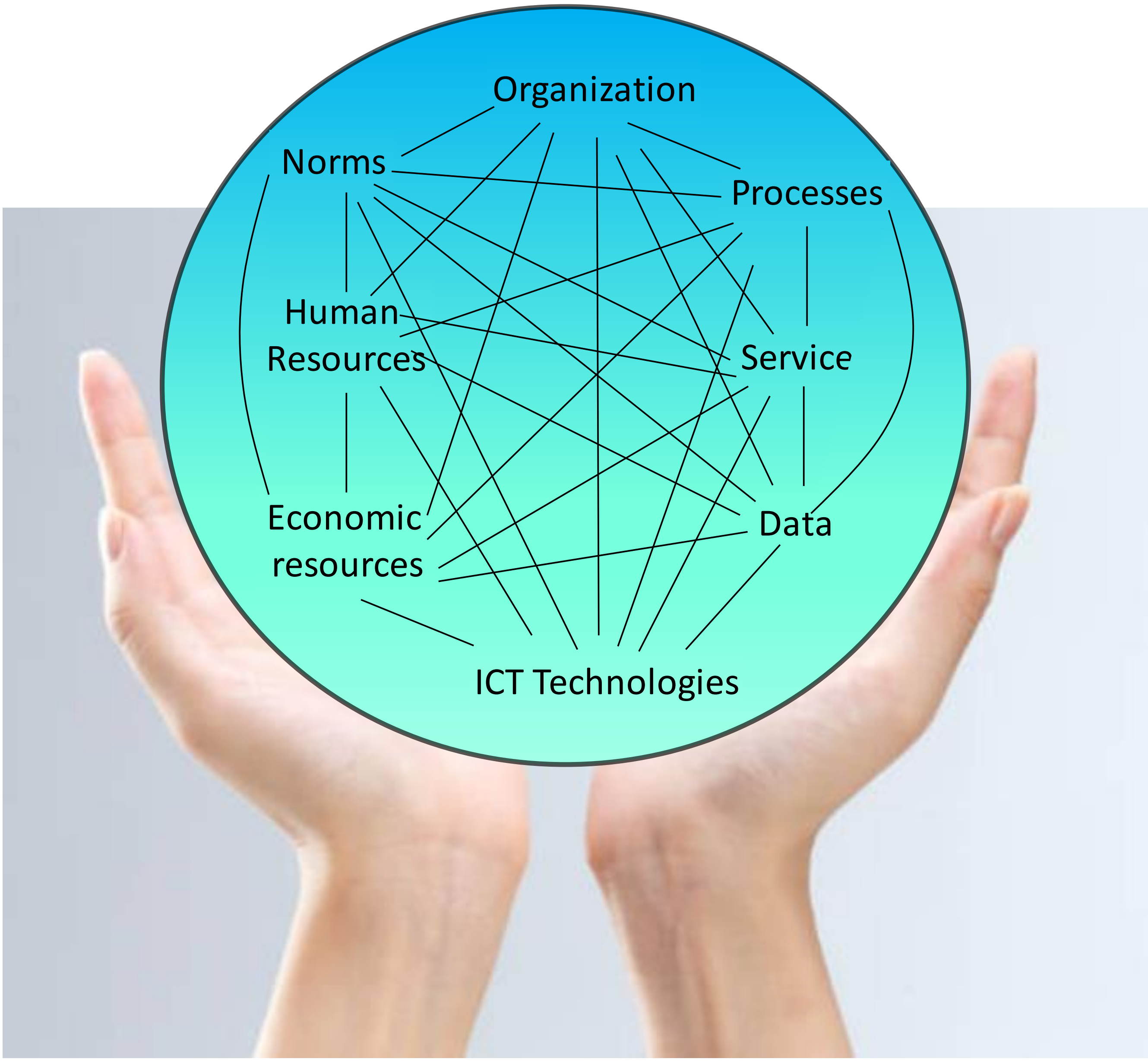
Come si progetta un Sistema Informativo
How to design an Information System

Carlo Batini
Università di Milano-Bicocca



Premessa/Foreword

Progettare un sistema informativo è molto difficile, *insegnare a progettare* sistemi informativi è quasi impossibile, ma siccome a me piace affrontare imprese impossibili, a questo punto, dopo venticinque anni che insegno sistemi informativi, e questo è l'ultimo, ho deciso di *scrivere un libro* su come si progettano i sistemi informativi. Il libro è scritto in inglese.

Le mani del disegno nel frontespizio sono nella mia immaginazione quelle di Paolo Ercoli, che tanto tempo fa, negli anni 80 del secolo scorso, per farmi comprendere I sistemi informativi e quali contenuti dovessero essere trattati e composti in un corso di sistemi informative mi fece un gesto con le mani, come se dovessero raccogliere, mettere insieme, amalgamare tanti contenuti, in una sintesi che soltanto l'esperienza poteva permettere. Questa esperienza l'ho fatta lavorando all'Aipa, la Autorità per l'Informatica nella Pubblica Amministrazione, e l'ho fatta perché Guido Rey mi ha chiamato a lavorare su alcuni dei sistemi informativi più complessi che si possano immaginare.

Milano, Giugno 2019

To design an information system is very complex, to teach how to design an information system is extremely difficult, but, since I like to face impossible challenges, after 25 years of teaching the course, first at University La Sapienza in Rome, then at University of Milano-Bicocca, in the occasion of my last year of teaching before retiring, I decided to write a book on *How to design an information system.* The book is written in English.

In my imagination, the hands depicted in the title page are those of Paolo Ercoli; long ago, in the 80s of the last century, to make me perceive the nature of information systems and which contents should be dealt with in a course on information systems, Ercoli made a gesture with his hands, as if they had to collect and put together so many contents, in a synthesis that only the experience could afford. I got such experience working at AIPA, the Authority for Informatics in Public Administration, and I got it because Guido Rey asked me to work on some of the most complex information systems that can be imagined.

Milan, Italy, June 2019

Dedica

Questo libro è dedicato a mio nipote Curzio, che ci ha lasciato nel 2014, a 51 anni. Curzio era iscritto alla Facoltà di Ingegneria della Università di Roma La Sapienza, fece l'esame di Sistemi Informativi con me, e gli detti 29. Che stupido che sono stato…. Ora questo libro è per Lui, e per Paola e Alessandra.

Contents of the book

Chapter 1 - Information Systems in Organizations: basic concepts

1.1 Introduction

Assume you are a student enrolled in a University, and that you make an exam. In the following box the activities that are performed to update the database of student’s exams are shown. We assume that the University under consideration is a bit old-fashioned, since paper registries are used by professors at the end of an exam session.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher. When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office. The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students: • have in their study program the course corresponding to the exam, and • are in good standing with University taxes. In case of errors or inconsistencies in the register, the office sends the register back to the teacher. When the records are complete, the grades of exams are updated in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

The above requirements show a simple example of information system, highlighting the interrelationship in the management of data of the three main components involved: the data, the organization and the activities performed.

In order to provide a definition of information system, we now introduce the following concepts:

- System
- Data
- Information
- Knowledge
- Organizational system
- Information System
- Computer System

An information system can be seen as a specialization of a *system*, which may be defined as a set of functional interacting components, where every component provides a contribution to a common goal.

Examples of systems are:

1. A Railway station
2. A University
3. The Human body

L. Von Bertalanffy in his book on Premise of General Systems Theory says that “*there exist models, principles, and laws that apply to generalized systems or their subclasses, irrespective of their* *particular kind, the nature of their component elements, and the relations or 'forces' between them*". E.g. all Universities are similar in providing services to students, with small or large differences.

In Figure 1.1 we see several examples of systems, pertaining to three different categories, respectively
natural systems, manufactured systems and socio technological systems.

Type	Description	Examples
Natural Systems	Biological, geological, or climatological phenomena that occur in the natural world	• Animal • Earthquake • Weather
Manufactured Systems	Production of artifacts conceived by human designers	• Automobile • Computer • House
Socio-technological Systems	Combination of natural and manufactured systems	• Business • Government • Services

Figure 1.1: Examples of systems, with their descriptions and examples

Data, information, knowledge

According to Boisot [1995] data are the result of a measure on a physical phenomenon, e.g. to read the number 37,5 on a digital thermometer, or else on a traditional analogic thermometer such as the one shown in Figure 1.2. *Information* is the result of assigning a meaning to data based on previous knowledge on the phenomenon; e.g. 37,5 is a body temperature, measured in Celsius degrees. We will frequently use the terms data and information interchangeably.

Data is a discrimination between physical states of things (black, white, etc.) that may convey or not convey information to an agent. Whether it does so or not depends on the agent's prior stock of knowledge.

Knowledge is an asset. An asset is a stock from which a number of services are expected to flow. This idea is not new. Sixteen-century alchemists went to great length to protect the secrets of their craft. In the 21° century, knowledge assets are coming to constitute the very basis of post-industrial economies.

Thus, whereas data can be characterized as a property of things, knowledge is a property of agents. Information establishes a relationship between things and agents. In contrast to information, knowledge cannot be directly observed, its existence can only be inferred from the action of agents.

Figure 1.2 - An old fashioned thermometer

Data and Information are primary resources used in public and private organizations and in our own life. ICT technologies and digital data have rapidly evolved in the last forty years, deeply changing our life and the administrative/business processes of organizations.

The most relevant technologies that create value from digital data are shown in Figure 1.3. Their novelty and boost to innovation comes in the modern world from the following characteristics (see [Grefen 2016] for a more comprehensive discussion):

- Mobile computing establishes an always-and-every where-connected paradigm
- Internet of things enables the integration of virtual and physical worlds
- Social networks allow Individuals to obtain access to easy-to-use devices connected to the Internet
- Cloud computing makes shared, remote computing facilities easily accessible
- Big data architectures enable scalability In big data processing.

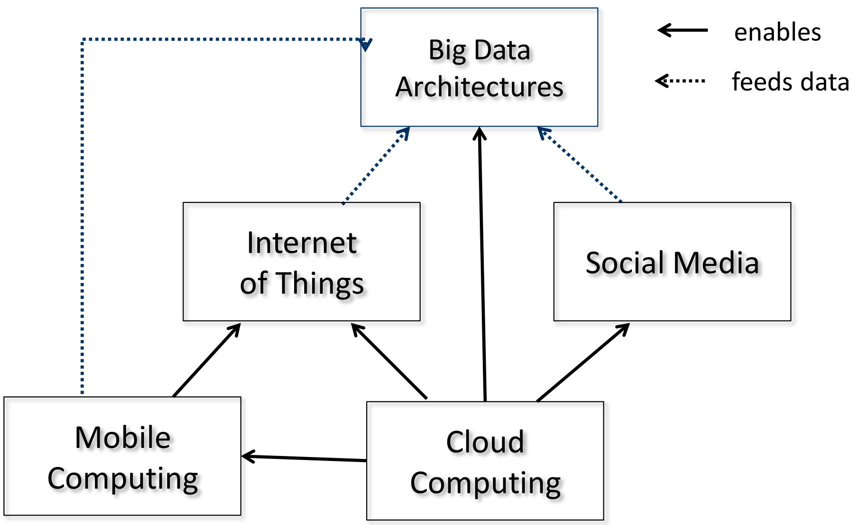


Figure 1.3 – The five main technologies that exploit digital data to create value

A first definition of Information system is the following. An information system of an organization is the set of information (paper or digital) that is

- acquired in input,
- stored,
- exchanged,
- processed inside the organization, and between the organization and the external environment
- produced in output.

1.2. Structure of Information Systems and Organizational Systems

A simplified representation of an Information system is shown in Figure 1.4, where the classical hardware, software, network and data technologies are shown as the enabling technologies that operate on the information resource.

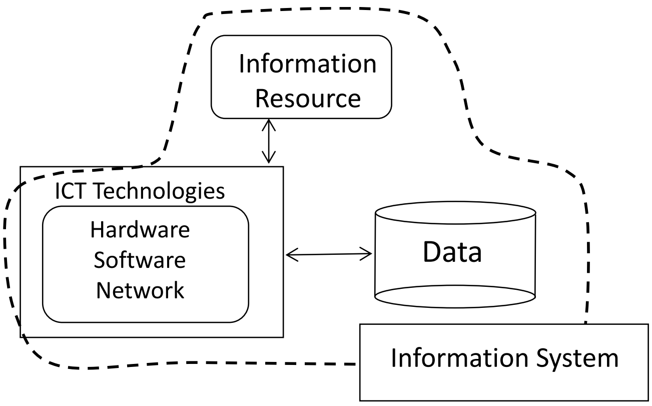


Figure 1.4 – Simplified structure of an information system

The efficiency and effectiveness (see later for a definition and examples) of information systems in achieving the goals of the organization is positively influenced by the usage of ICT technologies.

A computerized information system (CIS) or, more simply, a computer system, is a system making wide usage of ICT technologies in digital information processing. The CIS enhances and extends in time with the discovery of new ICT technologies and through the processing of more and more types of digital information. Going into more detail, the computer system has a layered structure that, moving from the user to the lower technological levels, can be classified in terms of:

- Web site
- Human interface and interaction devices
- Access channels
- Client side software applications
- Server/Web side software applications
- Middleware (e.g. DataBase Management Systems)
- System software (e.g. Operating System)
- Data base and document management systems
- Centralized and distributed hardware infrastructure
- Telecommunication networks

- Sensor networks.

An information system cannot exist in isolation; it has to be used by an organization. So, a concept very close to the concept of information system is the organizational system, that makes use of resources (information among others) and is compliant with the rules for the usage of resources adopted in an organization for achieving its goals, see Figure 1.5.

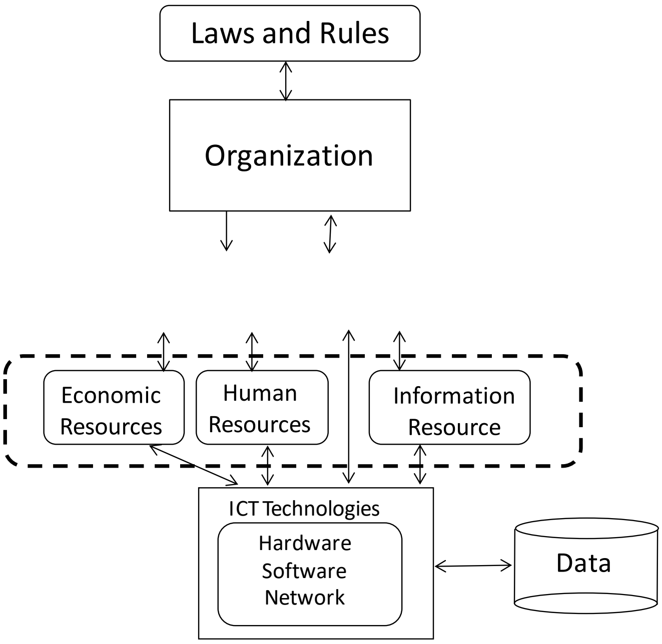


Figure 1.5 – Organization, resources, technologies

Organizational systems in their life cycle must obey a set of laws and rules, going from International Laws to, in case of Italy, European Union rules and laws, to National, Regional Municipality and finally internal Organizational regulations.

Going more in depth, an Organizational system, in order to use an information system for its goals, makes use of several resources, such as:

- Economic resources
- Human resources
- Information resources
- Technological resources
- Physical/Infrastructural resources (not represented in Figure 1.5 and in the following).

Resources transform the inputs to the organizational system into outputs carrying out a set of activities called *process*es. Processes are the activities performed in an organization by the different organizational units or else by external users, making use of the different resources (economic, human, technological. physical, etc.) available in the organization.

In a wider sense, the Information system can be seen as including, besides the information resource, other resources used by processes, and the processes themselves, see Figure 1.6. Such wider view allow planning the future evolution of the information system considering not only technologies, but also other resources and processes making use of them.

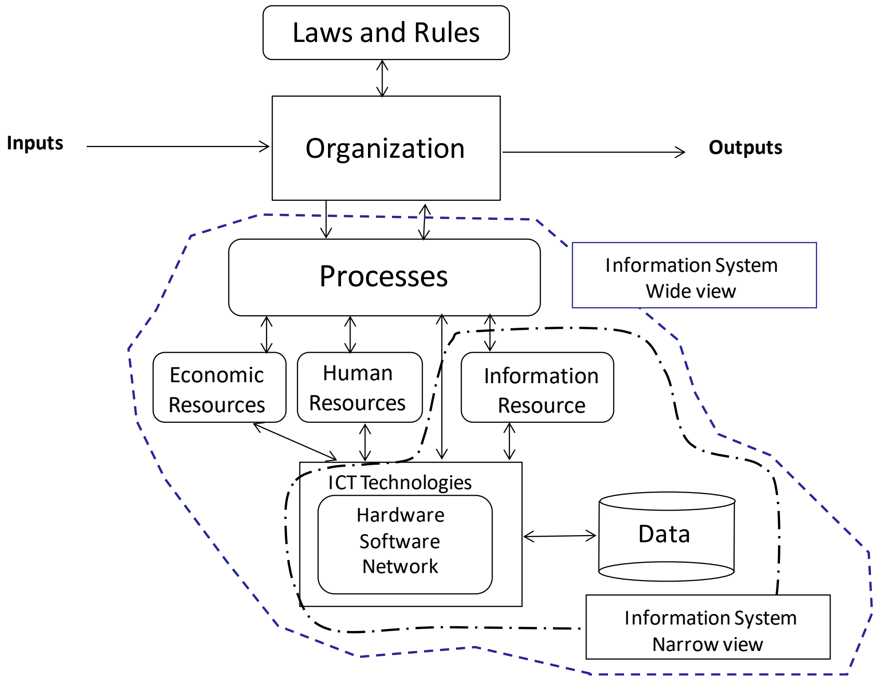


Figure 1.6 – Organization, processes, resources and technologies

Examples of processes in a University are:

- Enrollment of a student
- Preparing a lesson
- Teaching a course
- Pay a monthly salary to a professor.

The efficient and effective execution of processes in an organization is essential to allow the organization achieve its goals. To achieve such goals, processes make use of resources; among them, we will investigate human, technological and economic resources. Therefore, the information system and the organizational system are deeply intertwined.

Dear Reader, let us try to interact each other. We can afford a weak form of interaction in the following way. Time to time, I assign you a classwork, you try to solve it, and in the Appendix to the Chapter I provide a solution. In this way, you can more effectively understand if you succeed in understanding the topics I presented and learn by doing.

Exercise 1.1 - Focus on a University, and fill the following table with human resources and technological resources that you think are involved in performing the following processes.

Process	**Who**	**ICT technology**
Student enrollment	Student	Web site
Preparing a lesson		
Teaching a course		
Pay a salary		

Possible inputs to an Information system and outputs from the Information system are shown in Figure 1.7.

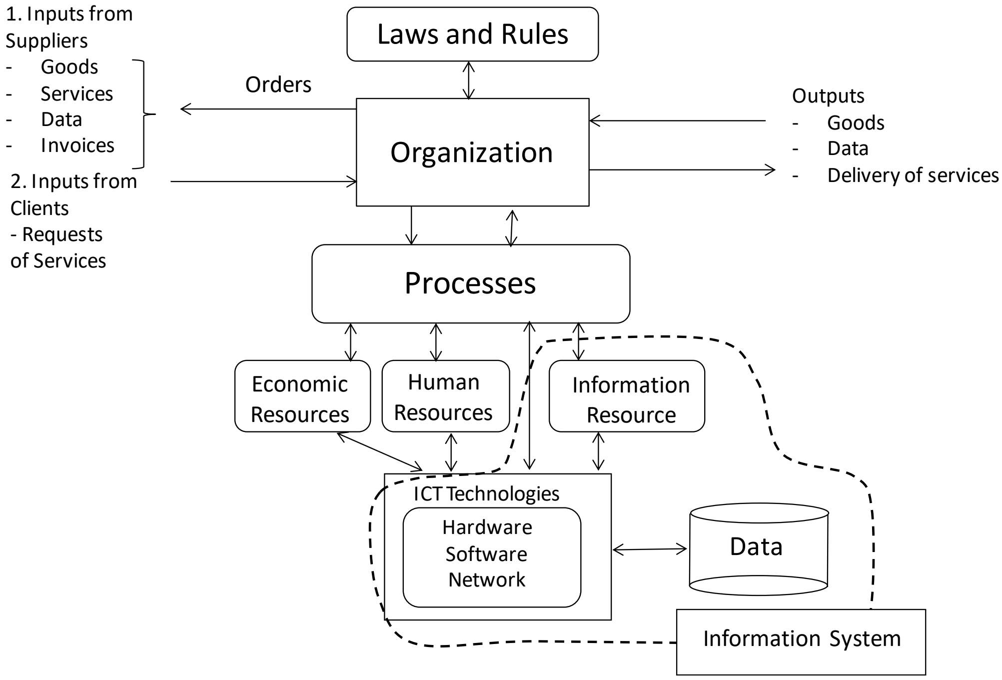


Figure 1.7 - Inputs to and outputs from an organizational and information system

Inputs are of two types, according to the subjects that interact with the organization; they correspond to a. suppliers, and b. clients. Suppliers provide the organizations with several types of products involved in the activities of the organization that can be of three types, a. goods, b. services, and c. data, or a combination among them. Such inputs are usually related to orders, and are provided together with invoices. Clients of the organization typically request services in input. Goods, services and data are at the corner of every interaction between an organization and the external environment.
In turn, the organization through the execution of processes provides in output goods, services or data which are delivered to clients or else to other organizations in response to their requests.

1.3. Goods, services, data

It is time now to better understand the difference between goods, services and data, the three fundamental engines of the economic and social exchanges in modern economies and societies. The whole book is dedicated to digital data managed in information system; the concept of good is in a sense quite traditional and embedded in our experience, so we decide to focus mainly on services.

A service consists in an activity or series of activities, of more or less intangible nature, that they have place in an exchange between a supplier and a customer, where the object of the transaction is an intangible good, so that both the supplier and the customer co-create and obtain value from the transaction. Examples of services are a reservation of a train seat, purchasing a train ticket, making a train travel from Milan to Rome, a reservation of a seat at Scala, Milan.

In Figure 1.8 we see several basic differences between goods and services.

Good	Service
A thing	An activity
Tangible	Intangible
Homogeneous	Heterogeneous
Core value produced in factory (production) • Production and distribution separated from consumption • Customers do not (normally) participate in the production process • Transfer of ownership • Can be kept in stock	Core value produced in buyer-seller interactions (co-production) • Production/Distribution/consumption are simultaneous processes • Customers participate in production • No transfer of ownership • Cannot (normally) be kept in stock

Figure 1.8 - Differences between goods and services

The most relevant difference concerns the tangibility; goods (e.g. oranges) have a physical consistence that usually is lacking in services. Another basic difference lays in the separation of production and consumption in goods: when you go to the supermarket, the oranges that you buy have been produced in the past, and are now consumed by you. While in services production and consumption are more simultaneous (when you go with a train from Milan to Rome, you obtain the service while the service is produced, by means of the train functioning). Finally, goods can be kept in stock, while services vanish while they are produced (if a train seat is not occupied during a train travel, such train seat cannot be recovered in the future).

Notwithstanding the above differences, often it is difficult to clearly distinguish between goods and services, and many artifacts we use have intrinsically a certain degree of “goodness” and of “serviceness”.

Exercise 1.2 - Provide a ranking of goodness vs serviceness from highest good-ness to highest service-ness among the following artifacts:

- Dinner at a nice restaurant
- Legal advice
- Blue jeans
- Socks
- Jewelry
- Dental Examination
- Running shoes
- Psycotherapy
- Car Rental
- Paper book
- eBook
- Plumbing repairs

See in Appendix 1.1 the ranking resulting from a survey described in [Jacobucci 1993].

Services are becoming more and more important in modern economies, and correspond to about 75%/80% of the gross internal product in advanced countries, while goods correspond to about 20% and agriculture to less than 5%.

For services, we can shortly discuss a concept that is also significant for good and data, the issue of quality. We show in Figure 1.9 the three activities to deliver a service of seat reservation, in which the seat is located based on the layout of available seats at Scala Theater in Milan.

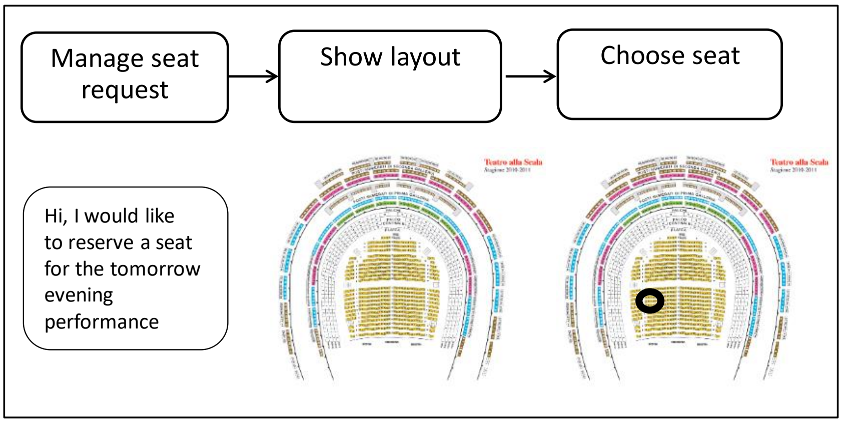


Figure 1.9 – Seat reservation service provided based on the layout of available seats

Surely a better service, or, as we can say, a service of a higher quality, is provided if we make available to the spectator the view of the stage from the seat (see Figure 1.10); since this surely, besides the price, is the most important aspect of the service.

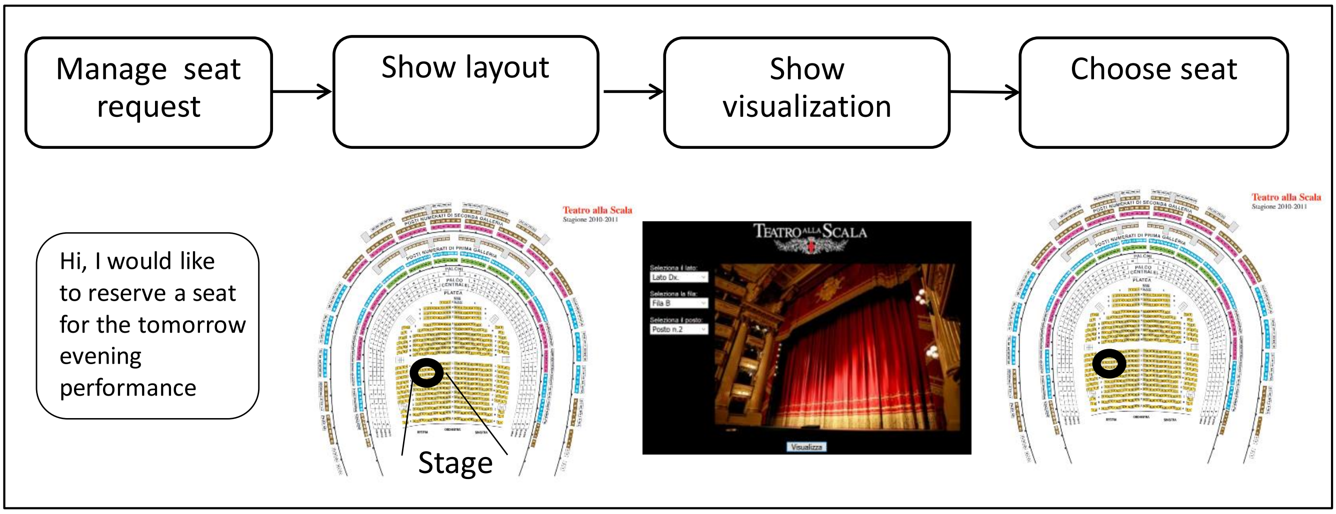


Figure 1.10: Reserving a seat at Scala, Milan

What about the basic features of data in comparison to features of goods and services? Where are placed data between jeans and psychotherapy in the ranking shown in Appendix 1.1? Well, we can state a few general points (the interested reader can go in more depth reading [Moody 1999]):

1. Data are not so visible as goods.
2. Data can be used and reused a lot of times, while goods and services perish.

3. Focusing again on a specific type of goods, namely books, the production of two paper copies of a book has a cost almost equal to two times the cost of one paper copy, while to produce two copies of a digital book has almost zero marginal cost w.r.t to producing one copy of the book.

Due to the mixed nature of goods and services, we can introduce (see [Grefen 2016]) the concept of *hybrid object*, a bundle of goods, services and data made of:
1. Supporting facilities – physical resources in place before a service can be offered, e.g. a ski lift
2. Facilitating goods – material purchased or consumed by the buyer, e.g. rented skis
3. Data made available from customer/provider, e.g. "Ski slopes open today"
4. Explicit services - aspects of the service & benefits readily observable by the senses. e.g. free seat in a coach.
5. Implicit services - aspects of the service & benefits that the customer may sense only vaguely. e.g. seat automatically assigned.

Exercise 1.3 – Browsing in the site of a University, e.g. the University of Milano-Bicocca in Milan, provide examples of layers and components of its information system. Use the following acronyms.

Layers and Components of an Organization and an Information System

1. EXT-ENV - External environment
2. US - Users requesting services
3. USSEG – User segment (e.g. Foreingn student)
4. CRS - Companies requesting services
5. NC - Networked companies
6. SG - Suppliers of goods
7. RS - Requests for services
8. SS - Supply of services
9. SER – Services provided
    - SERDE - Service description
    - SERME - Measured levels of service
    - SERPE - Perceived quality of service
10. LR - Laws and regulations
11. OR – Organization
    - CS – Central organizational structure
    - PS – Peripheral structure
12. IU – Internal users
13. PR – Processes
    - PRI Primary processes
    - SUP – Support processes
14. RES - Resources
    - ECON – Economic resources
    - HUM – Human resources
    - INF – Information resources
    - INFR – Physical resources
15. TECHN – Technologies
    - WEB – Website
    - SWAPPL – Software applications
    - SWMID – Software middleware
    - CHW – Central hardware
    - DHW – Distributed Hardware
    - I/O – Input output unit
    - NTW – Network
    - DB – Data Base

A possible solution is proposed in Appendix 1.1.

Exercise 1.4 – Look at requirements in Appendix 1.2, that describe the information system of a railway company, and classify requirements according to the above taxonomy. Examples are shown at the end of Appendix 1.2.

1.4. Languages and diagrams to model organization, processes and data

We start to perceive also from above the above examples that to design an Information System we have to represent clearly and precisely the organization, processes and data. We now introduce the languages that we will adopt in the book.

1.4.1 Modeling Organizations

First, how can we represent an organization? The traditional model adopted to represent organizations is the *organizational chart*. An example of organizational chart of a University is shown in Figure 1.11, where, besides the high level structure made of the Rector, the Senate and the Management Board, a simplified model of the Administration is shown, organized into the Student office and the Personnel office, and of Departments. Furthermore, the teaching and research structures of the University are represented duplicating the corresponding box, and highlighting the two structures that work in departments, namely, teachers/researchers and administrative personnel. Notice that the student office and the personnel office are examples of functional areas that correspond to functional specializations of the type of work performed in universities. The same holds for structures in Departments.

So, organizational charts can be represented by means of tree diagrams, where the hierarchical or functional relationship between a structure A and a structure B can be represented by means of the parent child relationship typical of trees.

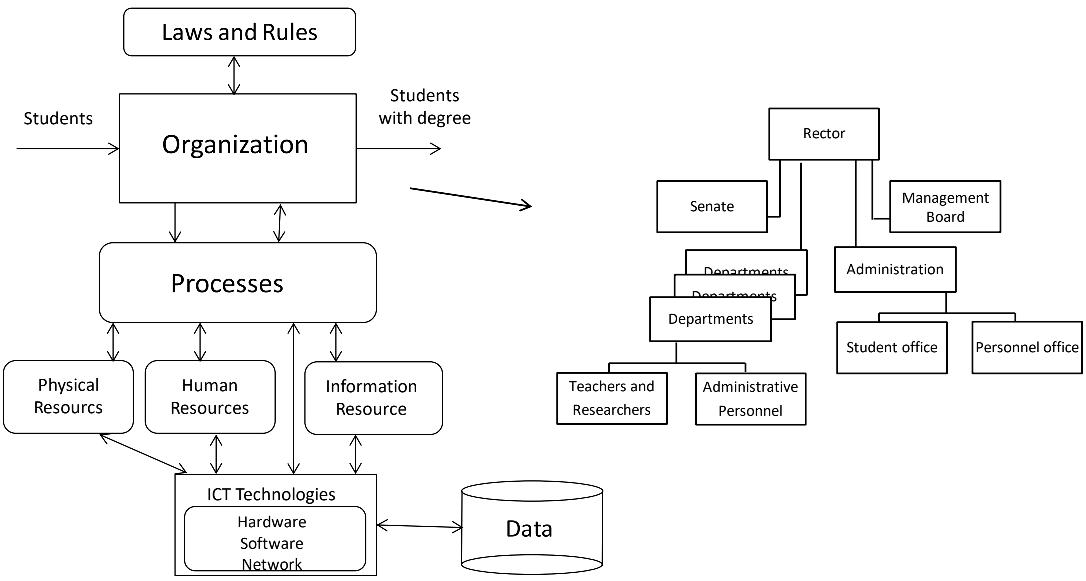


Figure 1.11 - Organizational system of a University, and its organizational chart

Another (equivalent) representation of organizations is an indented taxonomy, where the parent child relationship is represented by means of a two or more levels columnar tree, such as in the following box.

Rector Senate Board of Directors Administration • Student Office • Personnel Office Department 1 • Administrative personnel • Teachers & Researchers … Department n • Administrative personnel • Teachers & Researchers

1.4.2 The nature of processes

We now go more in depth on processes. Consider the following definition of a process. A process is the set of activities that, using available resources and technologies, an organization performs to produce a good or a service. An activity is a sequence of decisions and actions that produce an output. From the above definition, we argue that processes are performed in organizations, and make use of resources.

In any organization, many different processes are executed. Look at Figure 1.12 that represent several typical processes that are performed in a manufacturing company. Processes in the bottom represent the typical manufacturing production chain, made of inbound logistics, production, marketing, sales, and outbound logistics. Such processes are the so called primary processes since they represent the primary set of activities that have to be performed to transform input goods (e.g. a battery for a car) into output goods (the car).

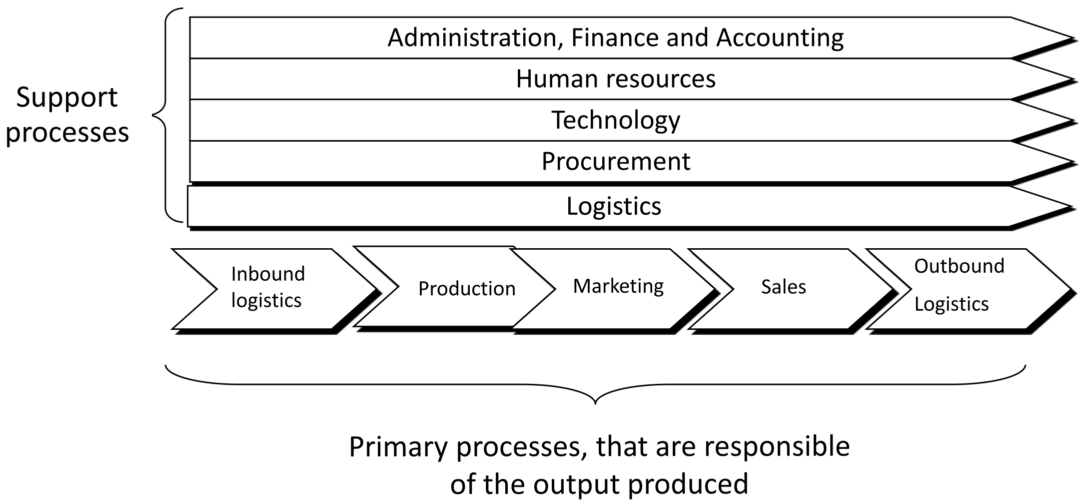


Figure 1.12 - Types of processes performed in a manufacturing company

Examples of primary processes in a University are orientation, teaching, making research, making exams, etc. Processes in the top part of the figure are not directly involved in output production, while they provide resources (economic, human, and technological) to primary processes; for this reason, they are called support processes.

As we see pictorially in Figure 1.12, processes *cross* the organization, since their goal is to transform an input (a good, a request of service, data) into an output, so it is intrinsic to a process crossing one or more functional areas, due to the specialization that such functional areas present. We show two examples of such double role of functional areas and processes in organizations.

The first example refers to the case study of exam registration in a University introduced at the beginning of the chapter. Which organizational units are involved in the process? See the requirements in which organizational units appear in **bold**.

When a student passes an exam, the **teacher** records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the **Department's student office.**
The **student secretary** makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.
When the **department students' secretary** has collected all the registers of the exam session, sends them to the **central University students office**.
The **central University students office** re-analyzes the registers for errors and inconsistencies, and checks that students:

- have in their study program the course corresponding to the exam, and
- are in good standing with University taxes.

In case of errors or inconsistencies in the register, the **office** sends the register back to the **teacher**.
When the records are complete, the grades of exams are updated in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

Three different organizational units are involved in the exam registration process, the teachers, the department student secretary and the central student office. We may represent the process mapped on the organizational structure of Figure 1.11, obtaining the workflow highlighted in Figure 1.13.

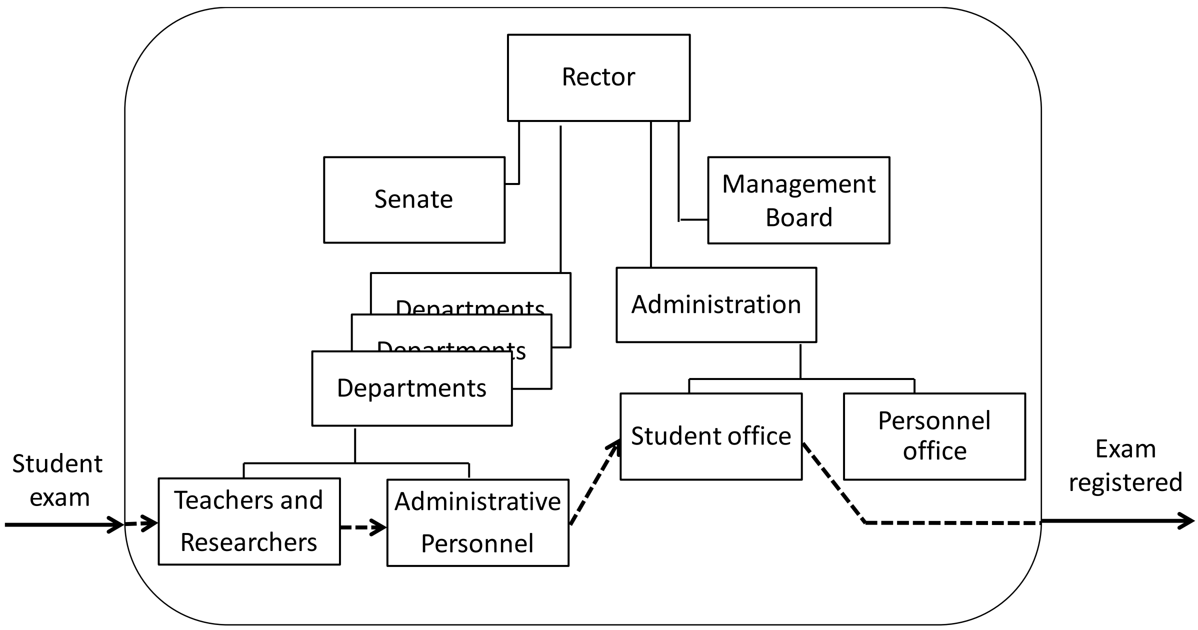


Figure 1.13 - The workflow of organizational structures involved in the exam registration process

As a second example, we consider one of the most popular processes executed in organizations that sell products to clients, the order-invoice process. In Figure 1.14 we show a possible workflow that involves three different organizational structures, the sales office, the accounting office and the production office.

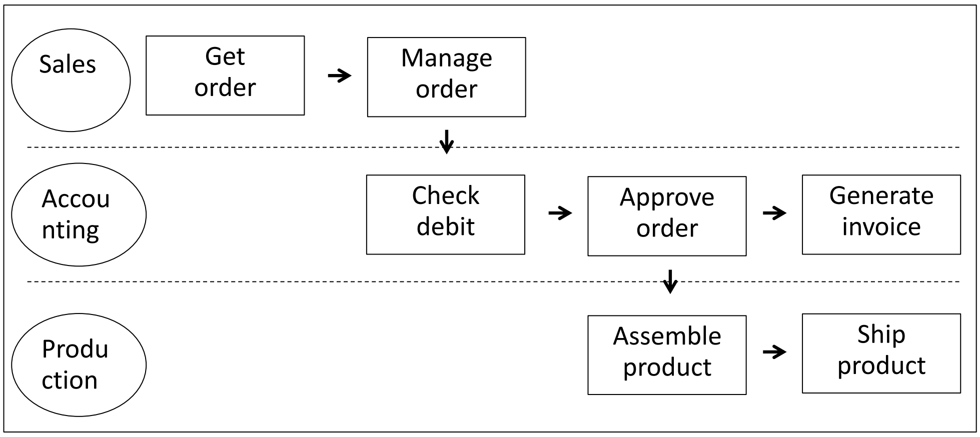


Figure 1.14 - The order-invoice process

1.4.3 Modeling Processes

Similarly to what we have seen for organizations, processes can be modeled in terms of a two/three levels taxonomy, representing processes, sub processes or activities, Looking at the requirements of the exam registration example, we may highlight processes in bold in the following box.

When a student passes an exam, the teacher **records biographical data of the student, course data and the grade** in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary **makes a first check of the completeness** of the registers, and, when he/she finds some void fields, **reports the incompleteness** to the teacher. When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office. The central University students office **re-analyzes the registers** for errors and inconsistencies, and checks that students: • have in their study program the course corresponding to the exam, and • are in good standing with University taxes. In case of errors or inconsistencies in the register, the office **sends the register back** to the teacher. When the records are complete, the grades of exams **are updated in the student's exam database**, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

The corresponding taxonomy is made of a single process, the exame registration, that can be seen as composed of five sub processes or activities, as shown in the following box.

Exam registration
    Fill the paper exam register
    Check of paper register
    Report incompleteness

Analyze the paper register Update the exam database

The above list of activities is a very poor representation of the requirements: e.g., we do not represent the control flow among activities, decision points, the flow of data between activities. Therefore, we need a richer modeling language; several languages have been proposed in the last twenty years for modeling processes, among them we choose in this book the Business Process Modeling Notation (BPMN). A description of the above requirements in terms of BPMN appears in Figure 1.15 where the main modeling constructs of BPMN are highlighted.

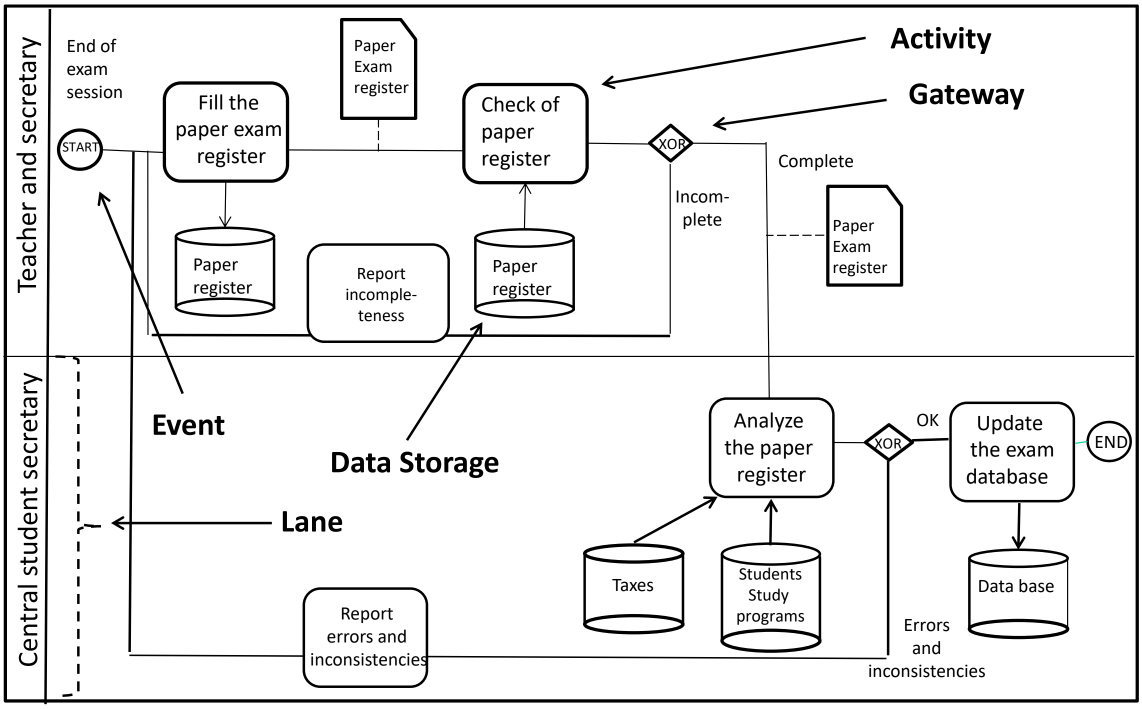


Figure 1.15 - A Business Process Modeling Notation description of the exam registration example

We provide now an intuitive explanation of the meaning of symbols, a more systematic introduction to BPMN appears in Chapter 3. Lanes represent organizational structures, in this case we have two lanes corresponding respectively to the teacher and department secretary and to central student office respectively. The process is activated and concluded by an event, represented by a circle; activities are represented by rounded squares; the flow of control in the process is represented by lines, while points of decision, called gateways, are represented by rhombuses.

1.5.3 Modeling Data

Besides the organization and processes, data are the third basic component of Information systems. Let us consider again the requirements of the exam registration case study; this time let us focus on the parts of sentences that focus on data. We highlight them in bold.

When a student passes an exam, the teacher records b**iographical data of the student, course data and the grade** in a **paper registry**. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.

When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students:

- have in their **study program** the course corresponding to the exam, and
- are in good standing with University **taxes**.

In case of errors or inconsistencies in the register, the office sends the register back to the teacher.
When the records are complete, the **grades of exams** are updated in the student's exam database, along with the **course passed and the grade**; in case of **Erasmus students**, the **country** of the student is represented.

Similarly, we can model the different types of data in the requirements by means of a taxonomy of data types and their properties, resulting more or less in the taxonomy of Figure 1.16.

Paper registry
Student
    Code
    Surname
Taxes
Erasmus Student
    Code
    Surname
    Country
Course
    Code
    Name
Exam
    Student
    Course
    Grade
Study program
    Student
    Course
Taxes
    Student
    Payment

Figure 1.16 - Taxonomy for types of data and their properties

A conceptual model widely diffused In the database community to represent data is the Entity Relationship Model. Using the Entity Relationship (ER) model we can describe a reality involved in the Information system in terms of a set of modeling constructs that represent classes of data (e.g. the class of Students, the class of Courses, etc.). As in the case of processes, we will provide more detail and definitions related to the Entity Relationship model in Chapter 3. In Figure 1.17 we see the data requirements involved in the exam registration example in terms of a diagrammatic representation usually adopted for the ER model.

The class of students is expressed by the entity Student and the corresponding rectangle; students have three properties called attributes in the ER model, namely Code, Surname and a binary attribute Tax paid? that informs whether the related student has paid the taxes or not. Other entities are Course, with attributes Code and Name, and Erasmus Student, a specific type of Student that has associated an attribute Country. Student and Course are related by the two relationships Study program, and Exams, each exam is characterized, besides a student and a course, by a grade.

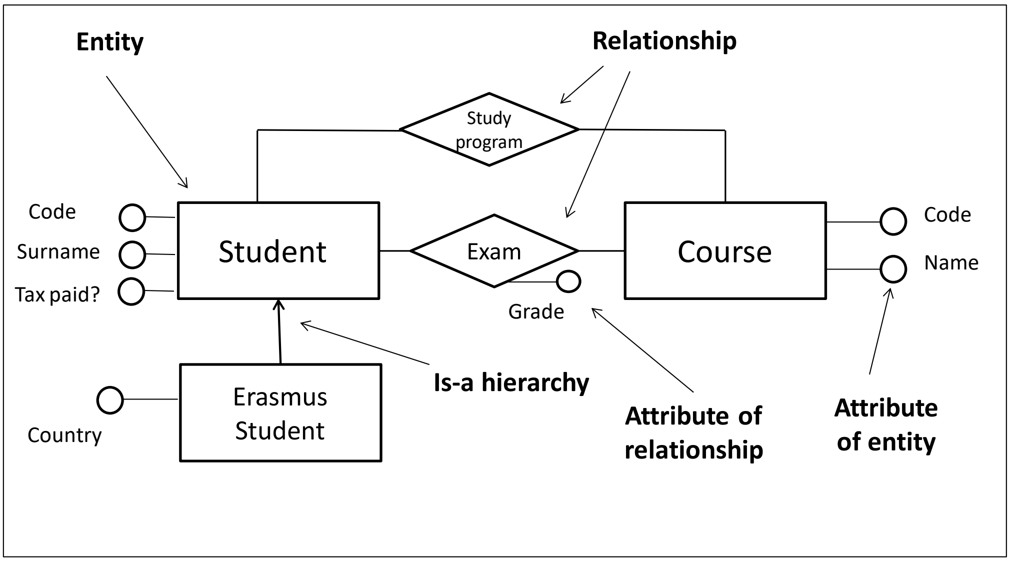


Figure 1.17 - An Entity Relationship conceptual data schema

1.5. Efficiency and effectiveness of information systems and processes

Processes are deeply linked to the services/goods delivered/produced by the organization, their final output being a good or a service provided to the end user. We have seen in the booking a seat at Scala example the following issue that a process that provides the user a view of the stage is of higher value to the user than a process that provides only a layout of seats.

We may say that the effectiveness for the user of looking at a video is higher than looking at the layout of seats. At the same time, providing the Scala web site with a view of the stage for each seat can be done either recording videos of the stage once for all or else equipping each seat with a web camera that provides the view on demand. The first solution is cheaper than the second one, so we may say that it is more efficient. From this example, we see that we need methods to be able to reason on two characteristics of processes, namely:

- efficiency, that corresponds to the minimization of resources used to produce the output, and
- effectiveness, the maximization of value to the end user provided by the delivered good or service.

We can similarly reason on the efficiency and effectiveness of the process for exam registration. The efficiency can be measured in terms of the human resources needed to register an exam; such indicator is influenced by the availability of a software application that aids the teacher and the secretaries in managing the data on exams. The effectiveness can be measured in various ways, e.g. in terms of the time passed between the end of the exam and the time the data on grade is inserted in the database. The time is influenced among others e.g. by the possible incompleteness of data in the paper registry, that leads to a recycle from the department secretary to the teacher and the consequent loss of time.

Efficiency and effectiveness of processes have a central role in the information life cycle. We will discuss in depth the two concepts in the book.

1.6. Evolution of Information Systems toward Networked eBusiness Information Systems

Information Systems are rapidly evolving and changing their internal organization. This is due to several reasons:

1. Larger and larger amounts of digital, processable data are available in information systems, leading to the phenomenon of big data.
2. The format of data is changing; in years 70' of the last century the great majority of data were structured, and managed in data base management systems (see Figure 1.18). Nowadays the great majority of data are unstructured, and are represented in a variety of formats such as unstructured texts of various types (think to the difference between the text in a scientific book and a twitter message..), images, maps, videos, etc.
3. The value for business processes of organizations lies in the ability to make heterogeneous data to coexist in the information system, and be linked and merged into integrated knowledge, leading to adopt new models of data based on knowledge graphs.

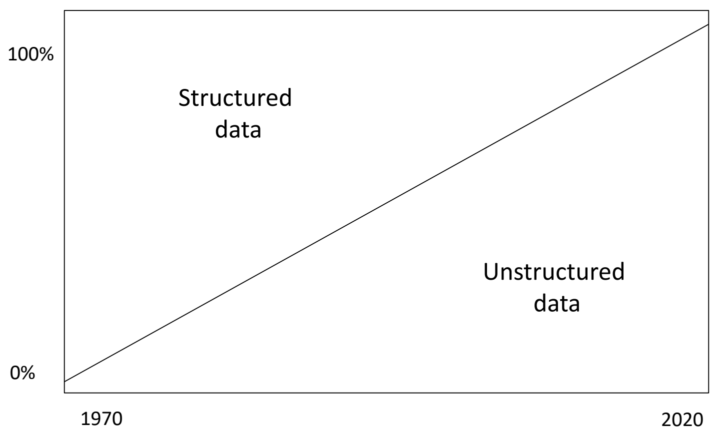


Figure 1.18 - Change of relevance of structured vs unstructured data in the last fifty years

4. The world is changing too, and globalization pushes for the dynamic creation of organizational networks that communicate with clients and users through social networks and web sites (see Figure 1.19).

For instance, think to Amazon and the extremely complex business-to-business networking needed to deliver digital and physical good to clients. New technologies seen in Figure 1.3 become the principal interface with clients; they enable the great transition from traditional to networked information systems, and networked e-business. In this context, inter-organizational core business activities exploit networked electronic business (e-business) technologies to create dynamic collaborations, enabling the integrated use of information technology for both communication and processing of information.

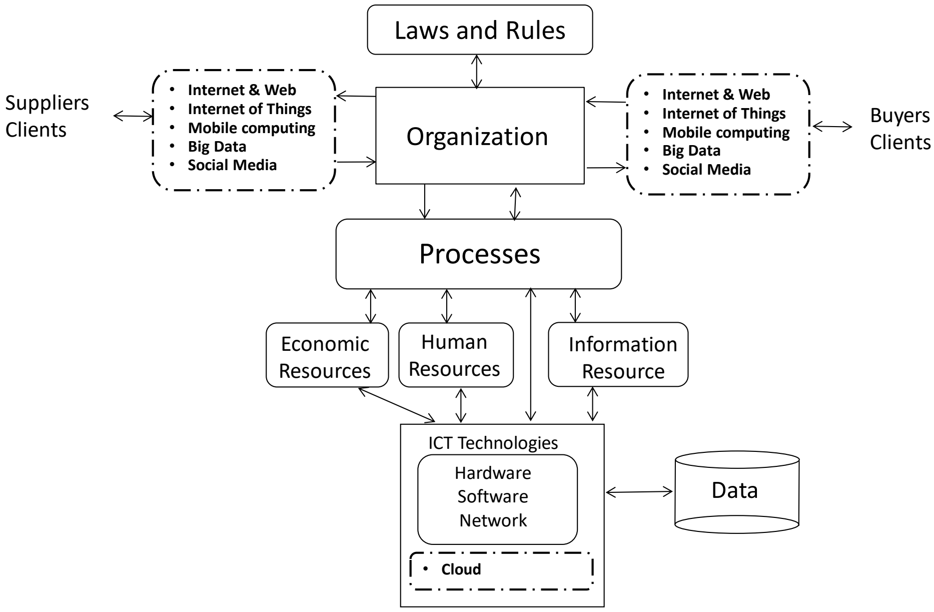


Figure 1.19 – The role of the big five technologies in modern information systems

1.7. Two alternative points of view in Information System Design

The reader has probably perceived that designing an information system is a complex issue that forces the designer to deal with several issues shown in the front page of the book and reproduced in Figure 1.20. Such issues are deeply interlinked each other, as shown in the figure.

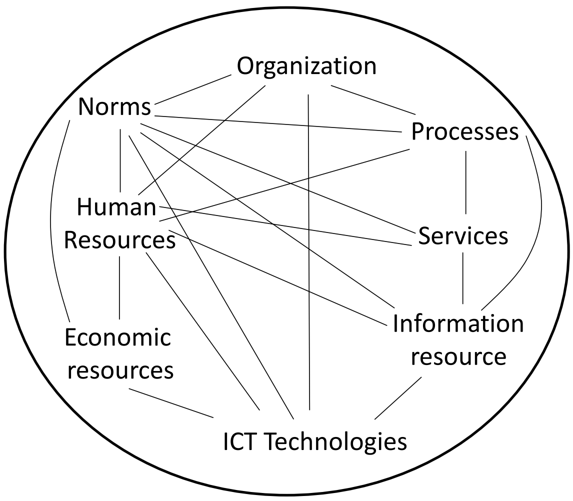


Figure 1.20 – Issues involved in information system design

Information system design can be performed according to two quite different strategies that may be classified as technology driven or service driven.

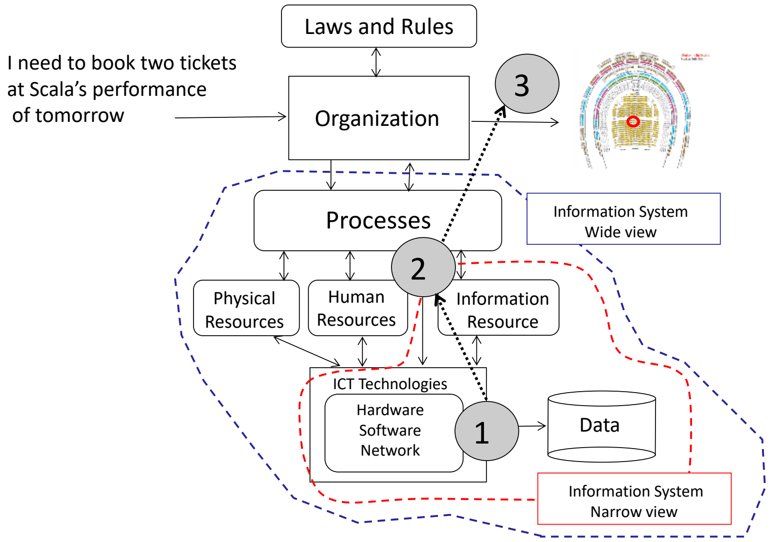


Figure 1.21 - Traditional technology driven information system design

In a technology driven approach (Figure 1.21), the attention is focused initially on technologies; we decide first e.g. if the system will be centralized or distributed, we design and size the hardware infrastructure in terms of servers and personal computers, and the organization of data in databases; then we define the activities to be performed by means of software applications, and finally we try to understand the effects on services (or goods) produced and delivered. Looking again to the Scala's example, in this case, since we have considered user needs only at the end of the design process, it is most likely that we would have conceived a process that delivers a service of lower value.

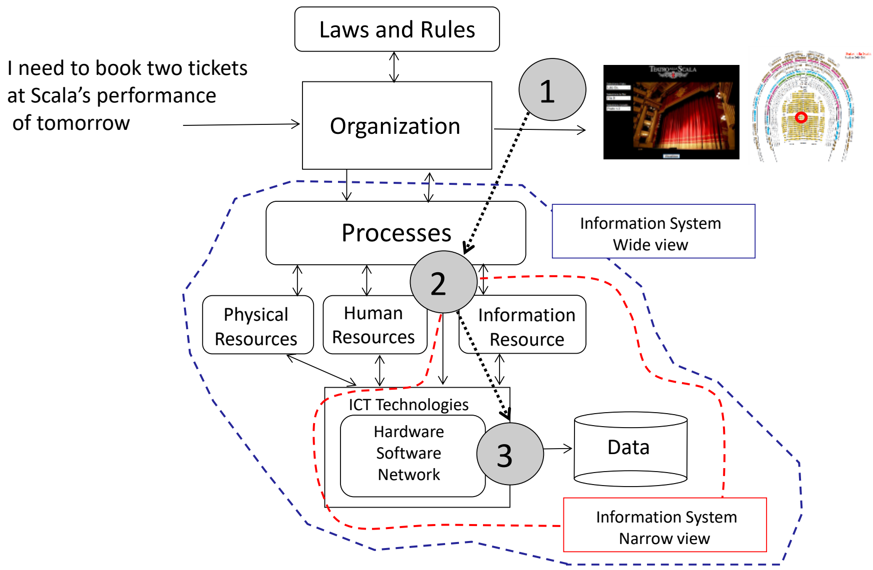


Figure 1.22 - Service oriented Information System design

If, on the contrary, we start the design from user needs (see Figure 1.22), there is a higher chance that service requirements will include requests such as experiencing the view of the stage from the seat.

1.8. The Information System life cycle

Figure 1.23 highlights all activities of an information system life cycle. The inputs are the requirements of users expressed in natural language; requirements are used to produce in output:

- the main processes performed in the organization
- the conceptual description of data used by processes
- an architecture and a sizing of all the technological components adopted,
- the related costs

that are chosen/needed to achieve the goal of maximizing the efficiency and effectiveness of processes performed in the information system and the value in use of related services.

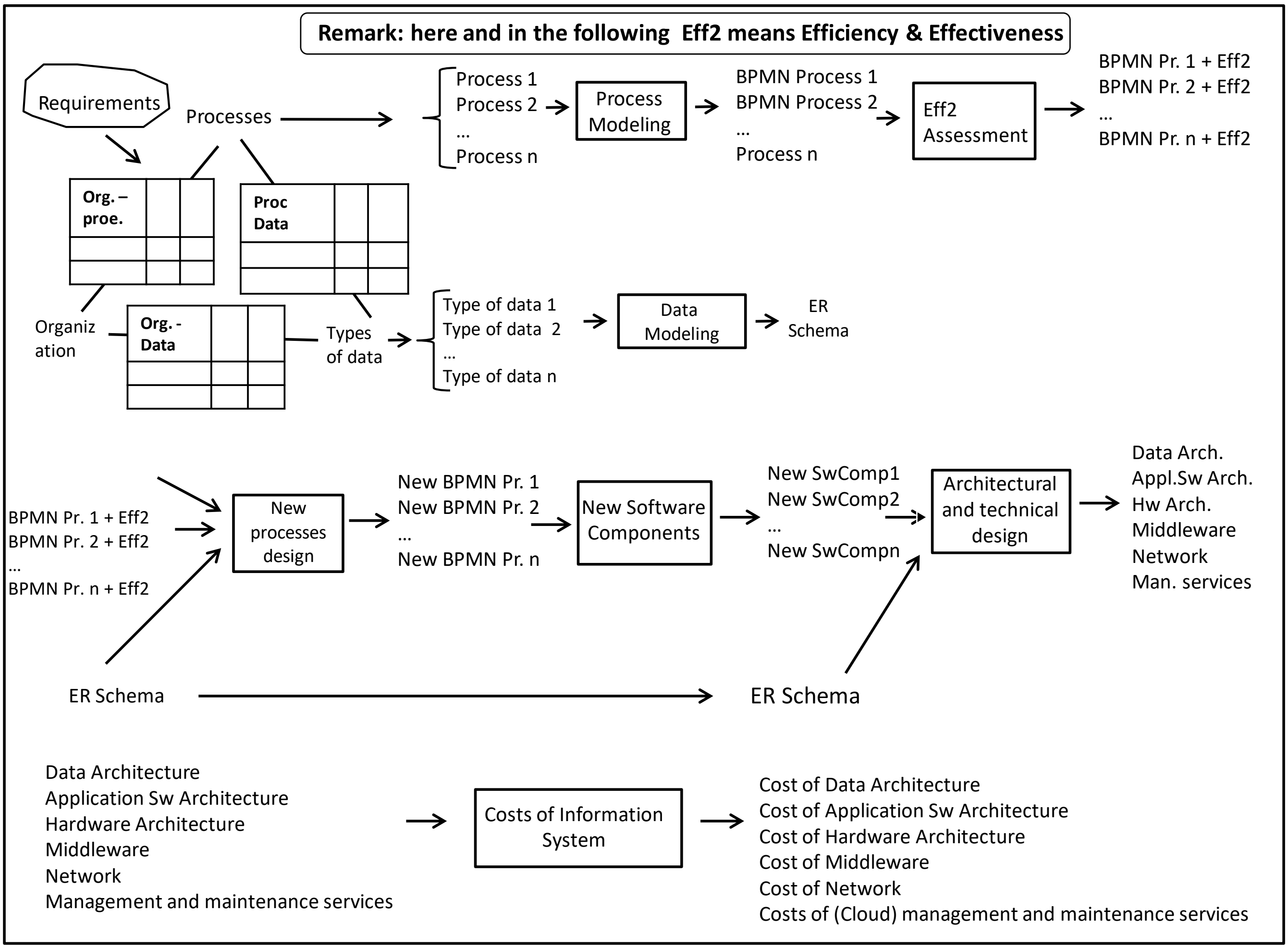


Figure 1.23 - The life cycle of Information System Design

The different activities of the Information system life cycle may be grouped into seven phases, see Figure 1.24, namely:

1. **State reconstruction**, whose goal is to build a complete picture of the organizational structures, the processes, the data, and of relationships among them (which data is used by each process, which process are executed by each organizational unit, which data are managed by each organizational unit), represented by means of matrixes (see Chapter 2). Organization, processes and data may be initially described by means of the taxonomies introduced above, and then making use of the BPMN and ER models.

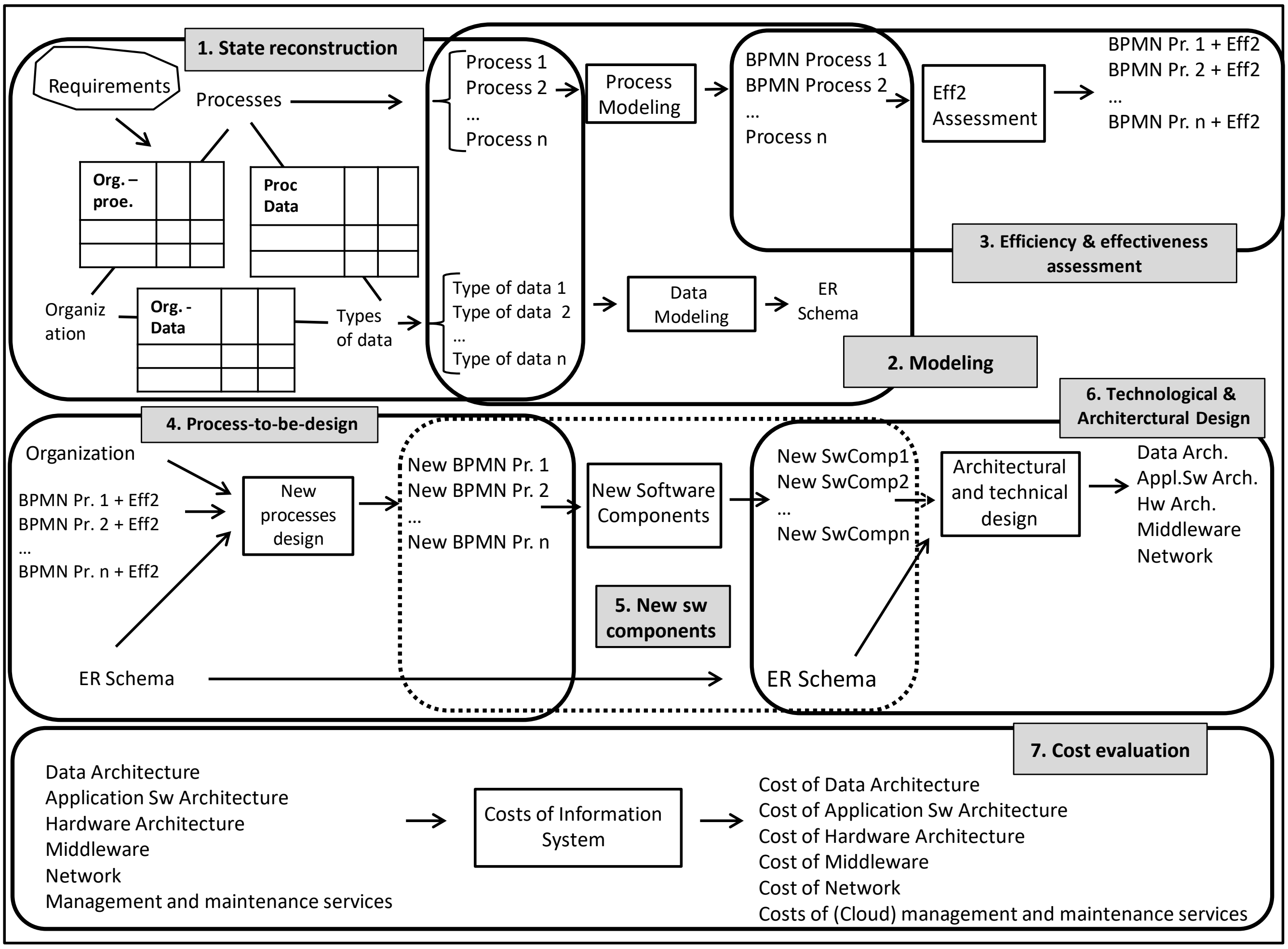


Figure 1.24 - The seven activities of the Information system design

2. **Efficiency and effectiveness assessment**, during which measures of efficiency and effectiveness of processes as-is (namely, the actual processes performed in the organization) are defined, that allow to assess their overall quality. When assessing quality of processes, indications are collected that provide cues for the subsequent possible process reorganization.

3. **Modeling**, whose goal is to build a semantically rich representation of processes and data in terms respectively of the Business Process Modeling Notation and the Entity Relationship Model.

4. **Process-to-be-design** where, based on previous assessment processes are redesigned taking into account ICT technologies that may improve their efficiency and effectiveness.

5. **New software components**, that starting from BPMN processes leads to identify functionalities to be implemented ad software applications.
6. **Technological and architectural design**, where technologies are mapped to the organizational structure, specifying the technological hardware, software and data architectures, e.g. whether hw/sw/data are managed centrally or distributed in peripheral organizational units.

7. **Costs evaluation**, where based on previous technological choices and sizings, economic costs of the information system production and operations are determined.

To be afforded, the above life cycle needs significant human resources. At the same time, the goals and context of IS design can be diverse; in Figure 1.25 we show four different scenarios related to IS design and the different activities involved in each of them.

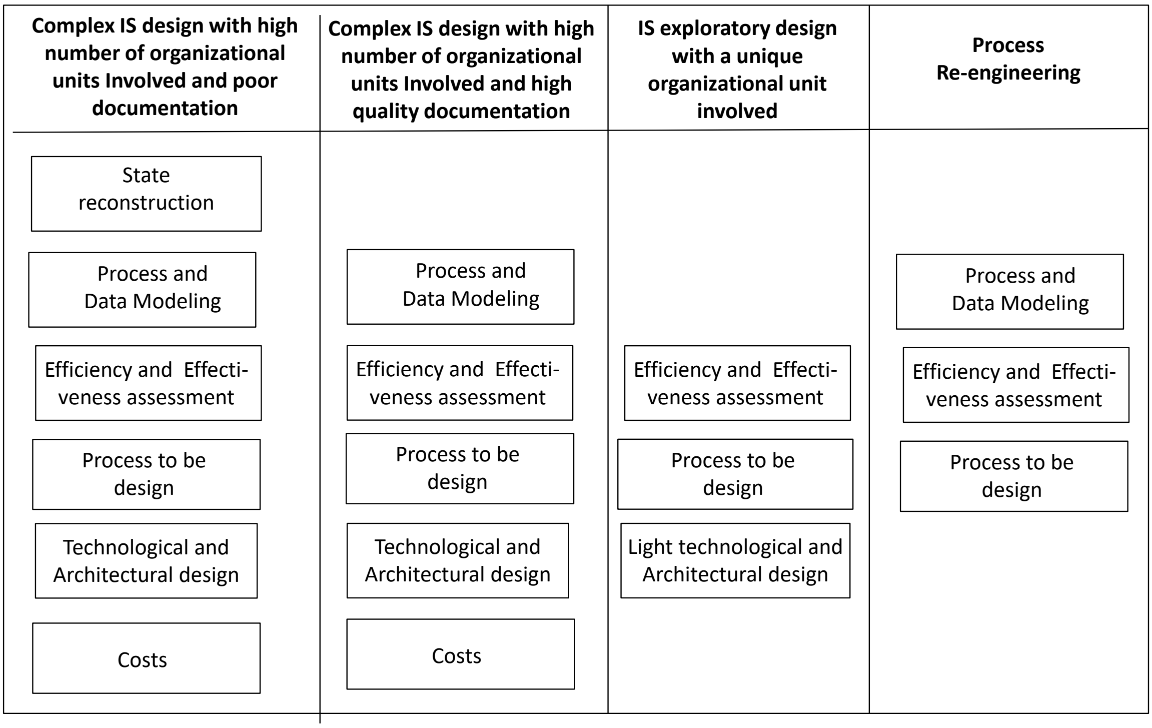


Figure 1.25 – Different scenarios for Information system design

They correspond to:

1. Complex IS design with high number of organizational units involved and poor documentation, in which we have to perform all phases.
2. Complex IS design with high number of organizational units Involved and high quality documentation
available, where due to the presence of rich documentation we may skip the initial State reconstruction activity. Both in the previous scenario and in this one, cost analysis may not be needed when the design activity is performed on a preliminary basis, to choose among possible different alternatives.
3. IS exploratory design with a unique organizational unit involved, where we may skip also the modeling activity, and due to the exploratory nature of the design, perform only a *light* technological and architectural design, where the choice of technologies and their allocation in the organizational structures is performed only at the macro level, or not performed at all.

4. Process Re-engineering, where the goal is to focus on specific relevant processes, on which an extensive assessment and re.-design activity is performed and the technological and architectural design is not requested.

Notice that we may follow a "faster" life cycle every time we want to realize a new process, not performed formerly, or else we do not want to waste time and resources in modeling the present state of the information system. In this case, we may also avoid producing the three matrixes, and focusing only on the relationship between processes and data, seeing Figure 1.24.

We provide in the following box the organization of the book in chapters; after the initial chapter in which we have introduced and motivated all relevant concepts on the information system structure and design life cycle, the chapters of the book are focused on the seven activities introduced above. A final Appendix focuses on an issue specifically relevant in Data Governance, the Data Architecture Governance activity, and on a conceptual tool that may be used to investigate the data architecture, the Repository of conceptual schemas.

We make use of two main case studies that we have introduced in the chapter:

- The Exam registration case study.
- The Railway company case study (see Appendix 1.2)

Reader, enjoy the reading!

Appendix 1.1 – Solutions to exercises

Solution to Exercise 1.1

One possible solution to Exercise 1.1

Process	Who & Which Technology
Student enrollment	Student + Web site
Preparing a lesson	Professor + eLearning Platform
Teaching a course	Professor + Projector + ..
Pay a salary	Administrative Staff + Salary SW Application + Personnel Data Base

Solution to Exercise 1.2

Producing a ranking of goodness or serviceness for services mentioned in Exercise 1.2 is an highly subjective matter. Anyhow, a reference ranking based on a survey for 49 artifacts that include the above twelve has been presented in [Jacobucci 1993], in the following figure we show the correspondence between 11 of the 12 artifacts and the ranking in [Jacobucci …], with the only exception of the e-Book, that at the time of the survey did not exist. I think that the reader agrees on the fact that a paper book is closer to the intuitive concept of good than a typical eBook, that, contrary to what happens for paper books, provides a set of micro-services such as finding all occurrences of a word in the text of a book, increasing or shrinking the dimension of characters, etc.

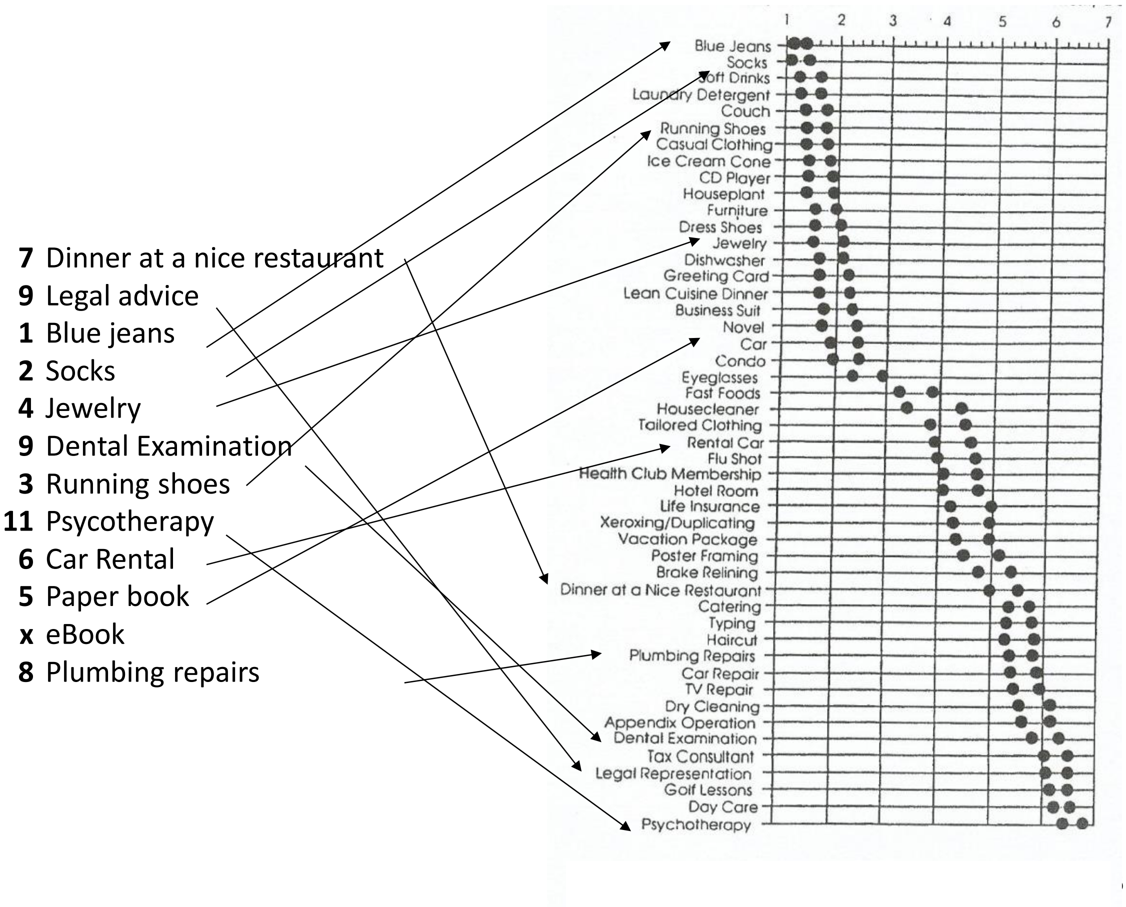


Figure 1.26 – Degree of goodness and serviceness of a list of artifacts that we experience in the real world

Possible Solution to Exercise 1.3

Layers at University of Milano-Bicocca, Italy

EXT-ENV - External environment: High schools, private/public school, international school sector (high, competitors), public enviroment, enviroment near by the university (print-points, bar, shops, food court…).
US - Users requesting services: High schoolers, international students
USSEG – User segment: e.g. Foreign students
CRS - Companies requesting services: private sector (e.g. Fastweb, Pirelli…), companies that need a high qualified workforce.
NC - Networked companies: Other Universities, the Ministry of University
SG - Suppliers of goods: food & beverage machines, Banca Popolare di Sondrio…
RS - Requests for services: students, researchers, teachers, retired personnel
SS - Supply of services: University itself through teachers, private companies, Banca Popolare di Sondrio.
SER – Services provided: Bachelor & Master degrees, Ph.D, Research
SERDE - Service description: The main goal of the University is to provide higher education, so all the services are in such way oriented to improve the overall education of the students
SERME - Measured levels of service: It's done through ICT
SERPE - Perceived quality of service: It can be measured via online/paper survey
LR - Laws and regulations: CdA (Management Board), Senate, CdS (Council of Students).
OR – Organization
CS – Central organizational structure: Rettorato, CdA, Senate, CdS, administration
PS – Peripheral structure: Departments
IU – Internal users: Teachers, Students, Administrative and Technical Personnel
PR – Processes
PRI Primary processes: Education, Research.
SUP – Support processes: Human resources management, Extra-curricular activities, others
RES - Resources

- ECON – Economic resources: taxes, funds from Regione Lombardia, funds from companies which require services or research…
- HUM – Human resources: student office, personnel office, secretaries.
- INFR – Infrastructural resource: Campus, Student Housing, warehouse.
- INF – Information resources: e.g. teacher class exercises on the blackboard.
- TECHN – Technologies

WEB – Website: unimib.it, Moodle e-Learning platform,
SWAPPL – Software application: Management of Unimib courses, study plans, exams, e-Learning. others
SWMID – Software Middleware: help desk, department's own software
CHW – Central hardware: Servers, computers, cameras,
DHW – Distributed Hardware: cloud systems, laboratories…
I/O – Input output units: Printers
NTW – Network: geographic network, local networks, Internet
DB – Data Base: Student's exams, student's biographic information, workers information, accountability & financial information…

Appendix 1.2

Exercise 1.4 - Modeling requirements of the information system of a Railway company

In the following box you find terms and acronyms referred to all the layers and components of an information system. Furthermore, you find a detailed description of the information system of a railway company. The goal of the exercise is to identify in the text the parts that correspond to the specific layers and components, in such a way to make you better understand their meaning and role in the information system structure. For each part of the text referring to a layer or component, you have to place in a box the text and at its beginning write the acronym of the corresponding layer or component.

Layers and components of an Organization and an Information System

EXT-ENV - External environment
US - Users requesting services
USSEG – User segment (e.g. young people)
CRS - Companies requesting services
NC - Networked companies
SG - Suppliers of goods
RS - Requests for services
SS - Supply of services
SER – Services provided

- SERDE - Service description
- SERRE – Service restrictions (currently not provided)
- SERME - Measured Levels of service
- SERPE - Perceived Quality of service

LR - Laws and regulations
OR – Organization

- CS – Central org. structure
- PS – Peripheral structure

IU – Internal users
PR – Processes

- PRI - Primary processes
- SUP – Support processes

RES - Resources

- ECON – Economic resources
- HUM – Human resources
- INF – Information resources
- INFR – Infrastructural (Physical) resources

TECHN – Technologies

- WEB – Website
- SWAPPL – Software application
- SWMID – Software Middleware
- CHW – Central hardware
- DHW – Distributed Hardware
- I/O – Input output unit
- NTW – Network
- DB – Data Base

Requirements of the information system of a railway company

We assume that a railway company has an information system that manages all the processes of the company, and in particular the processes of booking and purchasing tickets, accessible from the physical counters of railway stations, the affiliated agencies, a telephone call center, some self-service terminals located in the main railway stations, and a website. There are 2,000 physical stations, divided into 50 "large stations" where the train is used by more than 1 million ticket users per year, and 500 small stations, where the train is used by less than 1,000 ticket users per year. There are also about 20,000 employees at the stations, of which 3,000 in the large stations, and 2,000 in the small stations. Finally, there are 10,000 employees, including traveling staff and administrative staff. The network is made up of approximately 1,000 kilometers for high speed and 4,000 kilometers for normal speed trains. Finally, there are another 5,000 employees with administrative tasks (personnel, accounting, etc.) operating in the 20 regional branches distributed in the 20 Italian regions.

The main primary processes of the company concern:
1. The organization of journeys of travelers, through a fleet of about 100 high-speed trains, composed of 1,000 wagons. The high-speed users are around 1 million, with an average journey of 300 kilometers, while the users of "normal" trains are 5,000,000, with an average distance of 50 kilometers. Currently, this process, for the part of booking and ticket purchase, uses in addition to internal channels, about 1,000 external agencies, to which a fee equal to 10% of the ticket is paid.
2. The organization of travels for goods, through a fleet of about 50 trains.
3. The organization of special trains, through a fleet of about 20 trains, and 200 wagons.
4. The organization of night trains, about 20, which however cause significant security problems, and have high costs because external staff must be hired by a very expensive, specialized company.

The support processes are the usual ones for personnel, for administration and accounting, for logistics. With regard to staff, night-time rest has to be managed, which is currently 80 per cent in apartments owned by the company, and 20 per cent in hotels.

The main databases concern:
- The railway network, fairly stable over time.
- Trains available
- Railway timetable, which is updated every year according to the profitability of the routes.
- Train journeys of the three types.
- Staff
- Stations
- Lost items

The system has several restrictions.
- L1 does not allow you to make and modify bookings from mobile phones, allowing booking changes exclusively through the same channel as the initial booking;
- L2 does not provide services to particular categories of users, such as disabled motor users or users with a bicycle;
- L3 reservations are made by class, and it is the system that chooses the wagon and seat;
- L4 the website is not accessible to some categories of disabled people (the blind) and is available only in Italian;
- L5 Complaints can be sent only by ordinary mail, and not by e-mail; the answers to the complaints arrive on average after one month from their submission; the organization does not provide information on the process of examining the practice and reimbursement.
- L6 Self-service terminals are equipped with an interface considered difficult to use by customers and available only in Italian;

• L7 Currently the distribution of tickets booked and purchased is 70% at the counter and 30% online. This makes the costs of the front office high; the decision has been made to reverse these percentages, to allow savings in the front office structure. Moreover, we want to understand over the years what the actual savings will be.
• L8 does not allow travelers and internal managers to monitor service levels.
• L9 has high personnel management costs for personnel that must stay overnight.
• L10 there are about 10,000 complaints a year due to delayed arrival, which leads to a significant cost for the company.
• L11 up to now no historical series statistics have been collected, aimed at measuring, through a temporal benchmarking, the trend of a set of quality indicators.
• L12 no objective quality metrics are collected and there are no processes of measuring quality perception.

In addition, we want to improve the image of the company, through:

- the use of a larger number of channels,
- the provision of additional services in addition to the booking and purchase of tickets,
- the ability for travelers to book other services once they arrive,
- a multi-lingual version of the web interface and self-service terminals,
- a more rapid resolution of complaints,
- the introduction of initiatives focused on customer loyalty and improve the accessibility to disabled people of available electronic services.

The intention is to motivate staff by introducing forms of incentives linked to productivity and service levels, which must observe the overall quality, in both objective terms and the perception of users acquired from paper forms currently filled by travelers. The company then wants to make a comparative benchmarking with the competition. A survey was carried out on travelers, which led to these needs:
• G1 - travelers (all) would like to be able to book the exact place in the carriages, identified through the vision of the places already occupied, those places that are free and the direction of travel.
• G2 - the business users needs to be assisted for all the phases of the journey; for example, they can ask to book a taxi at the arrival station, or, if the train is delayed, to dynamically reconfigure the reservation of the next connecting train, or the taxi at the arrival station, and be informed of this new set-up reservations.
• G3 The motoric disabled need special places, with a space around the place larger than that of the able-bodied furthermore, they may have the need of a person accompanying them to assist them in the departure and arrival of the train.
• G4 - The owners of bicycles (SU3) are interested in carrying a bicycle, which must have a space in the train that must be booked with a specific procedure.

All previous needs lead to new services. Customers will no longer be indistinct, but must now be profiled according to the following categories:
1. Customers for whom no specific information on their profile is known;
2. Loyalty card customers, whose (of the card) release requests various types of information (for example information that allow them to be placed in the different segments defined previously).

It should also be noted that in the country where the company operates, a law was passed that obliges public and private service providers to make accessible the site through which reservations are made for visually impaired people. This creates a constraint to the system, in the sense that it must be adopted an interface that allows the blind to interact with the site as if he/she were a normal.
A second law that creates new opportunities is the law that regulates the production, provision and management of access cards to services, which enable the possibility to provide services for which access by the user requires a secure identification based on digital signature.

The set of Travel management activities, namely: a. booking, b. ticket purchase, c. request for refunds, d. complaint management, etc. is of responsibility of different organizational structures. Therefore, a user who has to perform all of them needs to interact with different organizational structures, or, in the case of interaction through the website, with different pages and services. For example, if you have purchased the ticket through an agency, in order to change it you must go back to the agency, it is not possible to do it with other channels. This reflects a software organization by sales channels, therefore “vertical” or “stove pipe”, not open to procedural changes.

• It is in the user's interest to be able to interact transparently with the various channels, to know the status of his/her requests or complaints, and do not be obliged to provide the same information to the system several times, as often happens in interactions with public information systems.
• In the existing information system there is already a database of reservations and ticket purchases, while there is not, as mentioned above, a database of registered users. This database is essential for managing all customer relationships; it will certainly be part of, and will have high priority, in the implementation of the new system.

With regard to train failures, which cause delays, we can assume that they are currently communicated via SMS by the staff on the train, and then arrive in an unstructured form. In the future, they will be communicated in semi-structured form in the XML tagging language for an immediate processing and a historical production of statistics on the types of faults.

Finally, user complaints, which will be sent by e-mail, may be marked in the fields Name, Surname, Day and Number of the train to which the complaint refers, while the other information will be represented by free text.
For all the previous points, the student can also access e.g. the Trenitalia website for checks on numbers and details.

Example 1

TECHN – WEB … and a web site.

Example 2 - Sometimes it may happen that you have to use boxes inside other boxes:

PRO - The main primary processes of the company concern:
PRI The organization of journeys of travelers

Appendix 1.3 – Solution to Exercise 1.4

OR - We assume that a railway company has an information system that manages

PR
all the processes of the company

PR-PRI
and in particular the processes of booking and purchasing tickets,

RES-INFR
accessible from the physical counters of railway stations,

NC - the affiliated agencies

RES – HUM & INFR & TECHN , a telephone call center

RES – INFR & TECHN -, some self-service terminals

RES – INFR located in the main railway stations,

TECHN – WEB
… and a web site.

RES – INFR - There are 2,000 physical stations, divided into 50 "large stations" where the train is used by

US - more than 1 million ticket users (utenti – biglietti) per year,

RES – INFR - and 500 small stations

US - , where the train is used by less than 1,000 ticket users. year

RES – HUM There are also about 20,000 employees at the stations
, of which 3,000 in the large stations, and 2,000 in the small stations

OR - . Finally, there are 10,000 employees, including traveling staff and administrative staff.

RES – INFR The network is made up of approximately 1,000 kilometers for high speed and 4,000 kilometers for normal speed trains.

RES – HUM Finally, there are another 5,000 employees

PRO – SUP with administrative tasks (personnel, accounting, etc.)

ORG – PER - operating in the 20 regional branches distributed in the 20 Italian regions.

PRPO - PRI
The main primary processes of the company concern:
1. The organization of journeys of travellers, through a fleet of about 100 high-speed trains, composed of 1,000 wagons. The high-speed users are around 1 million, with an average journey of 300 kilometers, while the users of "normal" trains are 5,000,000, with an average distance of 50 kilometers. Currently,

this process, for the part of booking and ticket purchase, uses in addition to internal channels, about 1,000 external agencies, to which a fee equal to 10% of the ticket is paid.
2. The organization of travel for goods, through a fleet of about 50 trains.
3. The organization of special trains, through a fleet of about 20 trains, and 200 wagons.
4. The organization of night trains, about 20, which however cause significant security problems, and have high costs because external staff must be hired by a very expensive, specialized company.

PRO – SUP - The support processes are the usual ones for personnel, for administration and accounting, for logistics

RES – HUM - With regard to staff, night-time rest has to be managed

RES – INFR ,- which is currently 80 per cent in apartments owned by the company

NC - , and 20 per cent in hotels.

TECHN – DB
The main databases concern:
• The railway network, fairly stable over time.
• Trains available
• Railway timetable, which is updated every year according to the profitability of the routes.
• Train journeys of the three types.
• Staff
• Stations
• Lost items

US - In addition, we want to improve the image of the company,

SER – SERME - through the use of a larger number of channels

SER – SERDE - , the provision of additional services in addition to the booking and purchase of tickets, the ability for travelers to book other services once they arrive, a multi-lingual version of the web interface and self-service terminals, a more rapid resolution of complaints, the introduction of initiatives focused on customer loyalty and improve the accessibility to

USSEG - disabled people

USSERV - of available electronic services.

HUM - The intention is to motivate staff by introducing forms of incentives linked to productivity and service levels, which must observe the overall quality, both in objective terms and the perception of users acquired from paper forms currently filled by travelers.

USPE - The company then wants to make a comparative benchmarking with the competition. A survey was carried out on travelers, which led to these needs:

USSEG
• G1 - travelers (all) would like to be able to book the exact place in the carriages, identified through the vision of the places already occupied, those places that are free and the direction of travel.
• G2 - the business users needs to be assisted for all the phases of the journey; for example, they can ask to book a taxi at the arrival station, or, if the train is delayed, to dynamically reconfigure the

reservation of the next connecting train, or the taxi at the arrival station, and be informed of this new set-up reservations.
• G3 The motoric disabled need special places, with a space around the place larger than that of the able bodied; furthermore, they may have the need of a person accompanying them to assist them in the departure and arrival of the train.
• G4 - The owners of bicycles (SU3) are interested in carrying a bicycle, which must have a space in the train that must be booked with a specific procedure.

USSERV - All previous needs lead to new services.

USSEG - Customers will no longer be indistinct, but must now be profiled according to the following categories:
1. Customers for whom no specific information on their profile is known;
2. Loyalty card customers, whose (of the card) release requests various types of information (for example information that allow them to be placed in the different segments defined previously).

LR - It should also be noted that in the country where the company operates, a law was passed that obliges public and private service providers to make accessible the site through which reservations are made for visually impaired people. This creates a constraint (vincolo) to the system, in the sense that it must be adopted an interface that allows the blind to interact with the site as if he/she were a normal.
A second law that creates new opportunities is the law which regulates the production, provision and management of access cards to services, which enable the possibility to provide services for which access by the user requires a secure identification based on digital signature.

PR – PRI - The set of TRAVEL MANAGEMENT activities, namely: a. booking, b. ticket purchase, c. request for refunds, d. complaint management, etc. is of responsibility of different organizational structures.

SERRE - As a consequence, a user who has to perform all of them needs to interact with different organizational structures, or, in the case of interaction through the website, with different pages and services. For example, if you have purchased the ticket through an agency, in order to change it you must go back to the agency, it is not possible to do it with other channels.

TECHN SWAPPL - This reflects a software organization by sales channels, therefore “vertical” or “stove pipe”, not open to procedural changes.

SERPE - It is in the user's interest to be able to interact transparently with the various channels, to know the status of his/her requests or complaints, and do not be obliged to provide the same information to the system several times, as often happens in interactions with public information systems.

TECHN DB - In the existing information system there is already a database of reservations and ticket purchases, while there is not, as mentioned above, a database of registered users. This database is essential for managing all customer relationships; it will certainly be part of, and will have high priority, in the implementation of the new system.

SERDE - With regard to train failures, which cause delays, we can assume that they are currently communicated via SMS by the staff on the train, and then arrive in an unstructured form, while in the future they can be communicated in semi-structured form in the XML tagging language for an immediate processing and a historical production of statistics on the types of faults.

SERDE - Finally, user complaints, which will be sent by e-mail, may be marked in the fields Name, Surname, Day and Number of the train to which the complaint refers, while the other information will be represented by free text.

References

C. Batini, M. Castelli, M. Comerio, M. Cremaschi, L. Iaquinta, A. Torsello, G. Viscusi - The Smart methodology for the life cycle of services, 2015, Creative Commons licence. freely downloadable at http://hdl.handle.net/10281/98632

M. Boisot - Information space: a framework for learning in organizations, institutions, and culture. London ; New York: Routledge, 1995

P. Grefen – Structured e-Business, 2006-2009, Eindhoven University of Technology, The Netherlands

P. Grefen - Beyond E-Business: Towards Networked Structures, Routledge 2015.

D. Jacobucci, «An empirical examination of some basic tenets in services: goods-services continua,» In Advances in service marketing and management, p. 1992.

D. Moody, and P. Walsh. "Measuring the Value of Information - An Asset Valuation Approach." ECIS. 1999.

Chapter 2 – State reconstruction

2.1 Introduction

In this chapter we deal with the phase of state reconstruction, see next Figure 2.1.

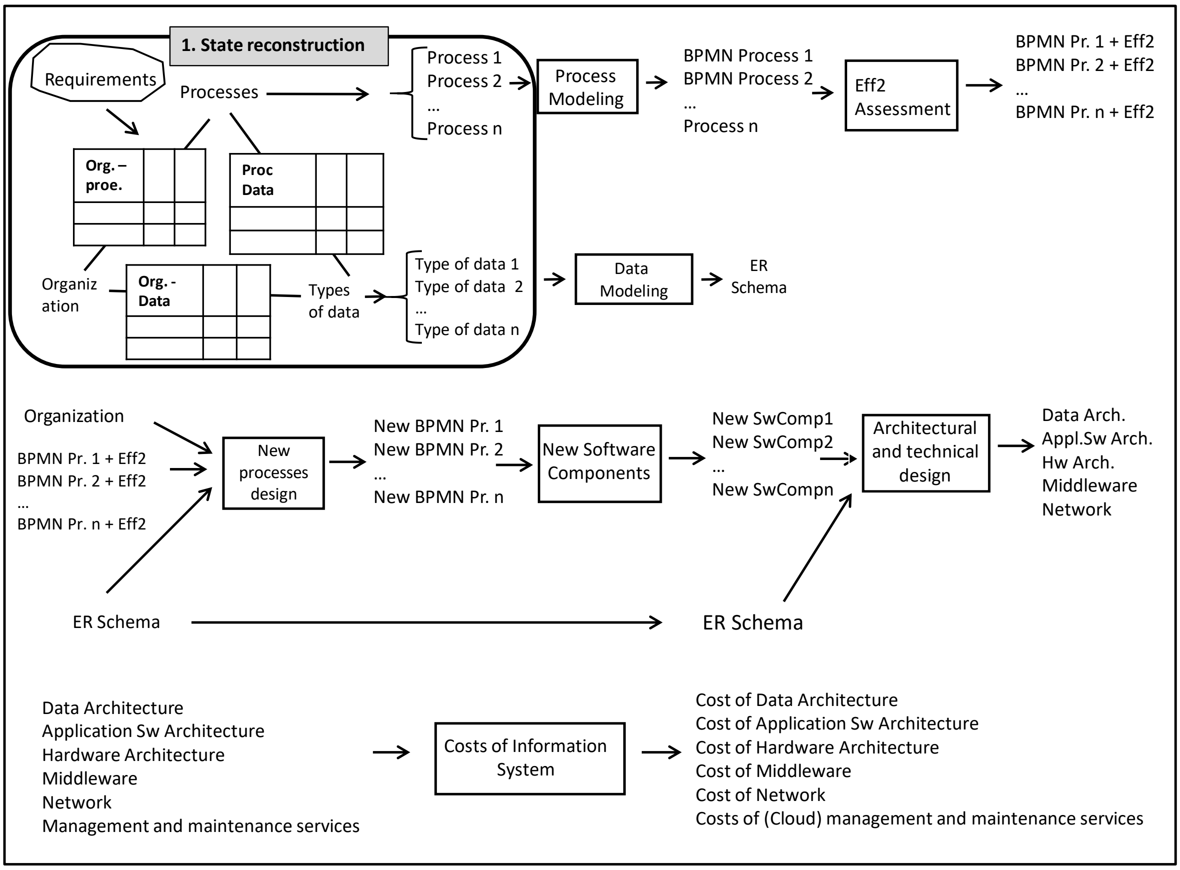


Figure 2.1 - The phase of state reconstruction in the IS life cycle

State reconstruction has the goal to achieve an integrated global view of the three most relevant components of an information system, a. organization, b. processes and c. data. This is done in two steps:

Step 1 - Initially, organization, processes and data are described with simple models, corresponding to a one or two level classification. Furthermore, each one of the three binary relationships existing among them is described by means of a two dimensional matrix, where correspondences (e.g. a process creates a data type) are expressed by means of a specific symbol in the related cell, see Figure 2.2 and the following sections.

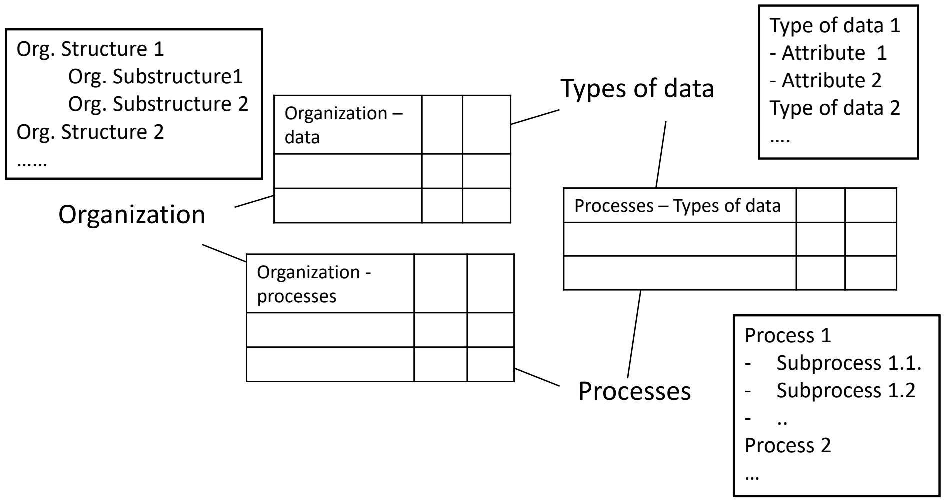


Figure 2.2 – Outputs of the state reconstruction phase

Production of such set of taxonomies and matrixes requests some effort, but the models adopted are sufficiently simple to enable a first feasible output of the state reconstruction activity.

Step 2 - When the knowledge requested on the organization, processes and data, and on their relationships, is not adequately described by classifications and matrixes, we may need to represent organization, processes and data with richer models than classifications, that correspond to:

- Organizational charts for organization
- BPMN model for processes
- Entity Relationship Model for data types.

The flow of activities performed in the first step of state reconstruction appears in Figure 2.3.

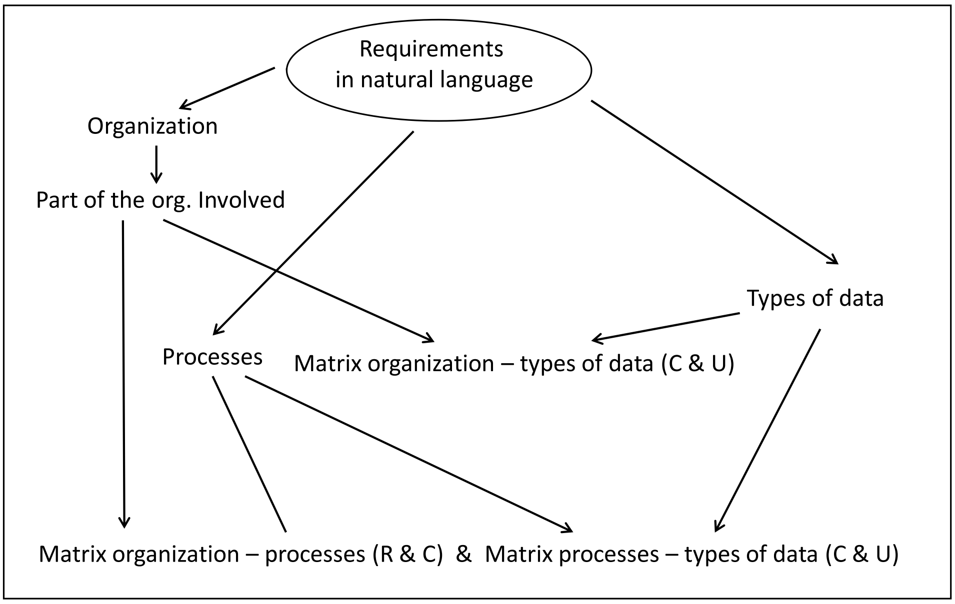


Figure 2.3 – First step of state reconstruction

We proceed now to discuss specific steps. We make use first of the Exam registration case study, whose requirements appear in Section 1 of Chapter 1.

2.2. Part of the organization involved

We have first to highlight the requirements that make reference to organizational structures, see the following box.

> When a student passes an exam, the **teacher** records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the **Department's student office**.
> The **student secretary** makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the **teacher**.
> When the department students' secretary has collected all the registers of the exam session, sends them to the **central University students office.**
> The **central University students office** re-analyzes the registers for errors and inconsistencies, and checks that students:
>
> - have in their study program the course corresponding to the exam, and
> - are in good standing with University taxes.
>
> In case of errors or inconsistencies in the register, the office sends the register back to the **teacher**.
> When the records are complete, the grades of exams are updated in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

Now we map such organizational structures on the organizational chart, that we reproduce in Figure 2.4 in a complete version, where both the structures involved in primary processes and those involved in support processes are represented.

Rector
Senate
Management Board
General Manager
Central Administration

- **Student office**
- Personnel office
- Research projects division
- Accounting division
- Infrastructures division
- Technological systems division

Decentralized units

- Department 1
    - Administrative secretary
    - **Students secretary**
    - Technical staff
    - **Teachers and researchers**
- Department 2
    - Administrative secretary
    - **Students secretary**
    - Technical staff
    - **Teachers and researchers**
- Department n
    - Administrative secretary
    - **Students secretary**
    - Technical staff
    - **Teachers and researchers**

Figure 2.4 – The organizational structure of the case study highlighted in the global organization structure of an University

2.3 Types of data

The types of data in the one level taxonomy are highlighted in the following box.

When a student passes an exam, the teacher records **biographical data of the student**, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office.
The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.
When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students:

- have in their **study program** the course corresponding to the exam, and
- are in good standing with University **taxes**.

In case of errors or inconsistencies in the register, the office sends the register back to the teacher.
When the records are complete, the grades of exams are updated in the **student's exam database**, along with the **course passed and the grade**; in case of Erasmus students, the **country** of the student is represented.

2.4 The Organization – Types of data matrix in the Exam registration case study

We now proceed to populate names of rows and columns of the Organizational structure – types of data matrix with a one level taxonomy of organizational structures and one level taxonomy of types of data (see Figure 2.5).

Org.-types of data	**Students Biographic data**	**Students – Study program**	**Students Taxes**	**Exam paper registry**	**Exam – Data base**
Teacher					
Department student secretary					
Central student office					

Figure 2.5 - Organization - types of data matrix with names of rows and columns filled

We have to give a meaning to the cells of the matrix. We fill the cells with two values:

- **Creates** (C) means that the organization unit in the row is responsible of creating values of the data type in the column; this means that is up to the organization unit to certify the quality of data, in terms of accuracy, completeness, currency and consistency with other data. Furthermore, the organizational unit establishes the right of access to data attributed to other organizational units.
- **Uses** (U) means that the organization unit in the row uses the data type in the column.

So, saying that the organization unit OU in the row is responsible of creating the data type DT in the column correspond to a transaction (changes the content of data).
Saying that the organization unit OU in the row uses the data type DT in the column corresponds to a query (extracts data without modifying the information content).

Exercise 2.1 - Fill the two cells with the question Uses or Creates? in the following matrix of Figure 2.6.

Org.-type of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Teacher	Uses or Creates?			Uses or Creates?	
Department student secretary					
Central student office					

Figure 2.6 – Matrix of Exercise 2.1

Here and in the following, try to formulate a solution; find the solution in the next page

Solution to Exercise 2.1

Org.-types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Teacher	U			C	
Department student secretary					
Central student office					

The first cell has value Uses, since the teacher uses Student biographic data to fill the paper registry (Remark: the teacher does not create such data, he/she simply copies such data in the paper registry). Therefore, for the same reason the second cell must be filled with a Create.

Exercise 2.2 - Fill now cells with U or C in the following matrix, see Figure 2.7.

Org.-type of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Teacher	U			C	
Department student secretary				U or C?	
Central student office	U or C?	U or C?	U or C?	U or C?	U or C?

Figure 2.7 - Matrix of Exercise 2.2

Solution to Exercise 2.2

Org.-type of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Teacher	U			C	
Department student secretary				U	
Central student office	U	U	U	U	C

The cell in the second row has to be filled with U since the department secretary simply checks the register for incomplete data. The cells in the third row are all U's except the last one, since the only data that are created by the central student secretary are the data inserted in the exam database, other data are only read or checked, see the following matrix.

Exercise 2.3 - Are there other cells that have to be filled with U's or C's?

Solution to Exercise 2.3

We can check the remaining cells by rows or by columns; in both cases we do not identify other usages or creations of data.

2.5 Processes

In order to identify processes, we scan the requirements to highlight in bold the sentences that express actions, and are candidates for processes, see Figure 2.7 where the processes are also listed at the end, sometimes suitably renamed.

When a student passes an exam, the teacher *records* biographical data of the student, course data and the grade **in a paper registry**. When the exam session ends, the teacher brings the register to the Department's student office.
The student secretary **makes a first check of** the completeness of the **registers**, and, when he/she finds some void fields, **reports the incompleteness** to the teacher.
When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office **re-analyzes the registers** for errors and inconsistencies, and checks that students:

- have in their study program the course corresponding to the exam, and
- are in good standing with University taxes.

In case of errors or inconsistencies in the register, the office **sends the register back** to the teacher.
When the records are complete, the grades of exams **are updated in the student's exam database**, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.
Fill the paper exam register
Check the paper register
Report incompleteness
Analyze the paper register
Report errors and inconsistencies
Update the exam database

Figure 2.7: Processes in the requirements description highlighted in bold

2.6 The Processes – Types of data matrix

Now we start to fill the cells of the processes – types of data matrix with the same symbols, C or U, we used for the organization – types of data matrix, with same meaning: C corresponds to creates, U to uses.

Exercise 2.4 - The reader is invited to choose C's or U's in Figure 2.8. Solutions in this chapter appear in the page after the exercise.

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	C or U?	C or U?	C or U?	C or U?	
Check the paper register					
Reports incompleteness					
Analyze the paper register					
Report errors and inconsitencies					
Update the exam database					

Figure 2.8 – Matrix of Exercise 2.4

Solution to Exercise 2.4

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register					
Reports incompleteness					
Analyze the paper register					
Report errors and inconsitencies					
Update the exam database					

Figure 2.9 – Matrix solution to Exercise 2.4

We provide the solution In Figure 2.9; the unique C corresponds to the type of data "Exam paper register", the remaining cells are filled with U's except the Student taxes type of data that is not involved in the process.

Exercise 2.5 - We have now to fill the part of the matrix in the rectangular box that refers to the "Exam paper registry" type of data, see Figure 2.10.

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register					
Reports incompleteness					
Analyze the paper register				**C or U?**	
Report errors and inconsitencies					
Update the exam database					

Figure 2.10 – Matrix of Exercise 2.5

Solution to exercise 2.5 - In this case the only C's make reference to a. the Report incompleteness and b. the Report errors and inconsistencies, that we may assume are performed highlighting the corresponding errors in reports, see Figure 2.10.

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register				U	
Reports incompleteness				C	
Analyze the paper register				U	
Report errors and inconsitencies				C	
Update the exam database				U	

Figure 2.11 – Matrix solution to exercise 2.5

Exercise 2.6 - Finally, we have to fill the two remaining areas of the matrix, see Figure 2.12.

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register				U	
Reports incompleteness				C	
Analyze the paper register				U	
Report errors and inconsitencies		C or U?		C	C or U?
Update the exam database				U	

Figure 2.12 – New question

Solution to Exercise 2.6 - In this case we add two U's to the row referring to the process *Analyze the paper register*, and a single C to the process *Update the exam database*, as can be straightforwardly understood from the meaning of the process. Look now to the area in the grey rectangle in Figure 2.13.

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register				U	
Reports incompleteness				C	
Analyze the paper register		U	U	U	
Report errors and inconsistencies				C	
Update the exam database				U	C

Figure 2.13 – Matrix solution to Exercise 2.6

Exercise 2.7 - Do you notice something strange in the framed part of the matrix in Figure 2.14?

Processes / Types of data	Students Biographic data	Students – Study program	Students Taxes	Exam paper registry	Exam – Data base
Fill the paper exam register	U	U		C	
Check the paper register				U	
Reports incompleteness				C	
Analyze the paper register		U	U	U	
Report errors and inconsitencies				C	
Update the exam database				U	C

Figure 2.14 – New question

Solution to Exercise 2.7 - In the frame there are no C's, so apparently there is an inconsistency, since no process is responsible of creating corresponding data. How is this possible? It is possible, under the condition that some other process, outside the *exam registration process*, is responsible of it. We will have to take notice of this when we will conceive the process-to-be.

We now move to the Railway company case study.

2.7 The Organization – types of data matrix in the Railway company case study

We have first to identify the different organizational structures, types of data and processes; in order to simplify following exercises, we assume they are as in Figure 2.15.

Organizational structures
- Controllers
- Office involved in reservation and ticket purchase
- Marketing office
- Pricing and discount campaigns office
- Train composition office
- Fidelity card office
- Standard and special trains division
- Statistical office

Types of data
- Seat reservation and ticket purchase
- Paper tickets and sms
- User profile
- Structure of trains in terms of coaches and seats
- Time table of train travels
- % of occupation of seats in trains in the last 10 years
- Taxi reservation at arrival in a station

Processes
- Ticket control
    - Control
    - Fine (multa)
- Seat reservation and ticket purhase
    - Reservation and purchase
    - Change of reservation
- Marketing
    - Long term marketing
    - Short term marketing
- Pricing
- Fidelity card management
- Train composition
    - Standard trains
    - Special trains
- Timetable design and production
- Statistics

Figure 2.15 - Organizational structures, types of data and processes that can be extracted from requisites of the railway company case study.

We fill now the organization - types of data matrix (see Figure 2.16) with C's and U's with the usual meaning. E.g. the controller uses all types of data, except the statistics on % of occupation, and creates data referring to taxi reservation.

Org. structure/types of data	Reserva-tions and tickets	Paper ticket and sms	User profile	Structure of trains	Timetable of trains	% of occupation of seats in trains in the last 10 years	Taxi reservation
Controller							
Office for reservation and ticket purchase							
Marketing office							
Pricing and discount campaigns office							
Train composition office							
Fidelity card office							
Standard and special trains division							
Statistical office							

Figure 2.16 – Organization / types of data matrix

Exercise 2.8 – Provide the complete matrix. The solution in the next page.

Solution to Exercise 2.8

Org. structure/types of data	Reservations and tickets	Paper ticket and sms	User profile	Structure of trains	Timetable of trains	% of occupation of seats in trains in the last 10 years	Taxi reservation
Controller	U	U		U	U		C
Office for reservation and ticket purchase	C	C	U	U	U		
Marketing office			U	U	U	U	U
Pricing and discount campaigns office			U		U	U	
Train composition office				C		U	
Fidelity card office			C				
Standard and special trains division					C	U	
Statistical office	U		U	U	U	C	U

Figure 2.17 - The organization – types of data matrix

We may add now to C's and U's data on the frequency of execution in a year of operations, see Figure 2.18. Notice that C's correspond to transactions, that change the content of the database and U's correspond to queries. These data will be useful in the sizing of the database and in its architectural design.

Organizastional structure / Types of data →	Reser-vations and tickets	Paper ticket and sms	User profile	Structure of trains	Timetable of trains	% of occupation of seats in trains in the last 10 years	Taxi reser-vation
Controller	**U – 50 10^6**	**U - 100 10^6**		**U - 10^6**	**U - 10^6**		**C - 5 10^6**
Office for reservation and ticket purchase	**C - 70 10^6**	**C - 50 10^6**	**U - 10 10^6**	**U - 70 10^6**	**U - 70 10^6**		
Marketing office			**U - 6**	**U - 6**	**U - 6**	**U - 2**	
Pricing and discount campaigns office			**U - 6**		**U - 6**	**U - 2**	
Train composition office				**C - 2**		**U - 2**	
Fidelity card office			**C – 2 10^6**				
Standard and special trains division					**C - 2**	**U - 2**	
Statistical office	**U - 6**		**U - 6**	**U - 6**	**U - 6**	**C - 6**	**U - 6**

Figure 2.18 – Frequencies of execution of C and U operations

2.8 The Organization - Processes matrix in the Railway company case study

We now move to the organization - processes matrix. We notice that the organizational structures refer in some cases to offices of the organization, in other cases to roles of employees such as controllers, see Figure 2.19.

Organizational structures or roles → Processes	Controller	Reservation and ticket purchase office	Marke-ting office	Pricing and discount campaign office	Train compo-sition office	Fidelity card office	Standard and special train timetable office	Sta-tistical office
1. Ticket control								
2. Fine								
3. Seat reserv. and ticket purchase								
4. Change reservation								
5. Short term Marketing								
6. Long term marketing								
7. Pricing								
8. Fidelity card management								
9. Train composition								
10. Seasonal timetable								
11. Standard train composition								
12. Special train composition								
13. Montlhy and annual statistics								

Figure 2.19 – The organization - process matrix

In this case, we change the meaning of the cells to verbs related to the actions associated to processes, where the action may be Decides or Contributes. This distinction is relevant since structures that decide are more important than structures that contribute in the process and play a more significant role in achieving the efficiency and effectiveness of the process.

We decide first to fill the rows and columns selected in Figure 2.20. These are the object of the next Exercise 2.9. See the solution in the subsequent page.

Organizational structures or roles → Processes	Controller	Reservation and ticket purchase office	Marke-ting office	Pricing and discount campaign office	Train compo-sition office	Fidelity card office	Standard and special train timetable office	Sta-tistical office
1. Ticket control								
2. Fine								
3. Seat reserv. and ticket purchase								
4. Change reservation								
5. Short term Marketing								
6. Long term marketing								
7. Pricing								
8. Fidelity card management								
9. Train composition								
10. Seasonal timetable								
11. Standard train composition								
12. Special train composition								
13. Montlhy and annual statistics								

Figure 2.18 – Rows and columns object of Exercise 2.10

Solution to exercise 2.9

Organizational structures or roles → Processes	Controller	Reservation and ticket purchase office	Marke-ting office	Pricing and discount campaign office	Train compo-sition office	Fidelity card office	Standard and special train timetable office	Sta-tistical office
1. Ticket control	D							
2. Fine	D							
3. Seat reserv. and ticket purchase								
4. Change reservation								
5. Short term Marketing								
6. Long term marketing								
7. Pricing								
8. Fidelity card management								
9. Train composition							C	
10. Seasonal timetable							D	
11. Standard train composition							C	
12. Special train composition							C	
13. Montlhy and annual statistics	C	C	C	C	C	C	C	D

Figure 2.19 – Rows and columns object of exercise 2.6 filled with C’s and D’s

Referring initially to controllers, they have a decisional role when they control the ticket of the passenger and in case it is not regular apply a fine. They also contribute to statistics that are useful for planning the evolution of the organizational and information system of the company.

As to the standard and special train timetable office, it has a decisional role in the production of the seasonal timetable, while it contributes to the process of train composition, since in rush hours trains with larger capacity are needed; this organizational unit too contributes to statistics.

We have seen in previous discussion that all organizational units contribute to statistics, while the decision on which specific statistics to perform is up to the statistical office.

Exercise 2.10 – Complete the whole matrix with D’s and C’s. The solution is in the next page.

We conclude here the chapter.

Solution to Exercise 2.10

Organizational structures or roles → Processes	Controller	Reservation and ticket purchase office	Marke-ting office	Pricing and discount campaign office	Train compo-sition office	Fidelity card office	Standard and special train timetable office	Sta-tistical office
1. Ticket control	D							
2. Fine	D							
3. Seat reserv. and ticket purchase		D						C
4. Change reservation		D						C
5. Short term Marketing		C	D	C				C
6. Long term marketing		C	D	C				C
7. Pricing		C	C	D		C		C
8. Fidelity card management		C	C	C		D		C
9. Train composition		C	C	C	D	C	C	C
10. Seasonal timetable		C	C	C	C	C	D	C
11. Standard train composition		C	C	C	D	C	C	C
12. Special train composition		C	C	C	D	C	C	C
13. Montlhy and annual statistics	C	C	C	C	C	C	C	D

Figure 2.20 – The final matrix as solution of Exercise 2.10

We see in the solution that a major role is played by the Reservation and ticket purchase office; the statistical office contributes to all processes except those processes performed by controllers. Marketing contributes to several processes, and decides on short term and long-term marketing.

Chapter 3 – Modeling processes and data

In the Information systems life cycle we have to provide a description of processes and data in terms of non-ambiguous and semantically rich models. The reason for this is that we need to reason on processes and data, e.g. we need to perform an assessment of processes, to produce a new process-to-be, to map processes and data respectively on new software applications and new databases. In Chapter 1 we have informally introduced two models for processes and data, the BPMN and the Entity Relationship model. Furthermore, due to the complexity of requirements, it is not enough to provide models to the designer; he/she needs also methodologies to be able to effectively use models for processes and data. The Modeling phase shown in Figure 3.1 deals with the above issues.

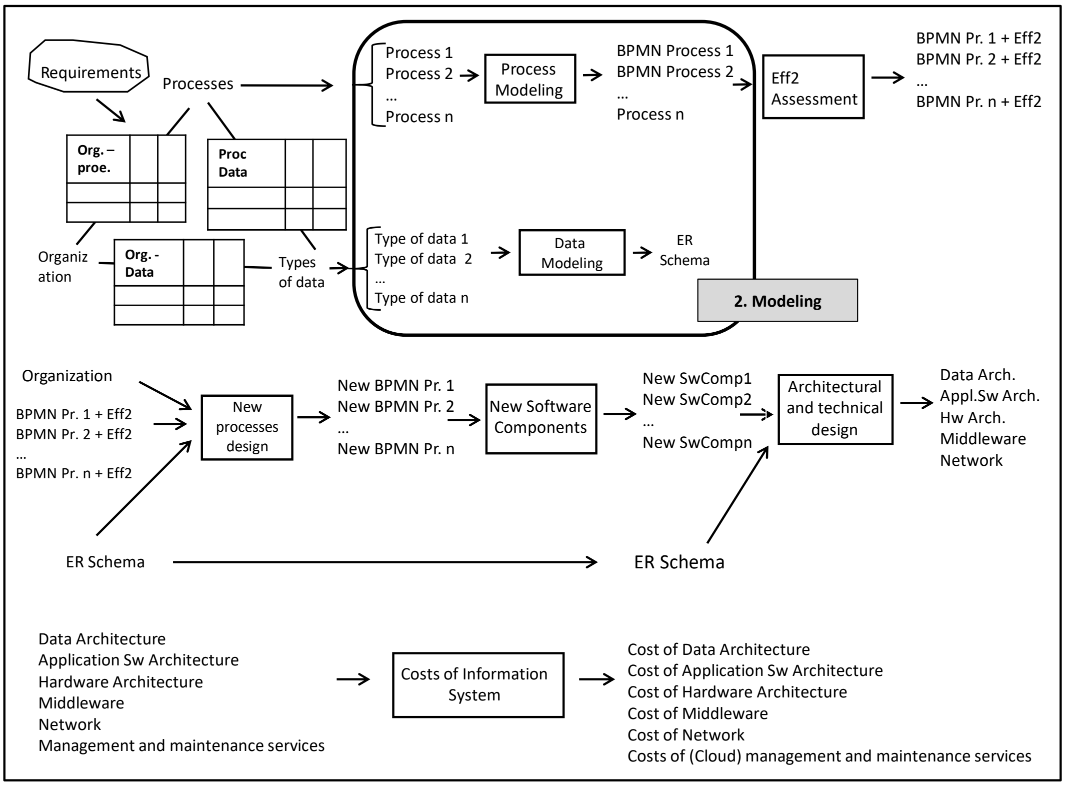


Figure 3.1: The Modeling phase in the information system life cycle

3.1 Introduction to models

Models are tools that we can use to represent the reality around us. More precisely, see Figure 3.2, the reality can be observed, in terms of objects and their properties. Depending on our goals, we can observe events, things, behaviors and relationships among them. Since we cannot reproduce the reality in all its details, we use models to filter relevant aspects of interest, and represent them by means of a set of modeling constructs; such modeling constructs are applied to *observables,* namely the aspects of reality that can be perceived. Focusing for clarity on data, observables may be:

- atomic (such as “37.5”); in this case they are modeled in terms of atomic data (37.5 as a value of a temperature), also called instances;

- grouped in classes (such as Students) and relationships between classes (such as Students pass Courses in Exams), and are modeled in this case as entities and relationships, resulting in schemas.

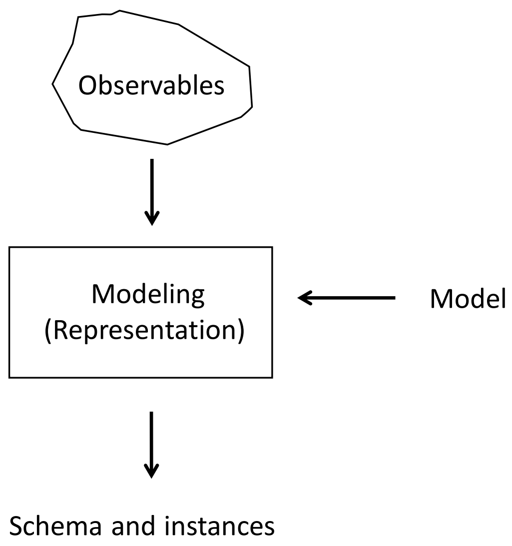


Figure 3.2 - The modeling process

One could naively believe that given a set of observables, the schema and instances corresponding to them are unique, namely, a unique schema and a unique set of instances can represent them. On the contrary, whatever model we can conceive, depending on our goals and on the cognitive process, we can conceive a variety of schemas that may convey different equivalent views of the same set of observables.

Models can be used during information system design, to allow the user of the system and the designer to share a compatible representation of the observables that are involved in the a. inputs to the (information) system, b. outputs from the system and c. activities to be performed in the system.

In Chapter 1 we have seen that we are interested to represent processes and data; we have introduced for this purpose two models, the Business Process Modeling Notation (BPMN) and the Entity Relationship Model (ER). In this chapter we go more in depth on BPMN and ER, showing the modeling constructs they provide and methodologies that enable the designer to model complex requirements in terms of BPMN and ER schemas. Anyhow, this chapter too has to be seen by the reader as introductory, since the modeling constructs of (especially) BPMN and (to some extent) ER are much richer than constructs addressed in the chapter. For this reason, we provide references to learning material that can be freely accessible/downloaded from the Web.

We address now the two issues of BPMN and ER, using in both cases the exam registration requirements, which we reproduce in the following box.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.

When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students:

- have in their study program the course corresponding to the exam, and
- are in good standing with University taxes.

In case of errors or inconsistencies in the register, the office sends the register back to the teacher.
When the records are complete, the grades of exams are updated in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

Both for BPMN and for the ER model we provide initially an introduction to the model, in terms of main modeling constructs and diagrammatic representations that can be used for them. Then, we will discuss methodologies that aid the designer in the modeling process. Such methodologies provide general strategies in terms of sequences of steps; such strategies, as we will see soon, are common to the BPMN and ER cases.

3.2 Introduction to BPMN

The diagrammatic representation of modeling constructs of BPMN appears in Figure 3.3.

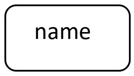
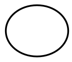
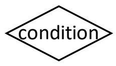
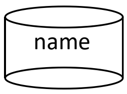
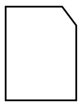
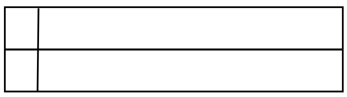
Construct	Symbol
Activity	
Event	
Gateway	
Data Storage	
Data Object	
Sequence flow	
Lane	

Figure 3.3: Diagrammatic representation of modeling constructs of BPMN

Let us now introduce the modeling constructs using as common example the BPMN process related to exam registration reproduced in Figure 3.4.

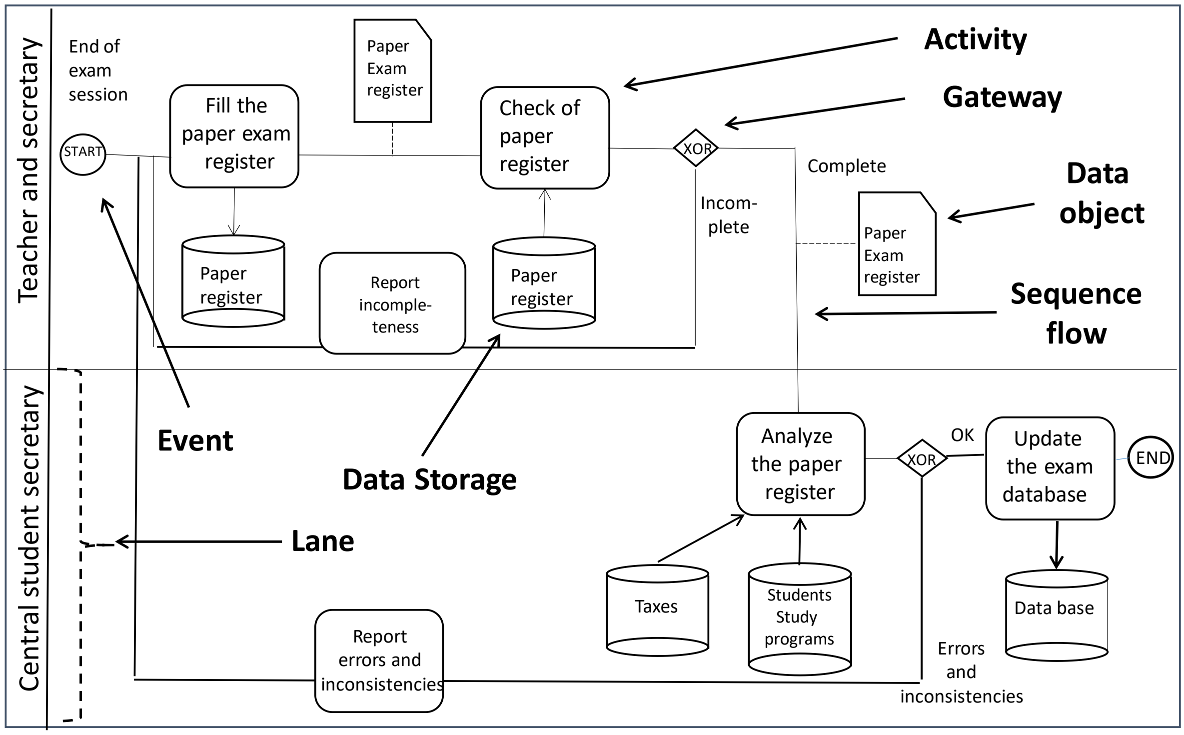


Figure 3.4 - The BPMN schema of the exam registration requirements

An **Activity** is an elementary process that has a data flow in input and can produce a new data flow and/or a data storage (see shortly) in output. In the example, the activity "Fill the paper exam register" results in an update of the paper exam register, and transfer the exam register to the activity "Check of paper register".

An **Event** is an occurrence of a fact in a given moment of time; e.g. the event "End of exam session" corresponds to the moment of time in which exams are ended, and the teacher starts the registration process.

A **Gateway** is a logical formula whose value, true or false, evaluated when the process reaches the gateway, and determines the evolution of the flow of activities. The XOR condition is one of the possible types of conditions in BPMN; when the flow of activities reaches the upper XOR condition in Figure 3.4. it is evaluated whether or not the paper register is complete, and according to the result of the evaluation, one of the "complete" vs "incomplete" flows is followed.

A **Data storage** is a dataset that is created/updated by an activity, and that can be permanently stored in paper format or in a data base, in such a way that data stored can be subsequently used by other activities. The data storage "paper register" is created by the "Fill the paper exam register" (notice the direction of the arrow) while it is read by the "Check of paper register" activity (notice the inverse direction of the arrow w.r.t. the previous one).

Lanes are parts of the BPMN process that are in charge to specific organizational units. In Figure 3.4 there are two lanes, in the upper lane teachers and department secretary are involved, while the central student office is involved in the lower lane.

Data objects are inputs to and outputs from activities. Data objects are used to represent documents, data or other objects that are passed between the activities in a process.

Sequence flows are the flows of the BPMN process that originate from the Start event and express the evolution of execution flow between events, activities performed by the lanes and decisions that are made (gateways).

Looking at the BPMN process of Figure 3.4, we observe that each modeling construct expresses a function that cannot be reproduced by other modeling constructs, so each one of them is essential in BPMN. At the same time, they are enough to use effectively BPMN in the book. The full set of modeling constructs of BPMN is reproduced in Appendix 3.2.

Exercise 3.1 - Produce a new BPMN in which three different lanes are distinguished, referring to

a. the teacher,
b. the department secretary and
c. the central student secretary.

The solution is in the Appendix.

3.3 Strategies for process schema design

Strategies to transform a set of requirements into a process are the following:

1. Oil stain, or inside out starts the design mapping requirements into one BPMN concept, e.g. an activity, a lane, etc. and then browses the requirements mapping the sentences into new BPMN concepts that are logically adjacent to the previous ones.
2. Priority based - From organization to processes to data; in this strategy, we start modeling the organization in terms of lanes, then we represent processes in terms of activities and flows of control, and finally we model data using data stores and data objects.
3. Top down starts mapping indistinctly all requirements into one BPMN concept (e.g. an activity whose name is “Exam registration”, that represents the whole set of requirements), and then refines such abstract description in terms of more concrete concepts.

The common aspect to the three strategies is the adoption of an attitude that ancient romans called “divide et impera”, that corresponds to fragment a complex goal, namely modeling complex requirements, into simpler ones. We show now the three strategies at work.

<u>Oil stain strategy</u>

The first part of requirements, reproduced in bold hereafter, can be mapped in the BPMN process of Figure 3.5.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers,…

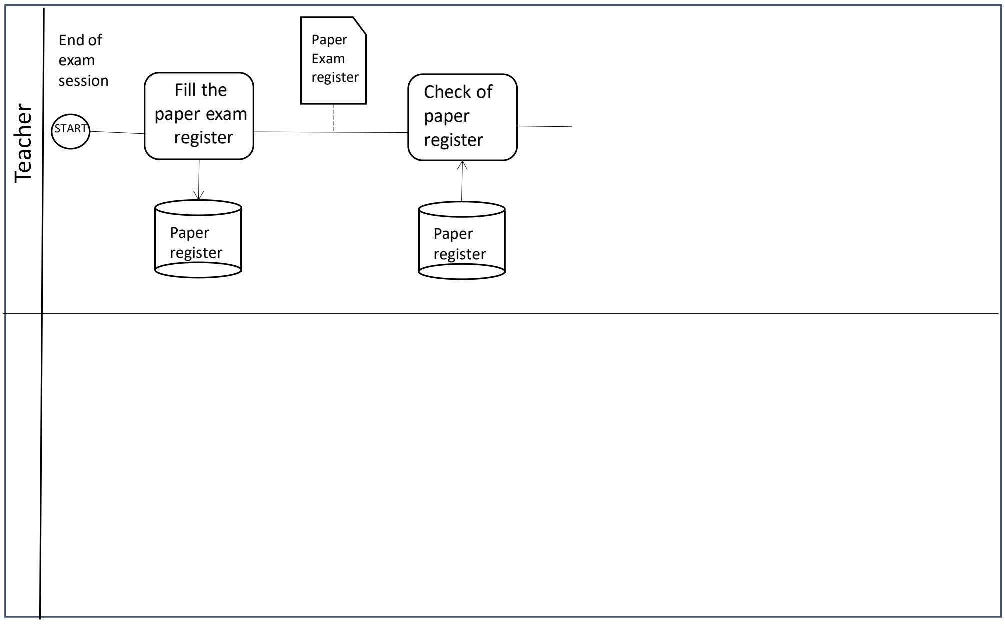


Figure 3.5 - First schema produced with the oil-stain strategy

Notice that we have modeled verbs such as “records” into an activity and “brings” into a sequence flow. Furthermore, so far we have represented activities that pertain to the teacher; so we have placed them in the teacher lane.

In the following box and in Figure 3.6 we go one step further, modeling the activities in charge to the department secretary; we decide to include the secretary in the previous lane, so that we have in a unique lane for all activities performed in departments, that are decentralized units.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers, **and, when he/she finds some void fields, reports the incompleteness to the teacher.**
When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.

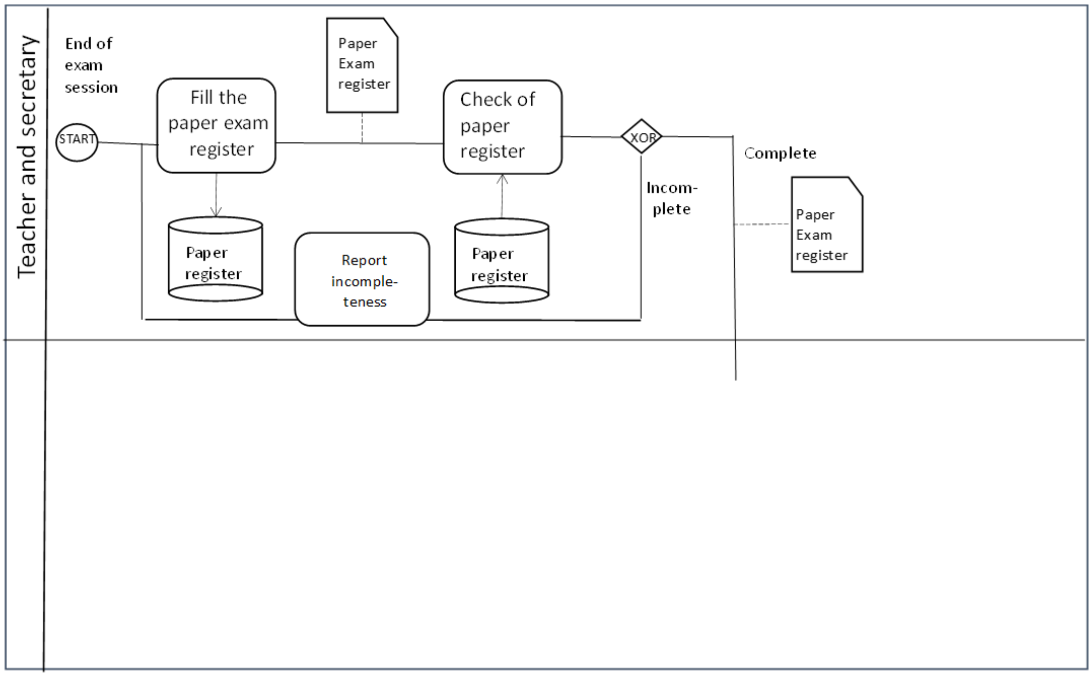


Figure 3.6 – Adding the activities in charge to the student secretary

Now we start to model the activities of the central student office, starting from the activity of analysis of the paper register and the data stores that are used by the activity, see Figure 3.7. Notice that in the requirements there is no mention of the office that manages the two data stores; we could also include such office in the BPMN, going more in depth on the nature of the two data stores, whether they are automatized and are supported by a software application.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office.
The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.
When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students:

- **have in their study program the course corresponding to the exam, and**
- **are in good standing with University taxes.**

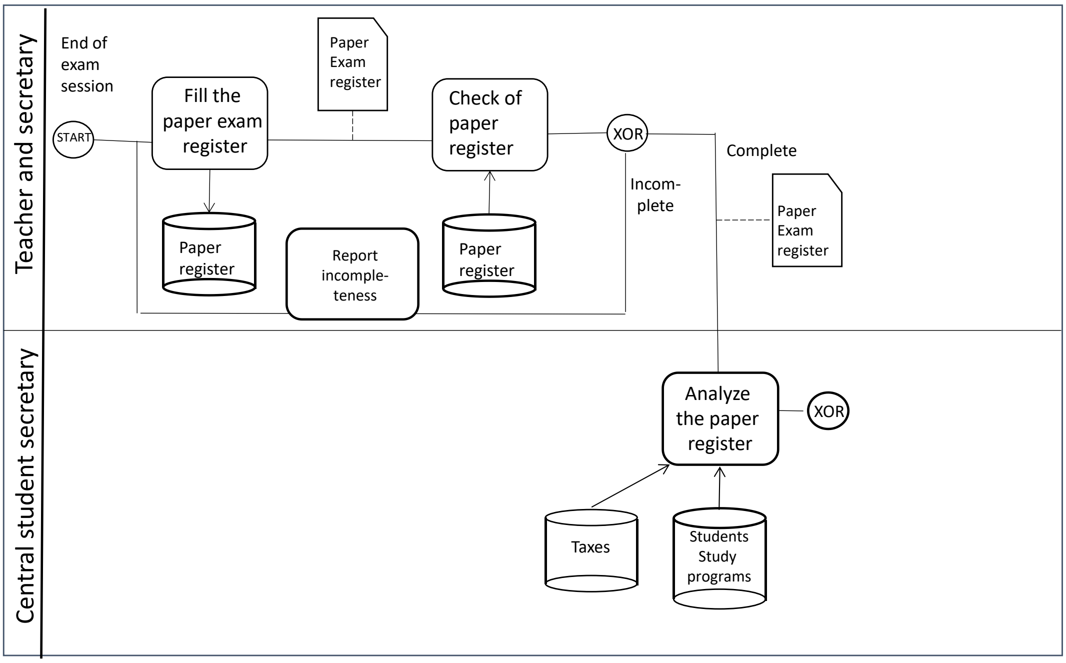


Figure 3.7 – Adding activities of the central student office

Finally, we model the remaining activities of the central student office, leading to the final BPMN that we have shown in Figure 3.4.

Priority based - First organization, then processes, finally data

In this strategy we discipline the concepts represented in the BPMN schema according to a set of priorities, modeling first the organization, then process and finally data. Such priorities guide the modeling process. Following this strategy the first BPMN results in a very simple representation, where we have to focus on the organization, identify the organizational structures involved and organize them into lanes, see Figure 3.8.

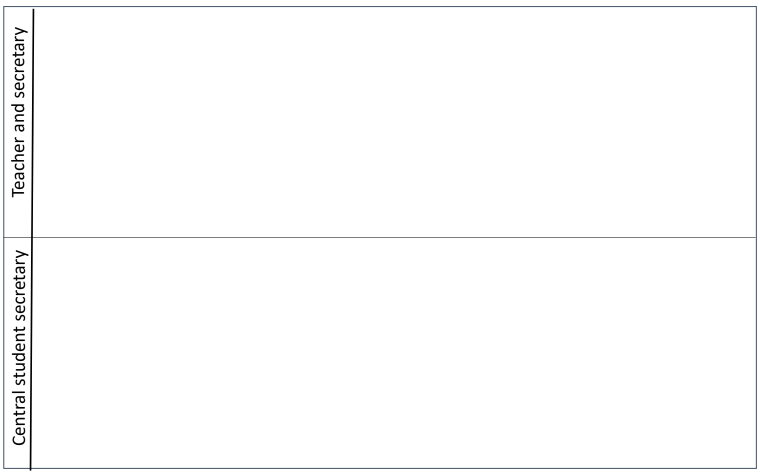


Figure 3.8 – First organization ….

Then we move toward events, gateways and activities, that all together makeup the core of processes, arranging activities in the different lanes, see Figure 3.9.

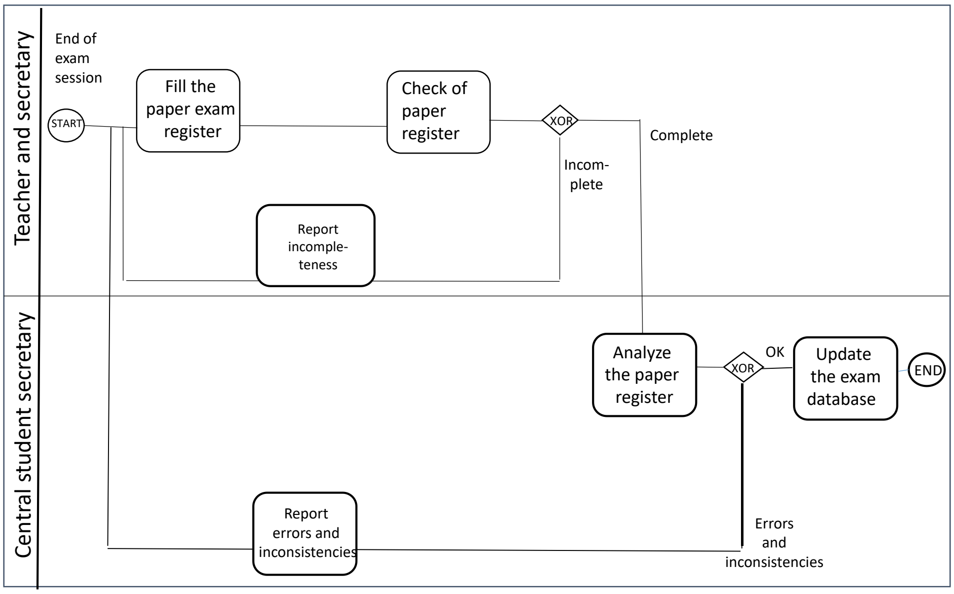


Figure 3.9 – ….. then processes…

Finally, we add data stores and data objects, leading to the BPMN of Figure 3.4.

Top down strategy

In the top down strategy we adopt a different point of view, representing initially the process with a very abstract BPMN schema, that is subsequently refined by means of transformations; such transformations act on BPMN constructs (typically activities), that are exploded into more detailed BPMN schemas. What distinguishes the top down strategy form the previous two strategies is that in this case we aim to model at each step ALL the requirements, although at different levels of detail.

As a first choice we could tentatively represent the process in terms of a unique activity called “exam registration”, but in this case we would not go a long way…; rather, we can represent the BPMN process in terms of two activities,
a. a generic activity “Manage the paper exam register” in the department lane and
b. a second activity “Analyze the paper register”in the central office lane,
together with a data store in the first lane, a gateway in the second one, and the start and end events, see Figure 3.10.

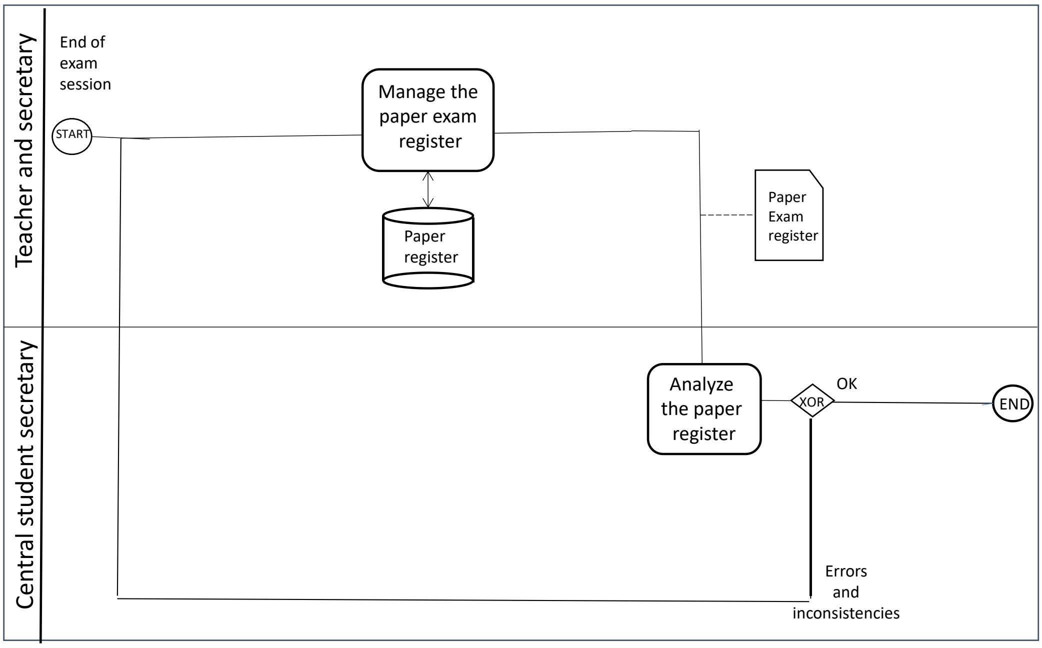


Figure 3.10 – First choice of abstract activities

We have now to choose one construct to which applying a refinement transformation. We can first focus on the activity “Manage the paper register”, that is at a higher level of abstraction than the “Analyze the paper register” (this is testified by the usage of the generic verb “manage”), and refine it as in Figure 3.11.

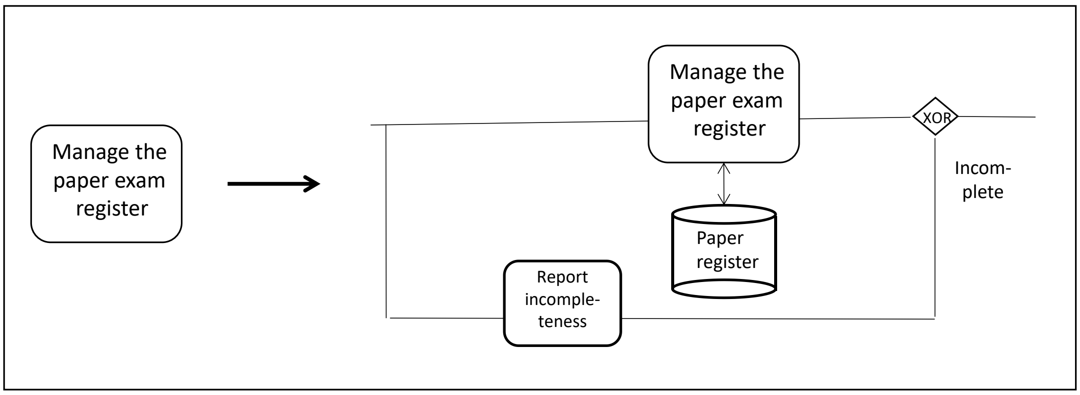


Figure 3.11 - Refinement performed on the activity “Manage the paper register”

The application of a second refinement to the activity “Analyze the paper register” leads us to the new BPMN schema of Figure 3.12. Notice that we have highlighted with dashed lines the parts of the schema that result from transformations on previous constructs.

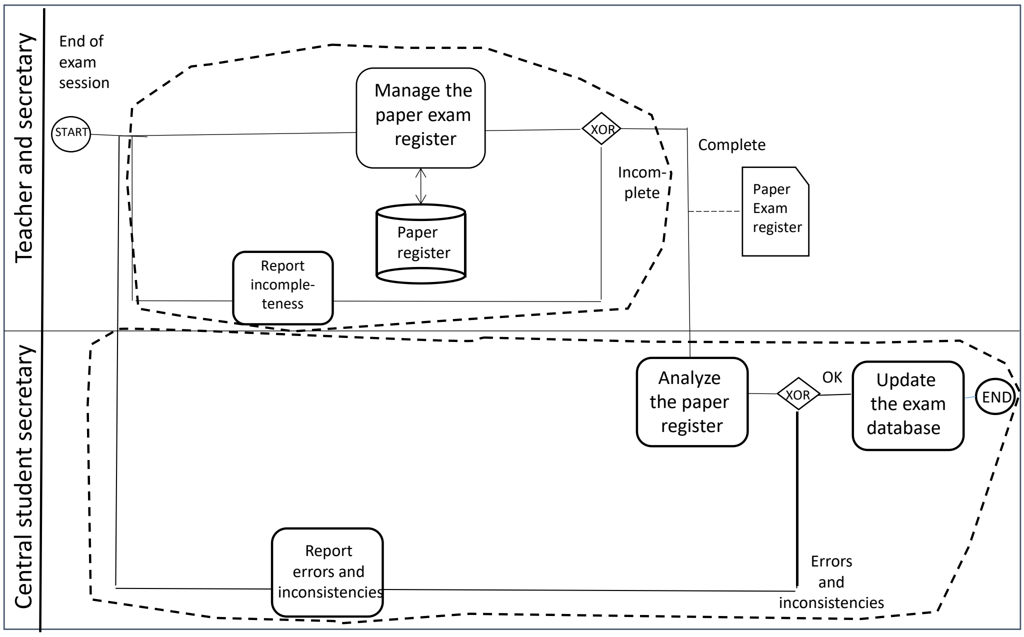


Figure 3.12 – First refinement in the top down strategy

New refinements on the BPMN process should act on the "Manage the paper exam register", that is yet too generic. We can perform the transformation in Figure 3.13, plus a second transformation on the lower lane, leading to the final BPMN schema that we reproduce in Figure 3.14 using dashed lines to highlight also in this case the transformations performed.

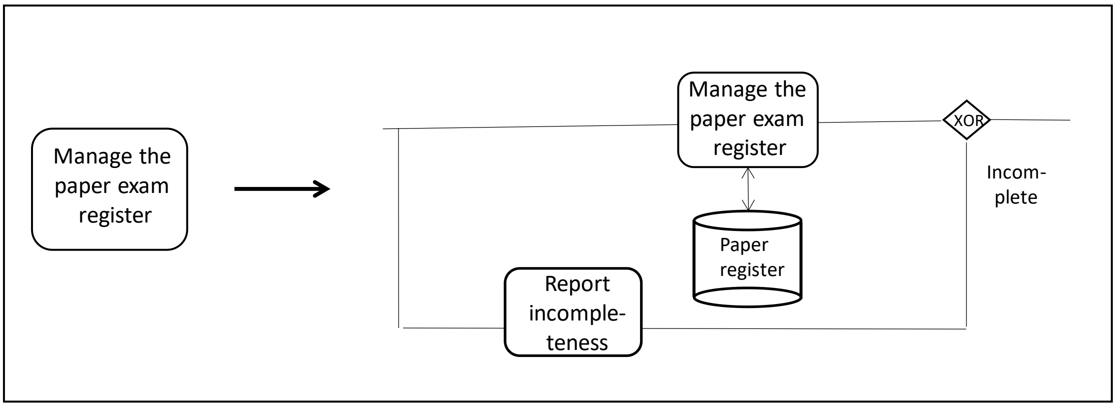


Figure 3.13 - Refinement performed on the activity "Manage the paper register"

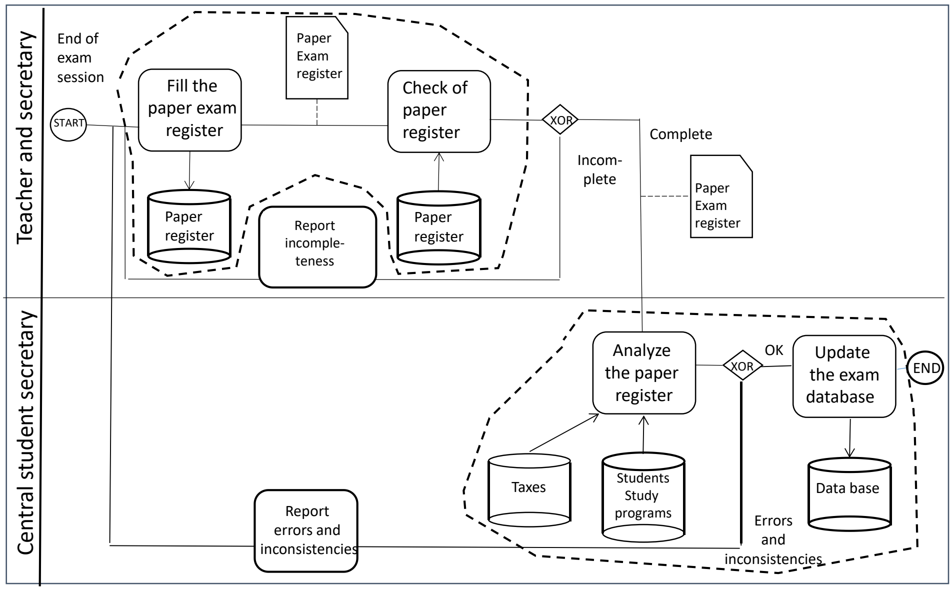


Figure 3.14 – Second and final choice of transformations

3.4 Introduction to the Entity Relationship model

We first show in Figure 3.15 the diagrammatic representation of constructs of the Entity Relationship (ER in the following) model.

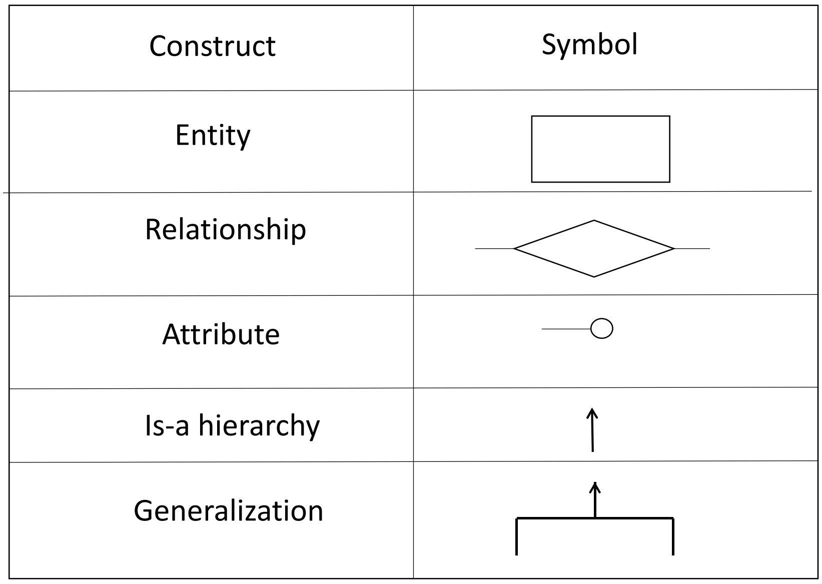


Figure 3.15 – Constructs of the ER Model and corresponding diagrammatic representations

We define now the different constructs, using as a common example the ER schema related to the exam registration requirements reproduced in Figure 3.16.

An **entity** is a class of observables of the real world, having unique identity and sharing common properties. In Figure 3.16, an example of entity is Course, which represents the class of courses that are passed in the exams by students; another entity is Student, which represents the class of students enrolled at University.

An **attribute of entity** is an elementary property common to all observables of the corresponding entity. E.g. the entity Course has two attributes, Code and Name, while entity Student has three attributes, referring to the code, the Surname, and a third attribute that states if the student has paid taxes or not.

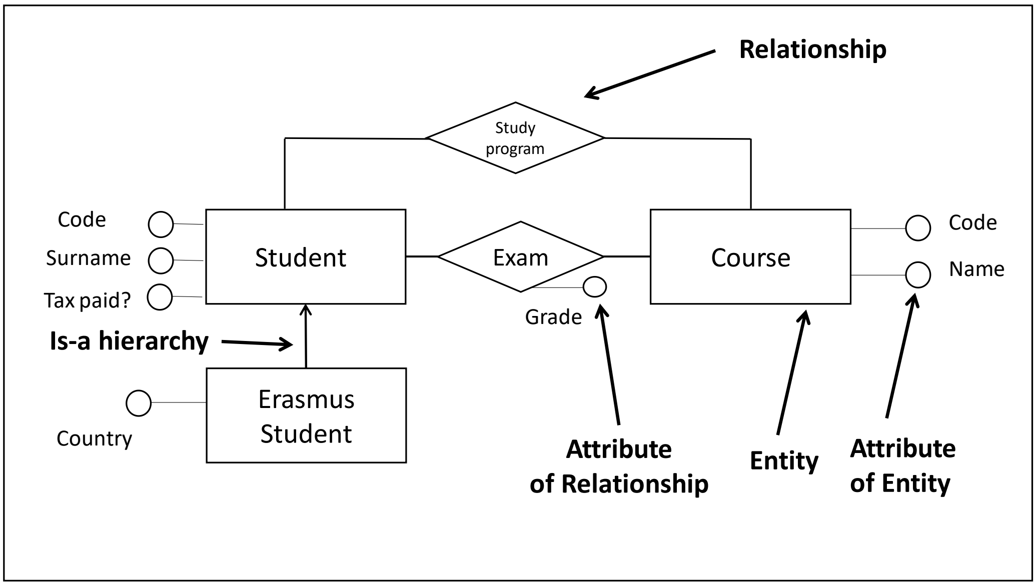


Figure 3.16 – The data schema of the exam registration case study with examples of modeling contructs

A **relationship** between two entities E1 and E2 is a class of pairs of observables that are elements of E1 and E2. In our example Exam is a relationship between pairs of students and courses that are logically related in exams passed. Study program is a relationship that relates students with courses in their own study program.

An **attribute of a relationship** is an elementary property of a relationship; in our example, a Grade is associated to each exam, that relates pairs of observables of a (specific) student and a (specific) course.

An **Is-a hierarchy** defined between two entities E1 and E2, E1 (called child entity) Is-a E2 (parent entity), states that each observable of E1 is also an observable of E2; in our example, each Erasmus Student is also a Student. Is-a hierarchies obey an important property, the inheritance property, that states that every property (attribute, relationship, a is-a hierarchy) of the parent entity in the hierarchy is also a property of the child entity; e.g. the Surname, a property of Student, is also a property of Erasmus Student.

A **generalization** among a parent Entity E and two child entities E1 and E2, is a pair of is-a hierarchies between E and E1 and E2 respectively, such that the union of observables in E1 and E2 corresponds to the set of observables in E. An example of generalization is the one with parent entity Student and child entities Italian Student and Erasmus Student.

Exercise 3.2 - Modify the ER schema of Figure 3.16 adding the representation of the following requirements: among Erasmus students, that we assume to come from different continents, distinguish those students that come from outside Europe, and represent for them the continent. Furthermore, represent, besides the grade of the exam also the date. Finally, represent the names and dates of birth of students. Solution in the Appendix.

3.5 Strategies for data schema design

For the Entity Relationship Model as for the BPMN model we may follow the oil stain strategy and the top down strategy, with similar philosophies than In the case of BPMN, while the Priority based strategy can be rephrased into "first entities, then relationships, than is-a hierarchies, finally attributes". We do not show examples of this last strategy, and we focus on oil stain and top-down strategies, while we show a third typical strategy, the bottom-up strategy,

<u>Oil stain strategy</u>

In Figure 3.17, we see a schema produced with the oil stain strategy, in which we have included three subsequent steps. In the first step, we represent the most important concept in requirements, namely the entity Student.

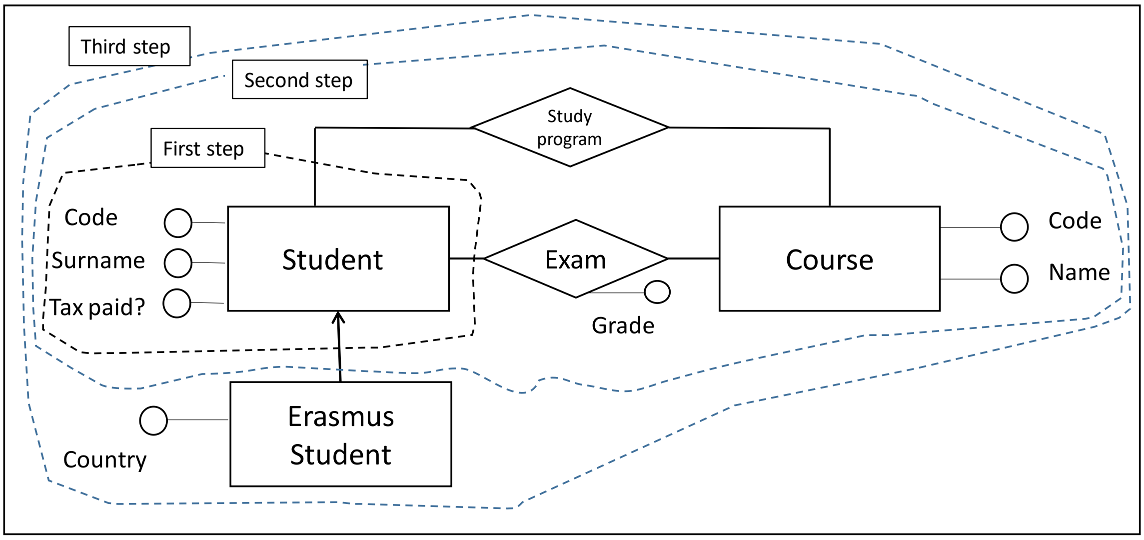


Figure 3.17 – Schemas produced in the different steps of the oil stain strategy

In the second step we move in the direction of the second relevant concept, namely, courses passed, we model courses with a second entity Course, and we represent the two relationships defined between courses and Students, Study program and Exam. In the last step we extend the entity Student to the more specific entity Erasmus Student, and we assign the attribute Country to the entity.

<u>Top down strategy</u>

In the tow down strategy we proceed, as for processes, from abstraction to concreteness, by initially representing the entity Exam, that can be considered a good abstraction of the requirements, whose general topic is exam registration, see Figure 3.18.

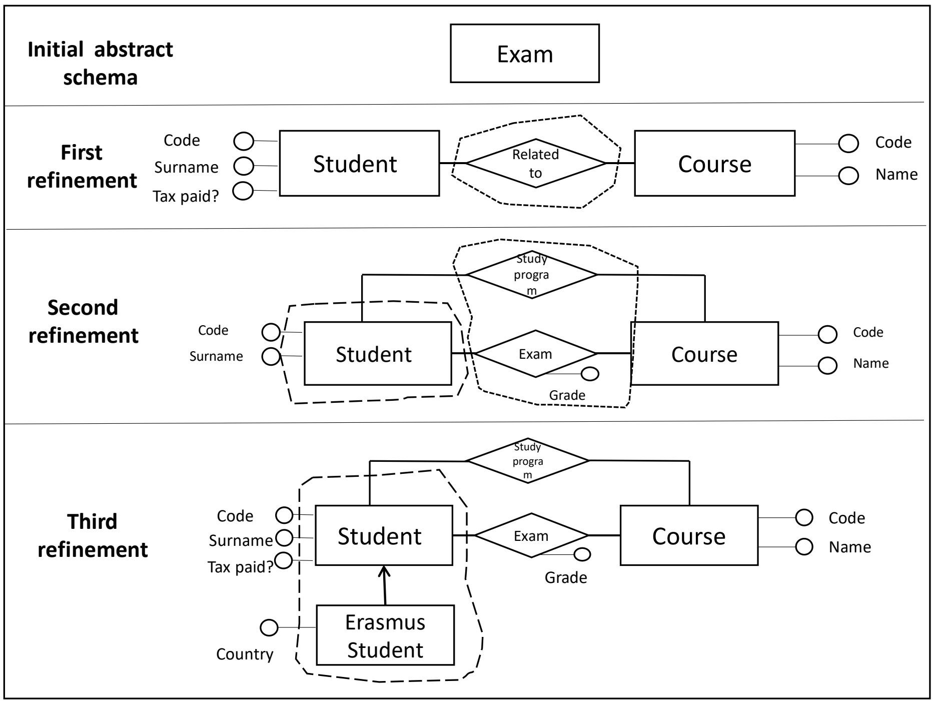


Figure 3.18 – The schemas produced in the different steps of the top down strategy

We then refine entity Exam into its constituent parts, Students and Courses that in the first refinement are connected by a unique generic relationship Related to; we also add to entities the respective attributes. In the second refinement, we split the generic relationship Related to into the two relationships Study program and Exam. In the third refinement, we introduce the entity Erasmus Student. Also in this case we highlight the parts of the schema resulting from refinements with dashed lines.

Finally, in Figure 3.19 we propose prototypical transformation patterns that can be applied in the top-down strategy for Entity Relationship schemas. Notice also that in Figure 3.18 we have applied transformation patterns that coincide with the first, second and fifth basic transformations shown in Figure 3.19.

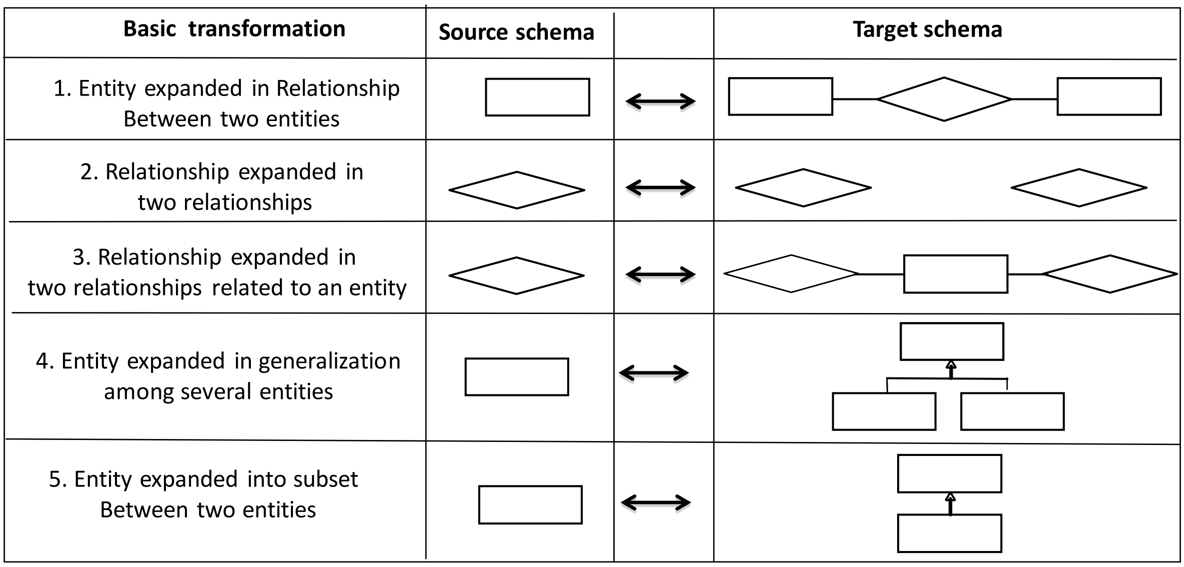


Figure 3.19 – Basic transformations in top down refinements in top down design of ER schemas

Bottom-up strategy

In the bottom-up strategy, we first identify elementary concepts of the schema, that in the Entity Relationship model correspond to attributes (see Figure 3.20).

Figure 3.20 – Bottom-up strategy: first step, attributes

Then we move to entities, whose attributes were defined previously, see Figure 21. Finally, we identify relationships, the most complex concepts in the model (see Figure 3.22). Apparently this is the most intuitive and easy strategy to be applied; this is not always the case, since the strategy is prone to potential restructuring on the schema, that are absent in previous strategies. E.g., in the second step we could add only the entity Student and assign to Student the four attributes on the left in Figure 3.20. If we subsequently identify Foreign Student, at this point we have to move the attribute Country to Foreign Student.

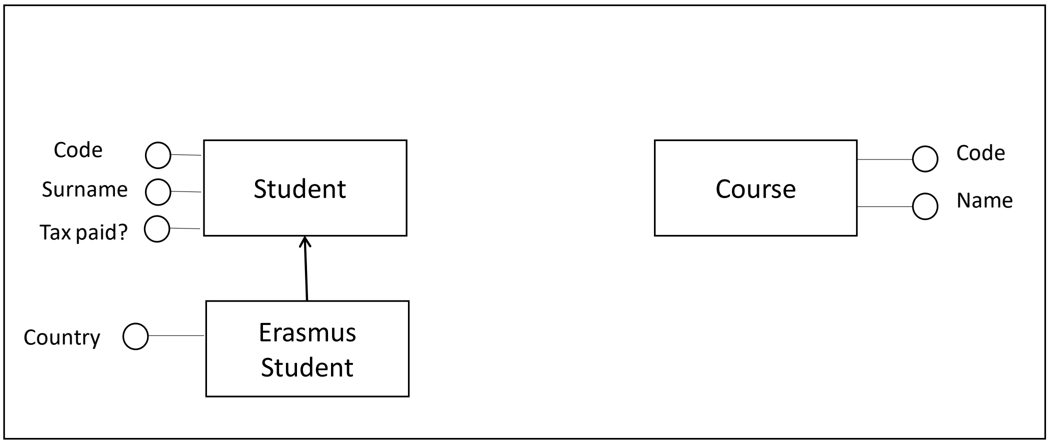


Figure 3.21 – Bottom-up strategy: second step, entities

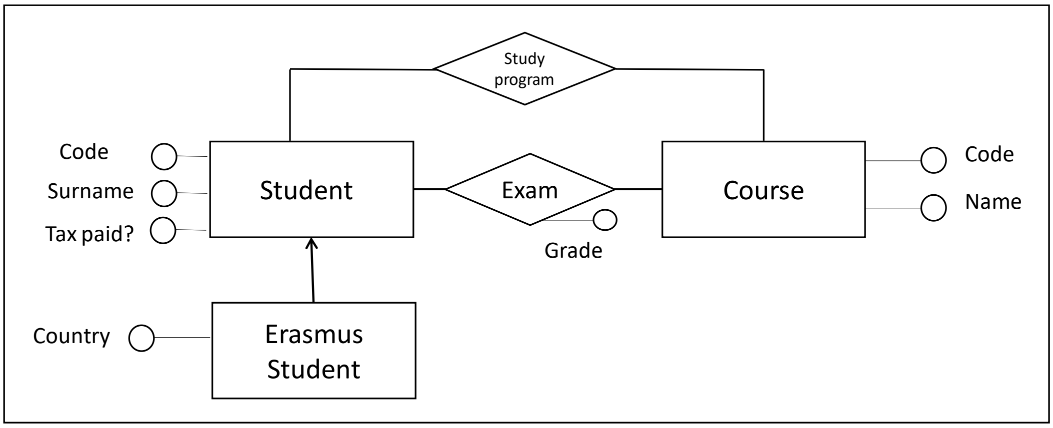


Figure 3.22 – Bottom-up strategy: third step, relationships

Appendix 3.1 – Solutions to exercises

Solution to Exercise 3.1

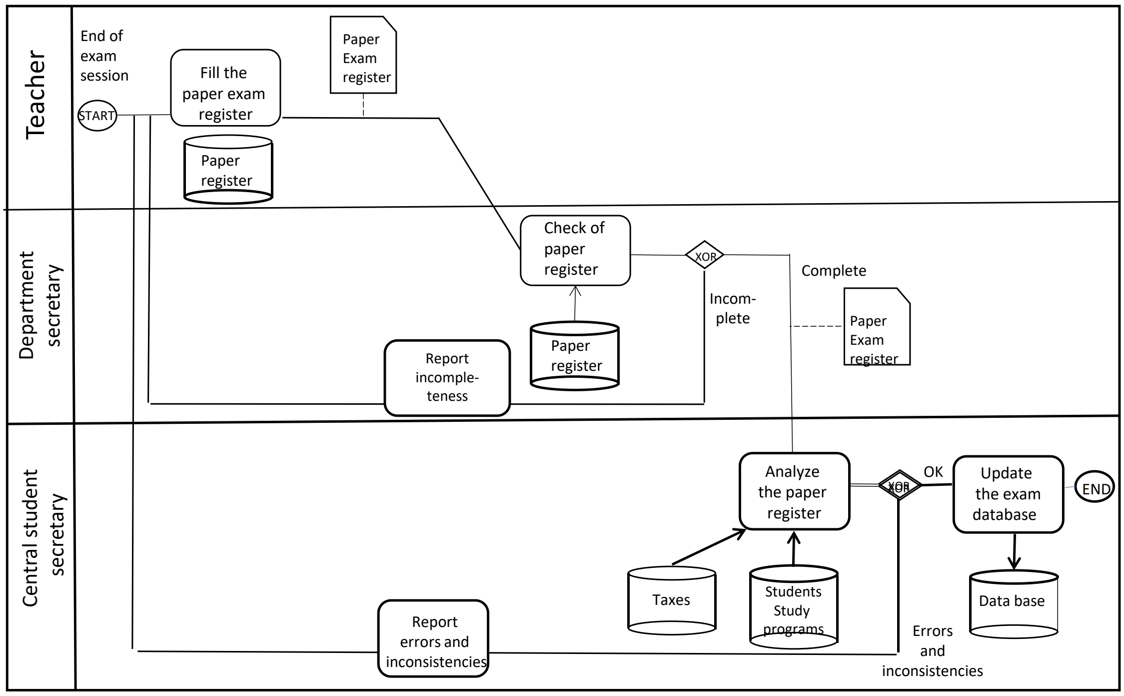


Figure 3.23 – BPMN solution to Exercise 3.1

Solution to Exercise 3.2

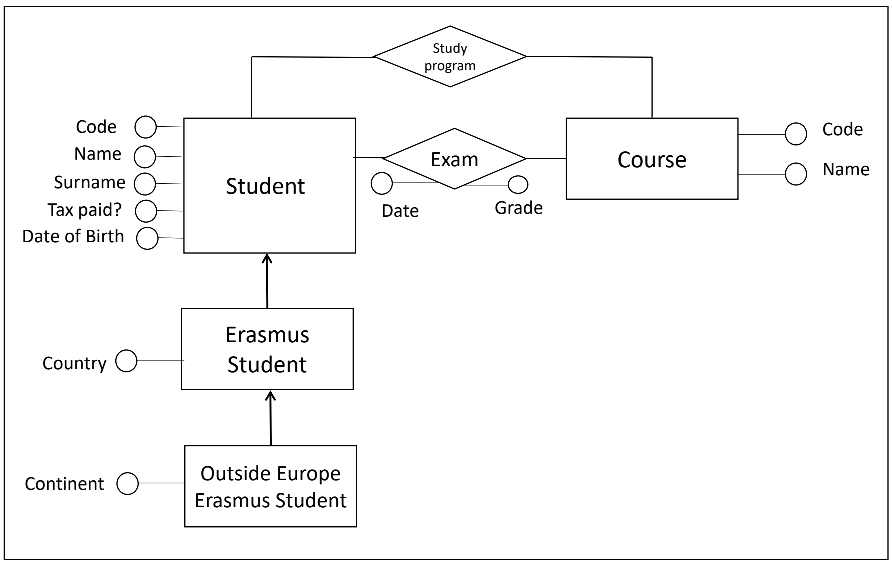
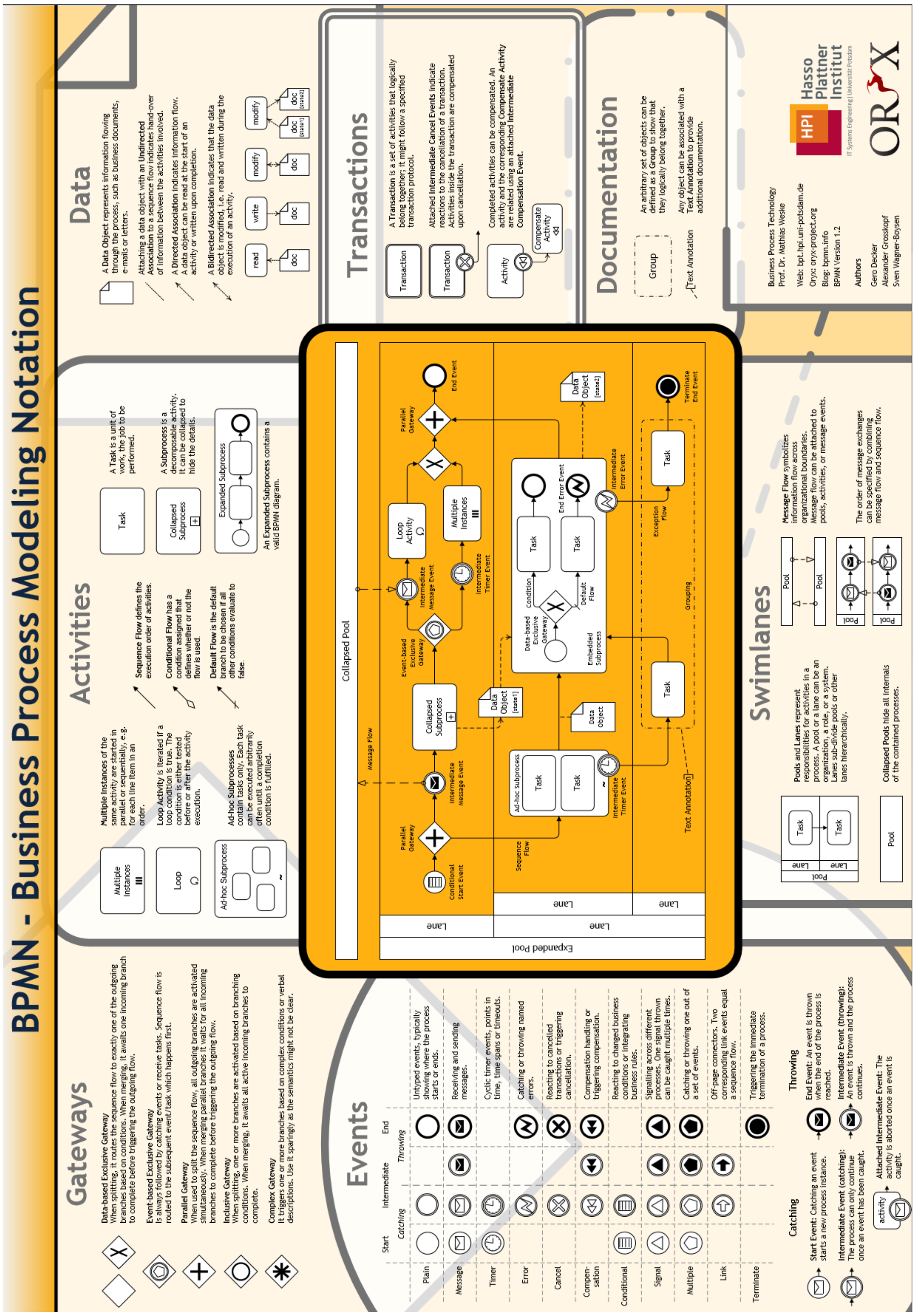


Figure 3.24 – Entity Relationship schema solution to Exercise 3.1

Appendix 3.2 – The full set of modeling constructs of BPMN



Appendix 3.3 - References for the BPMN model and the ER model

For BPMN are available in the Web a wide set of presentations and documents, especially those documents in the site of the Object Management Group. Such documents can be found looking in Google for the keywords “Introduction to BPMN pdf” or similar.

For a comprehensive introduction to the Entity Relationship model and conceptual design methodologies download from http://hdl.handle.net/10281/97114 the following documents:

- C. Batini - Database modeling and design: Part 2 – The Entity Relationship model
- C. Batini - Database modeling and design: Part 3 – Conceptual Database Design

You may also be interested to video lessons of the Mooc Course on Database Modeling and Design by Carlo Batini freely accessible at http://elearning.unimib.it/course/view.php?id=17573
Videos were originally “Engineered & Powered by Gruppo Spaggiari Parma S.p.A.” in 2015.

See also

C. Batini, S. Ceri, S. B. Navathe, - Conceptual Database Design: An Entity-Relationship Approach, Pearson, 1992.

Chapter 4 - Efficiency assessment of the process as-is through a case study

4.1 Introduction to efficiency

In this chapter we will start to address the issue of efficiency of processes and information systems starting from the case study of exam registration; in this way the reader may play an active role in the comprehension of concepts that are the basic pillars of the efficiency issue.

Efficiency is one of the two basic issues related to the assessment of processes and information systems, the second one being effectiveness. Efficiency and effectiveness can be introduced at a glance using the example of Figure 4.1, in which we focus on the care process of patients in a hospital. The process starts with a request of admission of a patient; it is made of several activities related to medical examinations, exams that are prescribed, therapy adopted, at the end of which the patient is discharged.

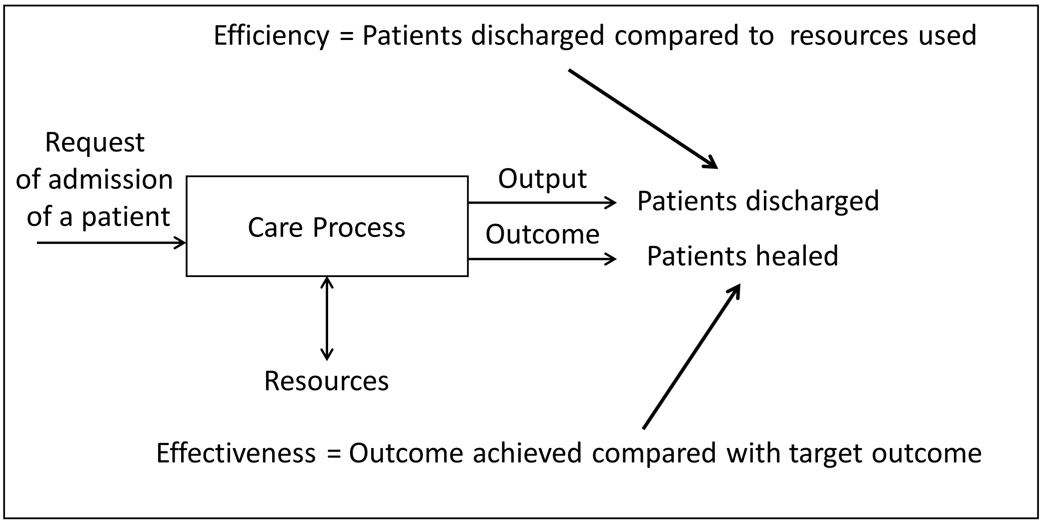


Figure 4.1 – Examples of efficiency and effectiveness related to patient care processes in a hospital

The quality of care in the hospital can be evaluated in terms of the following elements:
a. the amount of resources (human, technological, infrastructural) spent in the process of care of patient discharged; this category is related to efficiency;
b. the outcome of the process of care, namely, whether the care resulted in a healing of patients discharged; in this case we refer to effectiveness.

Notice that, intuitively, efficiency is easier to measure than effectiveness; in fact (see Figure 4.1) efficiency corresponds to the number of patients that are discharged compared to the amount of resources that have been used in the process of care. On the contrary, it is not easy at all to quantify the concept of “being healed” and the related outcome compared with the target outcome.

We will investigate the above issues in more depth in the next chapter; from now on, in this chapter we focus mainly on the efficiency issue, that we discuss intuitively based on the exam registration case study. The requirements of the case study are reproduced in the following box.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office.
The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher.
When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office.
The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students:

- have in their study program the course corresponding to the exam, and
- are in good standing with University taxes.

In case of errors or inconsistencies in the register, the office sends the register back to the teacher.
When the records are complete, the grades of exams are updated in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

A representation in terms of the BPMN language of the related process is shown in Figure 4.2.

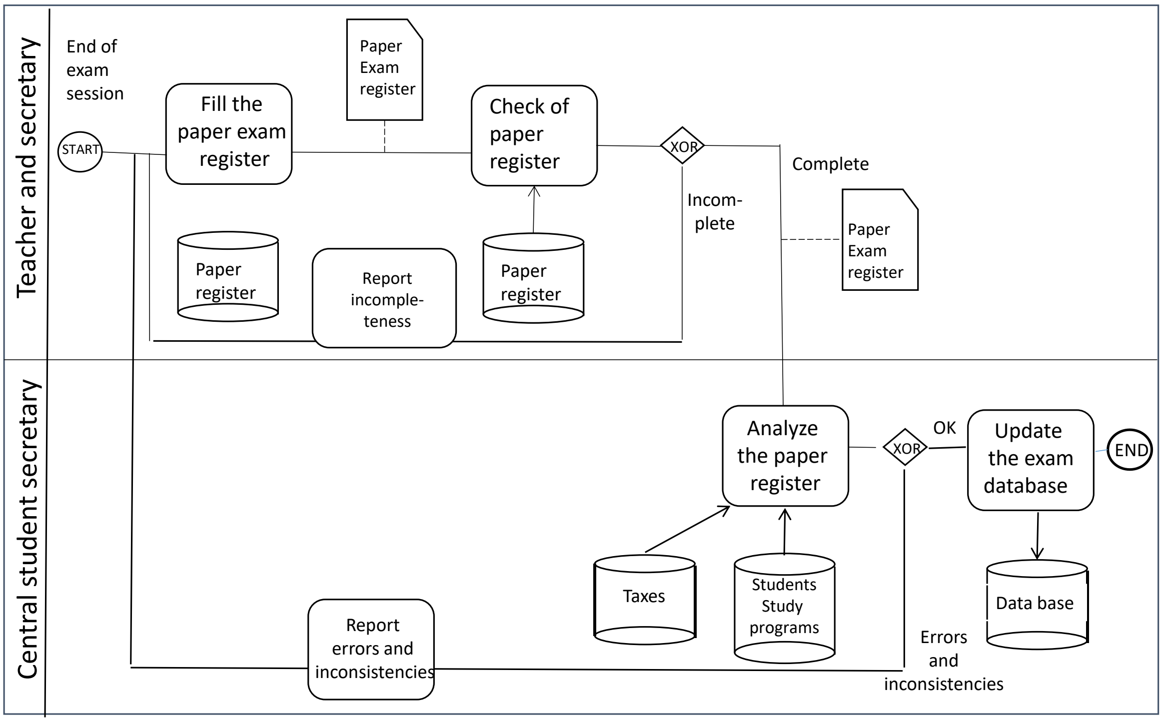


Figure 4.2 – The exam registration process

Following the above examples, in order to measure efficiency we may adopt the following general formula:

Efficiency of the process = Number of outputs produced in a given interval of time / Amount of resources needed in producing the output

Now, how can we measure the output of the process and the resources involved in the exam registration case study? The output of the process is easily identified as the exams and related grades that are inserted in the data base at the end of the process. Coming to the resources involved, we have several possible choices, among them a. human resources, b. economic resources (namely money spent), and c. time, resulting in the three following tentative definitions of efficiency and related measures of resources:

- Human resources efficiency – Human resources spent by the organization
- Economic efficiency – Money spent by the organization
- Time efficiency – Amount of time needed to produce an exam registration

Focusing first on human resources efficiency, we may measure the amount of resources involved as the number of hours that are spent by human resources, so that the formula for efficiency becomes

Human resource efficiency = # of registered exams / ∑ of hours spent by human resources

All the concepts involved in the measurement process are shown in Figure 4.3 (the symbol # means number). Notice that the formula we have considered corresponds to our intuition that efficiency increases when, on equal human resources, the number of exams registered is higher.

Input	Students enrolled in an exam
Output	Exams registered in the data base
Measure of output	# of registered exams
Resource considered	Human resources involved in the process
Measure of resource	# of hours spent by human resources
Measure of efficiency	Output / # of hours spent
Name of efficiency dimension	Human resource efficiency
Measure of efficiency	# of exams / # of hours spent

Figure 4.3 – Elements that are be considered in the evaluation of efficiency

We have now to go one step ahead and identify two different modalities in which the process occur, corresponding to

1. Absence of errors and inconsistencies in the process
2. Presence of errors and inconsistencies, referred to:

- Incompleteness of records
- Inconsistency in student's data on taxes or else in the study program

What do we mean by error? We mean the fact that the teacher misses to fill some data in the registry such as the date of birth of the student. While an inconsistency occurs if the central student office discovers that the student did not pay taxes or could not be enrolled in the exam due to the absence of the corresponding course in the study plan.

We may interpret the presence/absence of errors as a kind of desired outcome of the registration process, since errors in registrations delay the overall process, and consequently delay the moment in which the student have their exam registered. In this sense, errors are related to effectiveness, the more the errors, the lower is the effectiveness of the process as perceived by students.

Notice that while the process in presence of errors is the one we have reproduced in Figure 4.2, in the process without errors the two activities that are performed to produce feedbacks to teachers do not occur, resulting into the process shown in Figure 4.4.

Furthermore, besides human resource efficiency we are interested in economic efficiency, which corresponds to the money spent in paying wages to human resources involved, resulting in the four cases shown in Figure 4.5. Among them, we will discuss the three cases highlighted in bold in the remaining part of the chapter.

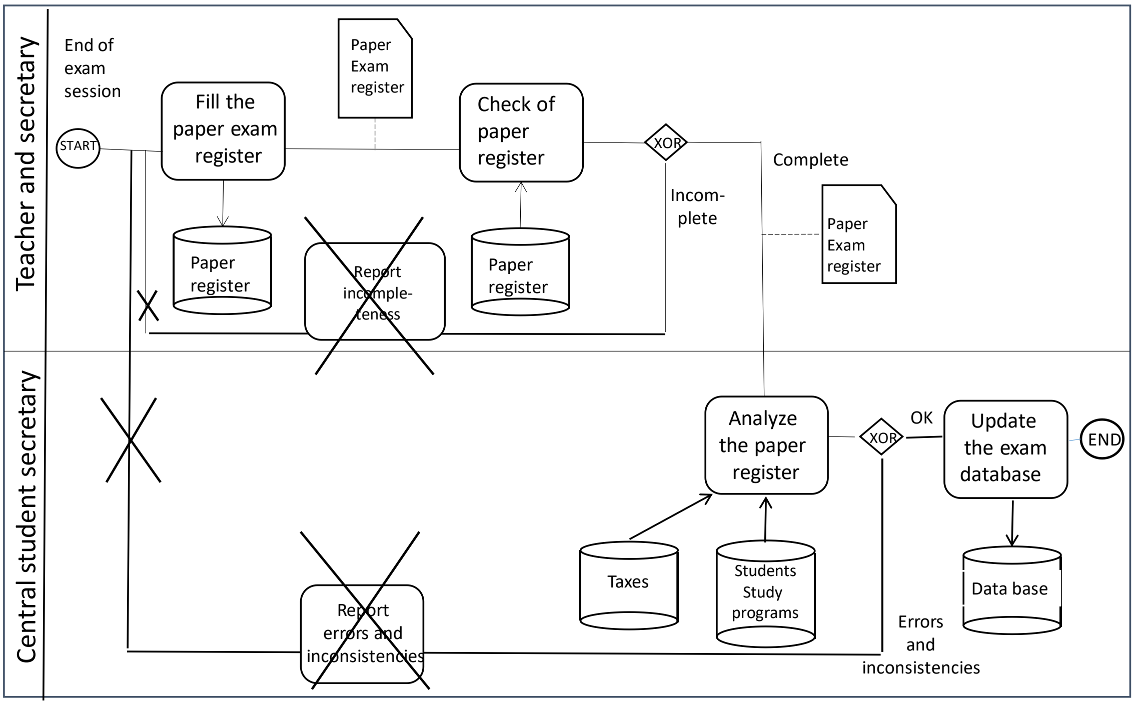


Figure 4.4: Process in absence of errors

Type of efficiency / errors →	Absence of errors	Presence of errors
Human resources efficiency	**X**	X
Economic efficiency	**X**	**X**

Figure 4.5: The cases that will be discussed in the following of the chapter

4.2 Human resources efficiency in absence of errors

We have first to specify the nature of activities, meaning:

- the types of human resources involved, and
- the types of activities, in terms of manual vs software activities.

In our case, three types of human resources are involved:

- teachers
- department student secretaries
- central student secretaries

The resulting process description (in which we have deleted activities not involved in the process in absence of errors) is shown in Figure 4.6.

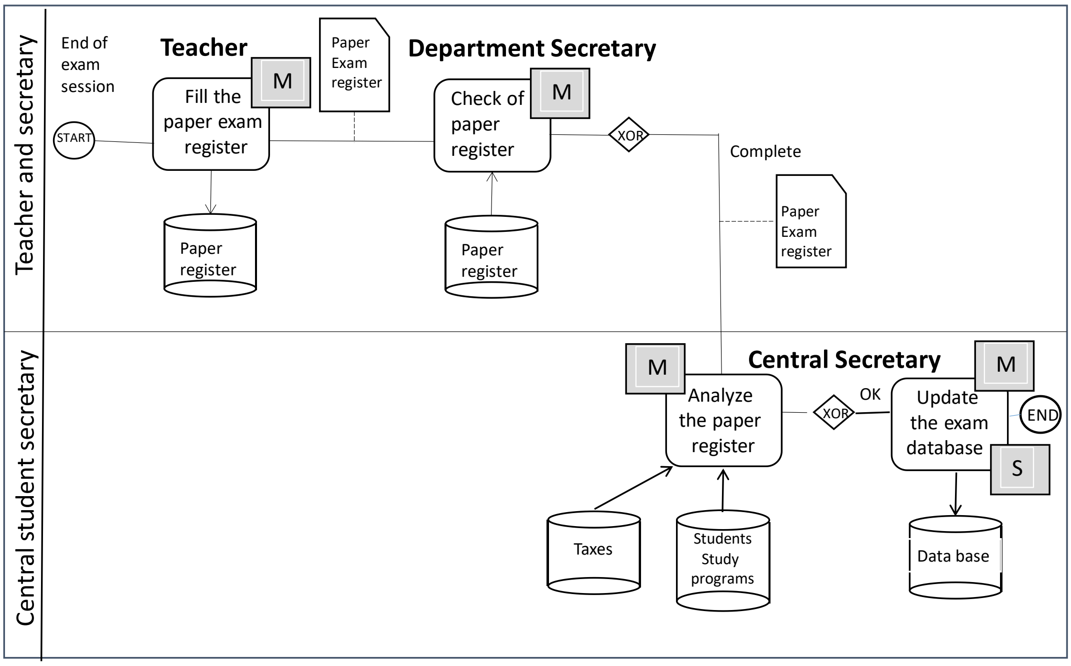


Figure 4.6: Process in absence of errors where types of human resources and types of activities (M for manual or S for software) are highlighted

We have now to choose the period of time during which we measure the above indicators; we assume a period of one year. Furthermore, we have to quantify the parameters that allow us to quantify the output: in this case we assume parameters and values in Figure 4.7.

Parameters to be considered in efficency evaluation	Value
Teachers	666
Average number of courses for each teacher	3
Courses	2.000
Exam sessions every year	5
Average number of exams for each session	20
Output: Number of registered exam	200.000

Figure 4.7 – Parameters and corresponding values considered in efficiency assessment

We have evaluated the output in terms of the number of teachers and the average number of courses taught, but we could also have started directly from the number of courses. Then, we have multiplied the number of courses by the number of the exams sessions (that in Italy are five) and the average number of exams for each session, coming to the final number of exams registered corresponding to 200.000.

We have now to focus on human resources involved. We assume to evaluate such parameters in the case of 100 exams managed in the process. We may make some interview, or else observe teachers and secretaries at work, and conclude that the time spent expressed in hours to manage 100 exams, is the time shown in Figure 4.8. In Figure 4.9 we evaluate the total time spent by the different resources each year.

Resource	**Activity**	**Time (in hours)**
Teacher	Insertion of data	1
Department secretary	Completeness check	0,2
Central secretary	Analysis	0,2
Central secretary	Insertion in the Data Base	0,8

Figure 4.8: Amount of time spent by the three human resources for 100 registered exams

Human resource / efficiency →	Hours for 100 exams	# exams	Total hours
Teacher	1	200.000	2.000
Department secretary	0,2	200.000	400
Central student secretary	1	200.000	2.000

Figure 4.9: Amount of time spent by the three human resources for 100 registered exams

A more effective representation of critical situations is obtained by projecting the human resources on the activities involved in the process; see Figure 4.10, where the human resources are highlighted with frames of different thickness.

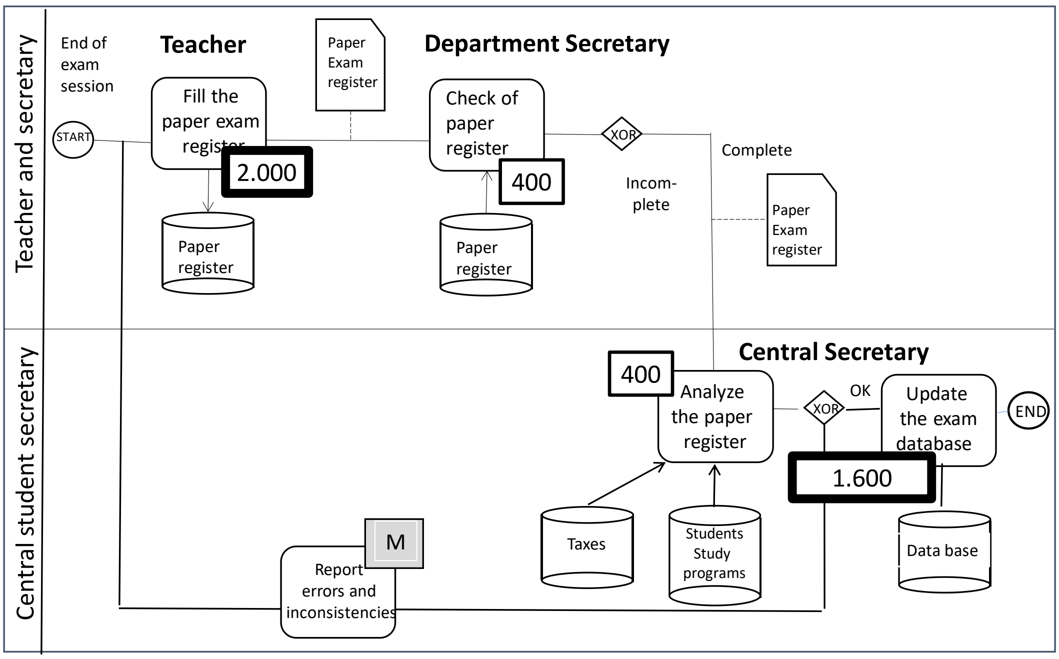


Figure 4.10: Human resources involved in the exam registration process projected onto the activities

4.3 Economic efficiency in absence of errors

We now move to economic efficiency in absence of errors. In this case, we have to start the evaluation from the hourly cost of the different human resources, to be multiplied by the total number of hours in a year, resulting in the table in Figure 4.11.

Human resource / efficiency	Hourly cost	Hours for 100 exams	# exams	Total hours	Total Cost in €
Teacher	40 €	1	200.000	2.000	80.000
Department secretary	15 €	0,2	200.000	400	6.000
Central student secretary	15 €	1	200.000	2.000	30.000

Figure 4.11 - The full set of parameters that are taken into account in economic efficiency

We can also evaluate a few synthetic figures, such as

- the total cost, equal to 116.000 euros
- the cost per registered exam = 116.000 / 200.000 euros, corresponding to 0,58 euros.

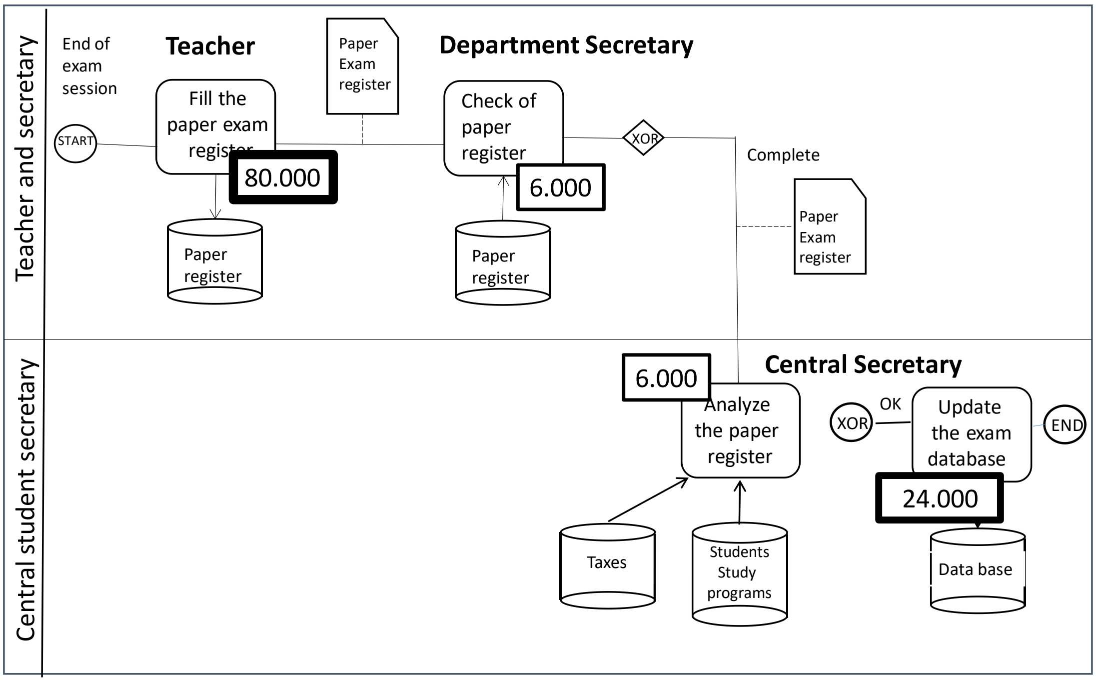


Figure 4.12: Economic costs in the exam registration process projected onto the activities

Finally we may project the costs onto the activities, see Figure 4.12. In this case, there is a noticeable difference between the costs of the activities in charge to teachers w.r.t. the costs of activities in charge of the central student secretary, due to higher salaries of teachers.

4.4. Economic efficiency in presence of errors

In this case, due to the presence of errors, the two loops present in the BPMN representation are executed a number of times that is related to the errors and inconsistencies discovered. We assume that after a new inspection of the process, focused on evaluating the percentage of errors, in the 5% of cases the two gateways lead to a feedback execution of the initial activity “Fill the paper registry”. We have now to evaluate the new number of executions of remaining activities.

We first evaluate (see Figure 4.13) the number of executions of activities in the first lane due to the first loop. Since for five times the execution of the process starts again, the “Fill the paper registry” is executed five more times, while the “Check of paper register” is not performed, since the probability of incompleteness after the correction is assumed negligible.

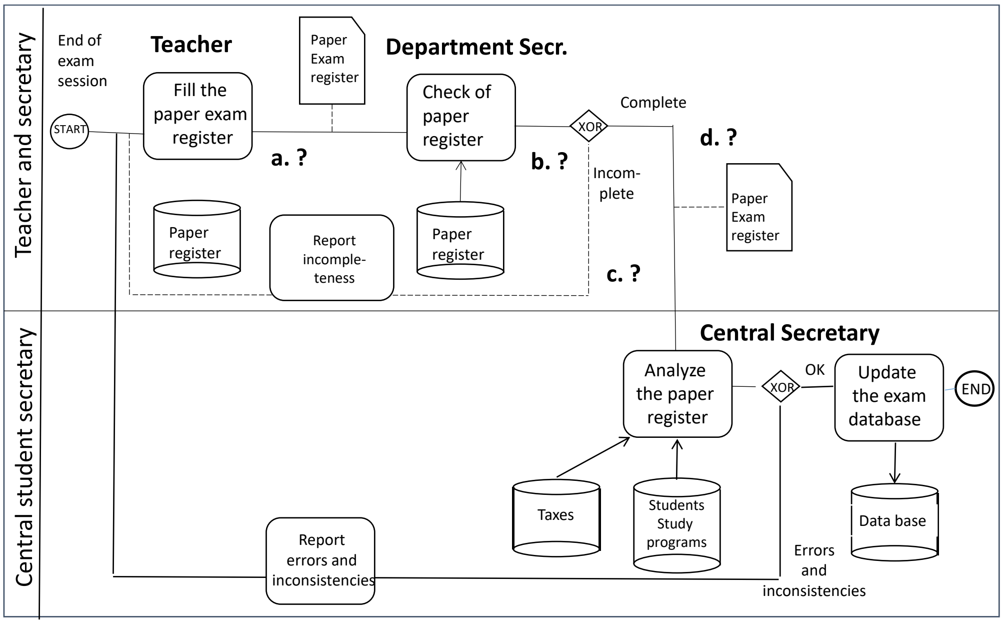


Figure 4.13 – Number of executions of activities in the first lane due to the first loop

We have now to evaluate the number of executions of activities in the two lanes due to the second loop; the reader is invited to provide a solution to this question.

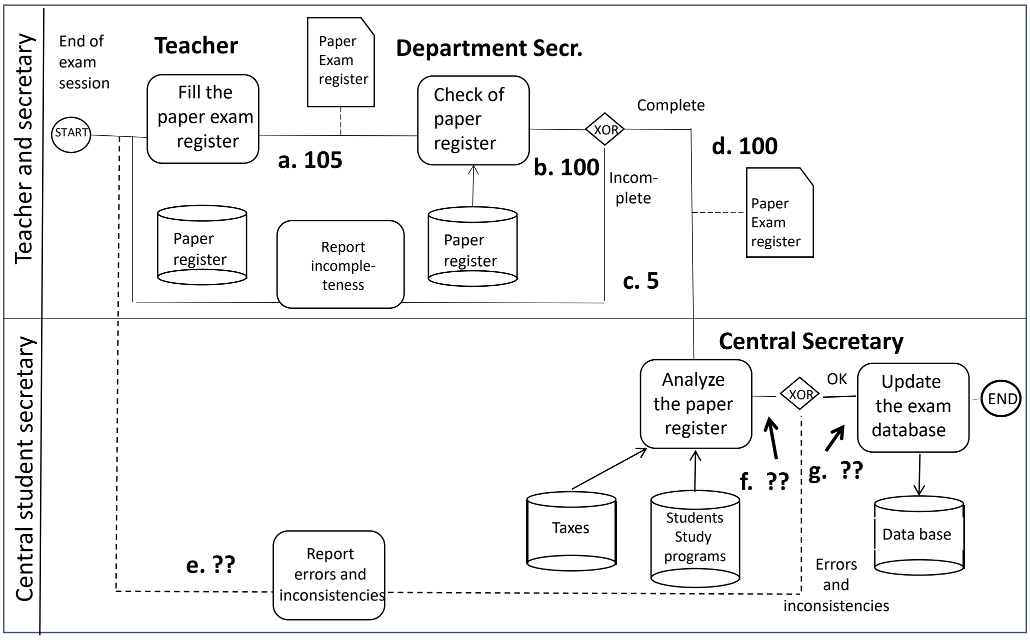


Figure 4.14 – Which is the number of executions of activities in the two lanes due to the second loop?

The solution appears in Figure 4.15.

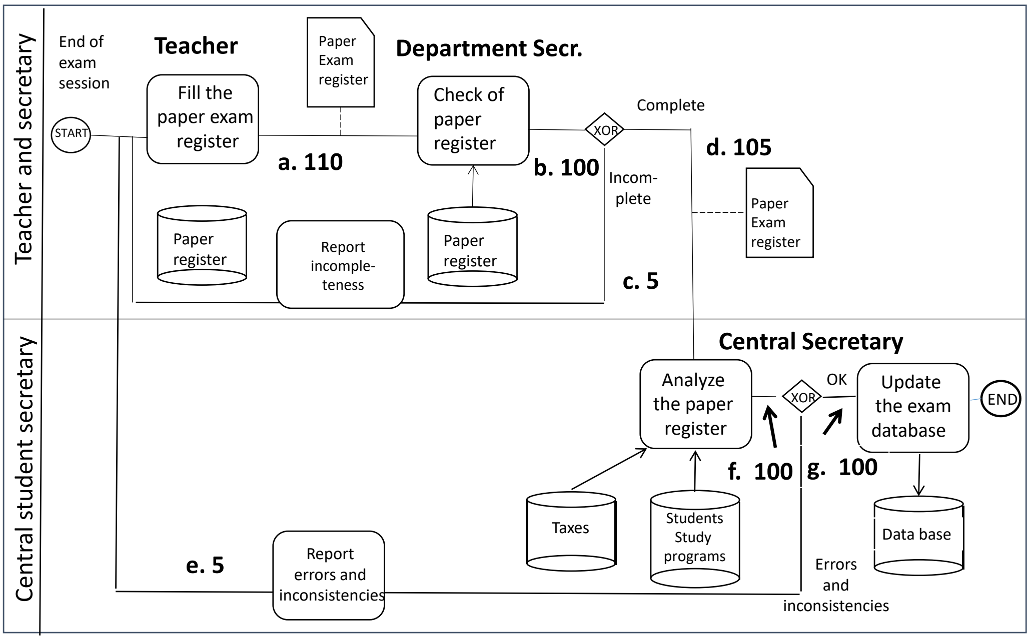


Figure 4.15 – Number of executions of activities in the two lanes

Now, due to the second loop, the “Fill the paper registry” activity is executed five more times, while the other activities are performed 100 times, for the same reasons discussed before. The new costs appear in Figure 4.16.

Human resource / efficiency	Hourly cost	Hours for 100 exams	# exams managed	Total hours	Total Cost in €
Teacher	40 €	1 → 1.1	200.000 → 220.000	2.000 → 2.200	80.000 →88.000
Department secretary	15 €	0,2 → 0,2	200.000 → 200.000	400 → 400	6.000 → 6.000
Central student secretary	15 €	1 → 1	200.000 → 200.000	2.000 → 2.000	30.000 → 30.000

Figure 4.16 – New costs

We evaluate also in this case synthetic figures, such as the total cost equal to 124.000 euros, with an increment of approximately the 7%, and the cost per registered exam is equal to 124.000/200.000 euros, corresponding to approximately 0,62 euros.

The incidence of costs related to the most costly resource, namely teachers, is even higher than before, and this is because it is in charge of teachers to perform all corrections.

The efficiency assessment so far has focused on aggregated figures of human effort and costs. We conclude that the most relevant component of costs is due to teachers, but we did not identify specific weaknesses in the process nature, that may allow us to orient the subsequent activities of the process-to-be design. An analytic detection of the reasons of inefficiency appears in Figure 4.17. We see that in substance the inefficiencies are due to:

1. late control of errors
2. usage of the most costly resource (teachers) in clerical activities and in the correction of errors that only in part are to teachers themselves.

1. The teacher must enter data such as Student Serial Number, Name, Surname, etc. that are already in possession of the administration.
2. Many activities of the Department and University central office could be easily automated, for example the verification of completeness of the minutes, leading to greater efficiency.
3. As a result, department secretaries could perform other activities with higher added value, e.g. student counseling.
4. The upstream process, acquisition of regisatration requests, is already automated, so the cost of adopting a digital register is very low.

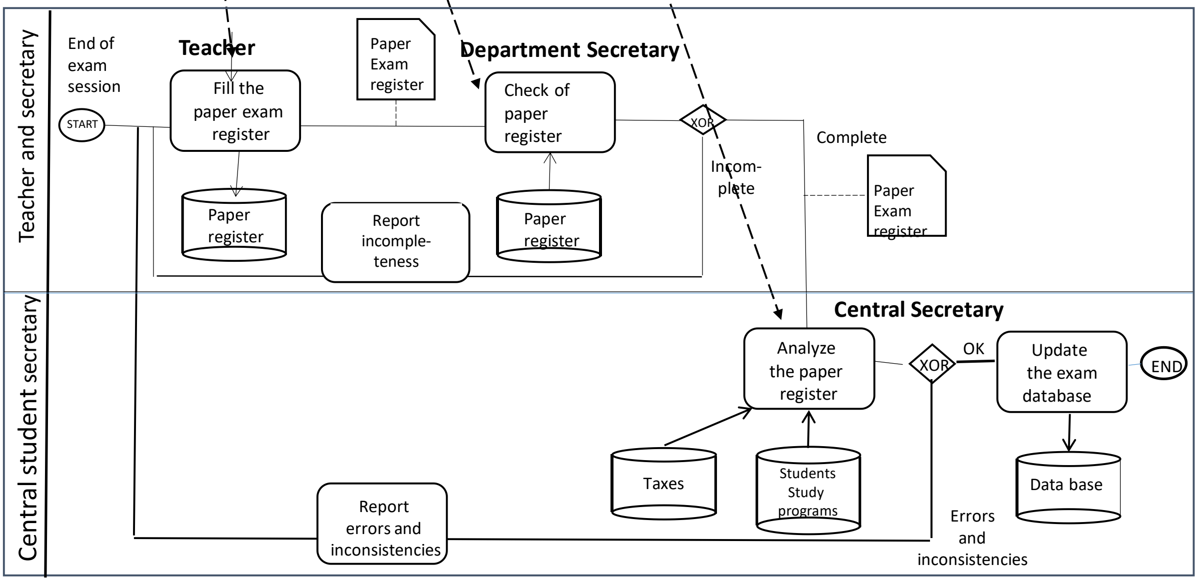


Figure 4.17 – Analytical detection of reasons of inefficiency

3. Presence of a lot of manual activities that could be avoided re-using digital data that are acquired in the initial enrollment of students at university (such as e.g. the date of birth) or else in the phase of exam enrollment by students (e.g. the student serial number).

The above analysis will be precious when in Chapter 6 we will design the process-to-be. We conclude here the chapter.

Chapter 5 - Efficiency and Effectiveness assessment

5.1 Introduction

In the previous chapter we have introduced the concept of efficiency through a case study. We observe now that efficiency and effectiveness can be seen as characteristics that refer to the more comprehensive concept of **quality**. When we deal with the quality of an organizational system and of an information system, we may refer to anyone of the layers we have introduced in Chapter 1, namely to:

- Laws and rules
- Organization
- Goods and services produced
- Processes
- ICT Technologies
– Data
– Software
– Hardware
– Network

For instance, we may be interested to the understandability of a law, the value in use of a service, the currency of data, the modularity and maintainability of a software application, the performance of a server. The interested reader may go in more depth on issues related to quality in [Viscusi, Batini et al 2006] and [Batini & Scannapieco 2016].

In this book we focus on quality of processes. Qualities can be expressed in terms of general categories, such as efficiency and effectiveness. A general definition of efficiency and effectiveness is as follows:

1. *Efficiency*: the amount of resources (including time) needed for output provision
2. *Effectiveness*: how close the output of the process achieves user's goals

As we may see, such definitions do not provide methods to measure quality. We may come closer to measuring quality through *dimensions*, corresponding to specific cases of such categories, such as human resource efficiency or economic efficiency. Dimensions can be expressed quantitatively, through a measure of a physical phenomenon (e.g. for the efficiency of a process, the time required to produce an output). There is a second way to measure of quality dimensions, through the perception of users, leading to the perceived quality dimensions, e.g. perceived efficiency or effectiveness, such as e.g. the little faces we may select to express our satisfaction once we leave a passport control office in an airport.

	Quantitative Assessment	Perceived assessment
Efficiency	**X**	**X**
Effectiveness	**X**	**X**

Figure 5.1 – Types of qualities and related measures considered in the chapter

In the previous chapter, we have introduced through a case study efficiency measured quantitatively; in this chapter, we will consider more systematically all cases appearing in Figure 5.1.

5.2. Quantitative efficiency

We introduce first the input/output view of a process shown in Figure 5.2, where, for simplicity, a process is composed of a linear sequence of activities.

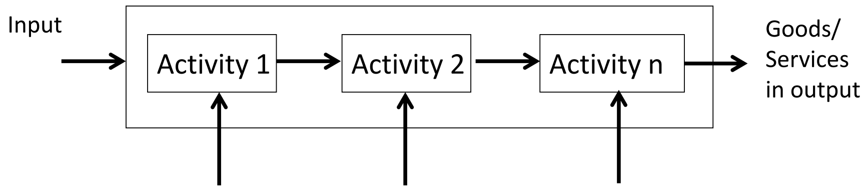


Figure 5.2 – Input/output view of a process

According to Figure 5.2 a process transforms goods or service requests in input into goods or services provided in output; to do so, it uses several types of resources: Economic, Human, Information, Technologies, Infrastructural, Time. Correspondingly, we may define several types of efficiency:

- Economic efficiency
- Human resource efficiency
- Information efficiency
- Technologies efficiency
- Infrastructural efficiency
- Temporal efficency

Also in this chapter we make use of examples to browse around the topics we aim to discuss. The first example concerns the process of care in a hospital.

Hospital case study - A hospital is a place where patients are admitted to treat illness or trauma. We assume to observe the hospital for a period of one calendar year. The generic care process in a hospital may be described as follows: patients are hospitalized in an hospital, then they receive treatment, and when the treatment is considered finished, they are discharged from the hospital. Let us distinguish in the above sentence the different events and activities.

Exercise 5.1 – Characterize the following parts of sentences

1. Patients are hospitalized in a hospital -> ??
2. then they receive treatment →??
3. and when the treatment is considered finished → ??
4. they are discharged from the hospital → ??

in terms of the following BPMN modeling constructs:

- Input
- Event
- Activity
- Output

Solution in the Appendix.

A general methodology for efficiency evaluation that we will apply in the following is shown in Figure 5.3.

1. Fix the **output** you want to consider
2. Provide a **measure** for output
3. Fix the **resource** involved in efficiency
4. Provide a **measure** for the resource
5. **Evaluate efficiency** as: measure of output / measure of resource

Figure 5.3 – A methodology to measure efficiency

Among the different types of efficiency we will apply such method in the following to temporal efficiency, human resoruces efficiency and economic efficiency. We address now each one of them.

5.2.1 Temporal efficiency

We apply the methodology of Figure 5.3, selecting as output the patients discharged in a year, and for resource the total number of days of hospitalization, see Figure 5.4.

Parameter	**Value**
Output	Patients discharged in a given period of time, e.g. a year
Measure of output	Number of patients discharged = 100
Resource considered	Days of hospitalization
Measure of resource	∑ days of hospitalization = 500
Measure of efficiency	Output / Time
Name of efficiency dimension	Temporal efficiency
Quantitative efficiency	0,2 patients a day

Figure 5.4 – Parameters to be quantified in the methodology of Figure 5.3 and related values

The reader is invited to associate a physical meaning to the quantitative efficiency value in Figure 5.4.

Did you succeed? it is not easy to associate to “0,2 patients a day” to a physical phenomenon…..

Therefore, we can adopt an inverted measure for quantitative efficiency, shown in Figure 5.5. This measure, “(average) number of days of hospitalization for each patient” is easy to understand, since it is a time, not its inverse.

Parameter	Value
Output	Patients discharged in a year
Measure of output	Number of patients discharged = 100
Resource considered	Time → Days of hospitalization
Measure of resource	∑ days of hospitalization = 500
Measure of efficiency	Output / Time
Name of efficiency dimension	Temporal efficiency
Quantitative efficiency	0,2 patients a day
More informative the inverse → Time / Output	5 days of hospitalization for each patient

Figure 5.5 – More informative measure of temporal efficiency

Notice that while in the previous formula the efficiency increases when the total number of days decreases, in the new formula the efficiency increases when the number of days of hospitalization decreases; that said, this is a more intuitive measure of efficiency.

5.2.2 Human Resource efficiency

As to human resource efficiency, we may select doctors as human resource considered, and adopt as measure of the involved resource the number of full time doctors, namely to the number of "heads" involved (see Figure 5.6).

Parameter	Value
Output	Patients discharged in a year
Measure of output	Patients discharged in a year = 100
Resource considered	Human resources → doctors
Measure of resource	# of full time doctors = 5
Measure of efficiency	Output / # of full time doctors
Name of efficiency dimension	Human resource efficiency
Quantitative efficiency	100 / 5 doctors = 20 patients for each doctor

Figure 5.6 – First measure of human resources efficiency

The problem in this approach is: what does it mean "doctors"? It may very well happen that one doctor works part time in the hospital and another one works full time, this measure does not distinguish the two cases and leads to two heads. It seems more meaningful to consider, instead of heads, the number of days the doctors have worked in the hospital.

In Figure 5.7 we show two different measures of the human resource efficiency, where as before we place the measure of the resource first at the denominator, have in this case the efficiency increases with the resources needed, and then at the numerator, characterized by the inverse trend.

Parameter	Value
Output	Patients discharged (dimessi)
Measure of output	Patients discharged in a given time interval = 100
Resource considered	Human → doctor days
Measure of resource	# of doctor days = 500
Measure of efficiency	Output / # of doctor days
Name of efficiency dimension	Human resource efficiency
Quantitative efficiency	100 / 500 doctor days = 0,2 patients for each doctor day
More informative measure	500 doctor days / 100 patients = 5 doctor days for each patient

Figure 5.7 – More informative measure of human resources efficiency

5.2.3 Economic efficiency

In economic efficiency the unit of measure of the resource is euros, we have to choose the human resource/resources whose wages to quantify. In this case, we can focus on doctors leading to the two measures in Figure 5.8.

Parameter	Value
Output	Patients discharged (dimessi)
Measure of output	Patients discharged in a given time interval = 100
Resource considered	Economic → Salary for doctor/day
Measure of resource	# of doctor days * payment of a day = 500 * 200 €
Measure of efficiency	Output / cost of doctor days in euros
Name of efficiency dimension	Economic efficiency
Quantitative efficiency	0,001 patient for an euro
More informative	100.000 euros / 100 patients = 1.000 euros per patient

Figure 5.8 – Measure of economic efficiency

We recall that In Britain, when she was Prime minister, Margaret Thatcher launched a reform of the hospitals to reduce health spending. Among other measures, the average number of days of hospitalization of the patients was considered as the measure of efficiency for the government plan, and it was decided that in order to reduce central government expenses the new target had to be *lower*.
Notice that the Thatcher reform was focused only on the efficiency and did not take into account the effectiveness of the care process.

Home work - Browsing in the Web, reconstruct in brief the issue of the Thatcher health reform, and how it was subsequently criticized.

5.2.4 Efficiency in a call center

We aim now to measure the efficiency of a call center dedicated to the reservation service of a medical examination. Let us consider the two phases of the related process, see Figure 5.9:

1. Taking charge of phone calls
2. Booking management.

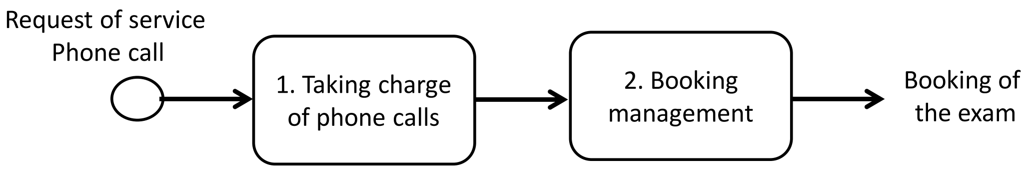


Figure 5.9 – Temporal efficiency in a call center

Taking charge of phone calls - In this case inputs, outputs and resource are as follows (see Figure 5.10):

- Input: Phone calls Pi
- Output: Responses to phone calls
- Resource: Time

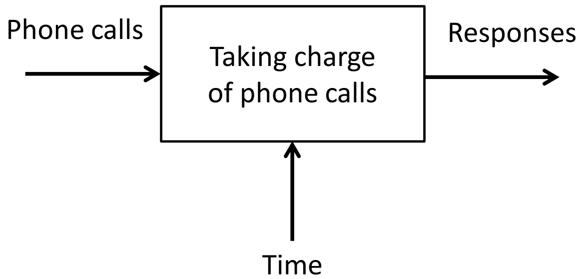


Figure 5.10 – Temporal efficiency of the first activity

The temporal efficiency can be measured as
Efficiency = average time to respond = $\sum_i \Delta$ (time at response to Pi, time at call Pi) / # responses
where Δ represents a difference. Here again the resources are in the numerator, and the outputs in the denominator, therefore the efficiency increases as the value of the ratio decreases.

Booking management - In this case inputs, outputs and resources chosen are (see Figure 5.11).

- Input: Requests Ri
- Output: Confirmations
- Resource: Time

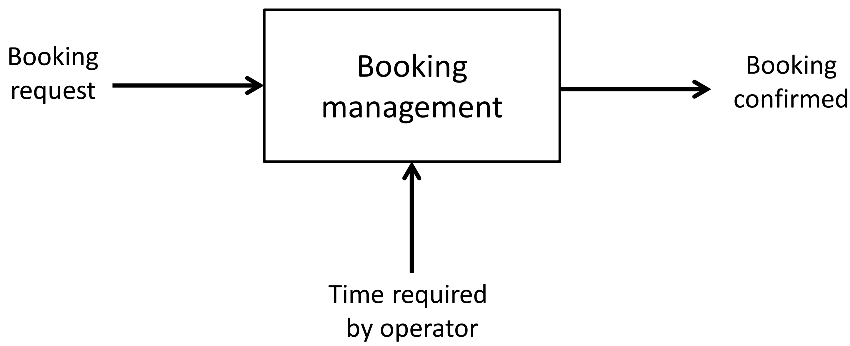


Figure 5.11 – Temporal efficiency of the second activity

The temporal efficiency can be measured as

Efficiency = average time to manage booking successfully = $\sum_i \Delta$ (time at confirmation to Ri, time at request Ri) / # confirmations. The efficiency of the whole process is
Efficiency = average time to respond to call + average time to manage booking succesfully

Exercise 5.2 - Measure the temporal efficiency of a single operator OPi in the booking activity, see Figure 5.12.

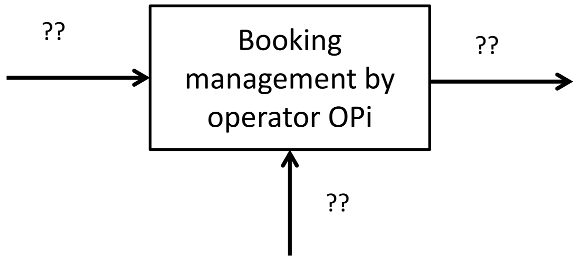


Figure 5.12 – Temporal efficiency of a single operator OPi

See the solution in the Appendix.

5.3 Quantitative Effectiveness

We now discuss effectiveness. Contrary to efficiency, which focuses on resources spent in the process, effectiveness is focused on users and their goals and needs. For this reason, in this section we focus on processes and information systems that provide services requested by users. The effectiveness concerns the ability of the system/process/service to supply the final user with what the user actually requests from the system, also called his/her goal or outcome, see Figure 5.13.

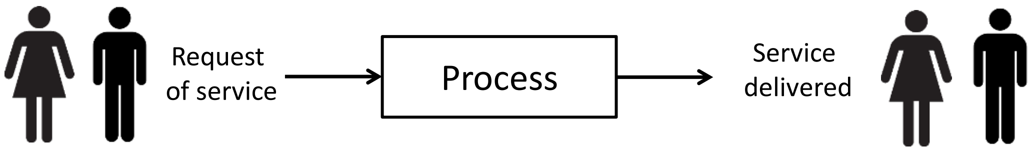


Figure 5.13 – A process providing a service to requesting users

Being so strictly related to user needs, the evaluation of effectiveness depends on the type of user (sometimes on the specific user) to whom the service is addressed, since each type of user obtains from the service a different value, called value in use. The value in use that a user perceives from the acquisition of a (good or) service can be quantitatively expressed as a function of benefits obtained and all sacrifices made to obtain the service:

Value in use (service; user) = f (benefits; sacrifices)

where the function f can be a ratio (/) or a difference (-). The benefits obtained can be expressed in terms of the degree of achievement of the goal of the user (see later), so they depend on the service and on the user. The typical sacrifices are:

- Cost of the service
- Time and resources spent to invoke the service
- Risk of failure in service delivery

We will now discuss four examples, at different levels of detail, exploring services related to the following issues:
a. the security of citizens in England;
b. the process of care in a hospital;
c. the fines to drivers of cars;
d. services provided by a call center.

5.3.1 Security of citizens in England

English city police provides detailed statistics on crimes committed in districts of all cities in England, an example appears in Figure 5.14 refereed to the city of Leicester. This information has a <u>positive</u> value e.g. for foreign Erasmus students that moves to an English city from abroad; they can exploit this information to rent an apartment in a safe district, where safe can be measured on a statistical basis. At the same time, such information has a <u>negative</u> value for property owners, that in case they rent a room or an apartment in a unsafe neighborhood, have the rental service depreciated, due to the decrease of the demand in the neighborhood.

All crime	1241
Burglary	75
Anti-social behaviour	317
Robbery	12
Vehicle crime	57
Violent crime	181
Public disorder and weapons	45
Shoplifting	172
Criminal damage and arson	102
Other theft	178
Drugs	48
Other crime	54

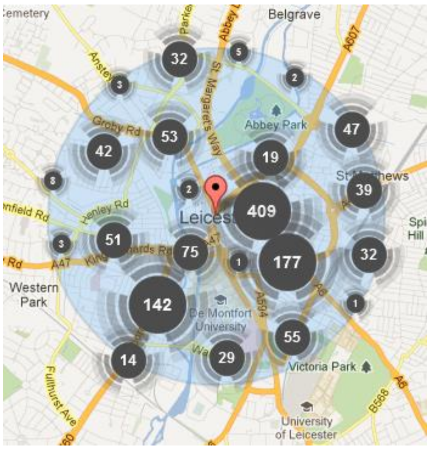


Figure 5.14: Crimes in an English city in a given year

In this case we have (at least) two users "segments" (as they are called in marketing): a. students in search of a safe apartment and b. property owners, whose goal is to rent apartments. Their goals and related values are in contrast, and cannot be reconciled.

5.3.2 The process of care in a hospital

A hospital is a place where patients are admitted to treat illness or trauma. The effectiveness of the process of care is a measure of the achievement of the goal of the patient; his/her goal is to be discharged by the hospital healed from the illness for which he/she was admitted to the hospital.

Hospitalized patients are not the unique users of the process of care, see Figure 5.15. Other users are the relatives of the patient, that have similar (but not identical) goals, and hospital managers, that can be seen as internal users of the process of care. The goal of hospital managers is certainly to increase the quality of care in such a way to make the hospital they manage more and more competitive, provided that the cost of cares and of medical care technologies is compatible with the available budget.

Type of user	Outcome
Hospitalized patient	Get out healed from the hospital (compatibly with the state of the disease)
Relatives of hospitalized patients	To get back their relatives healed from the hospital (compatibly with the state of the disease)
Hospital managers	Increase the quality of care with the same budget

Figure 5.15: Types of users

From the point of view of hospitalized patients, the effectiveness of a therapy for a given pathology can be measured as the number of patients healed for the pathology ratio total number of patients hospitalized for that pathology. But what does “healed” means? In order to choose the effectiveness measures for “being healed” we must look at the scientific literature on therapies.

In hospitals, to characterize the effectiveness of a therapy the concept of *appropriateness* is used. An appropriate therapy for a given pathology is defined as one that is expected to do more good than harm for a patient suffering that pathology. An inappropriate care service is one that is not expected to benefit the patient or, in the extreme case, may harm the patient. You may understand that, starting from the above definitions, measuring appropriateness could be (and is) very controversial…

Looking at the literature, we have a confirmation that do not exist shared and mature methodologies for measuring appropriateness. Therefore, the concept of appropriateness is often replaced by a more rough but easily measurable indicator, especially suitable for some types of serious diseases. The effectiveness of a therapy for a given pathology is measured as
= number of dead patients suffering the pathology / total number of patients hospitalized for that pathology.

In Italy, the magazine Wired used for several years in the past to compare the effectiveness of hospital on the site http://daily.wired.it/mappa_migliori_ospedali (2017: no longer operational), see in Figure 5.16 an excerpt of data published in 2017.

Hospital	Pathology	# of hospitalized	% of deaths
A.O Riuniti	Heart attack (HA)	240	10.09
A.O Riuniti	HA without angioplasty	109	20.62
A.O Riuniti	HA with angioplasty	69	2.15
Casa Sollievo	Heart attack	97	15.47
Casa Sollievo	HA with angioplasty	57	5.88

Figure 5.16 – An excerpt from the Wired site, 2017

In USA the HCAHPS (Hospital Consumer Assessment of Healthcare Providers and Systems) survey is claimed to be the first national, standardized, publicly reported survey of patients' perspectives of hospital care.

E.g. in https://health.data.ny.gov/Health/Cardiac-Surgery-by-Surgeon-Beginning-2008/dk4z-k3xb we may find data referred to hospitals in the State of New York, such as the dataset shown in Figure 5.17.

This dataset contains the a. number of cases b. number of deaths, c. observed, expected and risk-adjusted mortality rates for cardiac surgery by surgeon. Physician information is presented for each physician who:
a) performed 200 or more procedures during the three year analysis and/or
b) performed at least one cardiac surgery in each of the three years.

	Physician Name	Hospital Name	Procedure	Year of Hos	Number of Cases	Number of Deaths	Observed Mortality Rate	Expected Mortality Rate	Risk-Adjusted Mortality Rate	Lower Limit of Confi
1	Gorki H	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	21	1	4.76	2.01	6.47	
2	Gorki H	Lenox Hill Hospital	CABG	2009-2011	18	0	0.00	1.67	0.00	
3	Subramanian V	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	492	26	5.28	3.56	4.05	
4	Subramanian V	Lenox Hill Hospital	CABG	2009-2011	320	3	0.94	1.70	0.86	
5	Plestis K A	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	218	2	0.92	2.94	0.85	
6	Plestis K A	Lenox Hill Hospital	CABG	2009-2011	83	1	1.20	1.50	1.25	
7	Ciuffo G B	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	14	0	0.00	2.21	0.00	
8	Ciuffo G B	Lenox Hill Hospital	CABG	2009-2011	10	0	0.00	2.30	0.00	
9	Patel N C	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	747	13	1.74	2.19	2.16	
10	Patel N C	Lenox Hill Hospital	CABG	2009-2011	536	6	1.12	1.50	1.17	
11	Loulmet D F	Lenox Hill Hospital	CABG, Valve or Valve/CABG	2009-2011	22	2	9.09	3.31	7.50	

Figure 5.17 - Hospital ratings for the Hospital Consumer Assessment of Healthcare Providers and Systems (HCAHPS)

We have now to observe that the treatment of pathology often continues even after discharge, and that the previous measures do not take into account the period following the discharge from the hospital. Another measure of the effectiveness of a therapy for a given pathology that takes in account the following issues is:

Effectiveness = # of deaths for a given pathology, evaluated from discharge time to discharge time + six months / total number of patients hospitalized for that pathology

Another measure of effectiveness, which can be adopted for hospitals or specific processes of care, is shown in Figure 5.18. In this case, we compare two indicators, related to the % of patients that after an hospitalization for a pathology were subjected to a second one in the same hospital for the same pathology, and the number of hospitalized patients in a year.

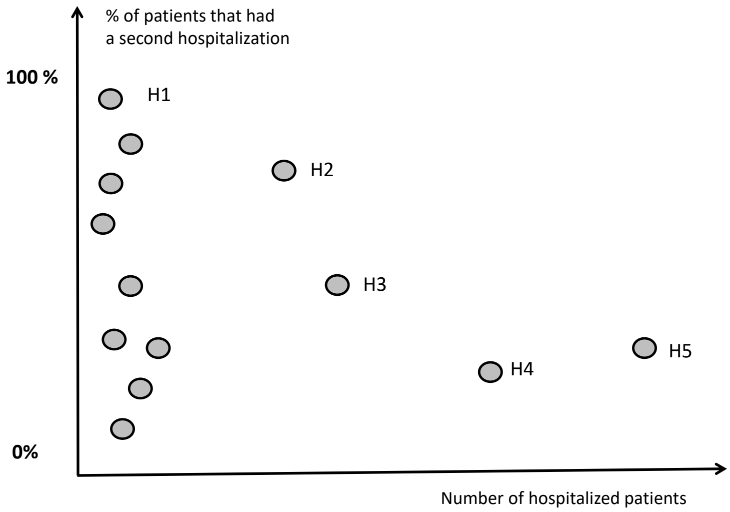


Figure 5.18 – Effectiveness of a hospital

The first indicator is an approximation (also called proxy) of the quality of care; in fact a second hospitalization is an indication that something didn't work in the first one. The second indicator is a proxy of the level of experience of doctors in the hospital; many hospitalized patients means that many different processes of care are performed, so much knowledge is collected and lessons are learned.

In Figure 5.18 hospitals H4 and H5 are the candidate for the best hospital; the indicator that can be used in this case to decide between H4 and H5 is

= % of patients that had a second hospitalization x number of hospitalized patients, or, better, the product of normalized values of the two parameters.

5.3.3 The process of fining car drivers and removal of cars

At least in Italy, especially in cities with an intricate urban road system and few parking lots, it is usual that cars are double-parked. So, traffic police officers in order to govern the phenomenon fine cars that are double-parked, and, in case cars obstruct the flow of cars or else the passage to pedestrians, they make them removed.

Users who receive a value (positive or negative!) from the fine and removal process, belong to types shown in Figure 5.19 together with their objectives

- Those to whom fines are made
- Those who transit by car in the streets where cars are wrongly parked
- Disabled people in wheelchairs
- Pedestrians
- The municipality (to whose income the fines contribute).

Type of user	Objective
Who get the fines	Pay as little as possible for infringements
Who transits by car	Transit in the streets having all the space available
Disabled persons in the wheelchair	Cross the road and easily go up or down the sidewalk
Pedestrians	Easily cross the road at intersections
Municipality	a. Increase revenues from payment of fines → Effectiveness as making money b. enforce the behavioural rules to ensure a smoother and safer traffic → Effectiveness as a social objective

Figure 5.19 – Types of users involved in the fines and removal of cars

Their objectives, as in the example of Section 5.3.1 may be conflicting. E.g. the subjective interest of the drivers that want to take a coffee at the cafeteria placed on the corner of two streets is to pay the least for fines and reduce the probability of removal; so, when they sip the coffee they may have a look at the street to check if a policeman is arriving. While the objective of disabled persons is to be able to cross the street in correspondence of the corner of sidestep, where usually the passage from the sidestep to the street is facilitated by a ramp.

5.3.4 Help desk services

In this case study we aim to find quantitative indicators of the effectiveness of an help desk. In the case study, the help desk is invoked by users of ICT technologies such as personal computers or printers, when they are in trouble in the usage of such technologies. Therefore, the goal of users is to have a quick response and a quick solution to their problems.

In order to conceive indicators for measuring the effectiveness of the help desk, we have first of all to look in detail its internal organization; an help desk is usually organized in three levels:

- The first level is the one that replies to user telephone calls, and manages initially the user request; such requests usually refers to one or more technologies in the technological stack that does/do not work correctly (e.g. a software application, a printer, etc.)
- The second level is involved when the first level is not able to give an answer.
- Suppliers of technologies are involved from the second level when the problem concerns a feature of the hardware/software that can be repaired only by the supplier that initially provided it.

We may define indicators that look at services provided by the help desk both as a whole and for each one of the levels. A list of effectiveness indicators is as follows.

- Maximum waiting time at call
- Percentage of incoming calls lost
- Percentage of calls resolved at the first level
- Average resolution time at the first level
- Percentage of calls resolved at the first + second level
- Average resolution time at the second level
- Percentage of unresolved calls passed to suppliers
- Maximum reaction time for suppliers
- Repair time to malfunction requested by suppliers

- Weekly operational period of time
- Operator education level
- Existence of customized profiles

Notice that some of the indicators influence directly the achievement of the goal of the user (e.g. the percentage of calls resolved at the first level), while other indicators influence the goal only indirectly (e.g. the operator education level).

We can go in more depth, providing for each indicator a threshold that should not be violated expressing the quality of service, see Figure 5.20. Notice that in several thresholds we did not adopt exact values, but instead levels that should be achieved with a certain probability. This choice is due to the stochastic nature of service provision, that suggests to avoid sharp formulas and adopt probabilistic ones.

#	Service level indicator	Treshold value
HD1	Maximum waiting time at call	< 30 seconds in 95% of cases
HD2	Percentage of incoming calls lost	< = 5%
HD3	Percentage of calls resolved at the first level	> = 70%
HD4	Average resolution time at the first level	< 3 minutes in 95% of cases
HD5	Percentage of calls resolved at the first + second level	> = 95%
HD6	Average resolution time at the second level	< = 5 minutes in the 85% of cases
HD7	Percentage of unresolved calls passed to suppliers	< = 5%
HD8	Maximum reaction time for suppliers	< = 30' in 99% of cases
HD9	Repair time to malfunction requested by suppliers	< = 8h in 99% of cases
HD10	Weekly operational period of time of the help desk	60 hours a week
HD11	Operator education level	Secondary school or higher degree
HD12	Existence of customized profiles	Availability of data on previous interactions with the user

Figure 5.20 – Indicators for effectiveness of help desk services, and corresponding thresholds

5.4 - Effectiveness & Efficiency together

We are often interested to analyze the effectiveness of processes and services not only in absolute terms, how we made in the previous section, but in its relation to the efficiency of process or service. In this case the formula for the evaluation is a function of both of them: f (Effectiveness; Efficiency). We show here a very tricky example. Consider in Figure 5.21 the ranking of universities in the world performed yearly by Shanghai Jiao Tong University (https://cwur.org/2016-17/Shanghai-Jiao-Tong-University.php). This ranking is referred to academic year 2016-17. Criteria that are adopted by Shanghai Jiao Tong University are based on three indicators:

- Number of former students (of the University) who took the Nobel Prize
- Number of Nobel prizes that are part of the teaching faculty
- Number of researchers with major scientific citations and studies published in specialized journals

The final score is normalized to 100. In the same year the Italian site Roars (www.roars.it) produced a new ranking (see Figure 5.22) based on the evaluation of a composite indicator for Effectiveness and efficiency, that uses the following formula:
f(Efficacy;Efficiency) = = Number of points in the effectiveness ranking / Yearly costs of management normalized to 100

Position	Nation	University	Score
1	USA	Harvard	100
2	USA	Stanford	73,3
3	USA	MIT	70,4
4	USA	UOC Berkeley	69,6
5	GBR	Cambridge	68,8
6	USA	Princeton	61
7	USA	CalTech	59,6
8	USA	Columbia	58,8
9	USA	University of Chicago	57,1
10	GBR	Oxford	56,6
11	USA	Yale	54,5
12	USA	UOC Los Angeles	50,7
13	USA	Cornell	50,5
14	USA	UOC San Diego	48,7
15	USA	Washington	47,8

Figure 5.21 – Ranking of Universities in the world performed by Shanghai Jiao Tong University, year 2016-17

Position	Nation	University	Score
1	Italy	Scuola Normale, Pisa	100
2	Italy	Ferrara	73,3
3	Italy	Trieste	70,4
4	Italy	Milano-Bicocca	69,6
5	GBR	Cambridge	68,8
6	USA	Princeton	61
7	Italy	Parma	59,6
8	Italy	Pavia	58,8
9	Italy	Perugia	57,1
10	Italy	Milano	56,6
11	GBR	Oxford	54,5
12	Italy	Torino	50,7
13	Italy	Politecnico di Milano	50,5
14	USA	UOC San Diego	48,7
15	USA	UOC Berkeley	47,8

Figure 5.22 – Ranking based on a combination of effectiveness and efficiency

It is very surprising that the first four Universities in the world on top of the ranking are Italian Universities! Is this a good news? Not at all.

Look at the next Figure 5.23, where two rankings on Universities in countries that are members of OECD (the organization for economic cooperation and development) are reported, based on the spending in % compared with the GDP (total and primary schools only) and the spending per student; you see that for both indicators Italy is in the second half of the ranking, showing that the previous figure was more a provocation than a contribution that should be considered seriously.

When we deal with effectiveness and efficiency together, we have to be very cautious in the validation and analysis of rankings.

Rank	Spending in % compared with GDP Total		Primary school		Spending per student	
	UE27	5,04	UE27	1,15	UE27	5650
	UE15	4,87	UE15	1,05	UE15	--
1	DK	8,28	SI	2,67	LU	--
2	SE	6,97	LU	2,06	AT	8293
3	CY	6,92	DK	1,93	DK	8244
4	MT	6,82	CY	1,89	SE	7204
5	FI	6,31	SE	1,83	UK	7084
6	BE	5,95	PL	1,69	NL	6703
7	SI	5,83	PT	1,65	CY	6684
8	FR	5,65	IE	1,6	DE	6503
9	PL	5,47	UK	1,44	BE	6501
10	UK	5,45	MT	1,4	FR	6364
11	HU	5,45	BE	1,4	FI	6225
12	AT	5,44	NL	1,37	SI	6056
13	PT	5,4	FI	1,31	IE	6012
14	NL	5,19	RO	1,26	IT	5908
15	LV	5,06	EE	1,23	MT	5882
16	LT	4,95	FR	1,14	ES	5718
17	EE	4,87	GR	1,13	PT	4703
18	IE	4,77	HU	1,09	GR	4606
19	DE	4,53	IT	1,09	HU	3842
20	BG	4,51	ES	1,09	CZ	3809
21	IT	4,43	AT	1,03	PL	3051
22	CZ	4,25	BG	0,92	EE	2868
23	ES	4,23	LV	0,79	LV	2746
24	GR	3,98	LT	0,74	SK	2699
25	SK	3,85	SK	0,66	LT	2475
26	(1) LU	3,81	DE	0,65	BG	1993
27	RO	3,48	CZ	0,62	RO	1454
	USA	4,85	USA	1,67	USA	10661
	JAP	3,52	JAP	1,26	JAP	7148

Figure 5.23 – Rankings based on expenditure for education for OECD countries

5.5 – Perceived qualities

In this section we address methods for subjective measurement of quality; such methods will be considered in a broad sense, including efficiency and effectiveness as specific categories of a more general and comprehensive category of quality.

First of all, looking at our experiences in the interactions with services and human beings offering them, we can come to the conclusion that some features of services and human beings that cannot be expressed with a quantitative measure based on a real life phenomenon. Think e.g. to the politeness of front office employees; how can we measure politeness with other means than our perception and subjective evaluation? Notice that in any case such subjective evaluation may change from user to user, and may also depend on his/her culture and habits.

Exercise 5.3 - Look again at the requirements of the Railway transport company and remember your experience of interaction with the personnel. Define at least five quality dimensions that is very difficult or impossible to evaluate through a measure of a real life phenomenon, and that can be evaluated through the user perception.

5.6 Measurement of perceived quality

We said that while quantitative measurement of quality is based on the measurement of physical phenomena, subjective quality is measured through user perception. The assessment of perception can be done through several possible procedures, among them:

- Unattended questionnaires - a questionnaire is sent to the users and there is no human support to them.
- Attended questionnaires - the questionnaire is filled by users with the support of a person
- Telephone interviews
- Focus group, a small group of remunerated people who, under the direction of an expert, answer questions and compare their perceptions during a meeting of a few hours.

In the following, we will focus on questionnaires. The questions in the questionnaire should correspond to the different modalities that users may adopt to evaluate subjective quality referring to dimensions, e.g. politeness. So in order to prepare the questionnaire we have to identify first the quality dimensions to be evaluated, then the modalities that users adopt in the evaluation of the dimension.

The perceived quality measure is the perception of the current level of quality for the dimension considered, achieved at the present moment, expressed in a scale of n values, that can be numeric (e.g. from 1 to 10), or on a scale of the type:

- Very Low
- Quite Low
- Quite High
- Very High

<u>Example of evaluation of services provided by an help desk</u>

Let us consider an help desk, that in this case is focused on personnel that assists users and interacts with them going to their offices. One possible quality dimension may be Reaction time of the help desk personnel. We can express the question this way:

Think to your interaction with the help desk personnel when you need help; what is your perception of the reaction time, namely the time between you request of intervention and the time of arrival in your office? The answer must be indicated on a scale of values that goes from [1] (very very low) to [7] (very very high) where [4] corresponds to an average grade.

A possible answer can be: my perception is [1] [**x**] [3] [4] [5] [6] [7] [Don't know]

In the previous example we have adopted a single level of evaluation, that is usually called the perceived quality. Depending on the quality to be evaluated, it may happen that the different users assign different meanings to the posed question, so that we may get an array of evaluations such as the one shown in Figure 5.24.

User	Perc. Grade
U1	3
U2	6
U3	4
U4	7
U5	2
U6	7
Average	4.83
Variance	3.81

Figure 5.24 – An array of perceived qualities

You see that evaluations are dispersed along the average value, as put in evidence by the high variance. When the variance is high, there is the suspect that the question was not clearly formulated and understood. An example of question on reaction time that has a high risk of dispersion is

Is the help desk very responsive when you call them?

In this case it is less clear what does it mean *responsive*, and the proposition *very* may lead to biased perceptions. Because of the above drawbacks, new references and related questions have been introduced in questionnaires, that reflect:

- the maximum realistically possible level, a reference that takes into account the "here and now", i.e. the resources associated to the service being evaluated. This is usually called the expected level of quality.
- similarly, the minimum threshold of acceptability, taking again into account the “here and now”. This is usually called the minimum level of quality.

Why taking into account the «here and now» in the above new measurements? We reply to the question by means of an example. In an organization of 150 people, help desk technicians were five as long as three months ago; recently three of them resigned or retired, and due to budget restrictions it is hard to replace them (this is the “now”). The two left technicians must do (this is the “here”) the work that formerly was done by five technicians.

Therefore, we come to the conclusion that it is more reliable to ask the users

- Perceived quality - Current level for quality that the user perceives as reached, expressed in a scale of values.
- Expected quality - the maximum level that the user thinks should be reached, in its relation to the available resources, expressed in a scale of values.
- Minimum quality, the minimum acceptable level that the user considers should be reached, in its relation to the available resources, expressed in a scale of values.

An example of evaluations on perceived, expected, minimum quality is shown in Figure 5.25, where we see that the variance on perceived quality is significantly decreased.

User	Min Grade	Perc Grade	Max grade
U1	2	4	6
U2	3	6	7
U3	2	5	7
U4	3	7	7
U5	2	4	6
U6	4	7	7
Average	2,67	5,50	6,67
Variance	0,56	1,58	0,22

Figure 5.25 – An array of evaluations of minimum, perceived, and maximum expected quality

5.7 – Weighted quality

In previous approaches and examples, we have seen that users may perceive differently the level of quality dimensions, dependently on their culture or other personal or social environment issues. This is true also for the relative level of importance they attribute to quality dimensions. Therefore, we have to adjoin to our apparatus of methods the possibility for the uses to assign different level of importance to different quality dimensions. This can be done with a questionnaire in which we have added a new column (see Figure 5.26) in which the user can distribute weights expressing the attributed importance, where the sum of weights is fixed to a given total, e.g. 100 points.

Quality dimension	**Weighted relevance (total 100)**
Skill of personnel	
Experience of personnel	
Politeness of personnel	
Empathy of personnel	
Accuracy of information provided to solve the problem	
Clarity of information provided to solve the problem	
Level of understanding of the user problem	
Reliability of personnel	
Avaliability of personnel when needed	

Figure 5.26 – Quality dimensions to which the users may attribute different levels of importance.

Home work – Blood tests are an experience quite diffused in our communities. Therefore, you may collect a group of relatives and a group of friends, to whom you may ask to fill the questionnaire in Figure 5.27.

Feature	Relevance (10 points total)	Perceived quality Scale 1-10	Minimum quality Scale 1-10	Expected quality Scale 1-10
Waiting time for the test				
Waiting time the day of the test				
Expertise of personnel for blood sample extraction				

Figure 5.27 – Questionnaire for level of importance of the different aspects of a blood test

5.8 Benchmarking

When we perform an assessment, the result of the assessment, either through quantitative measures or through perceived measures, is an *absolute* value, so does not give any information on the *relative* position of the process/service among similar ones. When we feel fever, we measure the body temperature and if the thermometer marks a temperature of 37, 5 Celsius degrees, we compare such temperature with the usual threshold for fever, namely 37 Celsius degree and we come to the conclusion that we have a moderate fever.

To compare with other references the value obtained in the assessment with other references, we need to perform a benchmarking. Benchmarking, compared to the assessment, evaluates the quality of the process/service in comparison with other similar processes/services.

There are at least three types of benchmarking:

- Internal - different similar processes within the same organization are compared; e.g. the amount of paper that is printed by different divisions in the organization.
- Competitive - processes are compared with other homogeneous competing organizations; e.g. the level of quality perceived by users in the seat reservation of a train by competing companies (e.g. in Italy Trenitalia and Italo-Nuovo Trasporto Viaggiatori).
- Temporal - the same processes of the organization over time are compared (e.g. how the quality of the seat reservation is perceived over the years.

We show other examples of the different types of benchmarking. An example of internal benchmarking in a University or else in a department is the quality of research publications, which can be measured with several indicators; here we mention:

- the number of citations, or (to avoid the proliferation of self-citations)
- the number of citations minus the number of self-citations.

An example of competitive benchmarking may be made among hospitals that provide care services, or else among public administrations that manage the budget on health services, the regions in Italy. We have seen general indicators in Section 5.2.2. Other indicators that are usually mentioned are

- the waiting time for booking a specialist medical exam;
- the mix of waiting times for a set of specialized medical exams (blood tests, heart tests, etc.)
- the cost of a syringe
- the cost mix of a set of products purchased by hospitals in the region.

A famous example of a “binary” competitive benchmarking is the Ford vs Mazda case. Ford and Mazda produce and sell cars and other vehicles. Ford had a division of 450 employees for order management. The Ford management sent a commission to visit Mazda. The Ford commission discovered that Mazda, similar to Ford in complexity, had an office of ten people to manage the same process!

Once come back from the visit, at Ford discovered that one of the reasons of the inefficiency was based on the management of partial orders (namely, orders that result from arrival of goods not coherent with the order); at Ford such orders were managed by creating a new residual order and starting a new process of order management. The result was that that to link the different orders related to the original one requested a lot of human effort. If the goods arriving at Mazda from the suppliers were not coherent with the order, the office sent back the whole package, and the cost was charger to the supplier, so the effort was much less. At Ford the process was made similar to the one in Mazda and people involved decreased to 125.

An example of temporal benchmarking is the functioning of an office that provides passports to requesting citizens. We can for example monitor over time the number of passports produced in a police station in a year, or else passports produced on the average per unit of staff, see Figure 5.28.

Year	# of back office employees	# Passports	# Passports per employee
2015	10	1.000	100
2016	11	1.430	130
2017	12	1.320	120

Figure 5.28 – Temporal benchmarking for passport production in the years

Home work on competitive and temporal benchmarking - Look at the document that you may find on Google on «Transports Metropolitans de Barcelona» referring to public transportation of Barcelona, and find at least five quantitative and five perceived/subjective quality indicators, evaluated for at least three years (temporal benchmarking).

Access now the London site of public transportations http://tfl.gov.uk/corporate/publications-and-reports and look for reports that provide and analyze identical or similar indicators to the case of Barcelona. In case iterate the search. Find a third European or American city providing indicators on public transportations of a city, and write a report in which a competitive benchmarking is performed, discussing also possible differences in the indicators and their values.

We conclude here the chapter.

Appendix 5.1 - Solutions to exercises

Solution to Exercise 5.1

1. Patients are hospitalized in a hospital → **Input**
2. then they receive treatment → **Activity**
3. and when the treatment is considered finished → **Event**
4. they are discharged from the hospital → **Output**

Solution to exercise 5.2

Inputs, outputs, and resources are as follows:

- Input: reservation requests Rij managed by OPi
- Output: confirmations
- Resource: Time required by OPi

Efficiency referred to operator OPi is =
$\sum_j$ Δ (time at confirmation to Rij, time at request Rij) / # reservation requests = average time for operator OPi to manage the request

Here again the resources are in the numerator, and the outputs in the denominator, therefore the efficiency increases as the value of the ratio decreases.

Possible solution to Exercise 5.3

A list of quality dimensions that is difficult or impossible to evaluate with objective measures is the following

- Empathy of personnel
- Politeness of personnel
- Clarity of information provided to solve the problem
- Level of understanding of the user problem
- Reliability of personnel
- Skill of personnel

Chapter 6 – Process-to-be design

6.1 Introduction

The goal of the phase of Process-to-be design (see Figure 6.1) is to transform the processes assessed in the previous phase, in such a way to achieve in the new processes new target objectives in terms of efficiency and effectiveness.

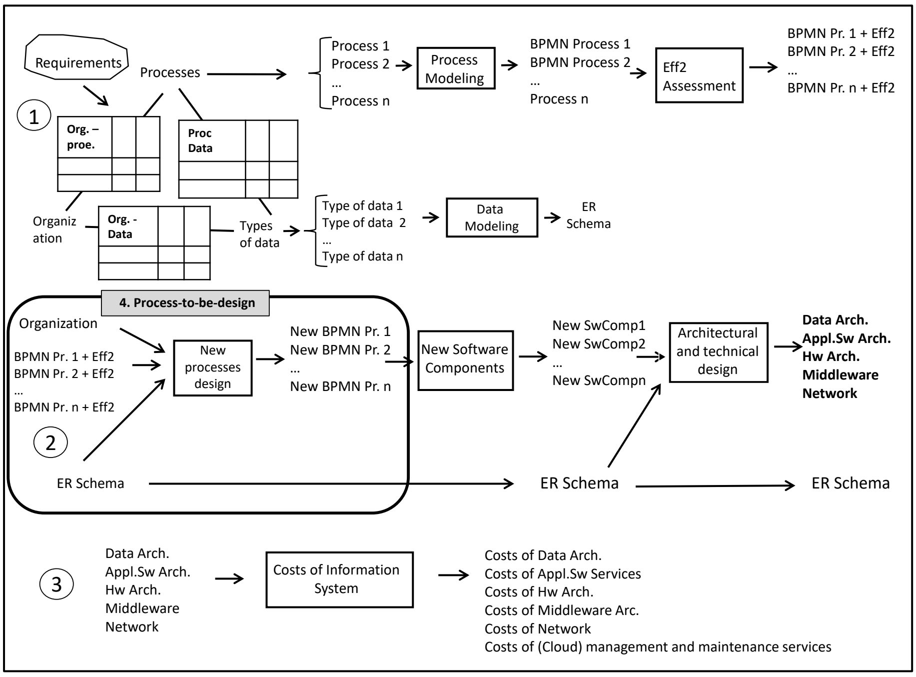


Figure 6.1 - The phase of process-to-be design in the IS life cycle

The process to be design is performed in two steps:

1. Fix quality targets of the new processes/information system
2. New Process-to-be design that achieved such targets.

Also in this chapter, we prefer to discuss the two case studies introduced in Chapter 1, instead of bothering the reader with abstract and blurred material. We first consider in section 6.1 the railway company case study, investigating a set of qualities that can be seen as specializations of effectiveness. For them we identify in the requirements shown in Appendix 2 of Chapter 1 actual values of qualities, and then we discuss possible target values and technologies that may be adopted to achieve them. Then in Section 6.2 we focus on the exam registration case study and investigate two different strategies to achieve improved quality values for efficiency and effectiveness.

6.1 Fix quality targets of the new processes/information system – the railway company case study

In the table shown in Figure 6.2 we consider several quality dimensions that can be seen as specific cases of effectiveness, together with actual levels mentioned in the requirements of Appendix 1.1, and possible future levels to be achieved through process reorganization and better usage of ICT technologies. We now comment each of them.

Quaity Dimension	Actual value	Target value
Response time to complaints	One month	One week
Cultural accessibility	One natural language	Four natural languages
Physical accessibility	4 channels + 40% selfs.	6 channels + 80% selfs.
Transparency	Very low	High
Usability	Low	High

Figure 6.2 – Actual and target quality levels in the railway company example

1. *Response time to complaints* is a typical example of effectiveness, since clients aim to have a fast response to their claims. In Figure 6.2 the objective to move from one month to one week is stated. In order to speed up the process, it could be worthwhile to build a digital document base for the management of document flows that report user complaints, and adopt a workflow technology that allows to make the offices involved in the process to interact more effectively.

2. *Cultural accessibility* concerns the diffusion among users of skills and capabilities required for an autonomous access to and usage of services. A number of metrics can be used to measure this dimension, such as:

1. the percentage of words provided in the description of the service whose meanings can be understood by an average user;
2. the number of natural languages in which the service is provided;
3. the perceived ease of use of the access channels.

Focusing on the second indicator, we may improve cultural accessibility organizing the Web site from adopting one language to four languages, chosen based on the estimated number of people in the world who speak them; the esteems may be obtained looking at statistics on the Web. To achieve such goal we may adopt free automatic translators such as e.g. Google translator with subsequent human intervention or else translation applications that are charged but have a better quality.

3. *Physical accessibility* measures the capability of the user to access the service offered taking into account his/her physical status and functions. Particularly important in the social domain is physical accessibility for disabled persons, the interested reader is addressed to [Viscusi et al 2010]. Here we measure physical accessibility in terms of the number of channels used to access the services. In the present system, four channels are available, that correspond to front offices of the company accessible at stations, agencies, the web site, and self-service terminals, that are present in the 40% of stations. In the future we may add two new channels, namely e-mails and sms, and extend self-service terminals to the 80% of stations.

4. As to *interface usability*, namely the usability of the different interfaces offered by channels, we assume that it has been measured using a questionnaire, and that users having at disposal five grades, a. very low, b. low, c. acceptable, d. good, e. very good, provided *low* as average level. It is clear that a new platform should be used for web site design that allows to implement a new graphical interface, and to increase to good the average level of responses.

4. Transparency of the organization, that concerns the volume of information that the company provides to users on the characteristics and evolution in time of service provision and on what they could expect or claim using the service. We assume that an assessment performed with a questionnaire on user complaints has evaluated transparency as very low. To increase transparency to *good,* we may assume to give users the trace of the service delivery process in at least three step (see Figure 6.3), for example:

Phase 1 - Request acquisition,
Phase 2 - Taking charge by the back office structure,
Phase 3 - In the case of claims involving refunds, preparation of the payment mandate.

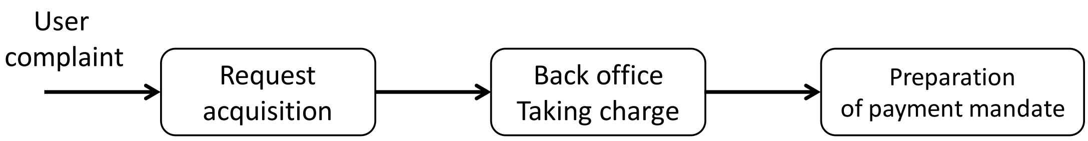


Figure 6.3 – The three enabling phases of transparency

6.2. Process-to-be design in the exam registration case study

In Figure 6.4 we reproduce the costs of human resources in the process of exam characterized by scarce automation discussed in Chapter 4. The corresponding cost per exam is a possible measure of economic efficiency, while the percentage of errors and inconsistencies that contribute to increase the costs and the time efficiency can be seen as a measure of effectiveness.

Human resources / efficiency	Hourly cost	Hours for 100 exams	# exams	Total hours	Total Cost in €
Teacher	40 €	1	200.000	2.200	88.000
Department secretary	15 €	0,2	200.000	400	6.000
Central student secretary	15 €	1	200.000	2.000	30.000

Figure 6.4 – Costs of the process-as-is

In this case we have to conceive first the new objectives, that we can express tentatively in terms of reduction in percentage of economic resources needed to perform the process or in terms of the reduction in percentage or errors.

In Figure 6.5 we show for the three types of human resources possible reduction objectives in terms of effort and corresponding economic resources. In fixing the objectives, we have in mind that the most critical resource are teachers, and furthermore that we should conceive a new process that excludes the involvement of department secretaries in the process.

Quality	Organizational Role	Effort in hours	Actual level	Reduction
Human resources/economic efficiency	Teachers	2.200	88.000	80%
Human resources/economic efficiency	Department secretaries	400	6.000	100%
Human resources/economic efficiency	Central secretaries	2.000	30.000	50%
Effectiveness: errors in the registry	Teachers		5% + 5%	100 %

Figure 6.5 – Target reduction objectives in process-to-be planning

As a starting point in the analysis, we may reconsider the level of automation of the activities in the actual process, see Figure 6.6; only the last activity that updates the exam database is partially automated; the

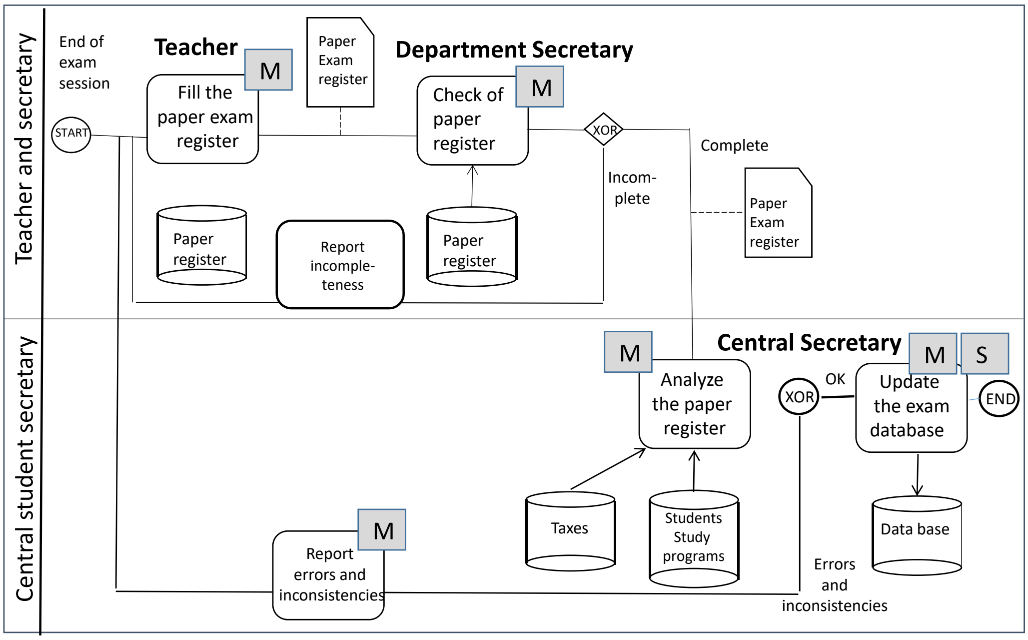


Figure 6.6 – Manual vs software activities in the process-as-is

automation is partial, since, as shown in Figure 6.7 it is composed of two sub activities, that correspond to insert manually data in a form, and exploit the database management system to permanently insert data in the database.

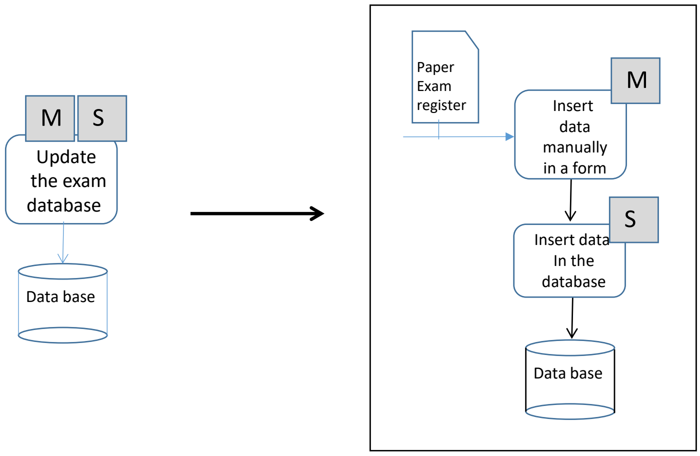


Figure 6.7 – Manual and software sub phases in the update the exam database

We move now to the second phase of process-to-be-design, in which we reply to the following question: may we change the process exploiting ICT technologies, achieving in such a way the new objectives?

Notice that in this phase besides ICT technologies we may intervene, on each of the layers of the information and organizational system, namely:

- Organization
- Process structure
- Human resources
- Information resource
- ICT technologies
- I/O devices
- Application software
- System software
- Centralized Harware
- Distributed hardware
- Network

Two different strategies may be adopted in this decision:

1. Limited and continuous improvement, that involves marginal interventions in some steps of the process (but, often,…. better little than nothing).
2. Significant improvement (or re-engineering) of the business process, that involves interventions on the structure of the process, generally favored by the adoption of new ICT technologies, which enable substantial improvements to its quality.

An example of limited improvement is the automation of the manual activity in Figure 6.7, to be performed using an optical reader, while for significant improvement, we may conceive the adoption of a. a digital register and b. digital signature. Such technologies allow teachers to:

1. record exams on pre-filled lists of students displayed on the screen;
2. send electronically signed registers to the central office by means of a secure channel.

Let us now discuss separately the two cases. For each one of the two cases we have to discuss:

- New requirements

- New ICT technologies
- New process
- New efficiency assessment.

6.3 Adoption of the optical reader

The new requirements are shown in the following box, it is straightforward to come to the conclusion the improvement of the process will be very limited.

When a student passes an exam, the teacher records biographical data of the student, course data and the grade in a paper registry. When the exam session ends, the teacher brings the register to the Department's student office. The student secretary makes a first check of the completeness of the registers, and, when he/she finds some void fields, reports the incompleteness to the teacher. When the department students' secretary has collected all the registers of the exam session, sends them to the central University students office. The central University students office re-analyzes the registers for errors and inconsistencies, and checks that students: • have in their study program the course corresponding to the exam, and • are in good standing with University taxes. In case of errors or inconsistencies in the register, the office sends the register back to the teacher. When the records are complete, the grades of exams are updated **using an optical reader** in the student's exam database, along with the course passed and the grade; in case of Erasmus students, the country of the student is represented.

The technologies for optical readers are made of:
1. the optical reader device;
2. off the shelf software for the optical reader functioning;
3. off the shelf software for Optical Character Recognition.

In Figure 6.8 we see the new process that differs from the previous one only in the last activity.

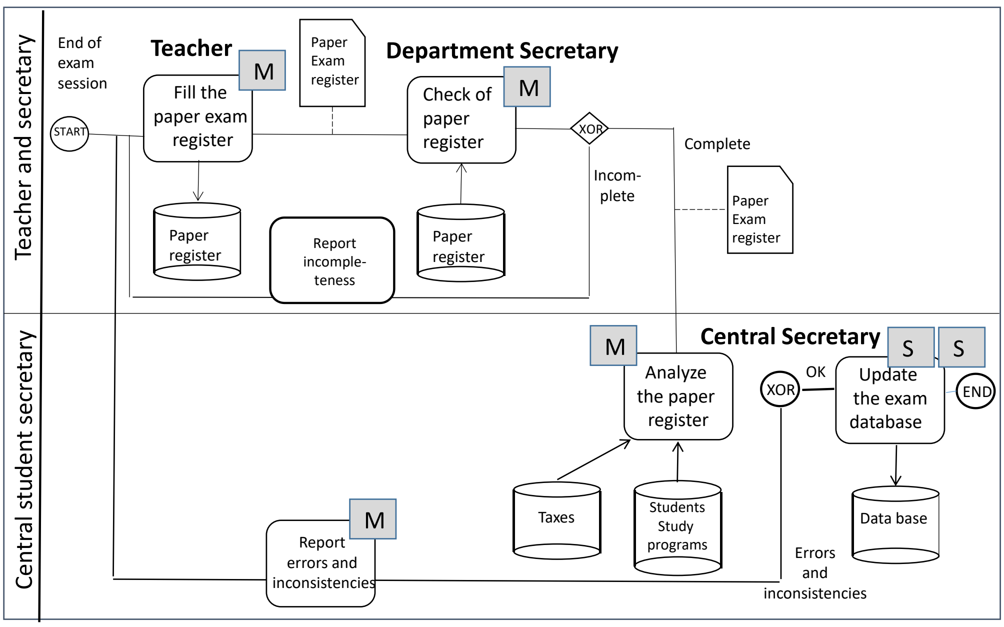


Figure 6.8 – The new process with the optical reader

Coming to new effort and new costs, the optical reader technology influences only the effort of the central secretaries in the update of the exam database. We may size the change in the effort for one hundred exams from one hour to 0,6 hours, leading to the new effort and costs shown in Figure 6.9.

Human resource / efficiency	Hourly cost	Hours for 100 exams	# exams	Total hours	Total Cost in €
Teacher	40 €	1	200.000	2.200	88.000
Department secretary	15 €	0,2	200.000	400	6.000
Central student secretary	15 €	0,6	200.000	800	18.000

Figure 6.9 – New human resources costs with the optical reader

The new total costs decrease from 124.000 to 98.000 euros, corresponding to 12.000 euros of savings, while the cost per registered exam decreases from 0,62 euros to 0.56 euros. Errors in the procedure are not influenced, so the new target levels of the process-to-be with optical reader are those shown in Figure 6.10.

Quality	Role	Actual level	Reduction
Human resources/economic efficiency	Teachers	80.000	0%
Human resources/economic efficiency	Department secretaries	6.000	0%
Human resources/economic efficiency	Central secretaries	30.000	40%
Effectiveness: errors in the registry	Teachers	5% + 5% = 10%	0%
Economic efficiency	All	104.000	11%

Fig 6.10 – New target levels of the process-to-be with optical reader

We have now to add the cost resulting from the adoption of the new technology. We assume to use the new technologies for a period of six years. The new costs can be evaluated:

- Cost of an optical reader = 200 euros
- Cost of reader management software = 100 euros
- Optical readers needed = 20
- Total cost = 6.000
- Years of use = 6

• Cost per year (amortization) = 1.000 → negligible w.r.t. the costs of human resources.

6.3 Complete reorganization of the process using automation

Also in this case we have to define:
1. New requirements
2. New ICT technologies
3. New process
4. New efficiency assessment

New technologies to be adopted deeply influence new requirements and the resulting process, see Figure 6.11. When a significant improvement is demanded, we can first focus on technologies, and then conceive how requirements change correspondingly.

Figure 6.11 – Process transformation with enhanced use of technologies

In substance, we may focus on the following technologies and usages:
- Adoption of the digital register
- Reuse of existing databases
- Design and production of new software applications
- Adoption of the digital signature to transmit securely data from the teacher to the central secretary office.

With reference to existing databases we may perform an investigation that leads us to identify the three databases in Figure 6.10, namely
- the database of the paid taxes,
- the study programs, and
- the student exam register.

Corresponding skeleton schemas are shown in the figure, represented with the simplified notation for the Entity Relationship model introduced in Chapter 3.

Database	Schema
Taxes	Student – *Paid* – Taxes
Study program	Student – *Study program* – Course
Exam register	Student - *Exam* - Course

Figure 6.12 – Databases that can be exploited in the process-to-be

Now we proceed to the production of new requirements, having in mind that we have to delete all manual activities and we have to substitute the registry with a digital one. In order to automatize the whole process, we have to develop software applications that

1. perform at the beginning of the process all controls on taxes and the study program that in the process-as-is are performed at the end of the process.
2. provide the teacher all data on students and courses that are prefilled in the exam register at the moment in which the student enrolls to the exam.
3. let the teacher fill only the grade obtained by the student in the exam.

We may progressively produce requirements and related BPMN as in the oil-stain strategy, see the following box and the BPMN process in Figure 6.13.

A new sw application should be developed for the university student office which, starting from

- the study programs,
- the database on taxes paid, and
- the students enrolled in exams,

enables the central secretary to send in advance to teachers the list of registered students in the exam session, with serial number, name, surname. Controls on student taxes and programs are made apriori.

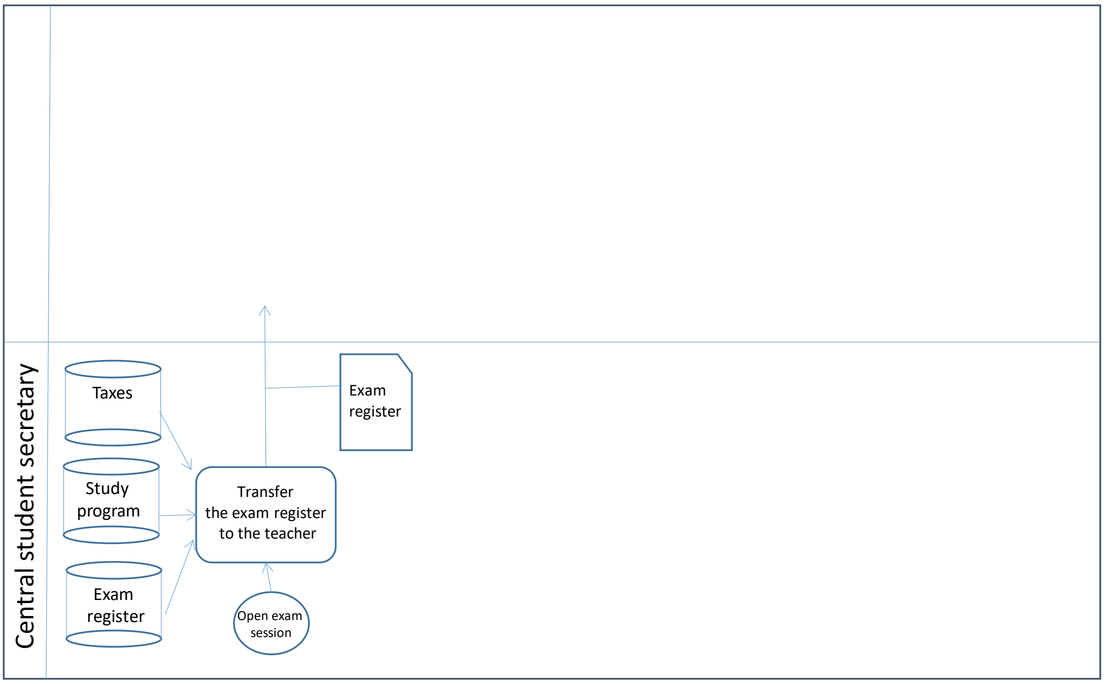


Figure 6.13 – First part of the process-to-be

We may now focus on activities of the teacher, which lead to the new requirements in the box and the related BPMN in Figure 6.14.

At the closing of an exam session, each teacher compiles the digital register related to his/her exams, in which he/she finds pre-registered information on the students enrolled in the session, and in which he/she must only enter the grade. At this point, the teacher sends the digital report to the university secretariat using a digital signature.

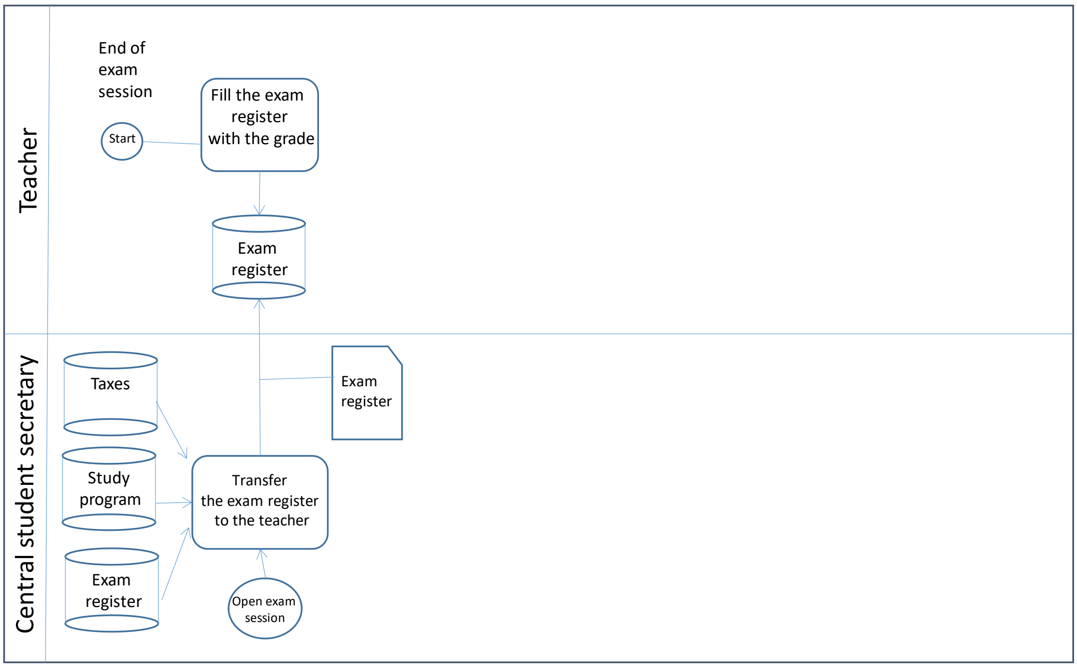


Figure 6.14 – The process-to-be with added the teacher activities.

The final part of requirements and the final BPMN process are shown in the table and Figure 6.15.

The central secretary, after formal checks that do not require feedback to the teacher, updates the student careers database.

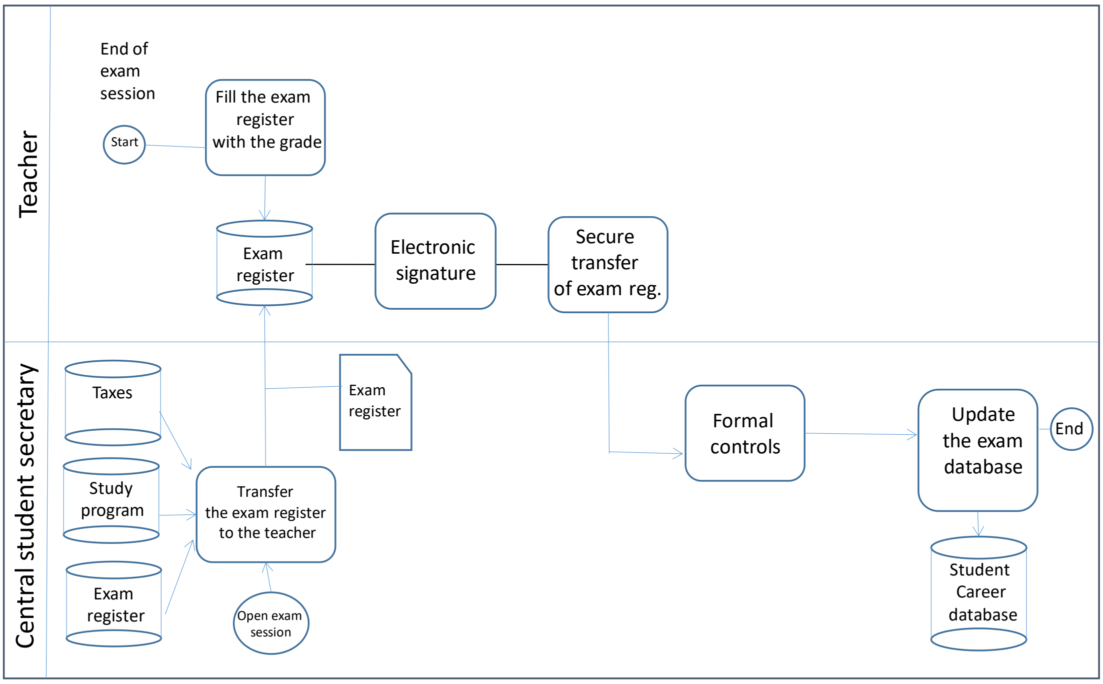


Figure 6.15 –Final process-to-be

Notice that now we have two start events, which are in charge respectively of the central secretary and the teacher. We have also made more precise the name of the database where to insert exam grades changing into "Student career database".

We have fully automated with software applications all the activities in the process, as it shown in Figure 6.16. Human resources involved in the process are now two, the teacher and the central secretary office; and we have eliminated from the upper lane the department student secretary, no longer involved in the activities, and whose effort can be focused on other tasks and services, such as the interaction with the students.

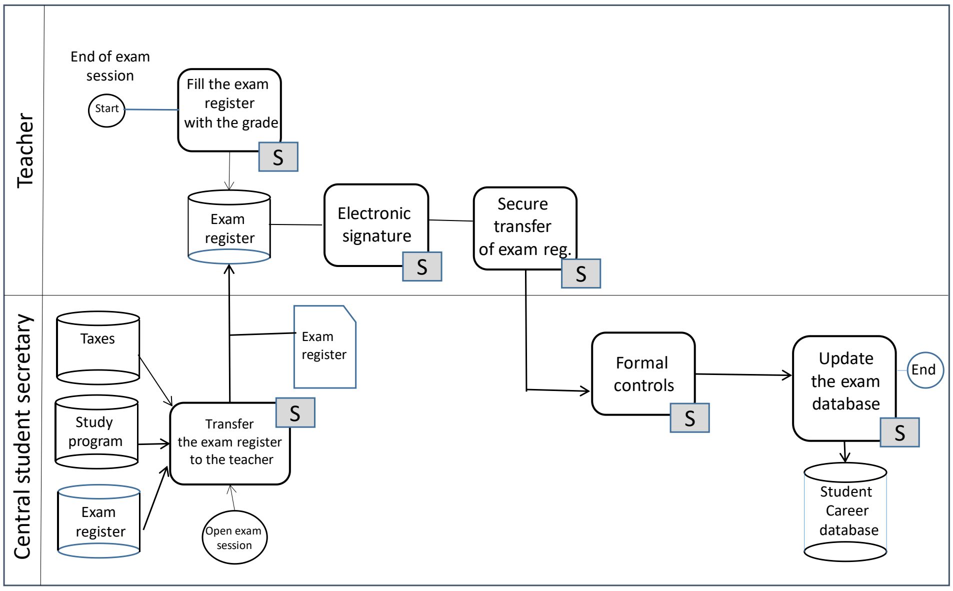
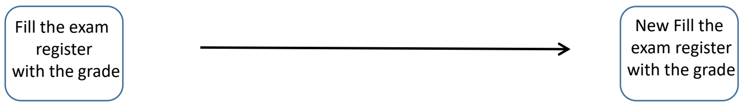


Figure 6.16 – All activities in the process-to-be are software based

Finally, while previously ten different values had to be filled manually by the teacher, now only the grade has to be filled as shown in Figure 6.17.



Student code	34276	Name	Carlo
Surname	Rossi	Date of Birth	7-5-1995
Course code	038	Course name	Data Bases
Prof Code	0443	Prof Surname	Batini
Date	3-2-2019	Grade	30

Student code	34276	Name	Carlo
Surname	Rossi	Date of Birth	7-5-1995
Course code	038	Course name	Data Bases
Prof Code	0443	Prof Surname	Batini
Date	3-2-2019	Grade	30

Legenda - To be filled / Prefilled

Figure 6.17 – All activities in the process-to-be are software based

The new costs related to human resources are shown in Figure 6.18. The cost of the teacher is 10% of the previous one, since he/she has to fill only one value on ten. The department secretary is no longer involved in the process; the central student secretary has in charge final checks that request 20% of the previous effort. Finally we have no longer costs related to incompleteness and inconsistencies.

Human resource / efficiency	Hourly cost	Hours for 100 exams	# exams	Total hours	Total Cost in €	Reduction
Teacher	40 €	1 → 0,1	200.000	2.000	80.000 → 8.000	90%
Department secretary	15 €	0,2 → 0	200.000	400	6.000 → 0	100%
Central student secretary	15 €	1 → 0,2	200.000	2.000	30.000 → 6.000	80 %

Figure 6.18 – New costs of human resources

Concerning the cost of technologies we distinguish between software and digital signature. As to software applications, we may assume the following costs:

- Software production: 50.000 euros
- Software maintenance: 5.000 for each year (10% of production)

For each year (assuming an amortization of 10 years)

- Cost per year = 5.000 + 5.000 = 10.000 euros

Concerning digital signature, we may assume that one single digital signature + software costs 10 euros a year. How many digital signatures are needed? We have:

- (approximately) 670 Teachers
- 30 Central secretaries

corresponding to a total of 700 digital signatures, leading to a cost for each year of 7.000 euros.

The total cost of technologies (application software + digital signature) for each year is 17.000 euros. New yearly human resources and technological costs in the process-to-be are shown in Figure 6.19.

Human resource / efficiency	Hourly cost	Hours for 100 exams	# exams	Total hours	Reduction	Total Cost in €
Teacher	40 €	1 → 0,1	200.000	2.000	90%	88.000 → 8.000
Department secretary	15 €	0,2 → 0	200.000	400	100%	6.000 → 0
Central student secretary	15 €	1 → 0,2	200.000	2.000	80 %	30.000 → 6.000
Software + Digital signature	-	-	-	-	-	17.000

Figure 6.19 – New yearly human resources and technological costs in the process-to-be

The overall yearly cost of human resources + technologies corresponds to 31.000 euros, leading to a cost per registered exam of 0,15 euros.

Chapter 7 – New software components

7.1 Introduction

The phase of New software components (see Figure 7.1) has the goal to identify, starting from processes designed in the previous phase, which are new software applications that have to be produced to achieve the efficiency and effectiveness objectives previously fixed.

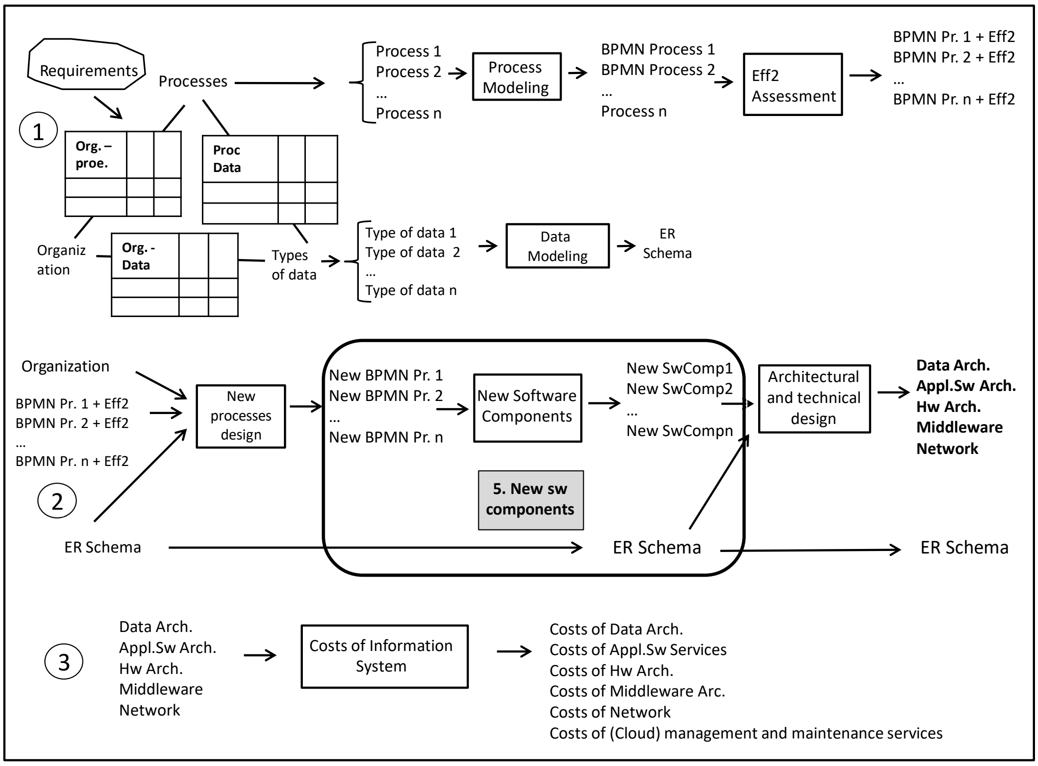


Figure 7.1 – The phase of New software components

So, in this step new software projects have to be identified, based on the new requirements. Another goal of the phase (see Figure 7.2) is the production

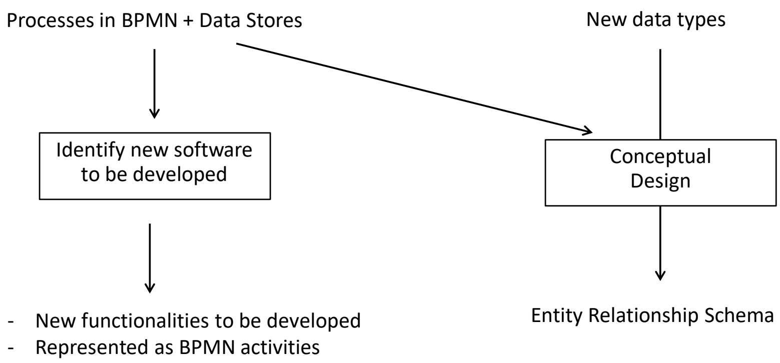


Figure 7.2 – Steps and outputs of the phase

of the Entity Relationship schema that represents the union of data stores present in the BPMN process. Notice that this aspect of our methodology has not to be taken for granted in practice, since usually software production is performed without a direct relationship with data design, that is considered as a sort of second class activity.

In figure 7.3 a methodology is shown that can be followed for this phase, while in Figure 7.4 we show the BPMN process in input to the phase in the exam registration case study.

1. Analyze activities described in BPMN
2. Identify activities candidate for a. new software development and b. software applications to be modified/enriched
3. Design the new Entity Relationship Schema
4. Identify basic software functions present in BPMN activities
5. Add sizing elements, such as frequency of execution and complexity

Figure 7.3 – A methodology for the phase

We now proceed to the discussion of most relevant steps.

7.2 Analyze activities described in BPMN and identify activities for software development

In Figure 7.4 we show the BPMN process of the exam registration case study in input to the phase.

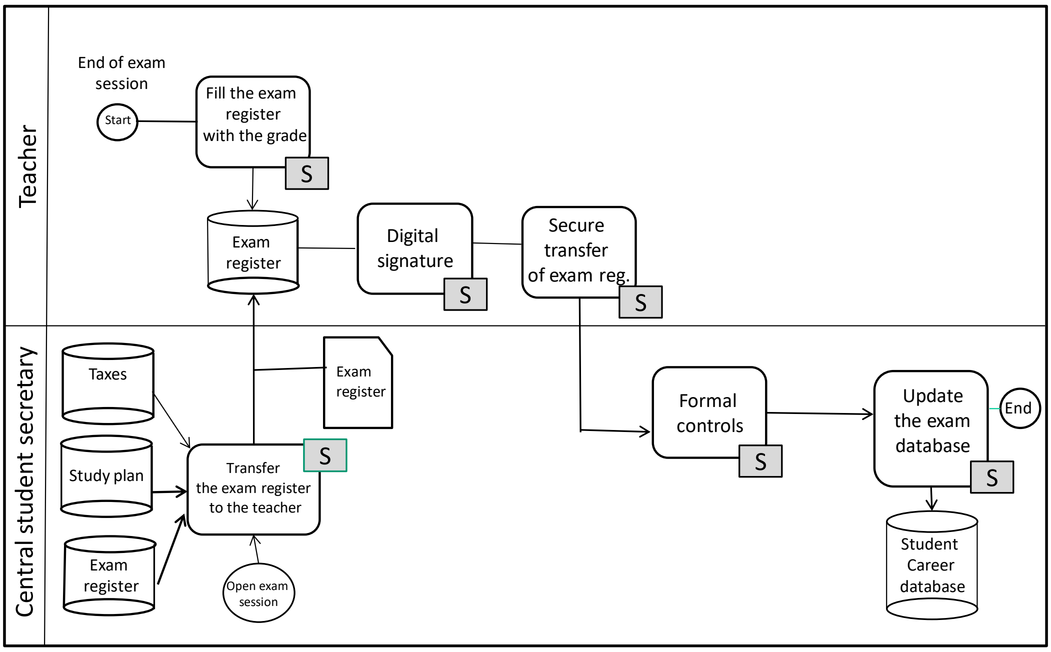


Figure 7.4 – The exam registration process

Identifying activities in the exam registration case study is straightforward, all the activities in figure 7.4 are candidates for software components, see Figure 7.5. We discuss the step for the railway company case study.

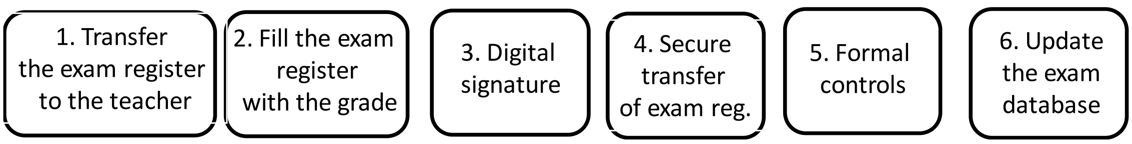


Figure 7.5 – Activities that are candidates for software components

In this case, we follow a simplified and more qualitative approach, looking at requirements and, without producing a BPMN description of processes, conceiving new services that lead to new software applications. Consider the following requirements:

1. The system does not allow making and modifying reservations through mobile phone, and allows reservation changes exclusively through the same channel used in the initial reservation. In this case new software applications and applications to be enriched with new functionalities are:
   - Application for reservations and changes to reservations by mobile phone (new)
   - Software application for channel "portability“, that allows to use and personalize the same software on different channels (to be enriched).

2. The system does not provide services for particular categories of users, such as visually impaired users or users with bicycles taken in the train; new software applications are:
   - Software voice interface application for visually impaired users
   - Software application for bicycle reservations.

3. Travelers would like to be able to book the exact place in the coaches, identified through the vision of the places already occupied, those places that are free and the direction of travel.
   - We have to modify the seats reservation application to enable above services (modify).

4. In order to establish with the users an additional secure access channel, it is necessary to open a certified e-mail channel, in which the electronic signature is adopted as a customer identification function; this is also needed to establish the legal validity of the documents sent by customers and to be able to send a receipt. This means that:
   - a certification authority must be created (this is an intervention at the organizational level)
   - mail systems must be equipped with software for identifying the user, signing the documents and sending a receipt, etc. (this is an intervention at the technological level).

7.3. Design the new Entity Relationship Schema

We have now to design the Entity Relationship schema corresponding to the set of data stores that are present in the BPMN process. In order to design the schema, we have to choose one of the design strategies discussed in Chapter 3.

In this case, we start from a very raw description of data, which are described uniquely by names of data stores. So we decide first to enrich the requirements providing a description in natural language for all of them; then we may proceed in this case using the bottom up strategy we have discussed in Chapter 3. In this case we apply the strategy by first building the Entity Relationship schemas of data stores and then we integrate the local schemas producing the global schema corresponding to the whole BPMN process, In figure 7.6 we represent data stores and their natural language description (first step). In Figure 7.7 we produce the local Entity Relationship schemas; notice that we have adopted a simplified diagrammatic representation and we did not represent attributes of entities.

The integration of the local schemas is made of several steps (for a comprehensive introduction to methodologies for integration the interested reader may see [Batini et al. 1984]). We have to find;

- possible homonymies (two concepts with the same name representing different classes of observables of the real world),
- synonymies (two concept with different names representing the same class of observables in the real world), and

- heterogeneities in the usage of Entity Relationship constructs, e.g. an entity in a schema is represented as attribute in a second schema.

Data store	Enriched set of requirements
Exam register	Relates courses, (code and name) with students (code and surname) that are enrolled in the current session in the course exam, with grade.
Taxes	Relates every student with code and surname, with the amount of taxes paid
Study program	Relates every student with code, name and surname, with the set of courses (with code, name, and year) in the study program
Student Career database	Relates every student with code, name, surname, and date of birth, with exams passed, the corresponding course, with code, name and grade in the exam. Furthermore, in case the student is an Erasmus student, the country is represented.

Figure 7.6 – Data stores and their natural language description

Exam register	Relates courses, (code and name) with students (code and surname) that are enrolled in the current session in the course exam, with grade.	**Student** — *Exam* — **Course** Grade
Taxes	Relates every student with code and surname, with the amount of taxes paid	**Student** — *Taxes* — **Part** Paid ?
Study program	Relates every student with code, name and surname, with the set of courses (with code, name, and year) in the study program	**Student** — *Study program* — **Course**
Student Career database	Relates every student with code, name, surname, and date of birth, with exams passed, with the corresponding course, with code, name and grade in the exam. Furthermore, in case the student is an Erasmus student, represents the country.	**Student** — *Exam* — **Course** Grade Date ↑ **Erasmus st.** Country

Figure 7.7 – Data stores and corresponding Entity Relationship schemas

In schemas of Figure 7.7 we do not have such cases, so we can superimpose the four schemas, obtaining the integrated schema of Figure 7.8.

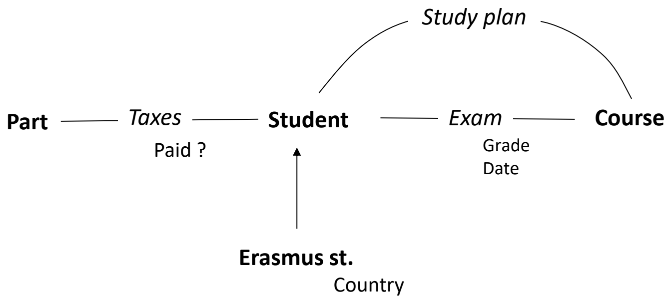


Figure 7.8 –The Entity Relationship schema resulting from the integration of local schemas

7.4. Identify basic software functionalities present in BPMN activities

In this step we must identify the specific basic software functionalities present in the BPMN activities. In Figure 7.9 we see in the left hand upper part the BPMN activities, in the right hand upper part a list of functionalities, and in the table:

- the organizational unit that has the ownership of the functionality (corresponds to the owner of the process in the organization process matrix of Chapter 2).
- Software applications, corresponding to the BPMN activities, but expressed with names more oriented to operational functionalities (the correspondence is made easier by the adoption of same numeric codes).
- For the first software application the corresponding choice of basic software functionalities, where the extract activity is performed through the "query" functionality and the "Send" activity is performed through file transfer.

BPMN activities	Basic software functionalities
1. Transfer the exam register to the teacher 2. Fill the exam register with grades 3. Electronic signature 4. Secure transfer of exam reg. 5. Formal controls 6. Update the exam database	• File transfer • Query • Update a table • Digital signature software • Join between tables • Off the shelf software

Ownership/ Responsibility	Software application	Main software functionality
Central secretary	1. Extract and send the digital registry	Query + File transfer
Teacher	2. Update the prefilled registry	??
Teacher	5. Digital signature	??
Teacher	3. Send the registry securely	??
Central secretary	4. Formal controls	??
Central secretary	6. Update the «student course of studies»	??

Figure 7.9 – BPMN activities to be expressed in terms of basic software functionalities

Exercise 7.1 - Complete the matrix for all other software applications identified by a question mark.

Solution in the Appendix.

7.5 Add sizing elements on the frequency of execution

To be able in the next phase of architectural design to decide between alternative architectures, we need to collect sizing elements on the application load. In order to acquire data on the frequency of software applications, we have to make some hypothesis on the activities are performed. Teachers proceed independently the one from the other, so to size their activities we have to consider the number of courses and the sessions in the year. The central office could proceed managing together all registries coming from the different teachers of the same department (this modality is called batch modality). We assume that there are 20 departments, see Figure 7.10.

Software application	**Number of runs in an year**
Extract and send the digital registry	600 teachers x 3 courses on the average x 5 sessions = 9.000
Update the prefilled registry	600 teachers x 3 courses on the average x 5 sessions = 9.000
Digital signature	600 teachers x 3 courses on the average x 5 sessions = 9.000
Send securely the registry	600 teachers x 3 courses on the average x 5 sessions = 9.000
Formal controls	5 sessions x 20 departments = 100
Update the exam database	5 sessions x 20 departments = 100

Figure 7.10 – Number of runs in a year of software applications

Appendix 7.1– Solutions to exercises

The solution to exercise 7.1 is shown in Figure 7.11.

Ownership/ Responsibility	Software application	Main software functionality
Central secretariat	Extract and send the digital registry	Query + file transfer
Teacher	Update the prefilled registry	Update a table
Teacher	Digital signature	Off the shelf - digital signature sofware (buy)
Teacher	Send the registry securely	File transfer
Central secretariat	Formal controls	Join between tables owned by different organizational units
Central secretariat	Update the exam database	Update a table

Figure 7.11 – BPMN activities and corresponding basic functionalities

Chapter 8 - Architectural and Technological Design

8.1 Introduction

In the chapter we discuss the phase of Architectural and Technological design (see Figure 8.1). The phase has in input new software components that have been identified in the previous phase and the ER schema resulting from previous activities, and produces in output the technological architectures related to data, software, middleware, hardware and network, and their relationships with the organizational architecture. Furthermore, it provides (to some extent) a sizing of technological components.

With the term architecture, in general, we mean a set of components (e.g. organizational structures, data bases, servers., etc.) in terms of which a layer of the organizational system or the information system is organized. We have seen so far several examples of architectures, e.g. processes divided into support processes and primary processes are in a sense an architecture.

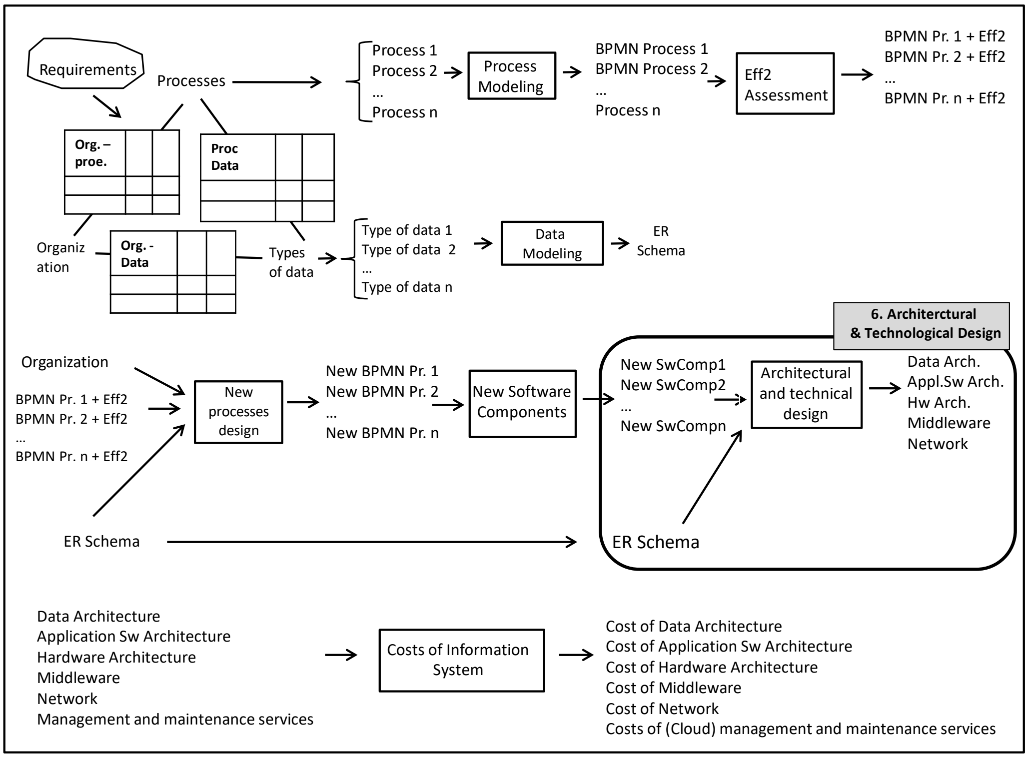


Figure 8.1 – Inputs and outputs of the Technological and Architectural design phase

The goal of the phase, as discussed in the following, is to identify design solutions in terms of

- Choice of technological architectures
- Sizing of technological components
- Choice of management services

Before addressing the three steps, in the next sections we will introduce preparatory material on technological architectures and components and services to manage them. We start the discussion with the organizational architecture, since all technological architectures will be based on it. Notice that we consider the organizational architecture as given, we address in the design phase only the technological architecture. In the chapter we will make of use the railway company case study.

8.2 Organizational architecture

In order to identify the organizational architecture we have to pinpoint in the railway company case study described in Appendix 1.2. Requirements referring to the organization. Examples of such requirements are shown in the following box.

• … There are 2,000 railway stations, divided into 50 “big stations” where the trains are used by more than 1 million user-tickets per year, and 500 small stations • … There are also about 20,000 employees at the stations, of which 3,000 are in big stations, and 2,000 in small stations. Finally, there are 10,000 employees, including traveling personnel and administrative staff. • ... self service terminals have an interface ...

Such requirements allow us to identify (see Figure 8.2) the following internal and external units referred to the organization architecture, where also the number of units is shown (e.g. there are 20 regional directions):

- Headquarters
- Regional directions
- Big stations
- Medium stations
- Small stations
- Agencies

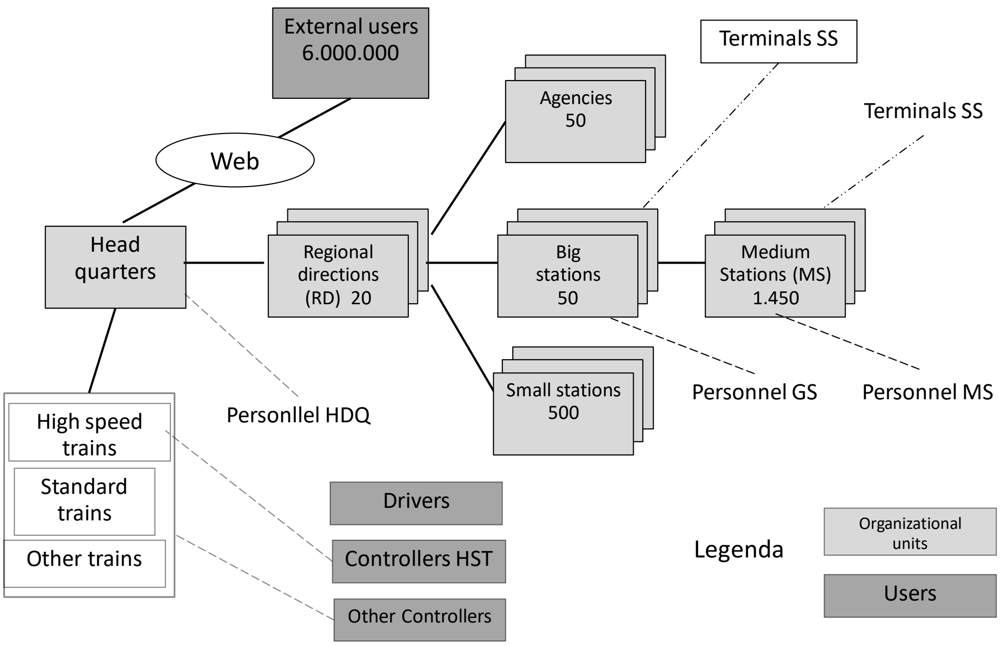


Figure 8.2 – Internal and external organizational units and Internal and external units in the railway company case study

Lines between units represent functional relationships (e.g. high-speed train are managed directly by headquarters since train routes cross different regions). Notice that we have shown the different functional relationships of medium stations with big stations and small stations with regional directions.

Furthermore, we have represented in Figure 8.2 internal and external users, identified several roles in the personnel, such as, generically, personnel in stations or controllers of High speed train and other controllers. Considering controllers, we may assume that controllers of high-speed trains are equipped with tablets that allow them to identify the seats occupied by passengers that got in the train in the different stations and ask them, and only them, the identification code assigned to their ticket. We also represent external users accessing services provided by the railway company through the Web.

Numbers are associated to all organizational structures and types of users, which represent their numerousness.

8.3 Types of technological components and services

Technological architectures are composed of two different types of "artifacts", that reproduce the distinction between goods and services we introduced in the first chapter, they are: technological components (corresponding to goods) and various types of services. A technological component is a device and an associated piece of software realized using ICT technologies, e.g. a computer, a printer, a software program, a telecommunication network. See Figure 8.3 in which we have refined the generic classification of technologies in terms of hardware, software, network and data technologies.

We will explain in more depth in the next sections the role of each box; here we provide a general introduction. In Chapter 1 we gave a generic definition of service, highlighting their intangible nature. In this chapter, we focus on services as activities pursued by human beings on a technological component to produce, maintain and manage such component.

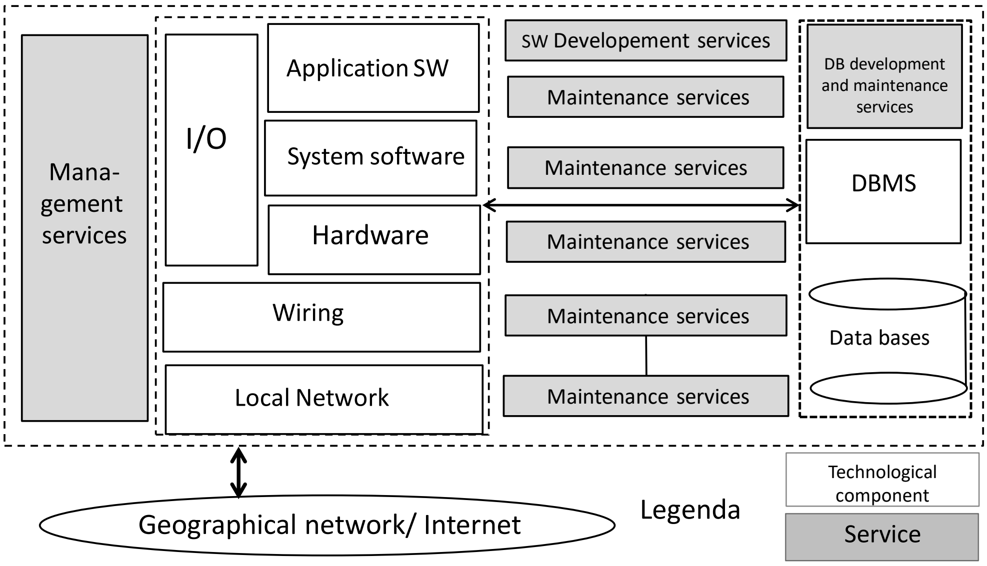


Figure 8.3 - Technological components and services in an information system

In the following sections, we will go in more depth on technological components and services depicted in Figure 8.3. Their allocation in the different units of the organizational structure is a relevant part of the phase of architectural and technological design.
We show in Figure 8.4 an example of allocation of components and services in a hypothetical organizational architecture made of three units (also called nodes in the following). You may notice that in Organizational unit 1 all components and services are present; this means that the unit has is the main node of the architecture. In Organizational unit 2 we have all technological components, and, among services, software development and maintenance services and database development services are not present, meaning, as we will see in more detail in the following, that no software production activity is performed in the node. In Organizational unit 3 we have a simple technological architecture and uniquely management and maintenance services, meaning that this unit is a minor unit in the organization. Finally, we underline the assumption that software production and other types of activities are performed inside the organization; this is the traditional choice, other typical and very frequent choices are to make a contract with a software house for software production and maintenance and other services, or else exploit cloud services. These issues will be dealt with in the next chapter. We have adopted this point of view in such a way that we can explicitly

provide a fully specified model of technological components and services, while we postpone to the next chapter the issue of the cost of such resources in a traditional setting and in a cloud setting.

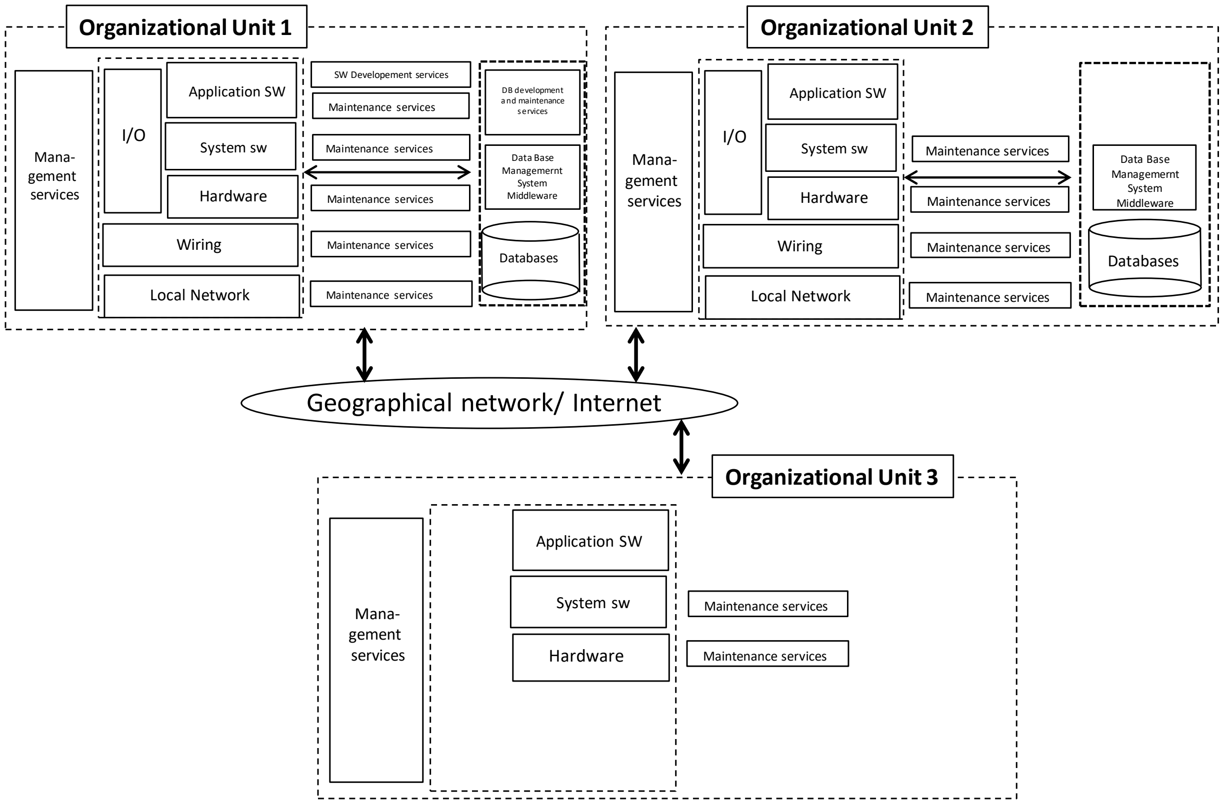


Figure 8.4 – Mapping of the technological architecture in a hypothetical organization made of three organizational units

We will now discuss components, architectures and services in more detail.

8.4. Technological components

Technological components correspond to application software, system software, hardware, input output units, wiring, local networks, and geographical networks. Application software is the software that is built on demand starting from custom requirements, and that addresses needs typical of specific users or groups of users, e.g. a software to manage accommodations for Erasmus student in a University. We will not deal in the following with a. application software, b. input output units, c. wiring, d. local networks, e. geographical networks; we will focus on system software and a specific but quite important and widely used hardware, namely servers.

8.4.1. System software

System software is the software built once for all by system providers, and maintained on time; system software realizes functionalities of large usage, such as e.g. access to secondary memory, or else a word processor. We distinguish four types of system software:

- Operating system, the software that operates in every computer, whose goal is to allow efficient usage of computer resources (e.g. memory) by software applications.
- Middleware, the software that is invoked by application software to perform repetitive activities, e.g. the Data base management system that manages concurrent requests of access to databases.
- Office automation, e.g. Excel

- Application development environments, that facilitate programmers to produce software applications (e.g. compilers, that translate high-level language programs into machine language programs.

8.4.2. Servers

A server is a software program and at the same time a device. In fact, the term *server* refers to a computer program or process (running program) that is part of the system software. Through homonymy, it refers to a hardware device used for (or a device dedicated to) running one or several server programs. The main function of servers is in providing functionalities to other programs and devices, called clients, see Figure 8.5, from which users access and interact with the system.

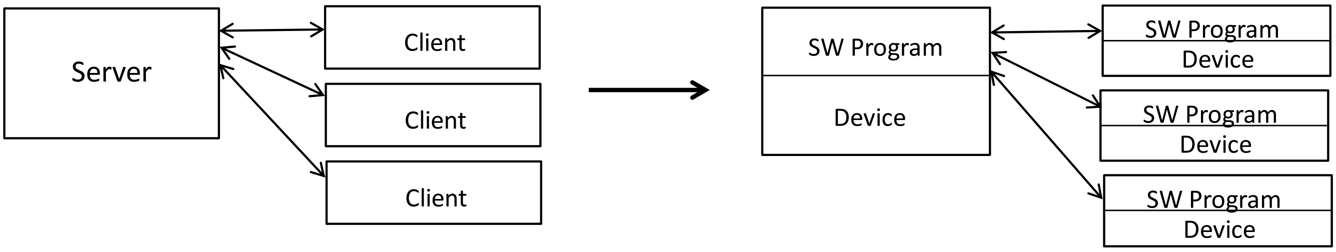


Figure 8.5 - Servers and clients in the hardware architecture

Examples of functionalities provided by servers are:

- sharing data or other resources among multiple clients, or
- performing a computation on request of one or more clients.

Clients may be personal computers or other servers. A single server can serve multiple clients, and a single client can use multiple servers. A client process may run on the same device or may connect over a network to a server on a different device, see Figure 8.5 as an example of servers & clients architecture.

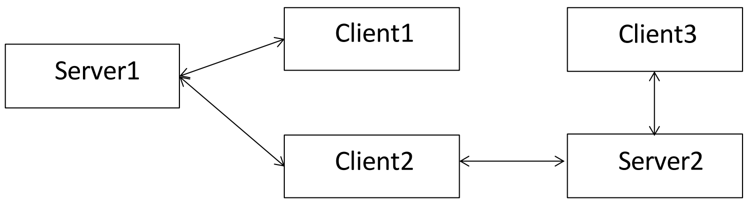


Figure 8.6 – Examples of servers and clients in an hardware architecture

As we said, servers can provide various types of functionalities. Main types of servers and related functionalities are:

- Database server – manages a database shared among several software applications
- Application server – provides the business logic for automated processes, namely, the software functionalities shared among client and servers. Typically, the business logic is made of three layers or tiers (see Figure 8.7):
  - First tier made of the browser application that runs on the client Personal Computer.
  - Second tier, the application server, that runs application programs (e.g. make a report).
  - Third tier, the database server (e.g. access data in the data base needed to make the report).

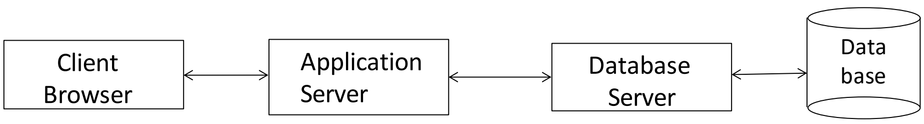


Figure 8.7 – The typical three tiers client server architecture

- Print server – provides shared access to printers
- Internet server – connects users to Internet to share information. Specific functionalities correspond to:
  - File server – manages requests from clients for files stored in secondary memory.
  - Domain Name Server - distributes network specific information such as IP addresses to allow e.g. sending an e-mail or a file.
- Mail server – manages an electronic mail directory
- Web server – presents information to the user in hypertext format and allows users to navigate in the Web.
- Proxy server – restricts access to data on the Web.
- Security server – manages all security issues (digital signature, secure e-mail, etc.)

Servers of various types, and less recently, computer mainframes, are the typical components of system software + hardware architectures. We may historically identify three types of such architectures:

- Mainframe / terminals (year 60s to today) – made of a single mainframe computer or centralized cluster of servers + a large number of connected terminals, on which simple transactions are performed (e.g. banking transactions, booking transactions).
- Client Server (years 80/90 to today) made of clients and servers. The software is distributed among
  - clients, e.g. personal computers connected by a local area network
  - servers, which can operate at different levels (tiers) of distribution in a local or geographical network
- Web based (late years 90s to today) - Client server solution that uses Web technologies (browser as the sole type of client, http protocol, etc.) to offer services to a variety of users accessing the Web.

8.5 Choice of Technological Architectures

In this step, we show how application software architecture, server architecture and data architectures can be designed and identified, in terms of their relationship with the organizational architecture. We will proceed by examples, without any presumption of being exhaustive. Anyhow, the reader should have clear the fact that the simple examples we provide are realistic, and not so far from real cases.

Remember that a technological architecture is an allocation of technological components and services in the different units of the organizational structure. Since organizations are logically and physically distributed, the IT system is also generally organized in distributed levels.

<u>Allocation of application software</u> - Assume we have two software applications that manage respectively the contractual economic treatment of employees and their holidays and illnesses. Being the contractual economic treatment common to all employees while holidays and illnesses dependent on single employees, we may choose the following allocation (see also Figure 8.8).

- Economic treatment of employees (ETE) in Headquarters,
- Holidays and illnesses (HI) by Region.

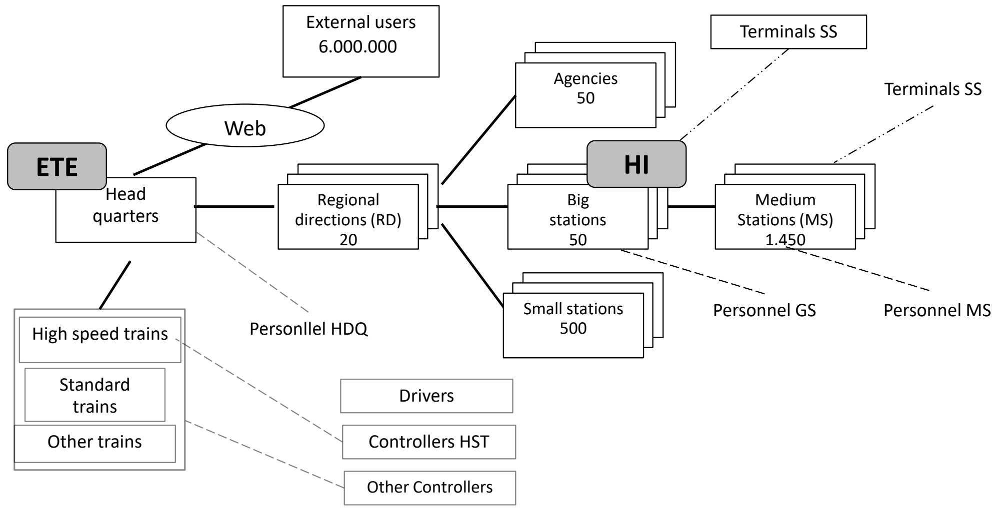


Figure 8.8 – Example of application software architecture

<u>Allocation of local networks</u> - Examples of technologies for local networks are wifi (WF) and fiber optics (FO). We can adopt wifi (WF) in stations and Fiber optics (FO) in regional directions and headquarters, see Figure 8.9.

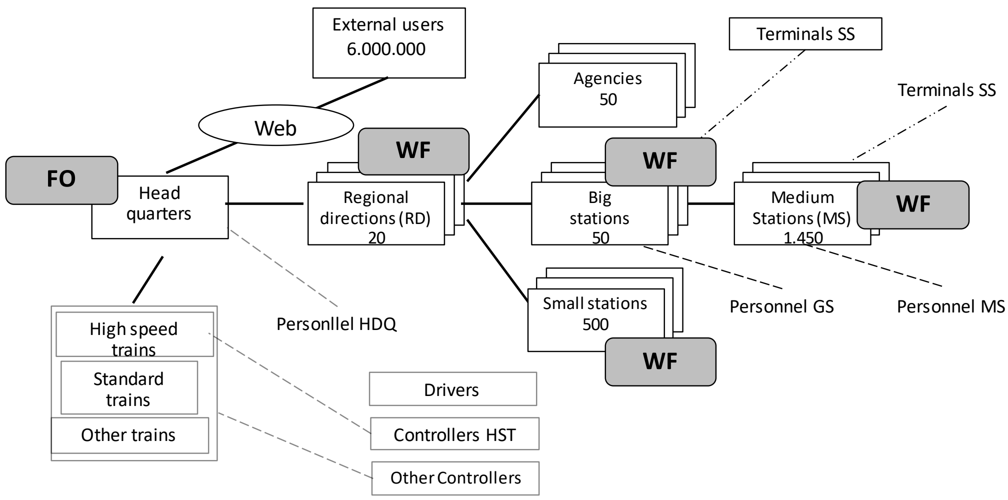


Figure 8.9 – Example of local network architecture

<u>Allocation of input output terminals</u> - In this case we extend of self service terminals from big stations to medium stations.

Exercise 8.1 - Allocation of servers of different types in the exam registration case study

Consider the following types of servers:

- Application
- Database
- Security
- File
- Proxy
- Security
- Mail
- Web

and software applications shown in figure 7.11 that we have identified in the New software components phase of the life cycle. We have associated to the first software functionality (query + file transfer) the related choice of servers in terms of application (extract) + file transfer (send) servers.

Exercise 8.1 – Complete the table with servers of types listed above, see Figure 7.11.

Ownership/ Responsibility	**Software application**	**Main software functionality**	**Server**
Central secretariat	Extract and send the digital registry	Query + file transfer	Application + File
Teacher	Update the prefilled registry	Update a table	
Teacher	Digital signature	Off the shelf - digital signature sofware (buy)	
Teacher	Send securely the registry	File transfer	
Central secretariat	Formal controls	Join between tables owned by different organizational units	
Central secretariat	Update the exam database	Update a table	

Figure 8.10 – Software applications and main software functionalities in the exam registration case study

<u>Allocation of servers and personal computers in the railways company case study</u> - In this case we have at least two possibilities

1. Fully centralized architecture
2. Distributed architecture

In the first case we may perform an allocation of a cluster of servers at the headquarters, and of personal computers in all other organizational units, see Figure 8.11.

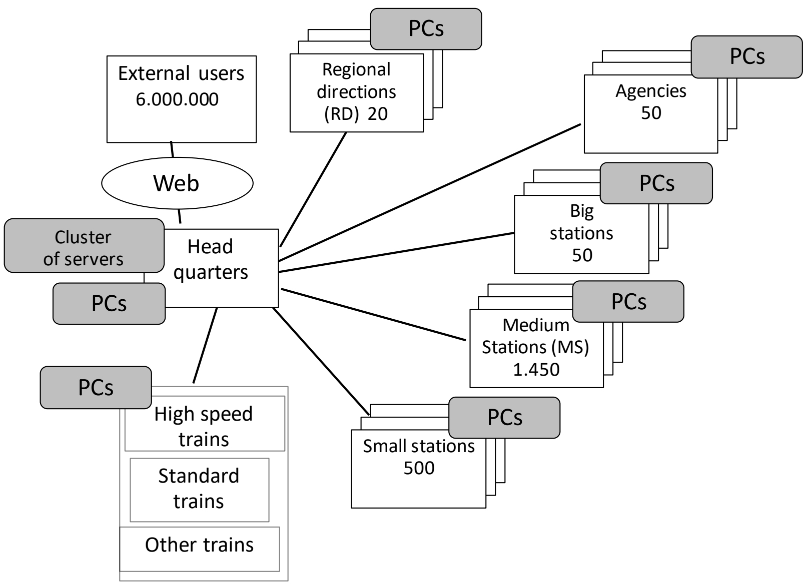


Figure 8.11 – Centralized hardware architecture in the railway company case study

A distributed architecture adopts servers, personal computers and I/O devices in all the organizational units. For what concerns PCs, tablets and self-service terminals, we may locate devices as in Figure 8.12, where we choose tablets instead of PCs for controllers of High speed trains due to their manageability and versatility. In small stations, that are unattended it is enough a smart phone to exchange data, sms, photos etc.

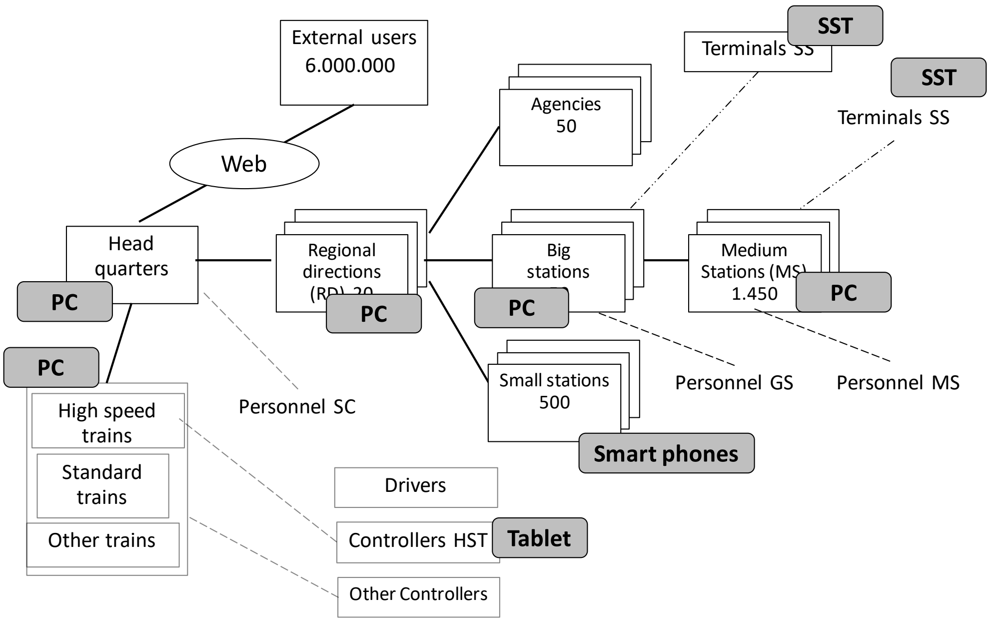


Figure 8.12 – PCs and I/O devices architecture in the railway company case study

Concerning servers to be allocated, we focus on the following types

1. Application server (App)
2. Data Server (Da)
3. Web Server (We)
4. File Server (File)
5. Backup server (Ba)
6. Print server (Pr).

A possible allocation appears in Figure 8.13; we see that all types of servers are allocated in headquarters, while in regional directions and big stations we do not allocate the backup server and the web server since we assume to perform centrally the backup and access to the Web. In medium stations we do not allocate also the data server, assuming that no data base is physically located there.

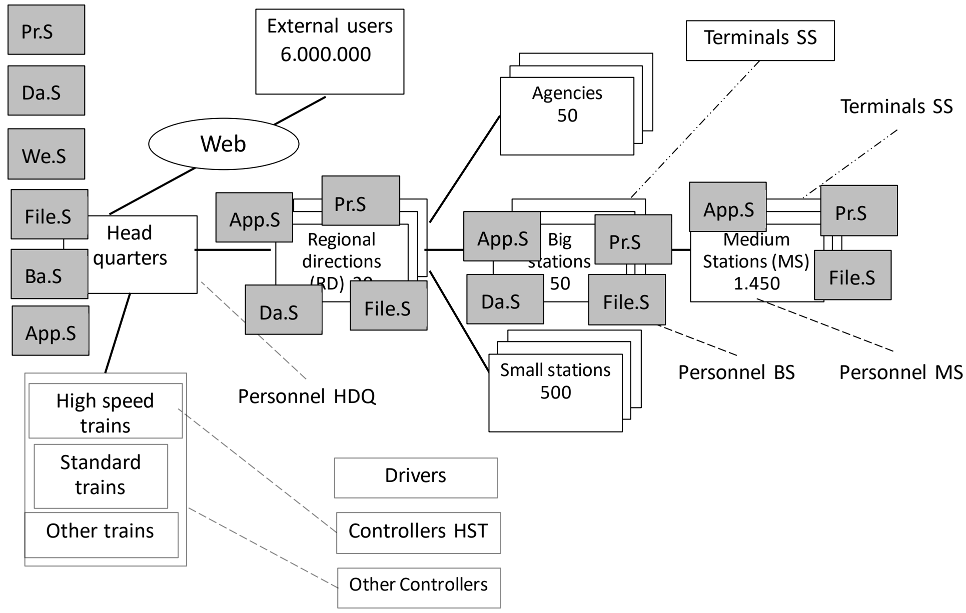


Figure 8.13 – Server architecture in the railway company case study

Notice that to enhance the resilience, that is the property by which in a networked architecture local nodes remain operating also in case of faults at central level, we may allocate smaller web servers and backup servers also in regional directions and in stations.

Allocation of databases in general - Moving to data architectures, we have first to go in more depth on the typical architectures that are of two types, centralized vs distributed. Furthermore we provide some detail on typical architectural migrations, that may be:

- From centralized to distributed
- From distributed to centralized, this migration is also called consolidation.

We remember that a typical database is made of three parts:

1. Data, such as “Mario Rossi”, that refer to observables of the real world one wants to represent in the database.
2. Schema, such as the entity Student, that refers to classes of observables of the real world represented in the database.
3. Data Base Management System (DBMS), the middleware used to access efficiently the database by queries and transactions.

The typical centralized data architecture is shown in Figure 8.14; it is made of a unique database, a unique global schema and a Data Base Management System.

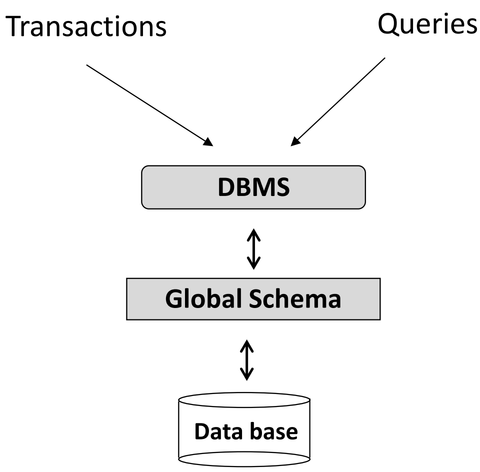


Figure 8.14 – Centralized data architecture

An example of distributed data architecture and related migration from a centralized to distributed architecture is shown in Figure 8.15.

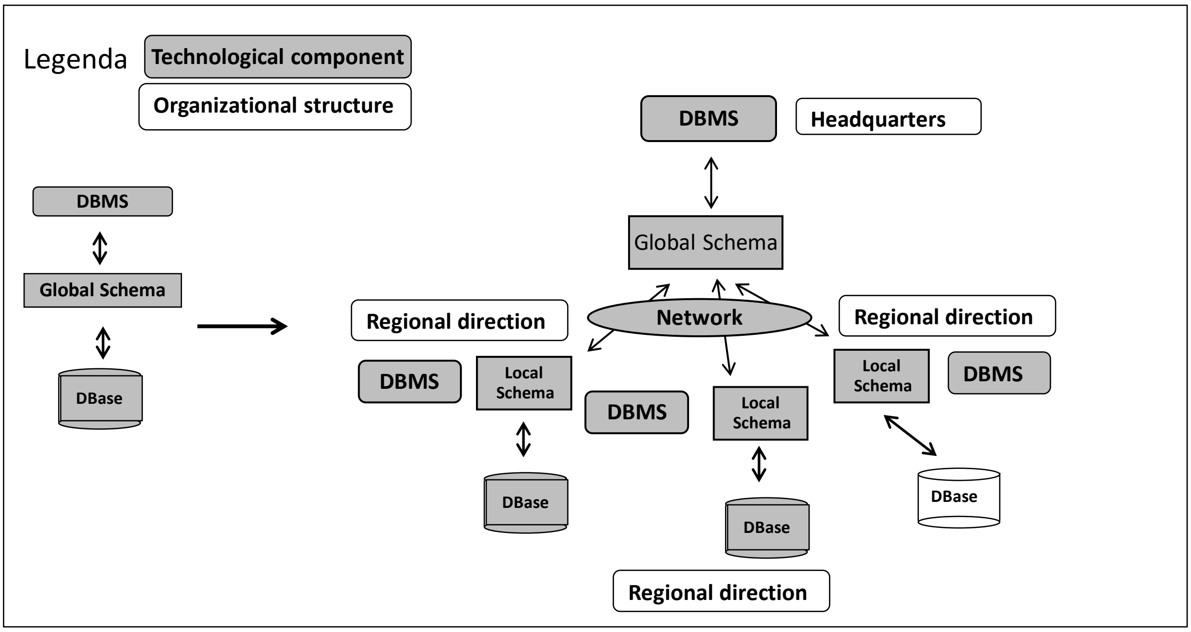


Figure 8.15 – Distributed data architecture and migration from centralized to distributed

Focusing on the data schema, in order to move from a centralized schema, which in the relational model is made of tables, we have to decide how tables migrate in the distributed nodes. We have two types of fragmentations; the first type is the vertical partitioning, that we exemplify in Figure 8.16. We assume to have in the centralized setting a table Employee with attributes Key, Node (where he/she works), Surname and Overtime; we can split vertically the table, resulting in two tables with common key and respectively attributes Place and Surname in Employee1 and Overtime in Employee2. We have to keep in both tables the key, otherwise we could not be able to reconstructs the original content of table Employee.

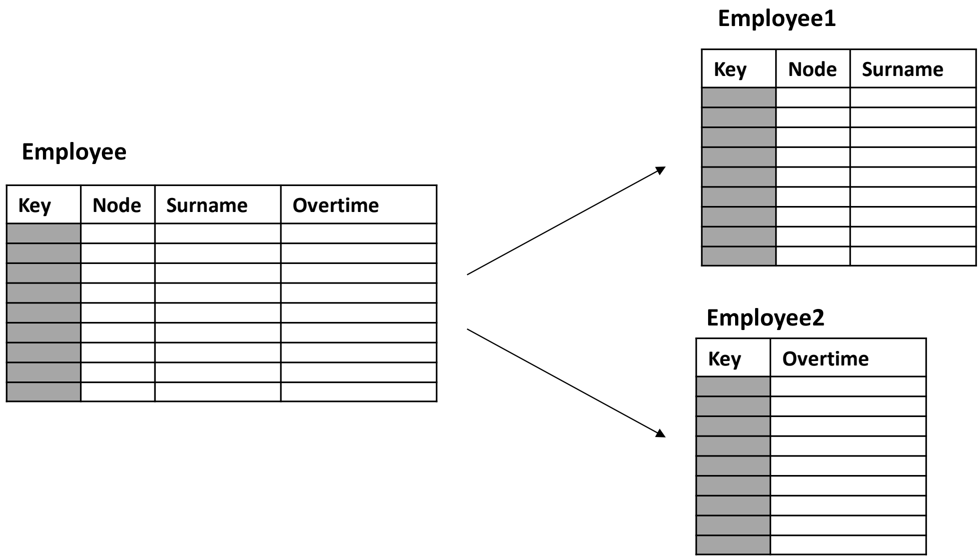


Figure 8.16 – Vertical partitioning

An example of horizontal partitioning is shown in Figure 8.17. In this case the partitioning is driven by the node where the employee works (corresponding to the regional direction). We can split horizontally the table in terms of n subtables, that represent the n regional directions.

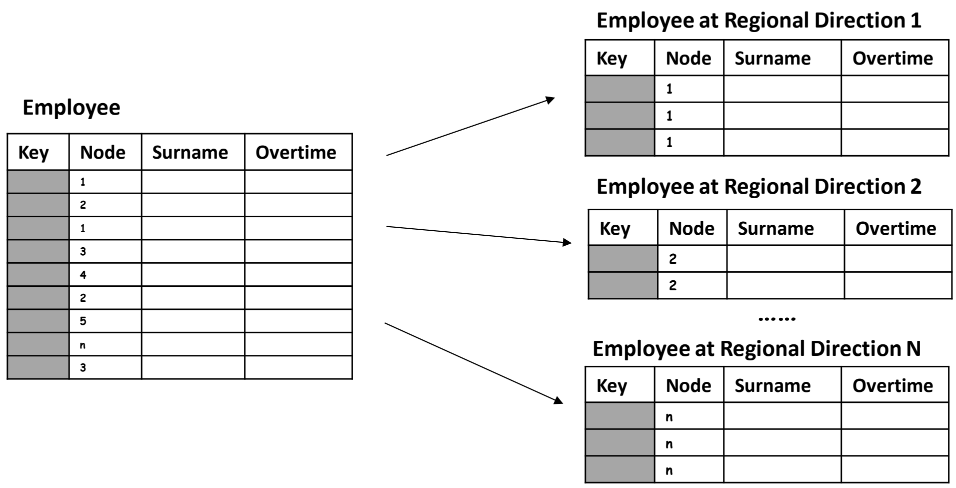


Figure 8.17 – Horizontal partitioning

The final architectural migration that we show concerns consolidation, where a cluster of distributed or autonomous set of databases is merged into a unique centralized database, see Figure 8.18.

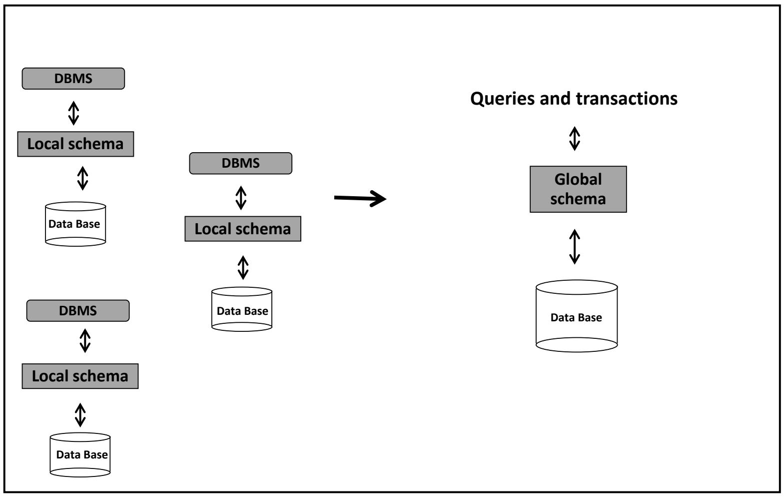


Figure 8.18 – Database consolidation

8.5 – Allocation of databases in the railway company case study

In the railway company database we may identify in the requirements, see Figure 8.19, the databases represented pictorially in Figure 8.20.

Main databases concern:

- Railway network, fairly stable over time.
- Trains available
- Railway timetable, which is updated every year according to the profitability of the routes and to social policies
- Reserved seats and tickets
- Personnel
- Stations
- Lost items
- Statistics

Figure 8.19 – Databases mentioned in requirements

Network	The railway network, fairly stable over time.
Trains	Trains available
Timetable	The railway timetable, which is updated every year according to the profitability of the routes
Tickets	The train travels, with class, coach and price
Personnel1	The database on Personnel, with personal data and data on holidays, illnesses, permits
Stations	Stations managed
Lost & Found	Lost property
Statistics	Statistics

Figure 8.20 – Databases involved in the railway company case study

In figure 8.21 we display the frequency of transaction updates for the different databases.

Data Bases	Number of Updates
Network	Once in a year
Trains	360 in a year
Timetable	Two times in a year
Tickets	100 times every minute
P1 - Personnel - Personal data	1.000 in a year
P2 - Personnel - holidays, diseases, etc.	100.000 in a year
Stations	1.000 in a year
Lost and found	10.000 in a year
Statistics on Personnel	1 every month

Figure 8.21 – Frequency of updates in the databases

We initially focus on two databases, Personnel and Statistics. Notice that Personnel initially undivided, but characterized by two partitions that have unequal transaction loads. A first choice on the two databases is to locale them centrally in the Headquarters node. In Figure 8.22 we show pros and cons of this solution.

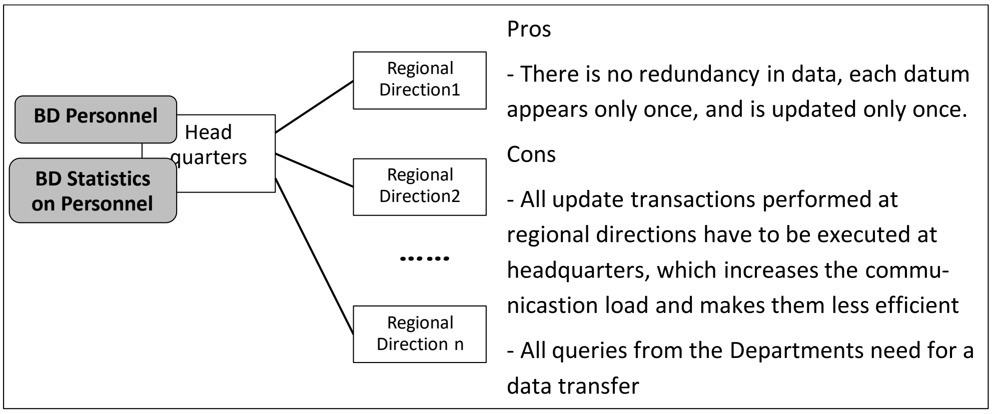


Figure 8.22 – Pros and cons of the centralized solution for Personnel and Statistics databases

A different choice is based on horizontal and vertical partitioning that can be performed on the Personnel database, see Figure 8.23; such partitionings are suggested by the unequal frequency of updates on personal data and holidays, illnesses etc. shown in Figure 8.21.

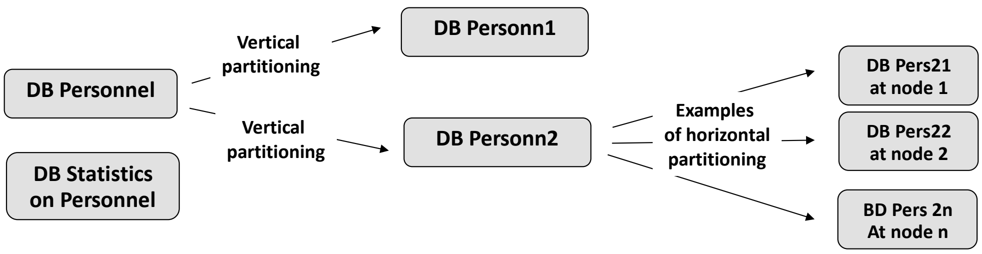


Figure 8.23 – Possible partitionings on the Personnel databases

We may apply both partitionings resulting in the allocation of Figure 8.24, where pros and cons of the solution are mentioned.

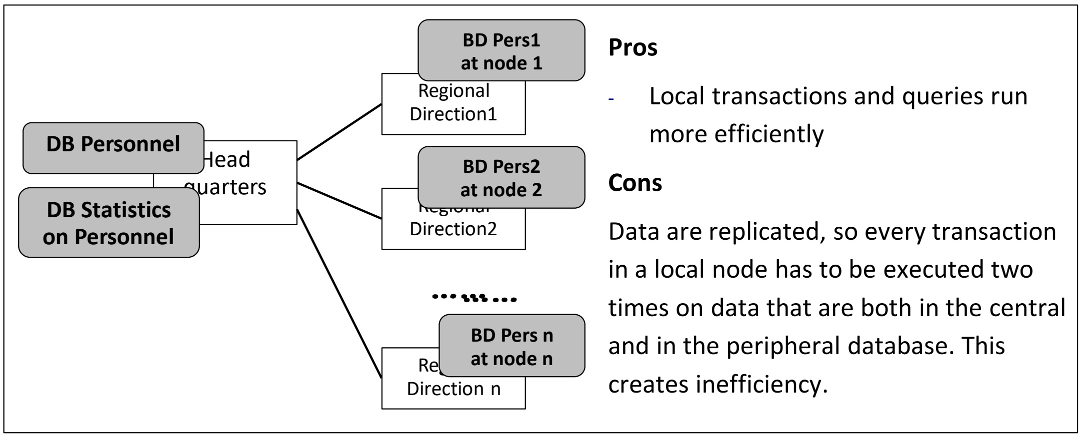


Figure 8.24 - Pros and cons of the distributed solution for Personnel

Methods that allow to choose one solution between the two are based on cost models for queries and transactions, that take into account both costs at the node, and costs to transfer data between nodes. We will not go into further detail.

Other databases can be allocated with similar arguments, resulting in the data architecture of Figure 8.24. You may notice that we have replicated in all nodes the Timetable database. This is because such database is updated two times in a year while it is frequently accessed in all nodes, so it is worthwhile to duplicate it in all nodes.

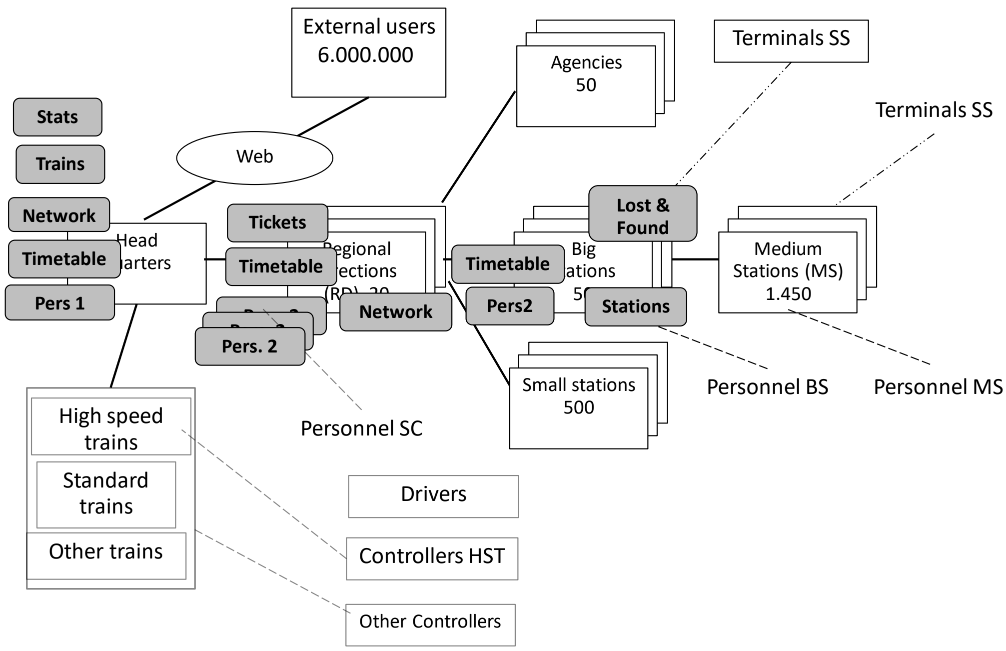


Figure 8.25 - Data architecture for the whole set of databases

8.7 Further elements for sizing the application software, input/output devices and hardware (servers + personal computers and tablets)

In order to size the technological architecture we have considered so far the frequency of queries and transactions. Other elements for the sizing the application software, input/output devices, servers, personal computers and tablets can be found in requirements; we reproduce the parts of requirements that mention useful information.

There are also about 20,000 employees at the stations, of which 3,000 are in large stations, and 2,000 in small stations. There are other 10,000 employees, including traveling personnel (6.000) and administrative staff (4.000).
The network consists of about 1,000 kilometers at high speed and 4,000 kilometers for normal speed trains. Finally, there are another 5,000 employees with administrative duties (personnel, accounting, etc.) who work in the 20 regional offices distributed in the 20 Italian regions.
The main primary processes of the company concern:

- The organization of travels for individuals, through a park of about 100 high-speed trains, consisting of 1,000 wagons. The high-speed users are around 1 million, with an average route per ticket of 300 kilometers, while the users of "normal" trains are 5,000,000, with an average distance of 50 kilometers. This process, currently, for the part of booking and purchasing tickets, uses in addition to the internal channels, about 1,000 external agencies, which are paid a fee equal to 10% of the ticket.
- The organization of travels for freights, through a park of about 50 trains.
- The organization of special trains, through a park of about 20 trains, and 200 wagons.
- The organization of night trains, about 20, which however raise significant safety problems, and have high costs because external personnel must be hired by a specialized company, very expensive

Limitations
…
There are about 10,000 complaints per year for delayed arrival, which lead to a significant cost for the company.

From such requirements we can fill the following comprehensive table, see Figure 8.26.

Sizings of structures, resources, users	Stations			Passengers of trains		Total
	Big	Medium	Small	Highspeed	Regional	Total
Stations	50	1.450	500			2.000
Employees at stations	3.000	15.000	20.000			20.000
Traveling + Administrative employees						6.000 + 4.000
Employees in Regional Directions						5.000
Reg. Directions						
Users				1.000.000	5.000.000	
High speed trains						100
Normal trains						Not known
Freight trains						50
Special trains						20
Night trains						20
Agencies						1.000
Claims						10.000

Figure 8.26 – Sizing of organizational structures, users and resources

8.8 - Management services

We now move to management services. An information system, in order to function in a state of

- efficiency (i.e. with reasonable use of resources),
- effectiveness (i.e. responding to the needs of end users),

must be overseen through a set of management services, provided by human and technological resources, which must be identified and sized at this stage. Management services belong to four types.

1. Operations management, that refer to Information technology (IT) infrastructures enabling:

- Functionality of systems - correspond to applying procedures to ensure the functioning, availability and restoration of systems, applications, databases, workstations and the communication network.
- Security – refer to procedures to ensure data integrity and access control to technological resources, e.g. databases.

2. System evolution - management of the evolution of the IT system in terms of
    - Changes in hardware and software configurations,
    - New installations,
    - Software distribution (e.g. versions of software for self-service terminals)

- Software license management,
- Warranty and maintenance contracts with suppliers.

3. User support - management and resolution of system problems (application and system support),
- Help desk - call center for internal and external users,
- Support - support for users in the usage of processing system functionalities,
- Maintenance - network, hardware and software application maintenance; software maintenance may be of the following types:
  a. Corrective (e.g. repair of flaws in application software)
  b. Adaptive (e.g. change of features in the interface)
  c. Improvement (e.g. improvement of disk performance)
  d. Evolutionary (e.g. change of norms for tax returns which leads to substantial changes in software applications).

4. Other - management of other requests coming from users such as e.g.
- Training
- Data and image acquisitions,
- Data entry
- Digital signature certification.

8.8 Cloud services

Nowadays, hardware and software resources in many organizations are no longer acquired by the market, and owned by the organization; the advent of cloud computing is radically changing the way such resources are managed, moving progressively from considering them as goods that are owned to services that are used.

Cloud computing is (from Wikipedia) the on-demand availability of computer system resources, especially data storage and computing power, without direct active management by the user. The term is generally used to describe data centers available to many users over the Internet. Large clouds, predominant today, often have functions distributed over multiple locations from central servers. We shortly mention this topic, without presumption of being exhaustive. The interested reader can find in the literature a huge amount of references on the cloud.

In Figure 8.27 we show possible subdivisions of IT resources between the organization (in grey) and the cloud (in white). In the traditional allocation of IT technologies, they are managed by the organization; this allocation is called “on premises”.

There are three possible evolutions of such arrangement called:
- Infrastructure as a service, where the cloud service and devices refer to the hardware infrastructure, including storage and networking.
- Platform as a service, where also the operating systems, the middleware and the run time execution of application software are managed in the cloud.
- Software as a service, where all the resources are managed in the cloud.

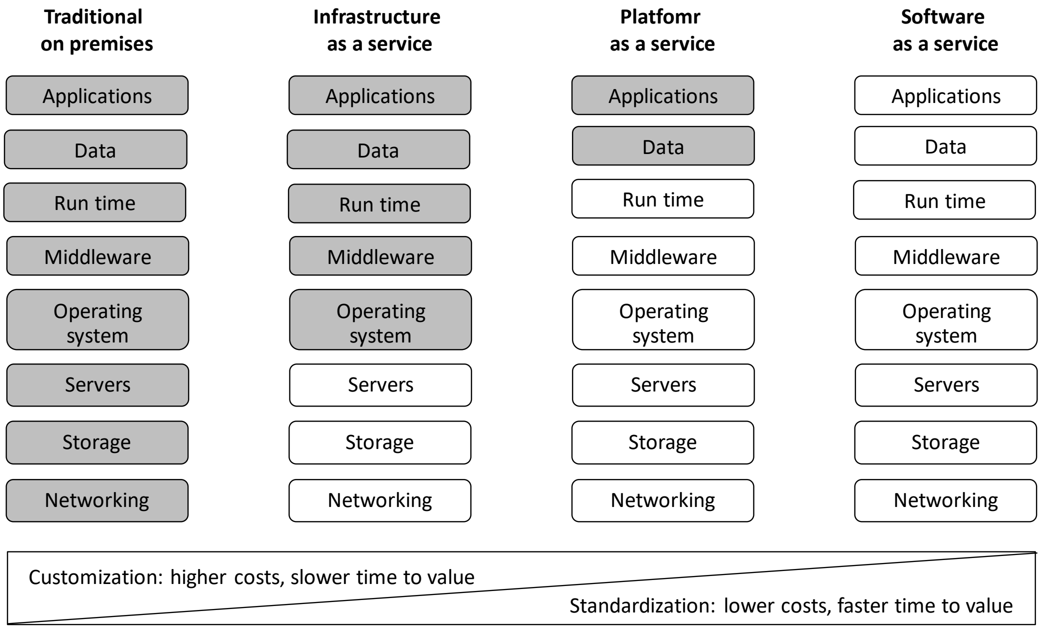


Figure 8.27 – Evolution of architectures in the Cloud

The cloud infrastructure in comparison with the traditions asset of IT resources result in a much wider flexibility and elasticity, meaning that can react rapidly to periodical changes in the application load. In Figure 8.28 we compare possible evolutions of computing power capacity in consequence of the evolution of demand capacity. In the traditional setting the capacity of hardware equipment, e.g. of servers, may evolve only by discontinuous steps, resulting in waste of capacity and consequent higher costs.

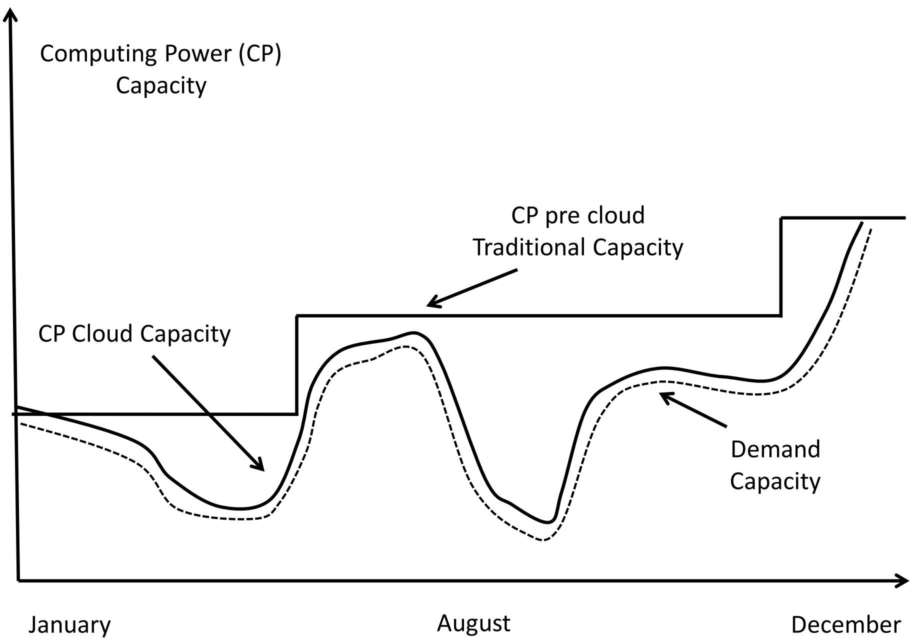


Figure 8.28 - Typical trend of computing power in a year as managed traditionally and in the cloud

In a cloud environment, being the cloud a resource that can be shared between different organizations, the cloud capacity may follow the demand capacity more closely, resulting in potential savings compared with the traditional solutions.

Appendix 8 – Solution to exercises

Solution to Exercise 8.1 - See Figure 8.29

Ownership/ Responsibility	Software application	Main software functionality	Server
Central secretariat	Extract and send the digital registry	Query + file transfer	Application + File
Teacher	Update the prefilled registry	Update a table	Database
Teacher	Digital signature	Off the shelf - digital signature sofware (buy)	Security
Teacher	Send securely the registry	File transfer	File
Central secretariat	Formal controls	Join between tables owned by different organizational units	Application + Database
Central secretariat	Update the «student course of studies» (carriera studenti)	Update a table	Database

Figure 8.29 – Choice of servers in the exam registration case study

Chapter 9 – Cost evaluation

Carlo Batini and Gaetano Santucci

9.1 Introduction

The phase of cost evaluation has the goal of determining the cost of the different technological components and services, selected in previous phases as the building blocks of the information system, see Figure 9.1.

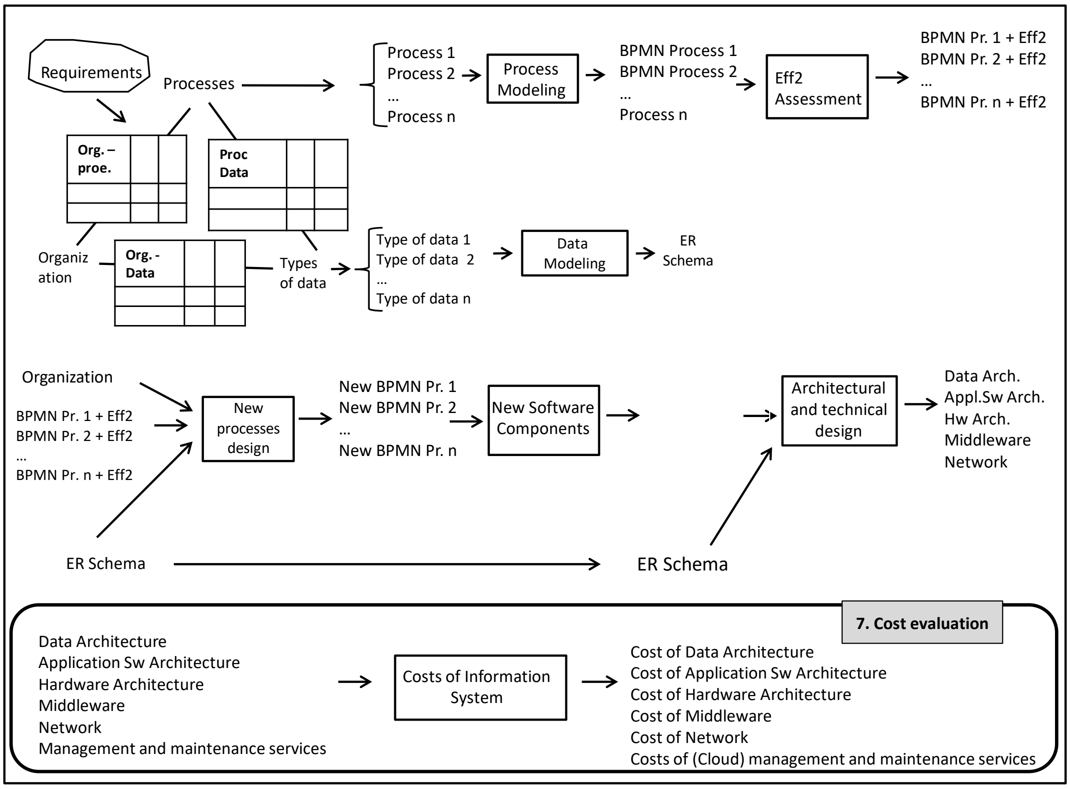


Figure 9.1 – The phase of cost evaluation in the information systems life cycle

In this introductory section, we deal with general issues related to costs, while in the following ones we focus on specific cost items. As a first issue, the application load influences information system costs. We recall that the application load of an information system is the set of software applications that run in the system, described with their types and the frequency of execution, see Figure 9.2. We have identified in previous chapters two types of basic operations used in software applications, namely transactions and queries.

Transactions modify the content of the database, while queries extract data that satisfy given conditions. Both transactions and queries are used in software applications in the modality called On line Transaction Processing (OLTP). In this case, the application load is typically made out of a huge number of transactions that have a simple structure (e.g. the "I need a ticket" transaction in Figure 9.3) or queries that extract useful information (e.g. the query "What is the delay?"). Such transactions and queries realize functionalities that make reference both to primary processes and support processes introduced in Chapter 1.

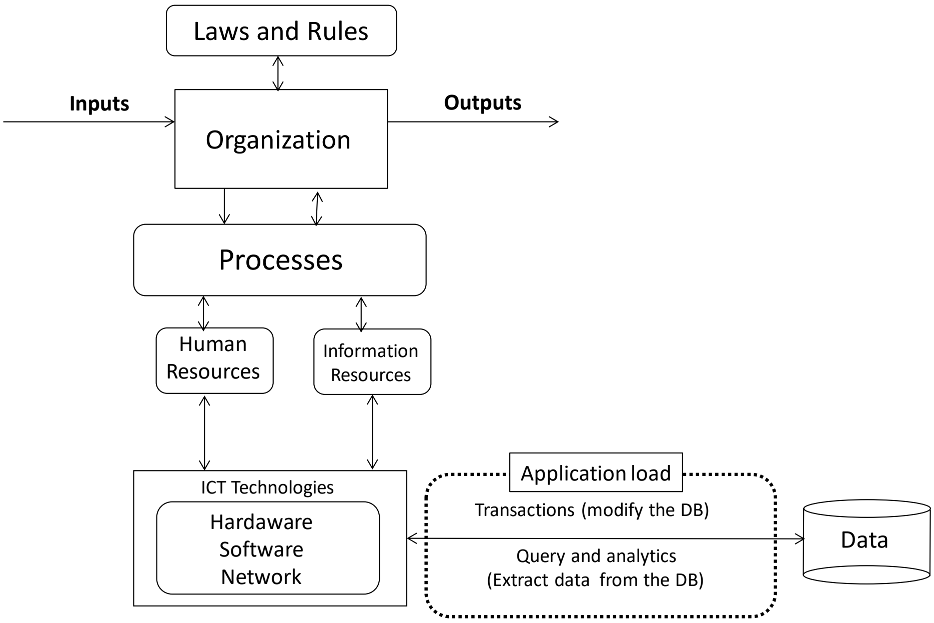


Figure 9.2 – The application load

Besides OLTP, there is another modality of usage of information systems, the On Line Analytical Processing (OLAP). In this modality the typical operations are complex queries or analytical functions, that usually operate on the full set of databases, or on relevant parts of them, see Figure 9.3. Such operations do not work on databases used by primary or support processes, but rather on a new database called Data Warehouse.

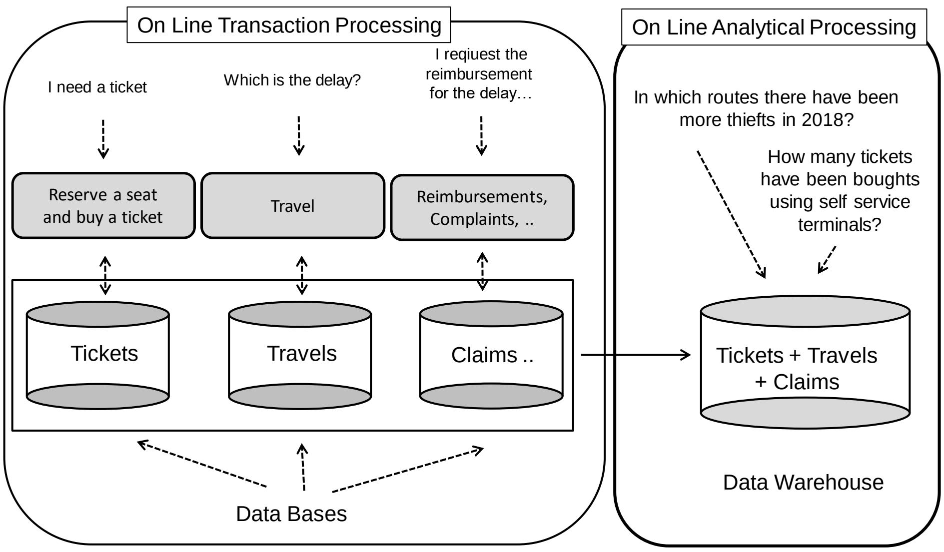


Figure 9.3 – On line transaction processing and on line analytical processing environment examples

In data warehouses, data that are dispersed in the operational databases are integrated to form a unique representation of data managed in the different databases, in such a way that it is possible to perform queries and other operations that span on the whole Information system (see examples in Figure 9.3). Organizations need to perform such computations for planning purposes, when they have to conceive the evolution of the organizational system and business processes. E.g., a railway company needs to know which are the travels that provide major revenues, or which are the travels in the week that are to be reinforced with more trains or more coaches, identified through the occupancy factor, namely the percentage of seats occupied. Such factor can be evaluated in the example of Figure 9.3 only by putting together data managed in the Tickets and Travels databases.

If we need the Data Warehouse to contain up-to-date data, it has to be fed with recently updated data with high frequency. Looking at the table in Figure 8.21, that provides the number of transactions in a year for the databases of the railway company case study, we see that critical databases for OLTP usage are the Tickets database and the Personnel P2 database. We will use the OLTP vs OLAP classification later in the chapter.

9.2 Make, Buy and Adapt

Procurement of technological components and services may be performed according to three different modalities, corresponding to a. *make*, when they are produced within the organization, b. *buy*, when they are acquired from the market, c. *adapt*, when they result from an adaptation of an artifact (a software program, a server, etc.). In Figure 9.4, we show main components and artifacts previously introduced, together with a classification in terms of the different types of procurement. With regard to *buy*, we separate between *buy on premises* and *get the service from the cloud*. It should be noted that, in order to obtain economic or operational advantages different arrangements of purchasing on premises may be acted upon, namely the purchase, the leasing or the rental of goods.

Component/Service	**Make**	**Buy on premises (good)**	**Get from the cloud (service)**	**Adapt**
Hardware - Server		X	X	
Hardware – Personal Computer		X		
Hardware - Printers		X		
Middleware – Operating System		X	X	
Middleware – DBMS		X	X	
Office automation (e.g. Word, Excel)		X	X	
Application Software	X	X	X	X
Management services	X	X	X	
Maintenance services	X	X	X	

Figure 9.4 – Components & services

Hardware, in its different components, is bought from the market. Servers can be either bought from the market and as a service from the cloud, as we have seen in Chapter 8. When they are bought from the market, their price is established by economic laws, such as the law of demand and offer; discounts may be applied when the number of units is significant. When it is bought from the cloud, pricing is made according to pricing models discussed in the next section. Personal computers and printers have to be bought on premises.

Management and maintenance services for hardware may be bought from the market, or else paid as a service. In both cases, the cost of the resource/service is influenced by the service levels that are specified in the contract; as an example, when the system has a fault, it could be requested in the contract that the provider of the service must intervene before a certain threshold of time, e.g. two hours. In addition, system software may be acquired on premises or as a service from the cloud, and it is subject to similar laws, that we will discuss later in the chapter. System software is also produced and distributed as open source, and in this case is free.

Concerning application software, there are generally three choices:
1. *Make*, i.e. through internal development
2. *Buy* when the development is outsourced to the market
3. *Adapt*, namely by adaptation of existing software (e.g. Sap, Buffetti in Italy, etc.).

Application software can be seen as an hybrid between a technological component and a service. Differently from previously discussed technological components, the price of application software significantly depends on requirements, spanning from simple applications that automate a single functionality of a process to extremely complex applications. For instance, a new competitor of the railway company in our case study has to create from scratch the whole set of applications of the information system. Maintenance services for application software are procured and priced similarly to the software they refer to.

9.3 Costs of hardware - servers

The price of servers can be established:
1. By performance index
2. Per number of users

Let us see the two cases. Here and in the following, costs mentioned refer to year 2019.

9.3.1. Pricing by performance index

There are three types of benchmarking of servers that can be performed with different application loads:
- Synthetic - express a set of performance characteristics through numerical values that cannot be directly related to the application load.
- Applications - measure performance with direct reference to a particular application context, e.g. OLTP or OLAP.
- Targeted - measure specific functional uses (e.g. Web server, database server, etc.).

Performance measures are:
- Throughput
- Response time
- Cost per transaction

<u>Throughput</u> - The throughput measures the overall performance of the system. For OLTP processing systems, throughput is typically measured in terms of *transactions per second* (TPS) or *transactions per minute* (TPM). Several types of application benchmarking exist that have the throughput as target, issued by the Transaction Processing Council and identified by a letter of the alphabet, TPC-A, TPC-B, TPC-C, TCP-D etc. We focus now on two of them, TPC-A and TPC-C.

The TPC-A benchmark measures performance in update-intensive database environments corresponding to simple OLTP applications. This benchmark uses a single, simple, update-intensive type of transaction to load the tested system. Environments are characterized by:

- Multiple on-line terminal sessions
- Significant disk input/output

Exercise 9.1 - As to the case study on the railway company, which is the most update-intensive transaction to be chosen for the TPC-A benchmark?

Exercise 9.2 - As to the case study on exam registration, which is the most update-intensive transaction to be chosen for the TPC-A benchmark?

The TPC-C benchmark is used for complex on-Line Transaction Processing applications; it is measured in transactions per minute (tpmC). It differs from TPC-A in the following aspects:

1. TPC-C is more complex than TPC-A, because of its multiple transaction types, more complex databases accessed and overall execution structure.
2. TPC-C involves a mix of five concurrent transactions of different types and complexity, either executed on-line or queued for deferred execution.
3. The database is composed of nine types of tables, with a wide range of record and population sizes.

Exercise 9.3 - As to the case study on the railway company, list five or four transactions to be chosen for the TPC-C benchmark.

Solutions to exercises appear in Appendix 9.1.

Response time - Response time measures the performance of individual transactions or queries; it is typically treated as the elapsed time from the time that a user enters a command or activates a function until the time the application indicates that the command or function has completed.

Cost per transaction - Cost per transaction is a financial measure that is typically used to compare overall operating costs among applications, database servers, or hardware platforms.

9.3.2 Pricing per number of users

Typical technologies for servers refer to the operating system on which they run. Servers adopting UNIX are more powerful and costly, while cheaper servers usually adopt Linux or Windows operating systems. Linux is adopted in the open source release or, more frequently, in the so-called “distributions”, implemented by commercial operators from the open source code; the operators take care of the updates, management and assistance to the customers (e.g. Red Hat, Suse).

Concerning pricing per number of users, we assist to a steady decrease in prices. Figure 9.5 provides typical cost per user for number of users ranging from 20 to 500, checked in year 2000, while Figure 9.6 shows prices collected in year 2019, a significant decrease may be noticed.

Users	Cost per user (in euro)
20	500
40-50	400
100	300
300-500	200

Figure 9.5 - Costs of servers per user, year 2000

Prices above are for servers characterized by an OLTP application load. Correction for application loads different from OLTP are as follows:
- Data warehousing (OLAP) Server have an increase in costs equal to 100%
- Web Server, with a decrease in costs equal to - 50%
- Network server, with a decrease in costs equal to -50%.

Typical discounts can be from 15% to 40%, depending on the order of magnitude of the supply.

Product	Price
NT Server 4.0 (for 5 users)	809 USD
NT Server 4.0 (for 10 users)	1 129 USD
NT 4.0 server edition for businesses (25-user version)	3 999 USD
NT 4.0 server edition for businesses (50-user version)	4 799 USD
literature kit NT Server 4.0	$ 69.95
license for 20 clients, prices of many	329 USD
license for a client	$ 19.95

Figure 9.6 – Costs from https://www.techulator.com/resources/2437-Microsoft-Windows-NT-Server-Comparison.aspx (2019)

9.3.3 Recent trends in server techologies

The concept of "server" has changed over time. Today, a server may be defined as an integrated suite of hardware and software components that offers specific services to other servers or clients.

Performances of the different technical solutions are very difficult to compare, and there are many technical and commercial obstacles among the different manufacturers in sharing the results of common benchmarks.

The estimation of server costs depends on the services released and on the technical configuration chosen (e.g.: a physical server with only one image of the operating system, a virtual server obtained from a physical server with virtualization software and an appliance that integrates hardware and software to provide services). Therefore, the estimation of the servers' costs may require an initial choice among the technical architectures.

In estimating servers' costs, we must take into account the different services required and, for each kind of services:

- defining one or more possible solutions;
- establishing the technical or functional performance required;
- consulting the offering and evaluate the cost of the solutions.

We show two examples of price evaluation using this method.

The first example concerns Oracle data base technologies and two possible technical solutions. Figure 9,7 shows prices in dollars taken from the Oracle US price list (2019).

	Storage Server Price	**Oracle Premier Support for Systems (Annual)**	**Oracle Premier Support for Operating Systems (Annual)**
Oracle Database Appliance			
Oracle Database Appliance X8-2S	18.500	2.220	1.480
Oracle Database Appliance X8-2M	30.000	3.600	2.400
Oracle Database Appliance X8-2-HA	77.000	9.240	6.160
Exadata Database Server			
Exadata Storage Server X8M-2 Extended (XT)	16.500	1.980	1.320
Exadata Eighth Rack Storage Server X8M-2 High Capacity (HC)	37.000	4.440	2.960
Exadata Storage Server X8M-2 Extreme Flash (EF)	53.000	6.360	4.240
Exadata Storage Server X8M-2 High Capacity (HC)	53.000	6.360	4.240

Figure 9.7 – Example of Oracle Engineered Systems Price List (currency US $)

Oracle Database Appliance is geared towards small and medium-sized business, whereas Exadata is for big enterprises. Oracle Database Appliance is a complete database system in a single box. It is an appliance and contains storage, servers, operating system and Web-based monitoring tool (database control) within one box. It provides a ready-to-use, clustered database solution that includes both hardware and software optimized to work together. Oracle Exadata is a pre-configured combination of hardware and software that provides an infrastructure for running Oracle Database. It consists of two layers based on Sun servers: a database layer and a storage layer connected through a network; then, the application and the Web layers have to be implemented by other systems. Therefore, the estimate must consider other costs.

The second example concerns firewalls, namely network security systems that monitor incoming and outgoing traffic using a predefined set of security rules to allow or block unwanted events. The various types of firewalls are distinguished mainly by the throughput and the number of simultaneous sessions managed. In this example we refer to the firewall prices of a framework contract for Italian Public Administrations (see Figure 9.8).

Firewall Level	Intrusion Prevention throughput	Firewall throughput	VPN throughput	# of sessions	new sessions per second	Supplier 1	Supplier 2
base	> 300 Mbps	> 800 Mbps	> 300 Mbps	> 700.000 sessions	> 15.000 sessions	€ 485,10	€ 498,58
intermediate	> 1 Gbps	> 3 Gbps	> 700 Mbps	> 3 millions sessions	> 75.000 sessions	€ 2.951,99	€ 4.966,80
high	> 2 Gbps	> 10 Gbps	> 4 Gbps	> 4 millions sessions	> 140.000 sessions	€ 3.279,72	€ 5.349,60
top	> 5 Gbps	> 20 Gbps	> 8 Gbps	> 7 millions sessions	> 200.000 sessions	€ 10.088,70	€ 11.020,20

Figure 9.8 – Firewall prices from an Italian Public Administration framework contract

9.4 Costs of hardware – personal computers

Personal computers are the usual devices to allow the access of the users to the IS.
We distinguish three types of desktop personal computers:
- Entry-level: for simple office automation applications
- Mainstream: for excel, database, data visualization
- Power user: for electronic publishing, application development, large databases, data warehouses, high-resolution graphics, etc.

Recently, laptop (portable) personal computers are replacing desktop personal computers for both home usage and office usage. Laptop personal computers may be classified in two categories:

- Desktop replacement personal computers: adopted when the need to move the device is low;
- High-mobility personal computers: adopted when the job of the user requires a frequent moving of the device.

In the last years the costs of desktop and laptop personal computers have been quite stable. At parity of costs, personal computers offer increasing computing performance and improve other characteristics (e.g. minor weight and longer autonomy of the battery for laptop, minor volume and more ports for desktop). Moreover, the characteristics of monitors are larger size, higher resolution, etc.

Current (2019) costs of desktop and laptop personal computers are shown in Figure 9.9. Prices of personal computers shown in the table include the operating system Microsoft Windows and 36 months of warranty.

Device	Cost in euro
Entry level desktop PC	500
Mainstream desktop PC	600
Replacement laptop PC	650
High portability laptop PC	800
Monitor 19”	100

Figure 9.9 - Prices of personal computers

The costs of Power users personal computers vary a lot depending on specific configurations.

Discounts are:
- 10 units → 10-20%
- 100 units → 20-30%
- 1.000 units → 30-40%

It has to be noted that access to a company's information system is increasingly common through wireless tablets and smartphone devices. However, these devices do not generally replace personal computers, but rather they are combined with them, allowing access to the system on the move and outside the office. In case of adoption of this access mode, the costs of the new devices are added to those of personal computers and it is necessary to guarantee an adequate wireless network inside the offices.

9.5 - Costs of system software

We distinguish four types of system software:
- Operating system
- Middleware
- Office automation
- Application development environments

The most common operating systems for servers are UNIX (in the different releases developed by the hardware vendors), Microsoft Windows, Linux (either in the open source version or in the so-called distributions, e.g. Red Hat and Suse) and MacOS. Licenses for servers are released on the basis of the number of clients connected or of the number of CPUs of the server. The prices are very different, depending on the number of users and servers; in the simplest cases, we may assume a percentage on the value of hardware, from 15% to 25%.

See sites of providers for the variety of technological characteristics that influence the price (e.g. basic Linux Red Hat enterprise € 2.000 per year; entry Microsoft Windows Server € 8.000 for perpetual licence and 3 years software assurance).

Middleware makes it easier for software developers to implement repetitive functionalities such as:
- Database access, corresponding to Data Base Management Systems (DBMS)
- Communication and file transfer
- Input/output drivers

so that they can focus on the specific purpose of their application.

Prices for middleware vary a lot depending on the products, the number of operating system images, the number of users and the configurations (e.g. Microsoft SQL Server € 12.000 for perpetual licence per processor, Oracle Database Enterprise Edition € 30.000 for perpetual licence per processor).

The basic price of an office automation suite is around 400-500 €. Discounts in percentage are shown in Figure 9.10.

Units	Discount in percentage
10	15
100	25
1.000	35
10.000	40

Figure 9.10 - Discounts in percentage on office automation suites
for different orders of magnitude of supplies

Software development environments are specialized software applications that facilitate the production of application software. Typical examples of development environment functionalities are:
- Compilers, that translate software programs from high-level symbolic programming languages into machine language.
- Editors
- Pseudocode generators, which translate the program specification into intermediate code
- Application load simulators, executing the program with an artificial application load to check performance
- Testing - allowing to check for errors in the program, that are discovered executing the program with several types of data in input.
- Computer aided Software Engineering suites, that assist the analyst/programmer in the whole software production life cycle.

Prices in euros of previous software components are shown in Figure 9.11.

Component	Cost in euro
Language compiler	250
Application generator	10.000
Computer aided Software engineering	20.000

Figure 9.11 – Prices of components or software development environments

9.6 Cost of application software

The cost of application software is traditionally evaluated starting from the number of lines of code. We discuss now this parameter. Consider the following program, made of three instructions:

1. SUM = 0
2. For NUMBER = 1 to 100
3. SUM = SUM + NUMBER

The program has three lines of code, that in this case coincide with instructions; in case the program was the following,

1. SUM = 0
2. For NUMBER = 1 to 100

3. SUM =
SUM + NUMBER

The instructions remain three, whilst the lines of code are five. The first line of code is an assignment instruction that assigns 0 to the variable SUM. The second and third lines of code cause the iterative sum of numbers spanning between 1 to 100 with the value of SUM, and assign the result to the variable SUM, resulting at the end of the execution in the calculation of the sum of the first 100 integer numbers in the variable SUM.

The estimation of the cost of a program starting from the lines of code looks contradictory at first sight, since we cannot esteem the cost of the program unless we have produced it. So, usually the evaluation proceeds as follows.

1. Fragment and refine the requirement functionalities until you come to define modules for which there is previous knowledge on the size of the software
2. Quantify the number of lines of code of each module
3. Define the total size in terms of lines of code.
4. Evaluate from the market the effort and corresponding cost of a programmer to produce one line of code.
5. Evaluate the overall cost of the program.

Several concerns arise in the use of lines of code for application software evaluation. Consider again the problem of calculating the sum of the first 100 integer numbers. Besides the previous program we can calculate the sum with the program:

SUM = 100 * 101 / 2

Under this light, we can perform the same calculation respectively with three lines of code and one line of code! For this reason, research has pursued the objective to evaluate the cost of programs with methods that are independent from the number of lines of code.

Function Points (FP) are a unit of measurement of software dimensions, proposed by A. Albrecht (IBM) in 1975, which is associated with a calculation method (FP Analysis) published in 1983. FPs measure the size of a software application, deriving it from the number of functionalities it provides to users, where functionality means every single elementary procedure for transforming an input into an output. A function point measures a single functionality (not a line of code) provided by the program. The two programs:

1. SUM = 0
2. For NUMBER = 1 to 100
3. SUM = SUM + NUMBER

and

1. Cont = 100 * 101 / 2

correspond to the same number of function points. The advantages of function point over lines of code lie in the fact that function points:

- Are sufficiently objective; we have seen that both previous programs correspond to one functionality

- Allow for an earlier evaluation, with regard to the realization of the code
- Are more linked to "what to do" (analysis phase, e.g. BPMN) than to "how to do it" (coding phase)
- Are quite independent from the adopted programming languages.

Now we address a relevant issue: how to calculate the number of function points of a software application?

The standard methodology to calculate the number of function points starts from the analysis of input files and output files of the functions to be developed and their complexity in terms of number of fields in files, assigning a specific number of Function Points at each input and output. During the last years, many different methodologies have been developed for this step; the reader can find in-depth material on the Web.

From function points, we have now to determine the effort in person days. This effort changes with the programming language (e.g.: Cobol vs Java) and with the available developing environment, since some languages have high level primitives that are more powerful than others in coding, and some developing environments make software libraries available, thus requesting less effort. The effort also depends on other characteristics of the software to be developed: the level of reliability, the usability, the response time, etc.

The number of hours or days required to develop one Function Point determines the productivity, i.e. the number of function points developed per hour or day. Using high level languages, the productivity varies from 1 to 2 Function Points per day. In the following, we assume for Java language the productivity of 1,5 Function Points per day.

The unit price per Function Point is evaluated taking into account the Productivity and the hourly rate of specialists.

9.7 Cost of services

We now deal with cost of services, focusing on hardware, system software and application software maintenance, and management services.

<u>Hardware maintenance</u> - Hardware maintenance of devices can be expressed in terms of % of the initial cost; furthermore, it varies among devices, and it is higher when the device has mechanical components. See in Figure 9.12.

Device	**Percentage**
Server	6
PC	7
Printer	12
Local Area Network	10

Figure 9.12 – Yearly percentage of cost of hardware compared with the initial cost

System software maintenance – Includes both the upgrade of the software with new releases and the assistance in case of errors and other malfunctions. In most cases, the maintenance is included in the price for a first period (one or more years). The usual percentage of yearly cost of system software is shown in Figure 9.13. We assume no cost for office automation software.

Component	Percentage
Operating system	6
DBMS	20
Application generator	20

Figure 9.13 – Costs of system software

Application software maintenance - Overall costs of the different types of software maintenance introduced in Section 8.8 of Chapter 8 vary in time, see Figure 9.14 that concerns corrective and adaptive maintenance.

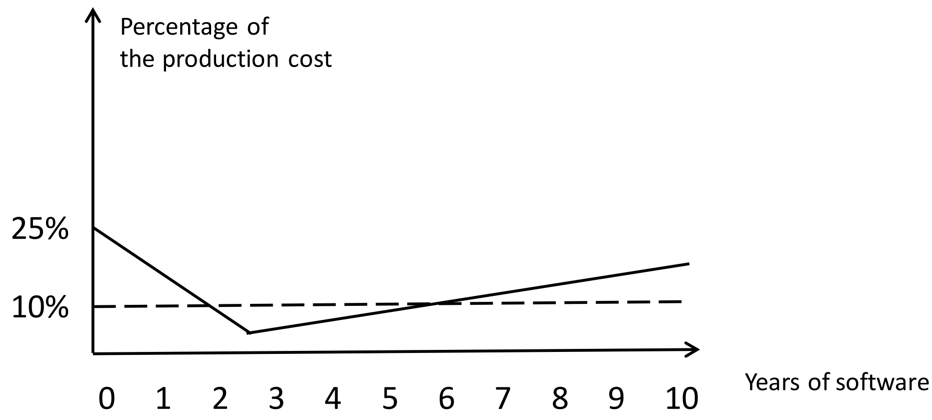


Figure 9.14 - Trends over time of maintenance costs per years after initial usage

In the first year, the costs of corrective maintenance and adaptive maintenance are predominant, since the debugging of software errors and request of users oriented to change some aspect of the interface are time-consuming activities. In the second and third year, costs tend to decrease, but - after the third year - costs tend to increase, due to several phenomena:

- the evolution of the system software might require changes and updates in order to assure the correct operation of the application;
- software gets old, even if no change is made on it; programmers tend to forget the meaning of choices made in coding,
- the documentation is rarely well curated, so it may become difficult or impossible for a new programmer to understand choices of original programmers involved
- the turnover of programmers is not accompanied by a serious and professional handover.

As a result, the trend of maintenance costs is the one shown in Figure 9.14. We can simplify the evaluation by assuming an average yearly cost of 10% for corrective and adaptive maintenance.

Moreover, the effort of evolutionary maintenance of application software may be relevant. Software requirements change over time, depending on the change of internal procedures or external events (e.g. changes in the rules governing the application sector of reference). The workload of evolutionary maintenance may be estimated in an average 20% of the development effort.

Network services - Include the costs to implement and manage the Local Area Network to ensure the linking of the different devices within the organization and the costs of the geographical network (Wide Area Network) to ensure the linking of different seats of the organization and the external connection.

With regard to network services, we deal only with the access to Wide Area Network services.

Different services of access may be required, distinguished in terms of technologies used (e.g. wired electrical, wired optical, wireless), access bandwidth and quality of service (bandwidth reservation, time of services, high reliability, etc.).

Figure 9.15 shows the monthly prices of different types of wired optical access services from the price list of a large contract of connectivity services of the Italian Public Administration.

Access bandwidth	**Monthly price in euro**
100 M	99,79
200 M	109,16
300 M	115,66
600 M	155,16
1 G	206,08
2,5 G	369,23
5 G	531,73
10 G	856,73

Figure 9.15 – Monthly prices of WAN access services

Management services - As for management services, we deal only with help desk services. The cost of help desk may be evaluated in terms of the number of users and corresponding PC based workplaces. Services can span from one operator for 30 PCs to one operator for 100 PCs. The total cost is obtained multiplying the number of operators by professional tariffs.

9.8 Pricing models in Cloud computing

Pricing models in Cloud computing are more flexible than traditional models, and every cloud provider has its own pricing scheme. In Figure 9.16 several pricing models are shown that can be adopted by a provider of cloud services; the numerousness and variety of such pricing models put in evidence the high flexibility in the procurement of cloud services in comparison with the traditional "on-premises" approach.

- Time based, pricing based on how long a service is used;
- Volume based, pricing based on the volume of a metric;
- Flat rate, a fixed tariff for a specified amount of time;
- Priority pricing, services are labeled and priced according to their priority;
- Edge pricing, calculation is done based on the "distance" between the service and the user;

- Responsive pricing, charging is activated only on service congestion;
- Session-oriented, based on the use given to the session;
- Usage-based, based on the use of the service for a period of time, e.g. a month.
- Content-based, based on the accessed content;
- Location-based, based on the access point of the user;
- Service type, based on the usage of the service;
- Free of charge, no charge is applied for the services;
- Pre-paid, the payment of the service is done in advance.
- Post-paid, the payment of the service is done after the use;
- Pay-per-use, customers only pay for what they use;
- Online, the accounting performed while the user makes use of a service.
- Offline, the accounting process is done after a service is used;

Figure 9.16 - Pricing models in the cloud

The adoption of cloud computing can yield numerous benefits.

- Primarily, cost reductions, as the procurement of cloud services requires lower investments for infrastructure, lower cost for maintenance, energy and physical space.
- In addition, cloud solutions allow for greater flexibility in the procurement of resources and easy scalability (increase or decrease); it is possible to request and obtain the service when needed and to easily manage application load changes.
- Moreover, it is possible to obtain increase of the processing power and of other resources (mass memory, network bandwidth, etc.) when needed, and the sizing of the resources can be carried out without necessarily referring to processing load peaks.

Having regard to the price lists from cloud computing service providers, we show a simple example of the economic benefits that can be obtained from the use of cloud computing.

Having regard to the price lists from cloud computing service providers, we show a simple example of a cloud computing solution (unitary prices are shown in Figure 9.17).

# CPU	RAM size	On-demand hourly price
4	8 GB	€ 0,104

Mass storage size	Monthly price
1 GB	€ 0,3

Figure 9.17 – In Cloud prices for computing resources

The formula to calculate the yearly cost of the computing resource is the following:

Yearly Computing resource cost = # hours per day * # days per year * hourly price
Monthly Mass storage cost = # months * # GB * Monthly price

Figure 9.18 shows the yearly cost for the usage of CPU/RAM for 24 hours per day for 365 days per year and the permanent usage of 500 GB of storage for 12 months.

CPU	RAM	STORAGE	Usage : 24 hours for 365 days	
			CPU yearly costs	Total yearly costs
4	8 GB	500 GB	911	2.711

Figure 9.18 – Examples of cloud computing yearly costs for CPUs and Storage

9.9 Cost evaluation case study

We now apply the above formulas and rules of thumb in a case study. Assume that an organization (see Figure 9.19) has three organizational structures located in nodes A, B and C:

- Node A has 300 users
- Node B has 100 users
- Node C has 50 users

A geographical network connects the three nodes and allows for the link of three offices and the Internet connection (bandwidths selected: 5 GB for Site A, 2,5 GB for Site B, 1 GB for Site C). Furthermore, 3.750 function points of software have to be developed in Java language. We assume that the devices are bought from the market and located on premises; we do not deal with cloud adoption.

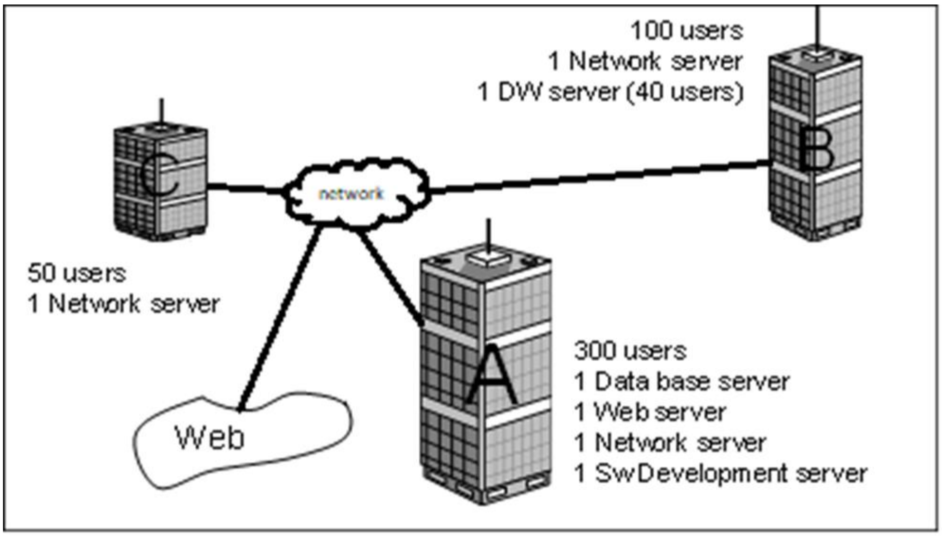


Figure 9.19 – The organization and related infrastructure considered in the case study

We have to consider three types of costs:

- Costs of purchasing (BUY) of components, namely servers, personal computers, printers and system software.
- Costs of software development (MAKE)
- Cost of operations for three years

leading to total costs for three years.

Costs of purchasing - As to the costs of servers, we evaluate in Figure 9.21 the costs per user, so for each server we have to specify the number of users at the node where the server is installed. At this point, we apply costs in Figure 9.5. Notice that we have to apply the corrections of section 9.3.2 for application loads different from OLTP. Finally, we apply a 20% discount, leading to the final price in Figure 9.20.

Site	Type	Users	Price per user	Total
A	DB	450	200	90.000
A	Network	300	100	30.000
A	Development	20	500	10.000
A	WWW	100	100	10.000
B	Network	100	150	15.000
B	DW	40	800	32.000
C	Network	50	200	10.000
Total				197.000
Final price with 20% discount				150.000

Figure 9.20 – Cost of servers

PCs can be acquired in the entry-level configuration with a standard monitor. As for printers, each employee is equipped with a personal low-cost laser printer, and each floor of the buildings is equipped with a fast printer: their cost can be easily estimated looking at a sales site, see Figure 9.21.

Type	Number	Cost	Total
Entry level PC	450	600	270.000
Laser printer	450	100	45.000
Fast printer	20	2.000	40.000
Total cost			355.000

Figure 9.21 - Costs of personal computers and printers

With regard to the system software, the cost of the operating system is evaluated as the 15% of the cost of servers, while the DBMS, the office automation and development environment costs are evaluated based on the number of users. See Figure 9.22.

Type	Parameter	Unitary cost	Total
Server OS	15%	150.000	22.500
DBMS	450 users	300	135.000
Office Automation	450 users	350	157.500
Development environment	10 users	10.000	100.000
Total			415.000

Figure 9.22 - Costs of system software

Costs of application software development - We apply the methodology based on function points, multiplying the number of function points by the unit price per Function Point, according to the Productivity and the Tariff per day, see Figure 9.23.

# of Function Points	3.750
Programming Language	Java
Productivity (FP per day)	1,5
Tariff per day	300
Unit price per function point	200
Total cost	750.000

Figure 9.23 - Costs of software development

Total costs of purchasing are shown in Figure 9.24.

Cost category	Cost
Servers	150.000
PCs	270.000
Printers	85.000
System Software	415.000
Total BUY	920.000
Software Development (MAKE)	750.000
Total	1.670.000

Figure 9.24 - Total cost of purchasing/development

Cost of operations for three years - Costs of operations per year refer to hardware maintenance, system and application software maintenance, geographical network services and management services. In the first three cases, we apply the percentages specified formerly in the chapter, leading to costs displayed in Figure 9.25, 9.26 and 9.27. The maintenance of Office Automation software is included in the licence price.

Type	%	Value	Total
Server	6	155.000	9.000
PC	7	270.000	19.000
Printer	13	85.000	10.000
Total			38.000

Figure 9.25 - Costs of hardware maintenance per year

Type	%	Value	Total
Operating System	6	22.500	1.500
DBMS	12	135.000	16.000
Office Automation	-	157.000	-
Development	20	100.000	20.000
Total			37.500

Figure 9.26 - Costs of system software and application software maintenance

Service	%	# Function Points	Yearly costs
Corrective and adaptive maintenance	10	3.750	75.000
Evolutionary maintenance	20	3.750	150.000
Total			225.000

Figure 9.27 - Costs of application software maintenance

Costs for geographical network are evaluated on the basis of the monthly prices shown in Figure 9.13 (rounded). Figure 9.28 shows.

Site	Access bandwidth	Monthly costs	# of months	Yearly costs
A	5 G	530	12	6.360
B	2,5 G	370	12	4.440
C	1 G	210	12	2.520
Total				13.500

Figure 9.28 - Costs of geographical network per year

Finally, costs of help desk are evaluated assuming one help desk operator per 50 users in the three nodes, see Figure 9.30.

Site	Number	Help desk personnel	Yearly salary	Total
A	300	6	30.000	180.000
B	100	2	30.000	60.000
C	50	1	30.000	30.000
Total				270.000

Figure 9.29 – Costs of help desk services per year

We may now calculate total cost of operations for three years, see Figure 9.30.

Type of cost	Cost per year	# of years	Total
HW maintenance	38.000	3	114.000
SS maintenance	37.500	3	112.500
Application sw maintenance	225.000	3	675.000
Network services	13.500	3	40.500
Help desk	270.000	3	810.000
Total	584.000	3	1.752.000

Figure 9.30 - Cost of operations for three years

Figure 9.31 summarizes the overall costs of the information system.

Type of cost	Cost per year	# of years	Total
Purchasing (BUY)	920.000	1	920.000
SW development (MAKE)	750.000	1	750.000
Operations	580.000	3	1.740.000
Total			3.410.000

Figure 9.31 - Total costs

Appendix 9.1 – Solutions to exercises

Solution to Exercise 9.1 - Seat reservation and ticket payment

Solution to Exercise 9.2 - Exam registration

Solution to Exercise 9.3
- Seat reservation and ticket payment from PCs – 5.000.000 per year
- Seat reservation and ticket payment from self-service - 10.000.000 per year
- Change of seat – 500.000 per year
- Ticket reimbursement - 100.000 per year
- On board personnel assignment – 500.000 per year

Appendix 9.2 – Home work

Exercise 9.4 - Create an excel sheet where you consider all the types of costs discussed in Chapter 9, in such a way that you can parametrically evaluate the costs of any information system, providing sizing data on the information system components and services, and without changing every time the sheet. Include in the excel sheet all the tables that have been used in Chapter 9 to determine costs. For servers, find in the Web data that are up-to-date, and mention the Web source of data. Self-evaluate your report on the basis of the completeness of elements considered and the degree of flexibility of the data in input. To test your excel file, evaluate costs of the information system defined in the following Figure 9.31.

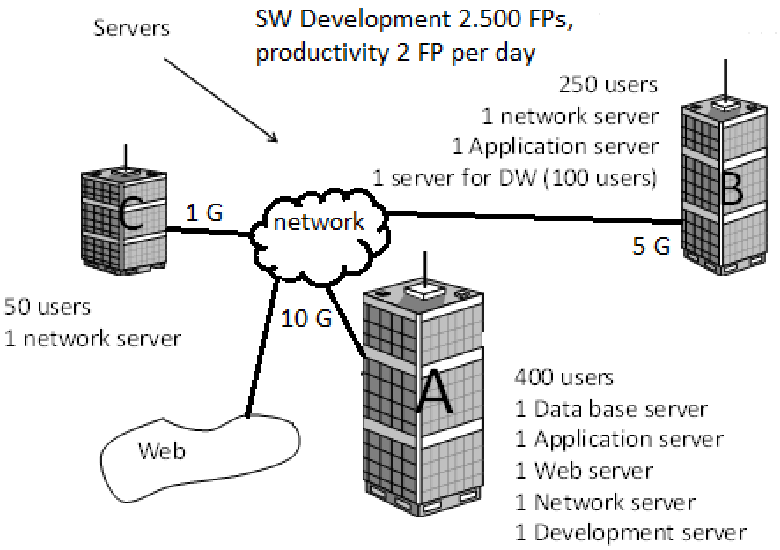


Figure 9.31 - The organization and related infrastructure considered in the case study

Appendix - Repositories of Conceptual Schemas

A1. Introduction to Data Governance

In this book I have tried to show that data and processes are equally relevant during the Information System life cycle. The number of data schemas and processes the result from the design activity can be huge. Therefore, we have to find a way to give a structure to plenty of data schemas and processes. In this Appendix we address such issue for data schemas, framing the problem in the more general topic of Data Governance.

Data governance can be defined as the formal orchestration of people, activities, and technology to enable an organization to leverage data as an enterprise asset. Several different issues related to data governance exist:

1. *Data quality*: the set of issues that allow to assess different dimensions of data quality (e.g., accuracy, completeness, currency, consistency) and to improve such dimensions by means of activities that may operate directly on data or else on processes that interchange or elaborate data. The interested reader may go in more depth on this issue reading [Batini and Scannapieco 2016].
2. *Data modeling*: the representation of classes of data in terms of a conceptual model, namely a model whose linguistic categories highlight the aspects related to the meaning of data, instead of their representation in a computer. To go in more depth on this issue read [Batini et al. 1984].
3. *Data integration*: The technologies that allow to query and access different independent databases as they were virtually a single, integrated database. To go in more depth on this issue you may read [Batini and Scannapieco 2016].
4. *Schema integration*: The process of harmonizing conceptual descriptions of data across heterogeneous databases. To go in more depth on this issue you may read [Batini et al. 1984].
5. *Data architecture governance*: The process that considers the overall architecture of data schemas, namely the representation of data in different databases of the organization, conceiving and maintaining on time an integrated representation of all the information asset of the organization.
6. *Data governance management*: The asset of responsibilities and activities and their collocation in the organization that allow to manage, monitor, govern, improve the quality and level of integration of data.

Although all of the above issues are relevant in data governance, for the reasons we said, in this Appendix we focus mostly on data architecture governance. We show the structure and internal organization of the Repository of schemas, the artifact that provides an integrated representation of all the information asset of the organization, in a way that can be understood by professionals and users together.

A2. The Repository of schemas as the integrated representation of the Information asset of an organization

Every organization, either a company or a public administration, has in its information system hundreds and sometimes thousands of databases that have been produced in the course of time. In order to reconstruct into a unitary conceptual representation the huge amount of database schemas, is to integrate all data schemas into a unique one. Unfortunately, when the number of schemas to be integrated is high, building the integrated schema becomes unfeasible, as Figure A.1 expresses in graphical/metaphorical terms.

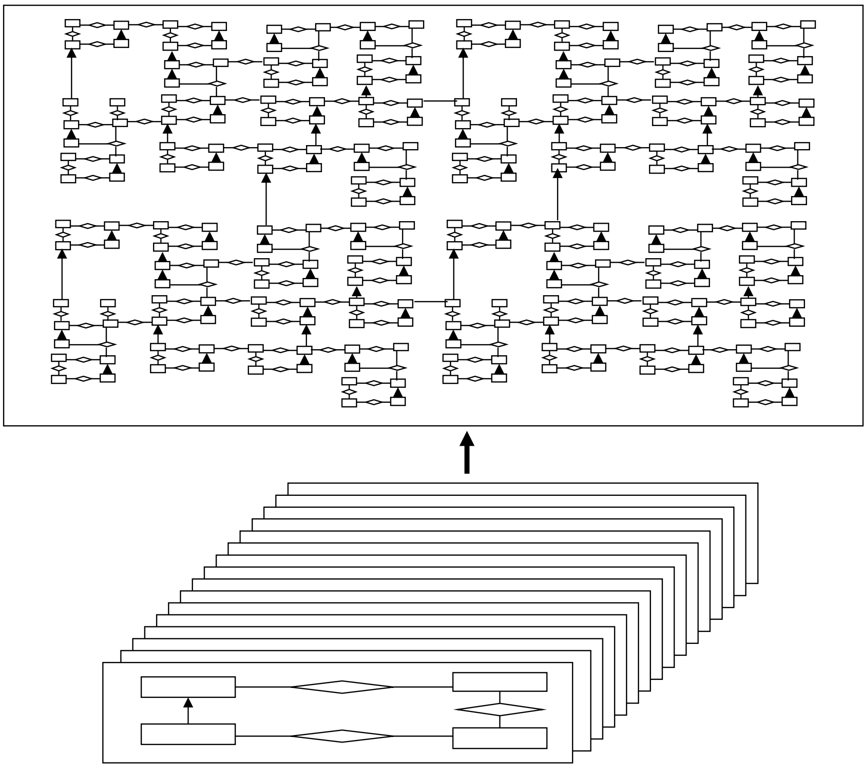


Figure A.1 - Integration in the large is unfeasible

To be able to feasibly integrate database schemas, we introduce the concept of *repository of schemas*. A repository of conceptual schemas is a set of conceptual schemas linked by relationships of *abstraction* and *integration*, see Figure A.2. More precisely:

- A conceptual schema ACS **abstraction** of a conceptual scheme CS, is a schema obtained by CS by collapsing groups of concepts in Cs with unique concepts in ACS; in this way, we obtain that ACS is a more compact description of CS. If we want to understand the information content of CS, we may first look at ACS and then look at the way in which CS is obtained as a transformation of ACS.
- An schema IS **integration** of two conceptual schemas CS1 and CS2, is a schema obtained from CS1 and CS2 in such a way that IS represents the whole information content represented by CS1 and CS2 separately; in this way we obtain that IS represents an information content that in the information system is represented by several schemas together.
- The inverse operations to abstraction and integration are **refinement** and **segmentation,** shown in Figure A.2.

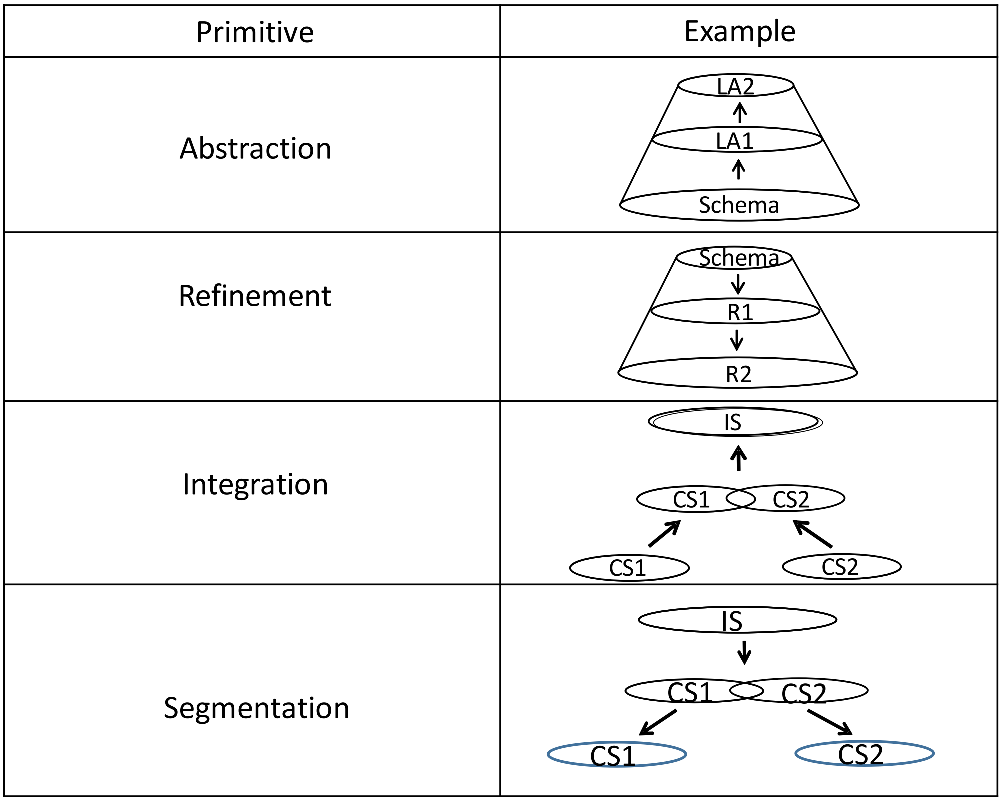


Figure A.2 - The four conceptual primitives used in the Repository of schemas

We now provide a case study whose goal is to construct a simple repository, starting from the three schemas shown in Figure A.3, referred to three of the most relevant databases in the Land department of an hypothetical Ministry of Finance. This department in several countries is in charge of the evaluation of real property, in order to determine direct and indirect tax assessment and to issue real estate certifications. Moreover, this department administers and records all state properties in regard to their financial affairs. Its responsibilities include the acquisition of new state properties; the disposal of properties when authorized; the care and supervision of state properties; and the maintenance of an inclusive inventory. The above activities are in charge of two offices, the general land office and the state property office.

Among the different databases managed by the Land Department, we focus on the following three:
1. General land office: Land database
2. General land office: Urban database
3. State property office: Mortgage registry database

Among them, in the following we consider the first three databases, whose schemas are described in Figure A.3. Notice that we represent only entities, relationships, IS-A relations, and generalizations; we do not represent attributes and names of relationships. Only in the case of urban schema we represent for the Urban real estate unit entity two attributes (in bold), the identifier Id and the attribute Address.

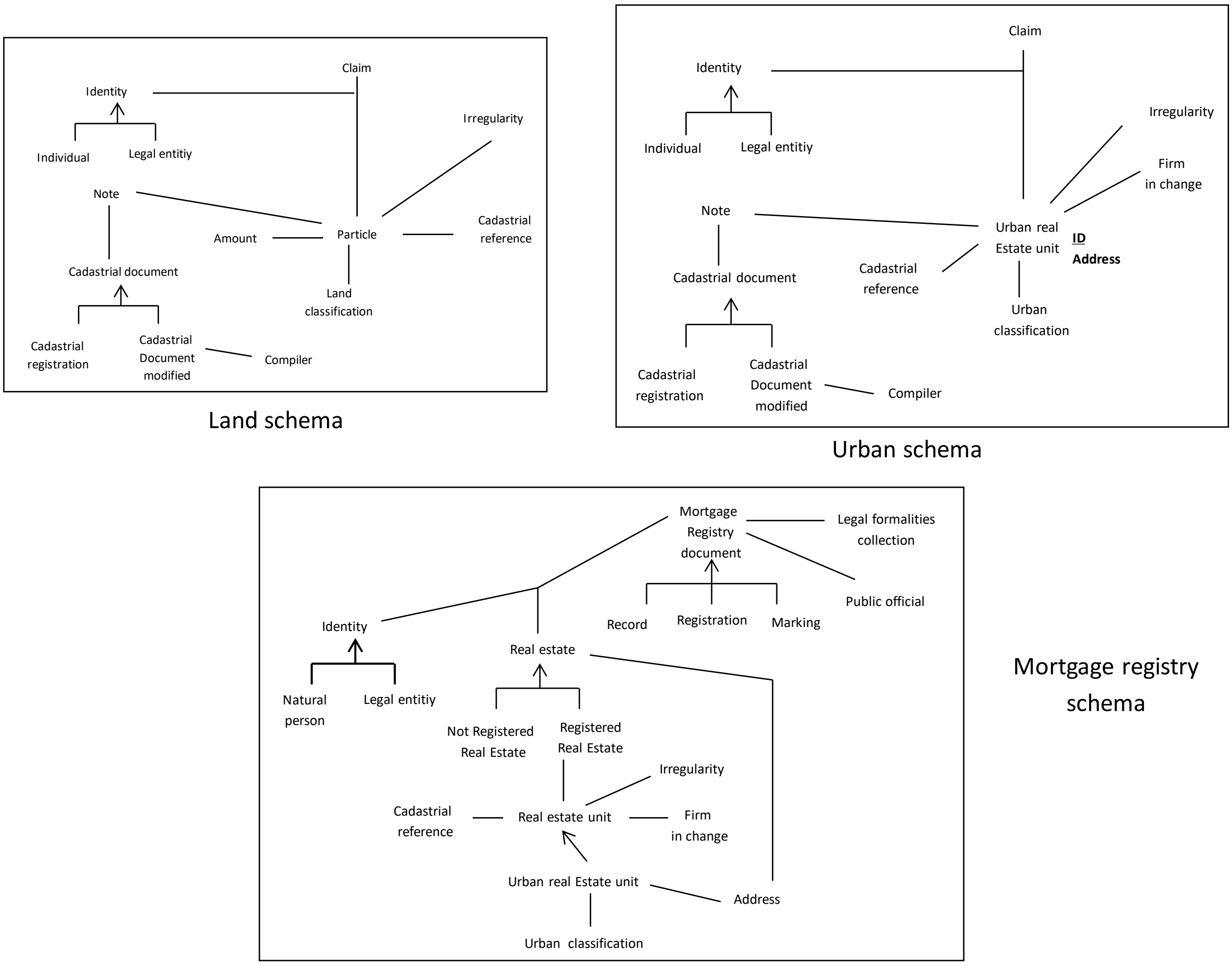


Figure A.3 - The three schemas in input to the repository

We investigate now separately the two main concept related to the Repository, namely Integration and Abstraction.

A3. Integration

Since databases in an organization Are usually created in the course of time by different designers, their conceptual schemas present usually redundancies (the same classes of observables, e.g. Students, are represented in several databases) and heterogeneities, e.g. the same class of observables is represented in two schemas with different names, leading to a synonymy. During integration redundancies and heterogeneities should be discovered and controlled, to give rise to the unitary representation consisting in the integrated schema.

The first activity to be performed in the integration step is conflict analysis, whose goal is to discover and solve heterogeneities among conceptual representations in different schemas. Two main activities may be distinguished:

1. *Name conflict analysis*, to establish naming correspondences for concepts. There are basically two sources of name conflicts: synonyms and homonyms. *Synonyms* occur when schema objects with different names represent the same concept while *homonyms* occur when the names are the

same, but different concepts are represented. Therefore, whenever synonyms or homonyms are detected, a concept renaming is required to solve the conflict.
2. *Structural conflict analysis*, to discover conflicts between different representations of the same concept. The use of an entity and an attribute to represent the same concept in two different schemas is a typical example of structural conflict.

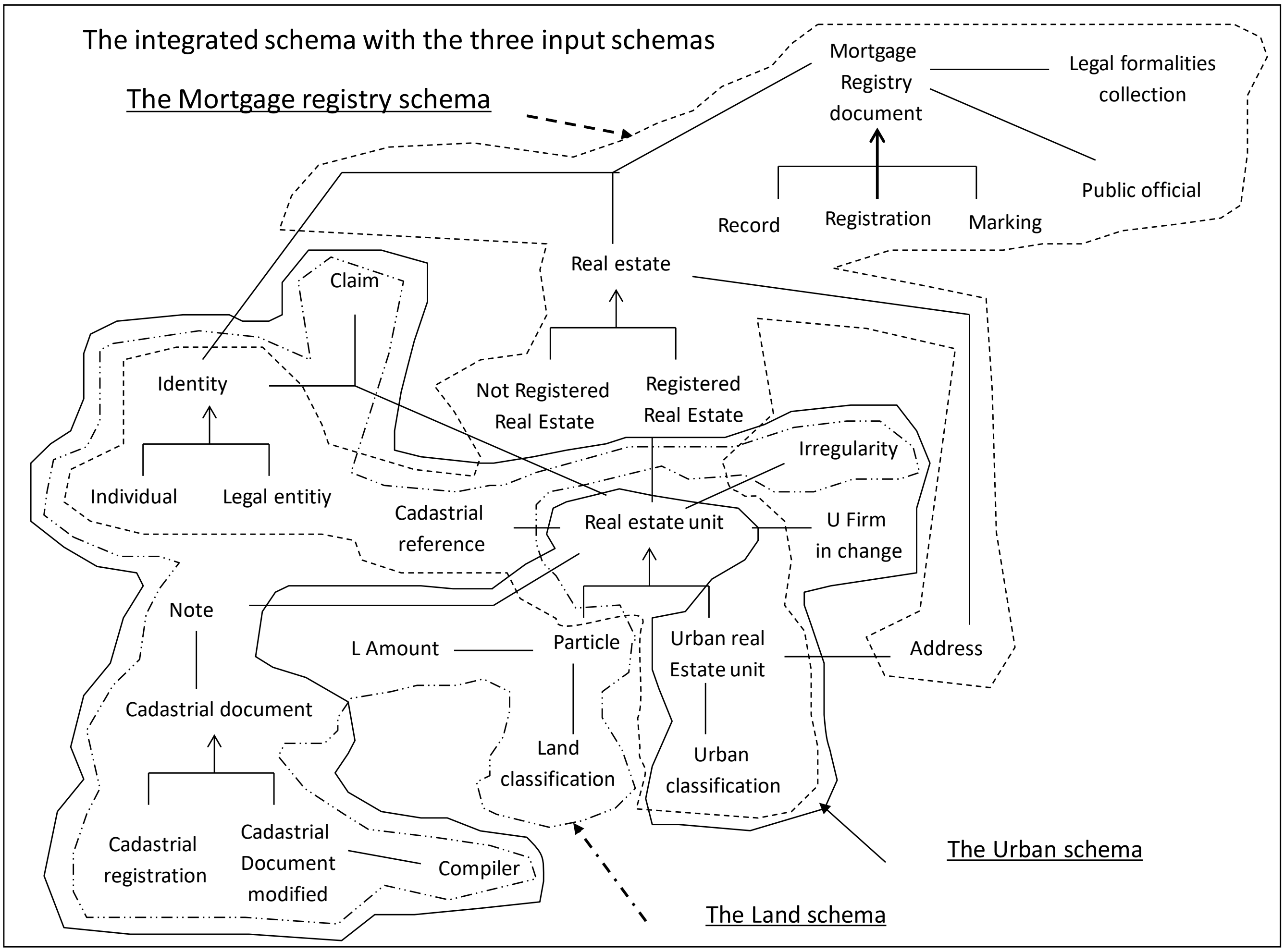


Figure A.4 – The integrated schema and the three input schemas

In our case study we have the following conflicts:
1. A synonym among Individual in the Urban and Land schemas and Natural person in the Mortgage schema; we choose the term Individual and consequently amend the Mortgage schema.
2. A structural conflict between the attribute Address in the Mortgage schema and the entity Address in the Urban schema; we choose for Address the entity type and consequently transform the attribute Address in the Mortgage schema into an entity.

We have now to detect *interschema properties*, that are properties (e.g. Is-a relationships) defined between concepts appearing in different schemas. Due to their cross-schema nature, these relationships could not be represented in the global schema and therefore require a specific analysis at this point.

In the case study, analyzing the Urban schema and the Land schema it is easy to discover that the entity Particle in the Land schema and Urban real estate unit in the Urban schema have many related concepts in common. Therefore, they are concepts in an Is-a relation with a common generalized concept, whose name is the Real estate Unit concept in the Mortgage Registry Schema.

The activity required at this point, called *schema merging*, is a simple superimposition of common concepts belonging to the amended schemas, thus building the integrated schema, see Figure A.4 in previous page. We represent in Figure A.5 the integration step, as a first building block of the repository we have in mind; the three columns on the right hand represent in the bottom row the three conceptual schemas from we started, and the schema on the left represents the result of their integration. We now move to abstraction.

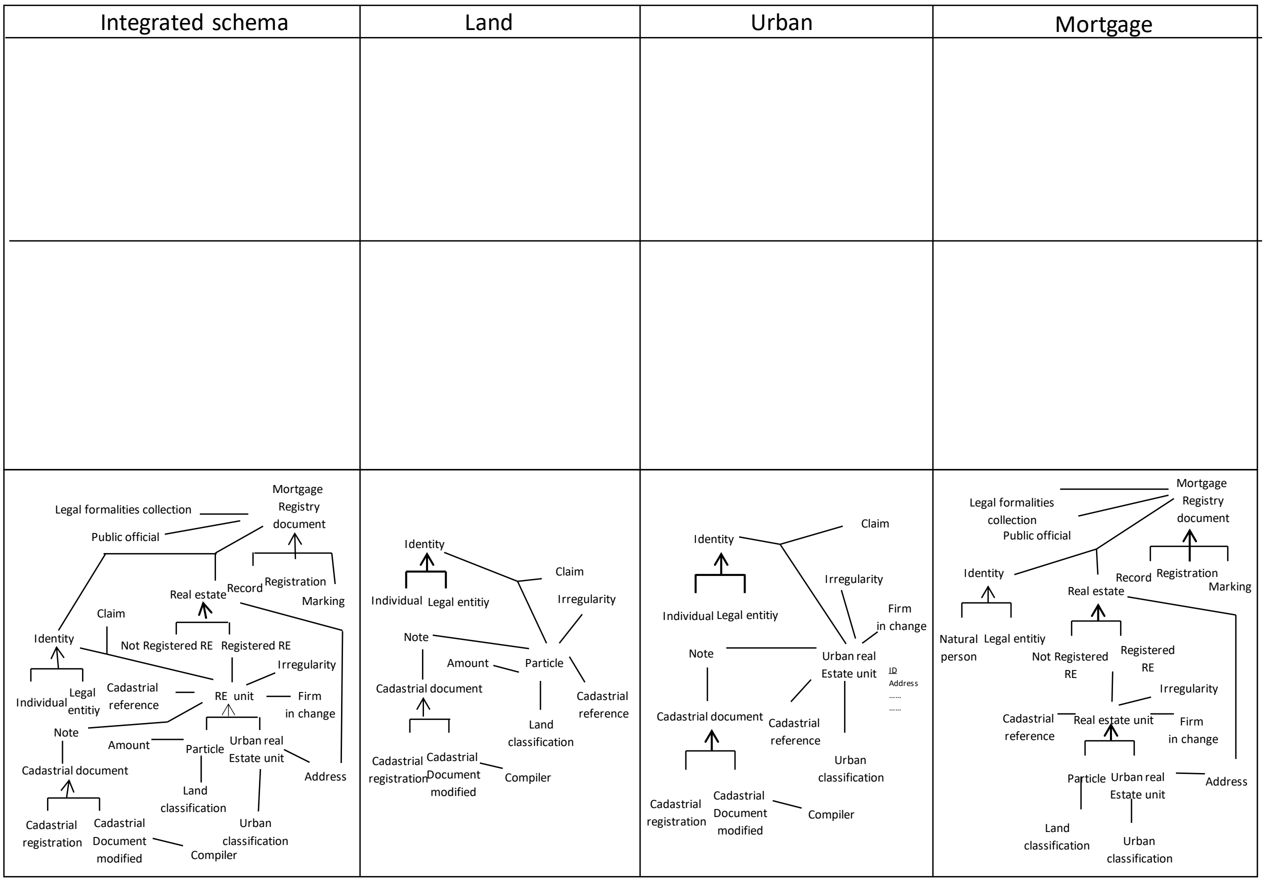


Figure A.5 – The Repository in the case study result of the integration of the three schemas

A4. Abstraction

The schema on the left of Figure A.5 looks too crowded of concept to be understood “on the fly”.

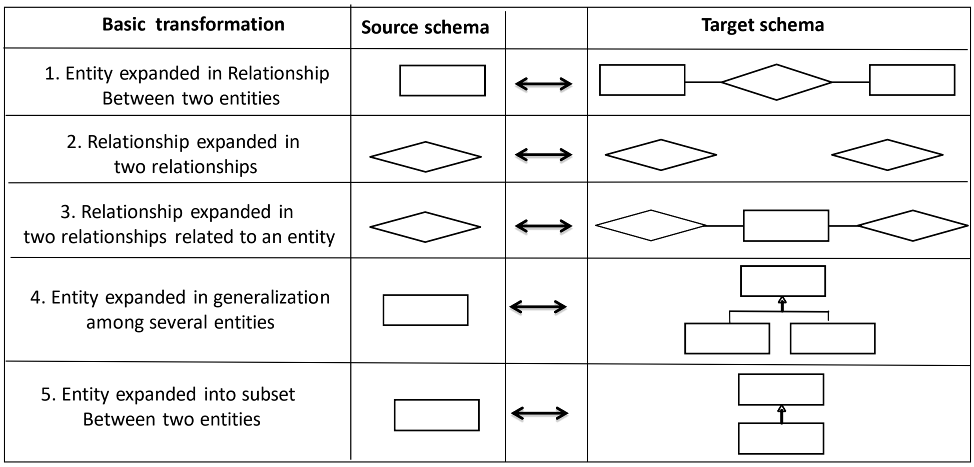


Figure A.6 – Refinement transformations in the top-down strategy

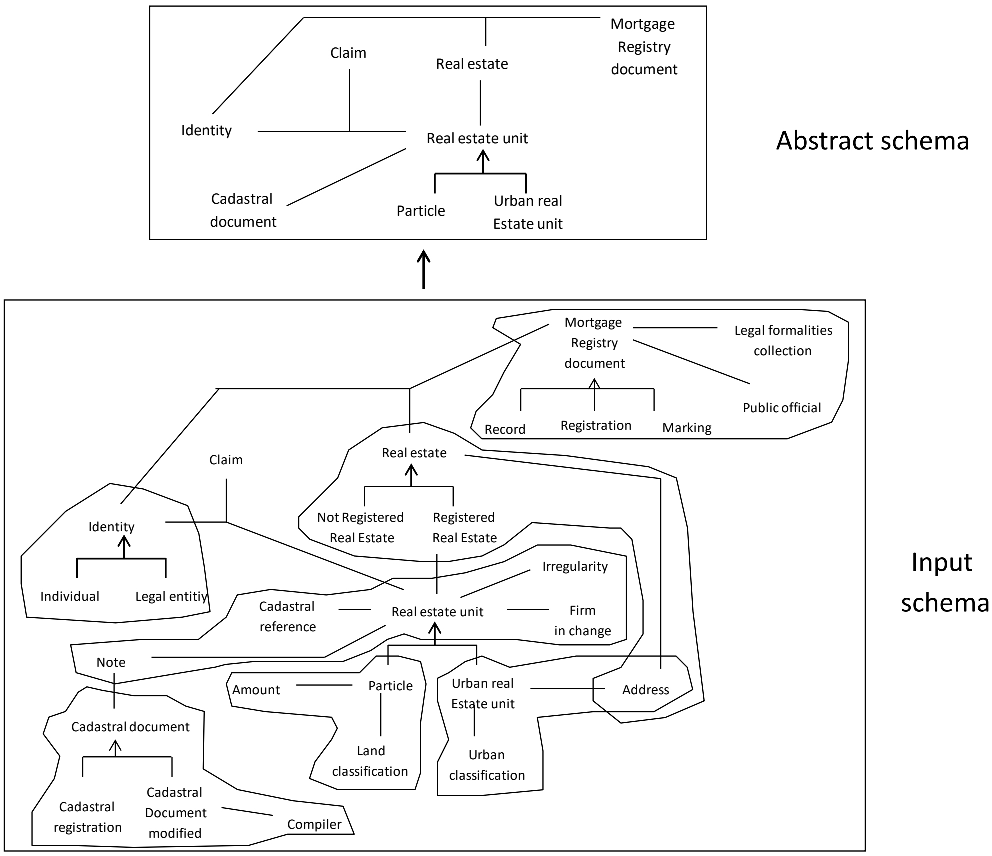


Figure A.7 – The abstraction operation in the case study

We apply an abstraction transformation to the integrated schema of Figure A.5, see figure A.6. Typical candidates for groups of concepts to be abstracted are generalization hierarchies, sometimes alone (as in the case of Identity (ancestor), Individual, Legal entity), other times together with less relevant concepts related to the generalization (as in the case of the group Cadastral document (ancestor), Cadastral registration and Cadastral document modified), to which the Compiler entity is added due to its unique relationship with the generalization.

Another group of concepts that can be abstracted concerns two relationships, among respectively: Urban Real Estate Unit and Urban Classification, and Urban Real Estate Unit and Address. In this case, the most relevant concept is clearly Urban Real Estate Unit that is chosen as the abstract concept in the upper level abstract schema. Notice in Figure A.6 that concepts that are clustered are shown in closed lines.

A5. The final Repository of conceptual schemas in the case study

In Figure A.8 we see the Repository populated with two successive abstractions.

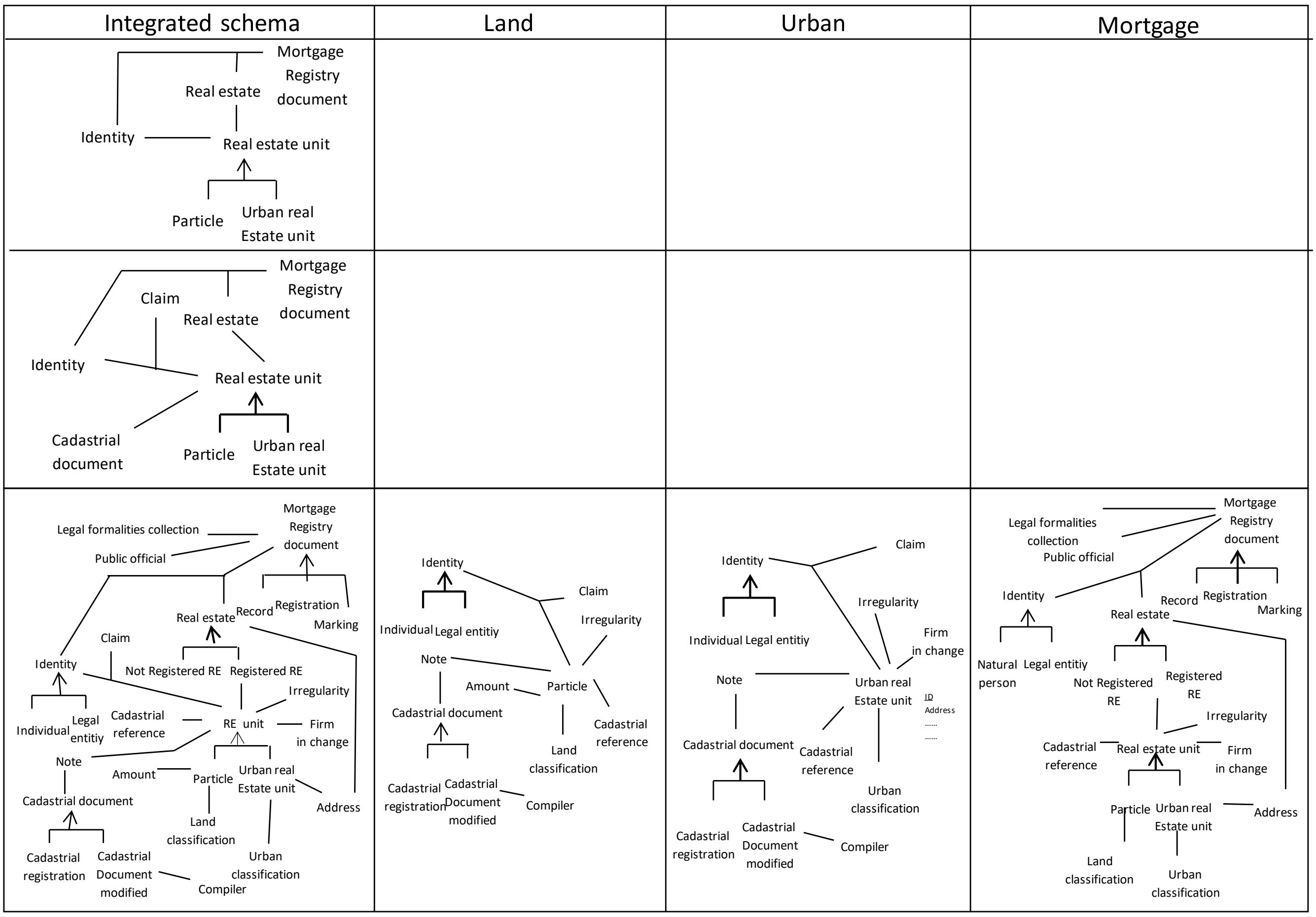


Figure A.8 – The repository of the case study where after integration we perform two subsequent abstractions

The Repository shown in Figure A.8 is a fragment of the wider Repository of the Italian tax system, shown in Figure A.8.

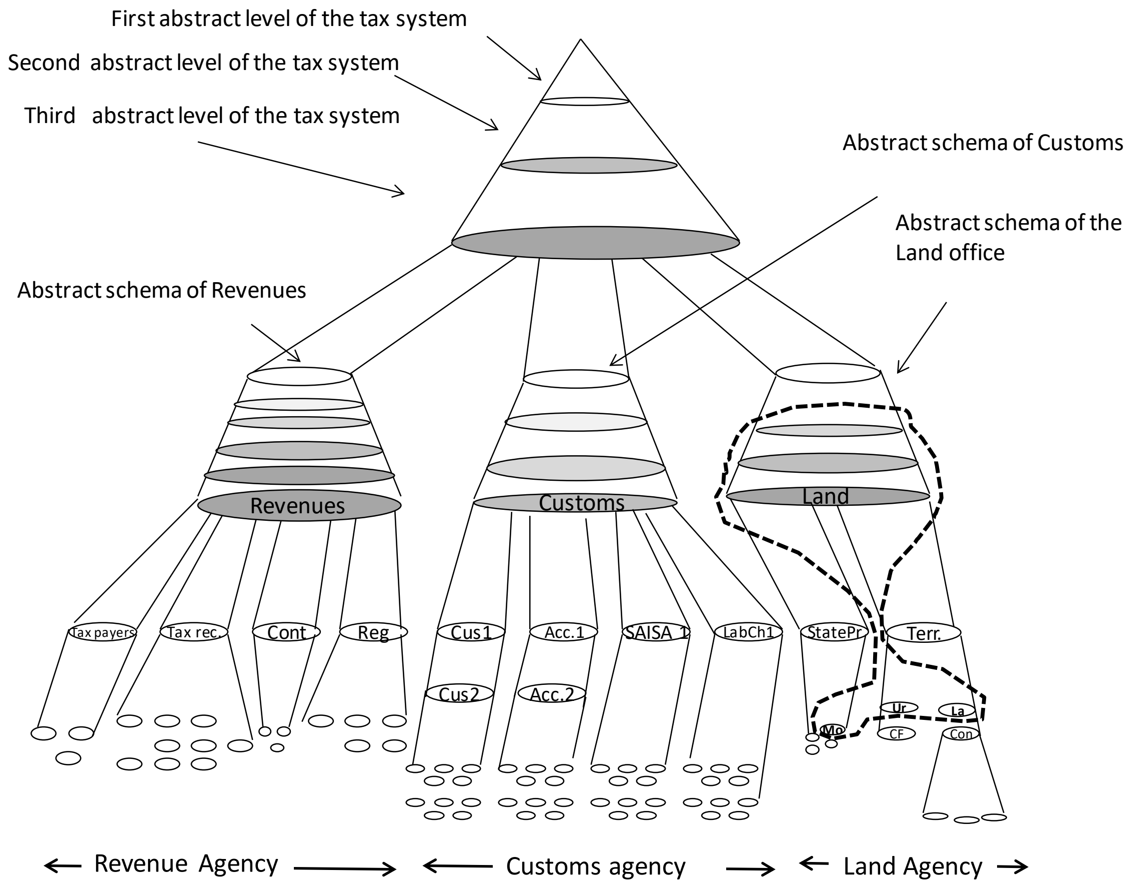


Figure A.8 - The repository of the italian tax system

You may see the three schemas of the case study identified by names Mo, Ur, and La, the integrated schema and the two abstract schemas represented with ellipses and enclosed in a dashed line. For a more comprehensive treatment of the concept of Repository of conceptual schemas, you may see [Viscusi, Batini 2010].

Conclusion of the book

Figure A.8 ends our book. Now you are able (I hope) to design an information system, assess the efficiency and effectiveness of an information system and processes as-is, conceive processes-to-be, model new software applications and data of interest in the organization, conceive technological architectures (hardware, input/output devices, software, data), evaluate the costs of operations, and build repositories of conceptual schemas.

You now are an **information system architect**, greetings from the author of this book, Carlo Batini.

References

C. Batini, S. Ceri, S. Navathe – Conceptual Database Design – Benjamin and Cummings, 1992

C. Batini, M. Scannapieco – Data and Information Quality: Dimensions, Principles and Techniques, Springer Verlag 2016.

G. Viscusi, C. Batini and M. Mecella – Information Systems for eGovernment, a Quality of Service perspective - Springer, 2010.