these users use every day.

As there are many uses and understandings of words depending on context, people culture, and habits… standard definition is a challenge.

Here are some business organizations whose goal is standardization:
- BPMI: Business Project Management Initiative, which merged with OMG (Object Management Group) in 2005
- WfMC : The Workflow Management Coalition (more than 300 members)
- WARIA: Workflow And Reengineering International Association, related to the BPMI and to OMG

## 5.1 Merise

The MERISE methodology (a waterfall methodology) was developed at the end of the seventies and was very popular in France until the early 90s (until the development of Object-oriented programming and UML). Its objective is to encompass the entire definition and implementation of an Information System. It includes three abstraction levels (from general to detailed: Conceptual, Logical/Organizational, Physical/Operational) and a five-step life cycle (master plan, pre-study, detailed analysis, technical analysis and implementation).

The MERISE Methodology deals with Processes according to two diagrams:
- At the Conceptual level, the MCT ("Modèle Conceptuel des Traitements" or Process Conceptual Model)
- At the organizational level, the MOT ("Modèle organisationnel des Traitements" or Process Organizational Model)

The MCT is focused on the relationships with **external** actors. It manages external events and information flows and describes the results of the Operations.

The MCT is "a representation, in a schematic form, of reaction events like: **Triggering Event -> Transformation in the Information System -> Result** ... and independent from any internal organization consideration".

Key concepts are:
- **Event**: Collection of facts, which might trigger an 'Operation' under the conditions defined in the 'synchronization'
- **Synchronization**: Boolean condition; describes the conditions to trigger an 'Operation'
- **Operation**: A set of actions, which cannot be interrupted (once the initial triggering event arises and the conditions are met)
- **Transmission Rule**: Set of Business rules or conditions which trigger the result
- **Result**: collection of facts produced by an 'Operation' under the conditions of 'transmission rules'.

The MCT is focused on what is achieved, not who does it. A high degree of invariance is looked for. The MCT has been highly influenced by the Petri Net approach.

Example of an MCT:



M.C.T. : Exemple "ouverture d'un compte en banque"
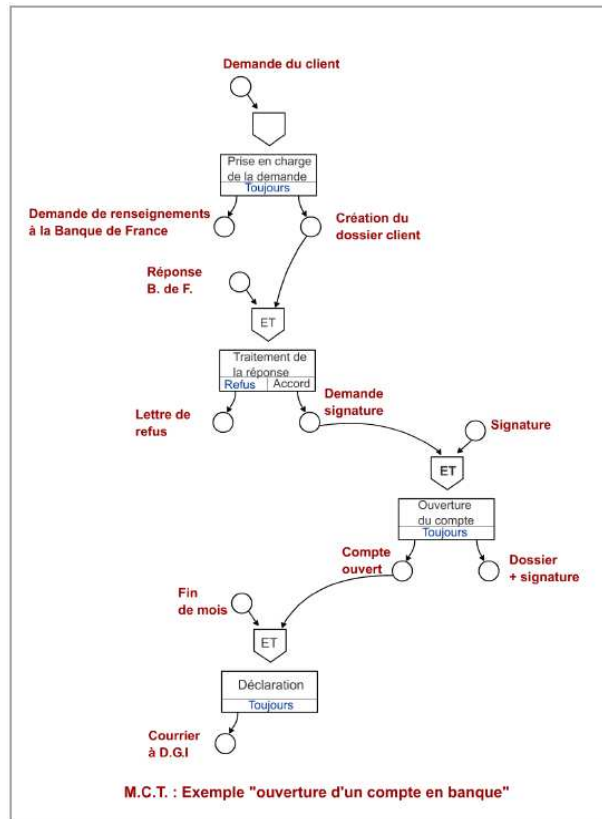
The MOT describes each 'Operation' of the MCT in detail in a set of 'Functional Procedures'. Each Functional procedure details the business rules which will have to be executed manually or coded in the Information System.

The MOT defines the activities of each job position (automated or not) and of each Organization Unit, while taking into account the "timing", the type of resources and the type of output.

Two examples:



Le graphe de circulation de l'information de l'"Admission" dans un hôpital
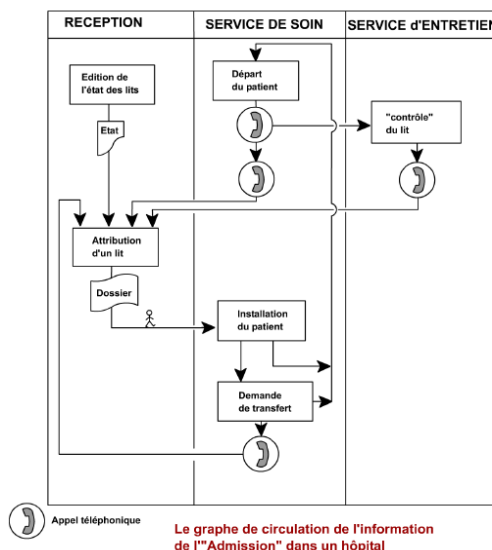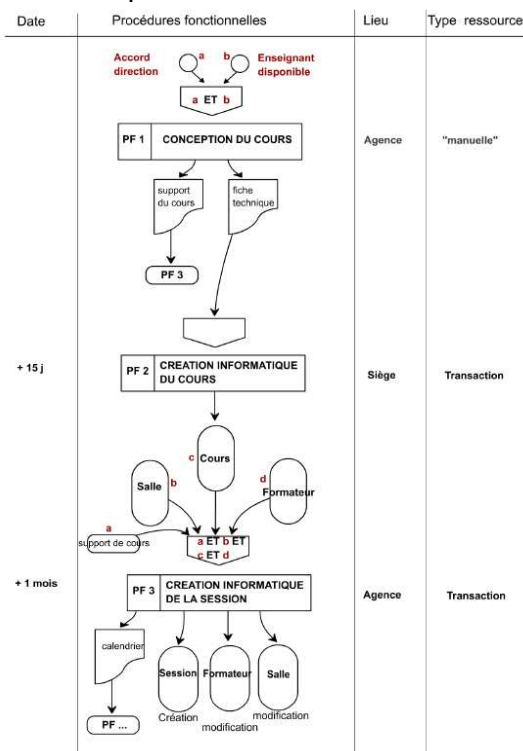
# 5.2 SADT/IDEF3

SADT, Structured Analysis and Design Technique, was developed at the end of 70s. It was first used for industrial systems. SADT is essentially a systemic approach. IDEF3 is the Process modeling part of the IDEF family of methodologies:

| Method | Objective | Purpose |
|--------|-----------|---------|
| IDEF0 | Function Modeling | Analysis |
| IDEF1 | Information Modeling | Analysis |
| IDEF1x | Data Modeling | Design |
| IDEF3 | Process Modeling | Analysis |
| IDEF4 | Object-oriented Design | Design |
| IDEF5 | Ontology Description | Analysis |

"The IDEF3 Process Description Capture Method provides a mechanism for collecting and documenting processes" (source: http://www.idef.com/idef3.html).
A Process Model is elaborated using a Business Scenario developed by Business experts, which describes what happens in the real world (equivalent to a Use Case in UML).

Two diagrams are used:
- The Process Description Diagram (called **Process Schematic**) is centered on Actions, named after the generic term "**Unit of Behavior.**"

| Model Entity | Definition | Graphical representation |
|--------------|------------|--------------------------|
| Unit of Behavior (UOB) | What happens in the system | |
| Link | indicates the precedence relationships (or more generally constraints) between UOBs | |
| Junction | provides a mechanism to specify the logic of process branching | |

Examples:
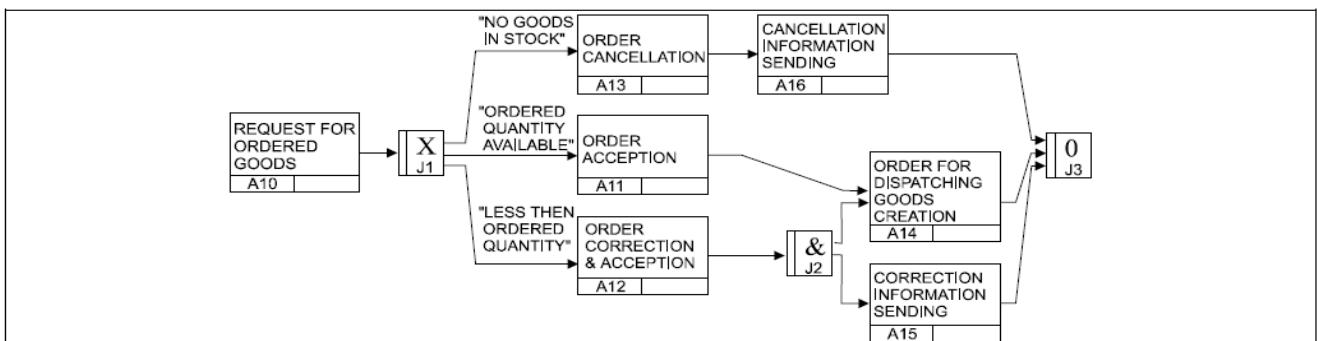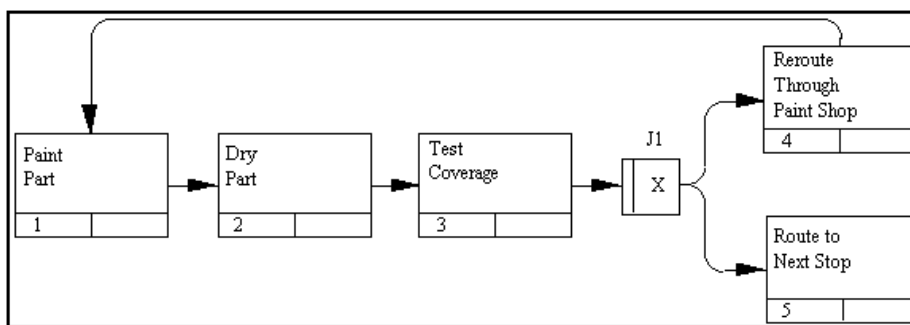
Each UOB can be broken down into a Process Schematic if it is too complex. Break-downs allow the user to capture descriptions at varying levels of abstraction.
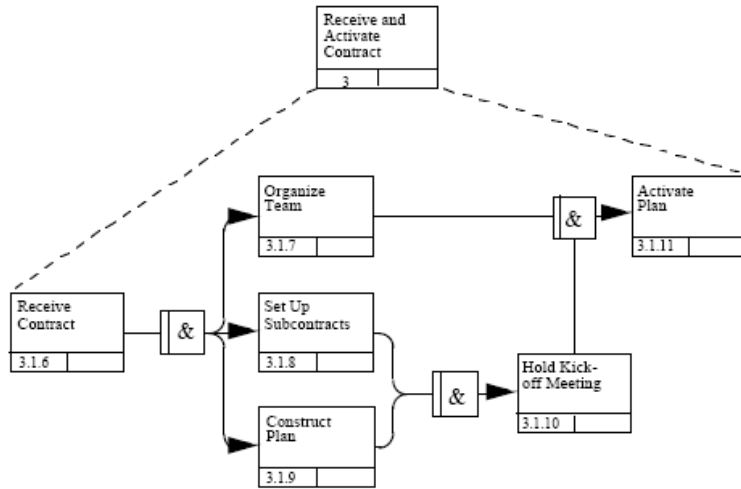


- The Object State Transition Network (called **Transition Schematic**) is centered on Objects, their states and transitions between states.

| Model Entity | Definition | Graphical representation |
|---|---|---|
| Object-State | A kind of object and its corresponding state | *A* (circle) |
| Link | Represents a transition between two states | → |
| Junction | Provides a mechanism to specify the logic of potentially multiple paths in state transition behavior. | (X) |

Example:
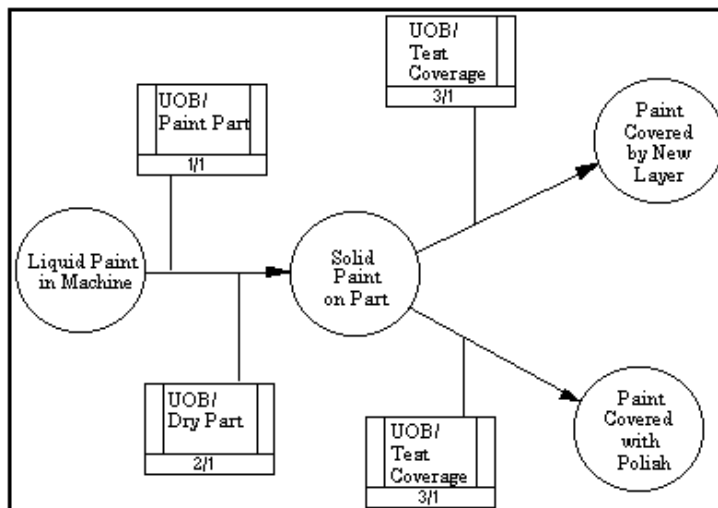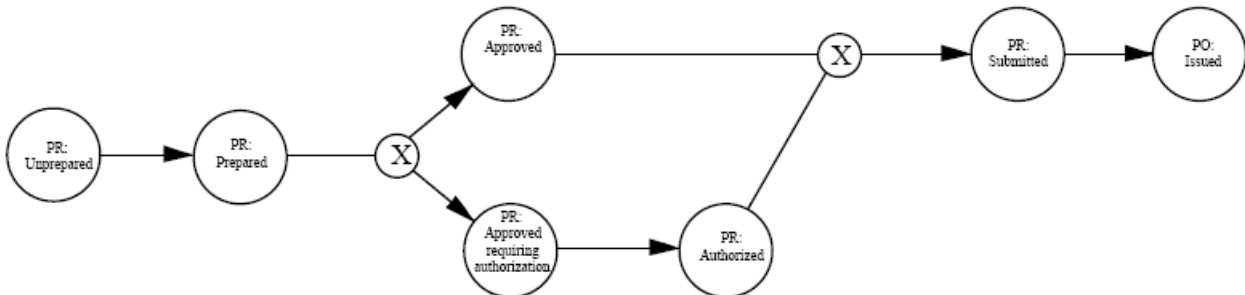
In a nutshell, here are all the building blocks of the two IDEF3 schematics:
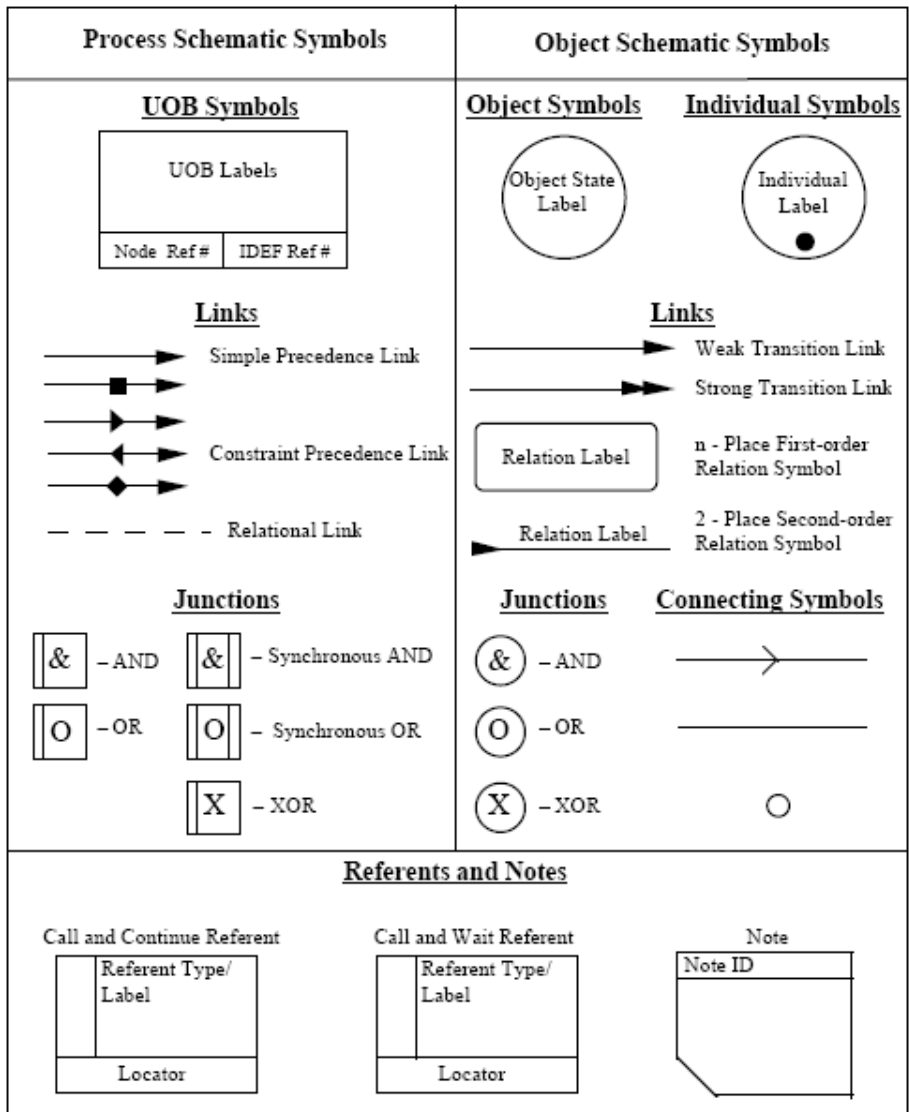


## 5.3 OSSAD

OSSAD was developed at the end at the end 80s under the CCE ESPRIT program. Its purpose was to help automation of office activities.

The OSSAD Method uses two types of Models to describe an Enterprise:
- The **Abstract Model**, which models the essential objectives of the Enterprise ignoring the means and the people, and
- the **Descriptive Model**, which represents the actual implementation of a system. There may be several Descriptive Models corresponding to one Abstract Model, because the essential objectives of an Enterprise can be realized many ways.

**The Abstract Model** emphasizes the value added by the Process. The main concepts (used to represent the goals and needs of the Enterprise) are:
- **Function (or Sub function):** A first division of the Actions and of the structure of an organization usually based on the need to co-ordinate, direct and control to obtain specified outcomes independently of the actual means used. It may be divided into Sub functions, with the number of such levels reflecting the need for detail.

- **Activity:** The finest division of Functions. It is the smallest amount of work that must be done for which a meaningful objective and/or easily definable output can be identified.
- **Packet:** A set of objects/data that passes between activities and between them and the external environment.

Visual representations are called a **Function Schema** or an **Activity Schema** depending on the level of detail at which the Model is represented.

**Function Schema:**

To be developed

The Sub function is represented as an Activity Schema. This level of detail may be adequate for the purpose of a specific study, in which case, it can be agreed and used. It may be detailed further if necessary. This Model can be viewed as a Model in its own right with its own boundary or it can be viewed as a refinement of the Function Schema.

**Activity Schema**:

To be developed

**The Descriptive Model** describes the Physical Organization (desks, paper, equipment, including the description of procedures) and the Organizational structure of Actors. The technological and work process structure is described in terms of Procedures, Operations, Resources and Facilities.

- **Operation**: The basic (most detailed) element relevant to the description of the Enterprise
- **Procedure**: A coherent set of Operations, undertaken by one or collaboratively by several Roles, which result in a meaningful objective being achieved or a Resource being output. An Activity in the Abstract Model maps onto a Procedure in the Descriptive Model
- **Resource**: Data or objects which are inputs to, or outputs from, Operations/Procedures
- **Facility:** Physical or technological support to perform work
- **Role:** An organizational charge covering a set of Operations performed by one or several individuals in a procedure
- **Unit:** An aggregation of Roles based on one or more organizational requirements for co-ordination or control. Units may be aggregated into higher level Units, culminating at the level of the entire Enterprise
- **Actor:** An individual who carries out a Role

A **Task** is a term that is occasionally used in the OSSAD context. It is not represented as a symbol in the OSSAD formalism. It is the set of Operations which are performed by a given Role and contained within a single Activity. Tasks are identified with the aid of the Role/Activity matrix.

In the OSSAD Methodology, a **Procedure Schema** is used to represent the relationships among a set of Procedures, linked by the flow of Resources (inputs/outputs from the various Procedures), using Facilities.

**Procedure Schema:**

Example to be developed

A Procedure, defined as a set of Operations, can be detailed at Operations level and represented as an **Operations Schema**. In OSSAD this is a pure Petri net representation.

Another representation, much more widely used in practice, is the **Procedure Diagram**. This representation shows how an Activity is actually implemented as a Procedure. It shows the Roles involved in the procedure, the sequence of Operations, the Resources passing between Operations and may show the Facilities used to support the Operations.

**Procedure Diagram:**

Example to be developed

Another complementary and widely used representation is the **Role Schema**. This shows in a simple form, with no reference to the temporal aspects, the communication between Roles.

**Role Schema:**

Example to be developed

Another form of representation is the matrix which is used to synthesize the Roles that are involved in an Activity or its implementation, the Procedure. The **Activity/Role matrix** is extensively used in OSSAD to analyze and then rationalize procedures.

Activities can be grouped Functions and an **Activity/Function matrix** prepared to analyze which Roles are central and peripheral to a Function, so that rationalization or redistribution of Roles can be studied.
**Activity/Role Matrix:**
Example to be developed

The OSSAD meta-model:
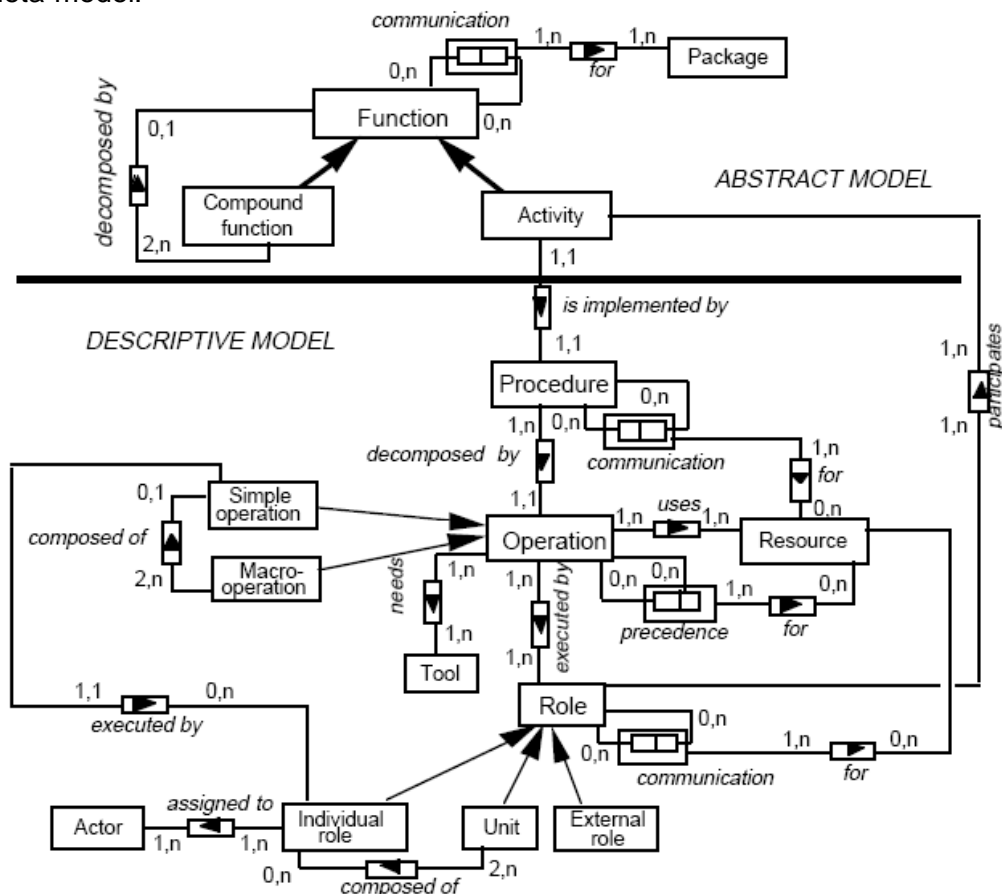


## 5.4  UML

UML is an object-oriented modeling technique resulting from the merging in 1997 of three prominent methods: OOD, OMT and OOSE. It can be used to describe an Information System using several types of representations:
- Structure diagrams (class, composite structure, component, deployment, object and package diagrams).
- Behavior diagrams (activity, state machine, and use case diagrams)
- Interaction diagrams (communication, interaction overview, sequence, and timing diagrams)

The structure diagrams depict a static view of the system and will not be described here. The behavior and interaction diagrams give a more dynamic view of the system. Hence, they can be used to model Processes.

### 5.4.1  Collaboration/Communication Diagram

Communication diagrams show the message flow between objects in an OO application and also imply the basic associations (relationships) between classes. The lines between the classes represent the relationships (associations, composition, dependencies, or inheritance) between them. Messages are

depicted as a labeled arrow that indicates the direction of the message, using a notation similar to that used on sequence diagrams.
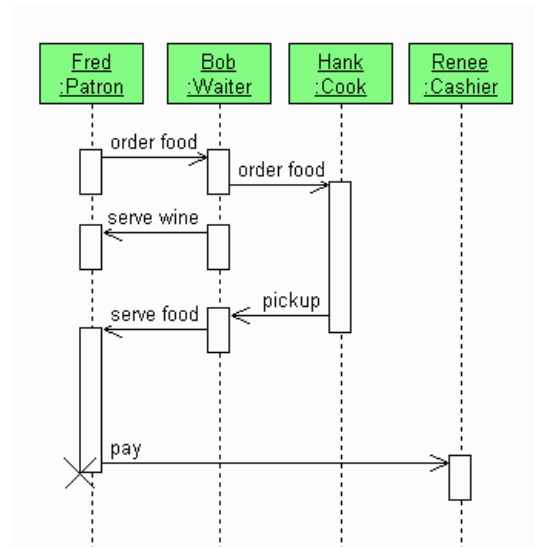
Example to be developed
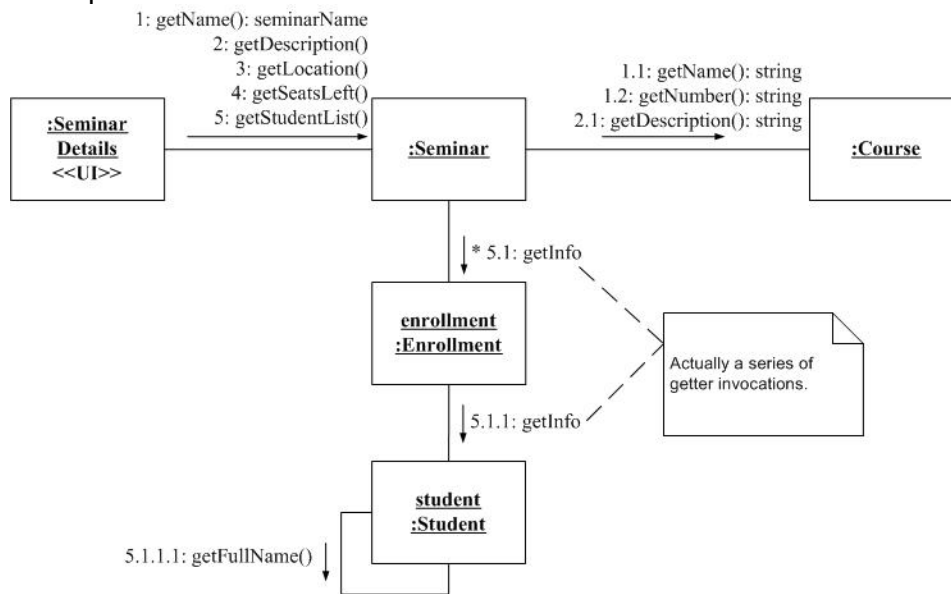


## 5.4.2  Sequence Diagram

Sequence diagrams contain the same information as Collaboration diagrams, but emphasize the sequence of the messages instead of the relationships between the objects.
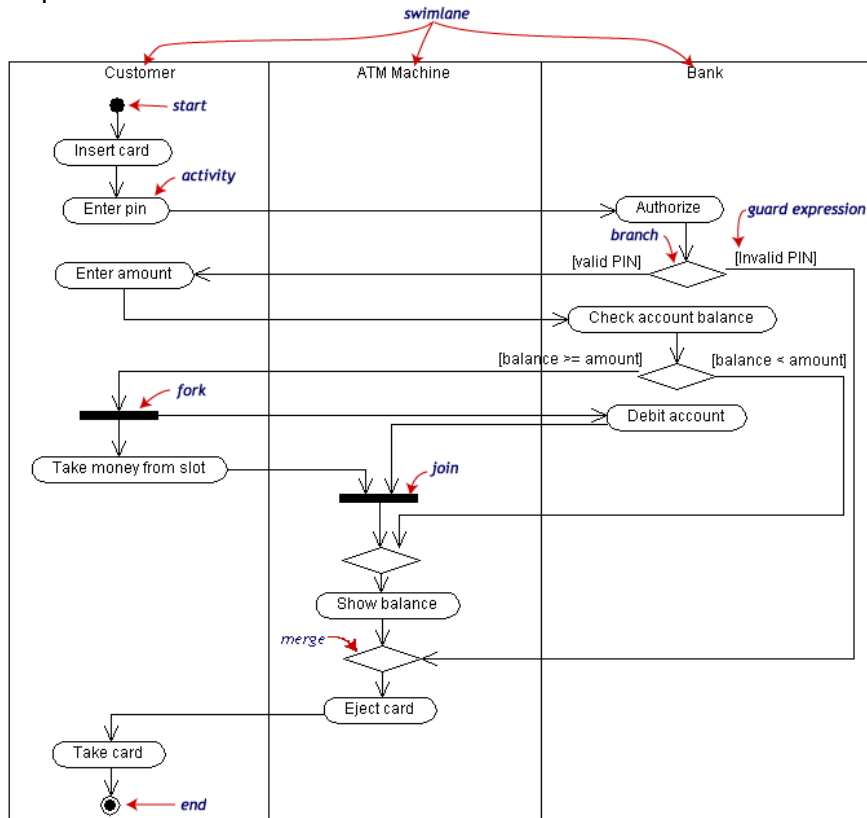Example to be developed



## 5.4.3  Activity Diagram

An **activity diagram** represents the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control. In many ways UML activity diagrams are the object-oriented equivalent of flow charts and data flow diagrams (DFDs) from structured development.
It is an oriented graph with conditional joins/forks. Several Actors can be represented using swim lanes. Transitions between Activities are triggered by Events.

| Action/Activity | AcceptEvent | InitialNode | ActivityFinal | FlowFinal | |
|---|---|---|---|---|---|
| CallBehaviorAction | SendSignal | Decision [cond1] ... [cond n] | Merge ... | Fork | Join |
| a) Actions | | b) Control Nodes | | | |

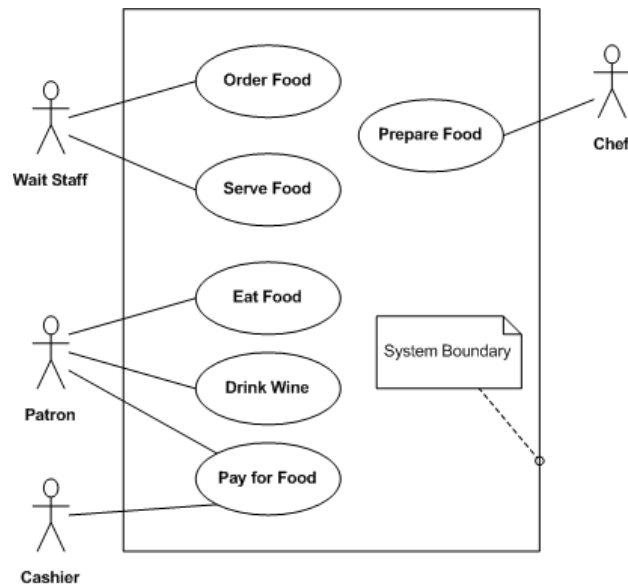Example to be developed



## 5.4.4  State Machine Diagram

A State Machine Diagram describes the possible states of an entity, based on its response to events.

Example to be developed
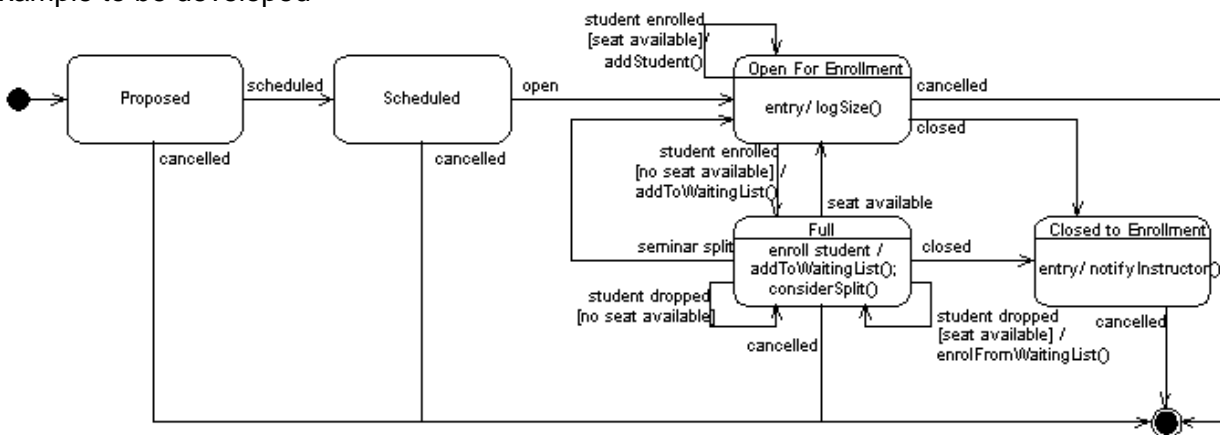


## 5.4.5  Use Case diagram

An UML use case is useful for describing a business scenario. The different Actors can be identified. The boundaries of the system are clearly defined and the main Actions and information flows are described.

Example to be developed

## 5.4.6 UML diagrams and Process modeling

Collaboration, sequence and use case diagrams can be used to represent a map of several Processes. This can help to define the boundaries between Processes and between Actors, and how they interact.

The State Machine diagram can be used to identify a list of Processes acting on a Business Entity. You can represent the various states of the Business Entity. The Processes can be derived from the transitions between states.

A detailed description of a Process can be obtained with an Activity diagram, which is probably the most useful UML diagram for Business Process modeling. The UML Activity diagram can be easily compared to the most recent Business Process Notation : BPMN.
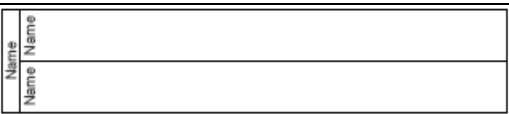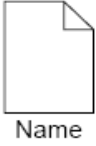
# 5.5 BPMN

Business Processing Modeling Notation was developed by Business Process Management Initiative (BPMI). BPMI merged with Object Management Group in 2005.
It includes three main concepts:

| Element | Description | Notation |
|---------|-------------|----------|
| Event | An event is something that "happens" during the course of a business process. These events affect the flow of the process and usually have a cause (trigger) or an impact (result). Events are circles with open centers to allow internal markers to differentiate different triggers or results. There are three types of Events, based on when they affect the flow: Start, Intermediate, and End. | |
| Activity | An activity is a generic term for work that a company performs. An activity can be atomic or non-atomic (compound). The types of activities that are a part of a Process Model are: Process, Sub-Process, and Task. Tasks and Sub-Processes are rounded rectangles. Processes are either unbounded or a contained within a Pool. | |
| Gateway | A Gateway is used to control the divergence and convergence of Sequence Flow. Thus, it will determine branching, forking, merging, and joining of paths. Internal Markers will indicate the type of behavior control. | |

Additional concepts include:

| Element | Description | Notation |
|---------|-------------|----------|
| Sequence flow | A Sequence Flow is used to show the order in which activities will be performed in a Process. | |
| Message flow | A Message Flow is used to show the flow of messages between two participants prepared to send and receive them. In BPMN, two separate Pools in the Diagram will represent the two participants (e.g., business entities or business roles). | |
| Association | An Association is used to associate information with Flow Objects. Text and graphical non-Flow Objects can be associated with the Flow Objects. | |
| Pool | A Pool represents a Participant in a Process. It also acts as a "swim lane" and a graphical container for partitioning a set of activities from other Pools, usually in the context of B2B situations. | |
| Lane | A Lane is a sub-partition within a Pool and will extend the entire length of the Pool, either vertically or horizontally. Lanes are used to organize and categorize activities. | |
| Data Object | Data Objects are considered Artifacts because they do not have any direct effect on the Sequence Flow or Message Flow of the Process, but they do provide information about what activities require to be performed and/or what they produce. | |
| Group (a box around a group of objects for documentation purposes) | A grouping of activities that does not affect the Sequence Flow. The grouping can be used for documentation or analysis purposes. Groups can also be used to identify the activities of a distributed transaction that is shown across Pools. | |
| Text Annotation (attached with an Association) | Text Annotations are a mechanism for a modeler to provide additional information for the reader of a BPMN Diagram. | |

Examples: