

Simona DINU

**Business Intelligence:
Architectures, Technologies and Implementations**



Simona DINU

**Business Intelligence:
Architectures, Technologies and Implementations**



Constanța 2024

Copyright © Editura NAUTICA, 2011

pentru prezenta ediție

Editura NAUTICA, 2011

Editură recunoscută de CNCIS

Str. Mircea cel Bătrân nr.104

900663 Constanța, România

tel.: +40-241-66.47.40

fax: +40-241-61.72.60

e-mail: info@cmu-edu.eu

Descrierea CIP a Bibliotecii Naționale a României

DINU, SIMONA

Business intelligence : architectures, technologies and implementations / Simona

Dinu. - Constanța : Nautica, 2024

Conține bibliografie

ISBN 978-606-681-180-4

004

338

Preface

The need for organizations to optimize their processes and achieve their organizational and strategic goals is a continuous challenge that they must take on. This desire forces them to make the most of business data to make the best decisions at the right time. But many companies have dispersed systems with their own data flows, coming from various sources and with different representation formats, which makes them difficult to manage, and ultimately this can result in the insufficiency of the current information management model. In this sense, there is a need to use some tools to make the identification, integration and analysis of these disparate business data more efficient, so that the information obtained from the data analysis is presented to the responsible people in the right format, easy to understand, accessible and when they need it, so they can make the right decisions quickly and efficiently. And the ability to make accurate and quick business decisions has become one of the keys to a company's success.

In this context, Business Intelligence (BI) systems appeared and evolved, offering a wide range of tools, architectures, methodologies and analysis technologies that facilitate the extraction, filtering, analysis and storage of large volumes of data generated in an organization, allowing access to the data stored by each of its functional areas and transforming this raw data into useful information and knowledge for complex decision-making processes.

This book offers a perspective on the field of Business Intelligence and explains the main concepts, methods and technologies used in BI processes, being useful to managers of organizations and specialists in the economic field concerned with improving the internal processes through the use of emerging technologies in the IT&C field, but also to all those who want to familiarize themselves with these new useful business tools to identify, discover, process and present business data in order to obtain valuable information.

The book is also useful as support for the "Business Intelligence" course held within the "Business Administration in Transport" master's program within the Faculty of Navigation Maritime Transport of the Maritime University of Constanta, but also for students and master's students from economic faculties in the country, for consolidating and deepening the profile knowledge. It is also addressed to researchers with concerns in the field of analysis of large data sets, but also to those with concerns in the field of mathematical modeling, the use of economic-mathematical methods and information technologies in the efficiency of various business processes.

This extensive study, which provides an overview of the Business Intelligence field, is presented gradually, being structured in the following chapters:

- the first chapter lays the theoretical and conceptual foundations of Business Intelligence in business processes, the evolution of Business Intelligence systems in recent years and their innovation trends. It also presents different data cleaning techniques, software tools used to ensure data quality and how to use them. Arguing the need for new approaches in information processing, with appropriate analytical tools, an overview of the Intelligent Document Processing technology and the key tools that sustain Intelligent Document Processing is presented next.
- the second chapter highlights the need and usefulness of modeling business processes to ensure the success of a Business Intelligence system. Also in this chapter, Business Process Management Systems are presented - systems that automate the management of business processes, but also the tools that

provide support and operational intelligence in the management of process flows. Process flows are visual ways of understanding what people do in their day-to-day activities and serve as a starting point for other studies, such as process improvement and cost estimation of different activities.

- the third chapter presents the Bizagi software. This application, which is designed for descriptive, analytical and execution modeling of business processes, enables the modeling of business flows, supports the processing of extensive documentation related to the process and allows the publication of all this documentation in various formats. The three components of the platform of this application are presented: Bizagi Modeler - the tool that allows the modeling of business processes, Bizagi Studio - the component that allows the automation of processes that are built with Bizagi Modeler and Bizagi Automation - the component that allows the execution of automatic business processes.

- chapter four offers an overview of Business Intelligence architectures and tools. The chapter first describes the main elements in the architecture of a BI platform, and then the Data Warehouse technology and the solutions for analyzing data stored in a Data Warehouse: the OLAP technologies. Next, an example of the implementation of an ETL solution with Pentaho Data Integration software is offered, and finally the Qlik Data Integration for Data Warehouse management application is described.

- chapter five presents the Microsoft Power BI integrative solution for Business Intelligence. The various functionalities of this software platform for business analysis are exemplified: the analytical tools that provide the possibility to access the data managed by a company, to merge the data sources, to analyze them efficiently and to convert them into graphs or reports, in order to present the information in various interactive, high-quality visualizations.

- in the sixth chapter, a data model is built using the Power BI Desktop working environment. Building a data model involves building visual representations of the connections between dataset tables, detailing the individual attributes contained in these data structures, and organizing the data into a structured format for analysis. Also, other functionalities of the application are presented: the creation of interactive visuals, of professional reports, advanced data filtering, etc.

- the last chapter is dedicated to the presentation of Big Data technology and the types of analysis dedicated to large data sets - descriptive, diagnostic, predictive and prescriptive analyses. Also, the connection between Business Intelligence, Big Data and Data Analytics is highlighted, and finally the advanced analytical features of the Microsoft Power BI platform are presented, which allow data analysis and the generation of information easily and quickly.

The book concludes with a rich bibliography useful to those who wish to expand their knowledge in the field of Business Intelligence.

Table of contents

Chapter I: Basic principles and elements of Business Intelligence	11
1.1 Business Intelligence: history and definitions.....	11
1.2 Why Business Intelligence?.....	11
1.3 Business Intelligence: transforming data into information and knowledge	12
1.4 Data quality.....	16
1.5 Tools and software for data quality	18
1.5.1 Data cleaning with OpenRefine.....	21
1.5.2 Operations and functions for cleaning data in Excel.....	31
1.6 Intelligent document processing.....	42
1.7 Key technologies behind Intelligent Document Processing.....	45
1.8 OCR tools to extract text from images or PDFs	48
1.8.1. Microsoft OneNote 2016	48
1.8.2. Cisdem PDF Converter OCR.....	51
1.8.3. i2OCR.com (Online).....	53
1.8.4 OnlineOCR.net (Online).....	55
1.8.5 OCR.Space.....	56
1.8.6 Google Docs	58
Chapter II. Business process modeling for Business Intelligence	61
2.1 Business processes - the key element in the analysis of any business.....	61
2.2 Process approach - important element in business modeling.....	64
2.3 Representation and modeling of business processes.....	66
2.3.1 Diagrams and design elements of the BPMN language	68
2.3.2 BPMN Best Practice rules.....	73
Chapter III. Bizagi software - a tool based on the standardized set of BPMN symbols used for modeling and automating business processes	75
3.1 Bizagi Modeler: application interface and structure.....	76
3.2 Modeling elements used in the Bizagi Modeler application	83

3.3 Bizagi Modeler operating mode.....	102
3.3.1 BPMN process documentation	102
3.3.2 Simulation in Bizagi Modeler.....	111
3.4 Automation of Business Processes: Process automation Wizard of Bizagi Studio software.....	114
3.4.1 Data model creation.....	115
3.4.2 Forms design	120
3.4.3 Definition of business rules that control the Process.....	122
3.4.4 Defining the participants for each activity.....	129
3.4.5 Integration Stage	131
3.4.6 Process execution.....	132
Chapter IV: Business Intelligence architectures and tools.....	133
4.1 Architecture of a Business Intelligence solution.....	133
4.1.1 Data Warehouse Technology	135
4.1.2 The multidimensional data model	136
4.1.3 Solutions for analyzing data stored in a Data Warehouse.....	139
4.1.4 Implementation architectures for OLAP techniques	143
4.2 Business Intelligence tools.....	144
4.3 Tools for implementing a Data Warehouse	151
4.3.1 ETL solutions with Pentaho Data Integration	151
4.3.2 Qlik Data Integration for Data Warehouse management	171
Chapter V. Microsoft Power BI for Business Intelligence.....	173
5.1 Microsoft Power BI tools	173
5.2 Key Features of Power BI Desktop.....	175
5.2.1 The Power BI Desktop user interface.....	175
5.2.2 Brief example of using the application by connecting to a data source and generating a report	177
5.3 Using the Power BI Desktop Query Editor.....	184
5.3.1 Power Query detailed views and transformations through contextual menus.....	186
5.3.2 Data cleaning and transformation operations in Power Query.....	187

5.3.3 Advanced data transformations.....	206
5.3.4 Power Query profiling tools for data analysis	226
Chapter VI. Designing a data model.....	229
6.1 Overview of Power BI for data modeling.....	229
6.2 Specific operations that use the Table Tools and Column Tools	232
6.3 Adding hierarchies and measures in the data model	239
6.4 Building relationships in the data model	245
6.5 Create interactive visuals.....	250
6.6 Visualize Data in a Matrix	262
6.7 Viewing reports on mobile devices.....	264
6.8 Filtering data in visuals	266
6.9 Using slicers.....	269
6.10 Using maps.....	271
Chapter VII. Types of analysis in the current business scenario – the connection between Business Intelligence, Big Data and Data Analytics.....	275
7.1 The new economic context and the impact of digital transformation on the business environment	275
7.2 Big Data: architectures, technologies and solutions	277
7.3 Advanced analytical features in Microsoft Power BI	287
Bibliography.....	293
Appendix 1.....	295
Appendix 2.....	305

Chapter I: Basic principles and elements of Business Intelligence

1.1 Business Intelligence: history and definitions

The concept of **Business Intelligence (BI)**, which is now considered critical in most companies, is not new. The concept has evolved over time and adapted to the evolution of the latest information and communication technologies, and to their implementation in companies, over the years, as well as to the most innovative business trends. In October 1958, Hans Peter Luhn, an IBM researcher at the time, coined the term in the article "A Business Intelligence System" as: "The ability to understand the relationships between facts presented in a way that guides actions toward a desired goal".

Today's Business Intelligence has evolved from decision support systems that began in the 1960s and developed in the mid-1980s. In the eighties, the concept of Data Warehouse appeared, as a data structuring model that provides a global, common and integrated view of an organization's data, with the following properties: stability, coherence, reliability and with basic information. One of the originators of the data warehouse concept is Ralph Kimball, who designed a methodology called "dimensional modeling" considered a standard in decision support systems.

It was not until 1989 that Gartner analyst, Howard Dresder, proposed a formal definition of the Business Intelligence concept: "Concepts and methods for improving business decisions through the use of evidence-based support systems". Since then, the concept has evolved by bringing together different technologies, methodologies and terms under its umbrella.

Today, Business Intelligence (BI) is a broad concept that includes computer technologies, specific business strategies, methodologies, and different analysis tools, which use a series of visual and mathematical models to identify, discover and process business data in order to obtain valuable information. Modern business intelligence and data analytics platforms have emerged to meet new organizational demands for accessibility, agility and deeper analytical insight. These modern platforms that provide users with detailed information about the state of the business are essentially supported by artificial intelligence technologies, machine learning and deep learning, data science, natural language processing and conversational voice technologies (such as bots, chatbots...) along with analysis of large volumes of data (Big Data). The information generated then leads to innovative ideas and provides new insights to support decision-making to improve the business and customer relationship.

BI technology has not stopped growing, with the use of data analysis now spreading across more and more companies and being increasingly appreciated by organizations. This increase in the popularity of data analysis in companies has led to an increase in the demand for analysts with this knowledge. This demand is growing a lot year by year, being a field of study with great projection for the future.

1.2 Why Business Intelligence?

Although many companies have adopted BI solutions and BI has become an established business tool, there are companies that are lagging behind their competitors because they underestimate the transformative power of BI. The ability to make accurate and quick business decisions has become one

of the keys to a company's success. However, traditional information systems (such as most management software, custom applications, and even the most sophisticated ERPs) typically have a very inflexible structure for this purpose.

Although the design of these systems is adapted to a greater or lesser extent to manage the company's data, it does not allow to obtain information from it, let alone extrapolate the knowledge stored in everyday databases. The main characteristics that limit these systems are:

- High rigidity when extracting data, so the user has to follow the predefined reports that were configured at the time of implementation and that do not always answer the current questions.
- Long response times, since complex data queries usually involve joining large operational tables, which translates into an uncomfortable waiting that hinders work flow.
- Lack of data integration, because many companies have several information systems, incorporated at different times, to solve different problems; this means that their databases are not integrated, which implies the existence of information islands.
- Absence of historical information because the data stored in the operational systems are designed to keep the company up to date, but do not allow the comparison of the current situation with a retrospective situation of years ago.

To overcome all these limitations, Business Intelligence relies on a set of analysis tools and methodologies that facilitate the extraction, filtering, analysis and storage of *data* generated in an organization to generate useful *information* and *knowledge* for complex decision-making processes.

1.3 Business Intelligence: transforming data into information and knowledge

One can note that in BI terminology, terms such as data, information, knowledge and intelligence are encountered. Although they might look similar, they have distinct meanings and play different roles. For example, even if data and information are words often used interchangeably, there is an important distinction between them.

Data: is the raw material that the computer processes to produce information: stored values, records at a transactional or operational level.

► **types of data:** - due to the processing limitations inherent in traditional data management systems, the predominant format accepted for processing within these systems was the text type format. Today, thanks to advances in technology, many forms of non-traditional data are available, such as images, audio, video, text, PDF, social media graphics, wearables, and more. This level of variety requires more work and analytical power to make it manageable.

- *Data in the form of text* exists in a wide variety of forms, with different levels and possibilities of aggregation: for example, there are financial data or accounting data, etc., which have an obvious numerical character and a possibility of direct processing, while the textual data requires an interpretation in a first stage, to be later processed (summed, centralized, clustered, etc.), and this interpretation that can be affected by the subjectivity of the operator.

- *Audio data* can be, for example, in the form of telephone conversations and voicemails.

- *Image data* in BMP, TIFF, PNG, GIF and JPEG formats, etc., can be photographic (for example images of company employees or marketing data that contains unstructured data in the form of product images) or non-photographic (graphs, diagrams) and are usually found together with other types of data formats.

- *Data in video format* requires more storage space and automatically involves decisions about how this form of data will be stored.

► **data provenance** - most businesses contain both internal and external data:

- *internal business data* (proprietary data related to employees, products, inventory, work in progress, project status reports, services or customers);

- *data from assessments and forecasts* (for example market data, competitor developments or the profit estimates that can be taken into account in the budget for next year);

- *external business data*: data from external, third-party or market sources (market indices, various financial records, specialized websites, applications, log files, devices, social media networks, etc.), which are relevant to the business.

► **data formats**:

- *structured data* (quantitative data - numbers and values): organized data, stored in tabular formats (e.g., excel sheets or SQL databases) that require less storage space.

Because structured data can be processed and analyzed by multiple IT tools, managers have more product options when using structured data. The specific and organized nature of structured data allows for easy manipulation and querying of that data. They can be processed quickly by using a relational database using Structured Query Language (SQL) - the programming language used to manage structured data.

The benefits of using structured data include the ease with which they can be processed, the simplicity with which searches can be performed within these data and the ease with which patterns and relationships that may exist between data can be identified.

- *unstructured data* (qualitative data - text files, audio and video files, etc.): data that is stored in its native format, being available in many file formats: e-mail messages, telephone conversations, posts on social networks, reports, presentations or forms that do not have a defined standard structure, chats, data from sensors, etc. These data collections are created both internally, but they also come from the company's external environment and subsequently introduced into the company.

Unstructured data that do not have a predefined data model and that, consequently, cannot be processed and analyzed by conventional data tools and methods. These types of data are best managed in non-relational databases (NoSQL), which require more space.

The advantages of unstructured data include:

- the possibility of a greater number of use cases, because the purpose of this data is adaptable.
- the company has more data to extract information from, which generates a greater variety of file formats in the database, because the data that can be stored is not limited by a specific format.
- not needing a predefined format for this data, it results in the possibility of being collected quickly and easily.

Compared to structured data, both have potential for use in the cloud, but structured data allows for less storage space and unstructured data requires more. Additionally, structured data can be used by the average business user, but unstructured data requires data science skills to derive accurate business intelligence.

- *semi-structured data*: data stored in JSON, CSV or XML files and which do not have a predefined data model and is more complex than structured data, yet easier to store than unstructured data.

Semi-structured data are those types of data that are unstructured, but which also have metadata that identifies certain characteristics: a structure that can help to identify them (such as, for example, the date of creation or the name of the author). Metadata contains enough information to allow data to be cataloged, searched, and analyzed more efficiently than strictly unstructured data.

An example of where semi-structured data comes from is websites, which contain a series of images that are categorized by file name and format, which may or may not give an indication of the "data" contained in the image.

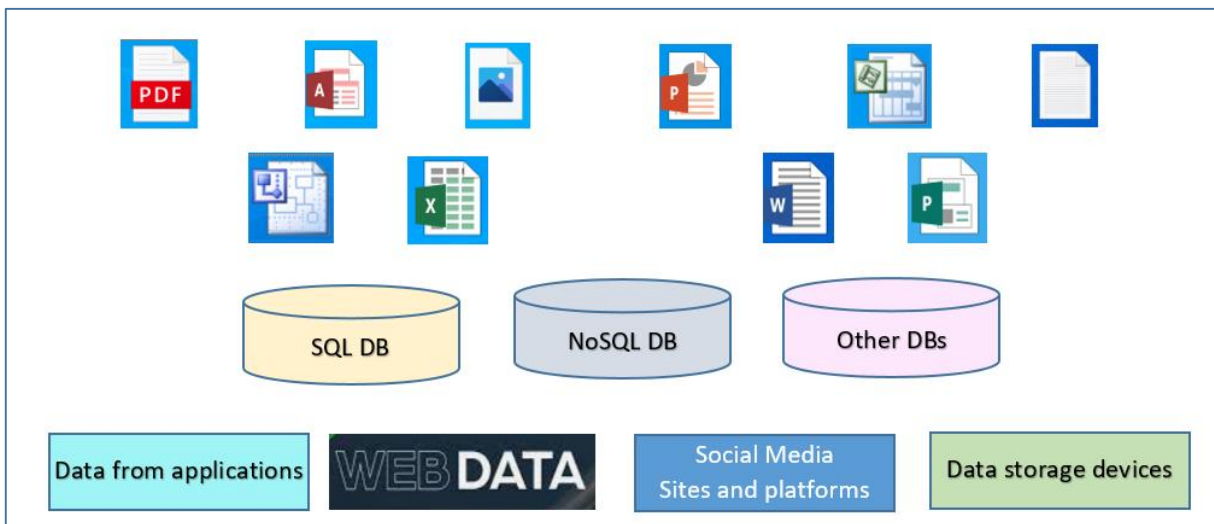


Figure 1.1: Data sources and formats

Data collected by organizations, relating to their employees, customers and suppliers, must be further processed, organized, structured or presented in a certain context to make it useful to the organization and to be considered information.

Information: can be defined as a set of processed data that has meaning (relevance, purpose and context), and that is useful to those who have to make decisions, by reducing their uncertainty.

Information is therefore the result of collecting and systematizing data in a way that generates relationships between data elements; in other words, data has the ability to produce information when context and meaning are added to it using various, mostly computer-based, techniques. Thus, information is relevant and meaningful data that has a well-defined meaning in a certain context.

Thus, it can be appreciated that business intelligence is the transformation of data into information, after it is analyzed and inserted in a certain context. Business intelligence involves the use of solutions developed with analytical technology that enables the transformation of data stored in files, databases or in large deposits into information that helps the various levels of an organization.

Business intelligence systems are primarily based on structured, quantitative information, usually organized in a database. But, these systems also handle unstructured information, for example analyzing e-mails and web pages through text extraction methods.

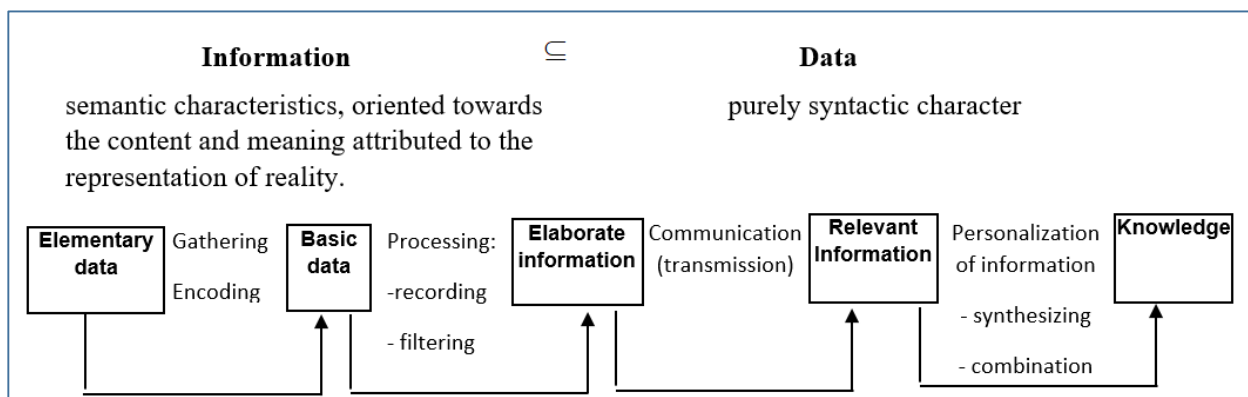


Figure 1.2: Turning data into information and knowledge

Knowledge: data in raw form does not provide any benefit to the end user. To decide on a certain action, knowledge is required. Knowledge is a mixture of experience, values, information and know-how that serves as a framework for incorporating new experiences. Knowledge is a step beyond information and involves an "understanding of information based on recognized patterns in a way that provides insight into the information" [Loshin, 2013].

In organizations, knowledge is often found not only in documents or data stores, but also in organizational routines, processes, practices, and standards. Knowledge represents an essential resource of an enterprise, being in fact the most important resource that determines the company's competitive advantage.

As a resource of an organization, knowledge has certain characteristic features that distinguish it from traditional resources [Szymczyk & El Emary, 2023]:

- *dominance* - refers to the role and strategic importance of knowledge within the organization: the effective use of knowledge supports the competitive position of enterprises
- *inexhaustibility* - knowledge, unlike other resources, does not run out, but on the contrary, its value increases with use.
- *simultaneity* – refers to the possibility of knowledge being used simultaneously by several people in different places in the enterprise or by several organizations and people in many places at the same time.

- *non-linearity* - there is no directly proportional relationship between the number of knowledge resources held and the results obtained: in different organizations, the same amount of knowledge can produce effects of a totally different magnitude.

- *incommensurability* - knowledge is often informal in nature, e.g. experience, intuition, organizational culture. It follows that knowledge is a difficult resource to measure, although, of course, attempts are made to quantify it.

Insights represent the ability to synthesize and derive decisions and potential actions based on a deep understanding of data, information and knowledge. Through learning and logical thinking one can identify trends, patterns, correlations and understand more about the problem.

Identifying the source of the data, determining how the data is collected, cleaned and structured, and then identifying the procedures needed to transform the data into a unified, meaningful format for optimal use by business intelligence systems are vital to business and to support organizations in making decisions for the future. Optimally, knowledge can be extracted from data through the active application of mathematical models and mathematical optimization algorithms. The adoption of mathematical models increases the efficiency of the decision-making process and leads to a deeper understanding of the business process analyzed to achieve strategic objectives. These objectives can be formulated through so-called key performance indicators (KPIs), which allow the measurement of business performance.

Another advantage is the fact that a mathematical model developed for a certain decision-making problem is applicable to other subsequent situations to solve similar problems.

1.4 Data quality

According to [Manjunath et al., 2010] one of the main reasons for the failure of Business Intelligence projects is the poor quality of data from databases and data warehouses. Such a system contains unstructured transactional data, which comes from different sources, having their own formats for storage (some being compatible, but others not) and which must be transformed and then loaded.

One of the main concerns of a company focuses on the quality of the data to be processed (financial, personal data, resources, money, sales, etc.) and which represent essential inputs of the company. The poor quality of the data from which significant information will later be extracted has an impact on the companies' business: they can affect the quality of the company's processes (which can generate higher internal costs), they can alter the quality of the products and services offered (which can endanger its reputation), but it can also generate security risks. Therefore, data quality should be considered a key element in the data inputs used in Business Intelligence analytics.

The evaluation of the qualitative level of the data is carried out by evaluating the attributes of the data residing in files, in databases or in large warehouses (Data Marts or Data Warehouses) managed at the company level. For this purpose, a control/a set of controls should be established to locate the errors in the data and not allow their erroneous loading. Regardless of whether they are performed manually or automatically, the checks must take into account different levels of detail and varying time periods, verifying that the uploaded data coincides with those of the original data sources;

According to the expert Richard Wang, from Total Data Quality Management Massachusetts Technology Institute [Wang & Strong, 1996], data should be analyzed according to their intrinsic, contextual, representational and accessibility aspects:

► The attributes related to the concept of **intrinsic data quality** are: accuracy, objectivity, uniqueness, credibility and reputation of the data.

- **data accuracy**: defines how much that data effectively, correctly and without errors represents something from the real world, compared to a certain reference value.

- **data objectivity**: defines the extent to which the data express the facts in a way that is free from bias, prejudice or partiality. In the economic context, the objectivity of the data is important, because there are situations in which the economic data are adjusted, the data from social statistics are exaggerated or undersized to alter the reality of the facts.

- **data uniqueness**: refers to the ability to uniquely refer to a particular entity given a particular context. This means that entity has no duplication and there would normally be a key attribute that allows it to be identified.

This concept is associated with the fact that certain file or database entities must be unique and not at risk of duplication and misuse.

- **data credibility**: defines the extent to which the data in question represents something that can be used as a credible and risk-free piece of information.

- **data reputation**: defines both the extent to which data are trusted (in terms of its source or content), but also how much that data, when viewed externally, can improve or damage the reputation of the area, company, project it serves or use.

► The attributes related to the concept of **contextual data quality** are: relevance, value-added, timeliness, completeness, appropriate amount of data.

- **data relevance** (importance in a given context of the data): the extent to which the data is applicable and useful for its particular activity.

- **value-added**: defines the extent to which data is important, beneficial, offers advantages from its use and can add value to the company's business.

For example, data converted into marketable information (as long as confidentiality is ensured) can be made available by selling it for use by customers and thus become an excellent source of business.

- **timeliness**: the extent to which the data is requested and generated at the necessary frequency and its age is appropriate for the respective activity.

- **data completeness**: the extent to which the data are sufficiently broad, complete and without gaps, serving the purpose of the respective activity.

- **data in the appropriate amount**: the extent to which data is stored in the amount desired and necessary to perform the desired analysis.

► The attributes related to the concept of **representational data quality** are: interpretability, ease of understanding, representational consistency, referential integrity, concise representation and ease of operation.

- **data interpretability:** the extent to which the language, symbols and units in which the data were produced or written are appropriate and the definitions of the data are clear, so that they can be interpreted correctly, in a context that allows the application of rational analysis, leading to correct conclusions.

- **ease of understanding:** the extent to which the data is clear, unambiguous and in a friendly, easy-to-understand form that allows it to be easily understood without the need for experts. Also, the representation and visualization in any graphic form that facilitates their interpretation.

- **representational consistency:** the extent to which the data elements are presented in a syntactically and semantically correct format and are compatible with previous data.

- **referential integrity:** a concept widely applied in the database world, characterized by the consistency of values between two data that have been defined in different entities in order to define an implicit form of relationship between them.

- **concise representation:** the extent to which data is represented in a succinct, compact and to-the-point way, but without losing its informational attributes.

Concise representation involves avoiding detailed, extensive, and tedious data, as seen in some text or descriptive fields.

- **ease of operation:** the extent to which data is represented in a way that allows easy work with it: update, move, aggregate, customize.

This attribute is of great importance with the increase in the volume of unstructured data from images, XML files, audio, etc.

► The **accessibility quality** refers both to aspects of data security and confidentiality: how data is secured through access, encryption protocols and services, etc., but also aspects of data access: the extent to which data is available or easily and quickly retrieved.

1.5 Tools and software for data quality

It represents technologies used by companies to identify and correct data deficiencies, to generate standardized data, which contain very few mistakes or errors, the ultimate goal being to provide support to end-user organizations in operational business processes and decision-making.

The Gartner company, specialized in IT research, defines several data quality characteristics that providers must offer in their solutions and tools: such as profiling and visualization, parsing and standardization, cleansing, matching, enrichment and monitoring.

► **Profiling and data visualization:**

Data profiling tools provide intuitive ways to examine data for accuracy and completeness.

Dedicated applications must provide functionality for attribute analysis (eg minimum, maximum, mean, percentiles, frequency distribution or other types of descriptive statistics) and dependency analysis (between tables and between files).

The instruments must present their results in tabular or graphical form. The results must be stored, with possibilities for trend analysis by comparing several stored series.

► **Parsing and standardization:**

Parsing and standardization tools must allow the identification and extraction of textual components from a wide spectrum of data types, such as names, addresses, e-mails and other related information for the purpose of their analysis based on rules that check whether the data follows a specific model.

For example, parsing tools locate and identify individual elements of information in data sources and isolate them in destination files. Such as separating the full name into surname, first name, or address into: street, number, floor, etc.

Standardization tools apply conversion routines to transform values into defined and consistent formats by applying standardization procedures and defined by business rules. For example: replacing the diminutive names with the corresponding names or standardizing the addressing formulas.

► **Verification and Validation:**

In order to obtain accurate and error-free data, the data received must be verified to ensure that it exactly matches the source it came from. Verification is mainly done by checking the format, by comparing them with the original source, by checking with other time periods and sources of statistics, by checking data that seems highly implausible or checking for duplicates

Data validation is a process used to compare a set of data against a set of business rules to determine whether it conforms to the data requirements of the business (with applicable standards, rules and conventions): whether the data is accurate, complete and reasonable. There may be rules for checking formats, integrity, reasonableness, identifying anomalies or different types of errors, and evaluating data by experts in the field.

► **Matching and deduplication:**

Matching and deduplication tools must allow data to be matched within and between defined sources. Matching involves comparing, identifying, or merging related records in two or more data sets. The matching functionality should not be limited to specific data types and domains, nor limited in terms of the number of attributes that can be considered in matching scenarios.

Data matching techniques can be used to identify and consolidate duplicate content, ensuring a unique and accurate view of the data.

Among the most used matching algorithms are:

- *Levenshtein Distance Algorithm*: is used to measure the similarity between two strings. The Levenshtein distance is calculated as the minimum number of insertions, replacements, or deletions needed to transform a string to another string.

For example, let $s_1 = \text{"core"}$, $s_2 = \text{"corn"}$.

$LD(s_1, s_2) = 1$, as “e” in string s_1 has to be replaced with “n” in string s_2 to make them similar.

- *Jaro Similarity* is another measure of similarity between two strings, which is calculated using the following formula:

$$JS = \begin{cases} 0, & \text{if } m = 0 \\ \frac{1}{3} \left(\frac{m}{|s_1|} + \frac{m}{|s_2|} + \frac{m-t}{m} \right), & \text{if } m \neq 0 \end{cases} \quad (1.1)$$

where:

m is the number of matching characters

t is half the number of transpositions

$|s_1|$ and $|s_2|$ are the lengths of the two strings we are comparing, strings s_1 and s_2 respectively.

For example, let $s_1 = \text{"extern"}$, $s_2 = \text{"xeetrn"}$.

Both the strings have 6 matching characters, but the characters do not have the same order: the number of characters that are not in order is 4, so the number of transpositions is 2.

Therefore, Jaro similarity is calculated as:

$$\text{Jaro Similarity} = (1/3) * \{ (6/6) + (6/6) + (6-2)/6 \} = 0.8$$

- *Jaccard similarity* measures the similarity between the values of two or more attributes in a dataset and is defined as the cardinality of the intersection of the defined sets divided by the cardinality of the union of them. It can only be applied to finite sample sets.

If A and B are two data attributes, the Jaccard distance can be computed by using the formula:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \quad (1.2)$$

For example, let attribute A has values $\{U, V, T, X\}$ and attribute B has values $\{J, U, V, P, Q, R\}$, then Jaccard similarity is calculated as:

$$J(A, B) = \{U, V\} / \{U, V, T, X, J, P, Q, R\} = 2/8 = 0.25$$

So, in this example, based on the Jaccard distance computation, attributes A and B have 25% similarity. It can be seen that the more similar the attributes, the higher the percentage of similarity.

► **Monitoring:**

Monitoring tools must provide continuous access and evaluation mechanisms to track the behavior of the data, ensuring that the data is suitable for the purpose of the analysis and that it will continue to respect the defined business rules.

► **Enrichment:**

The data management solution must enable internal information to be combined with additional data sets from internal and external sources to expand business understanding. Although the company has useful data, this technique brings benefits, being possible to obtain richer records about products, services and customers, etc.

► **Data cleaning:**

Data cleanup tools must provide rules for modifying data by dealing with syntax (format) and semantics (content of values) to ensure compliance with defined business rules.

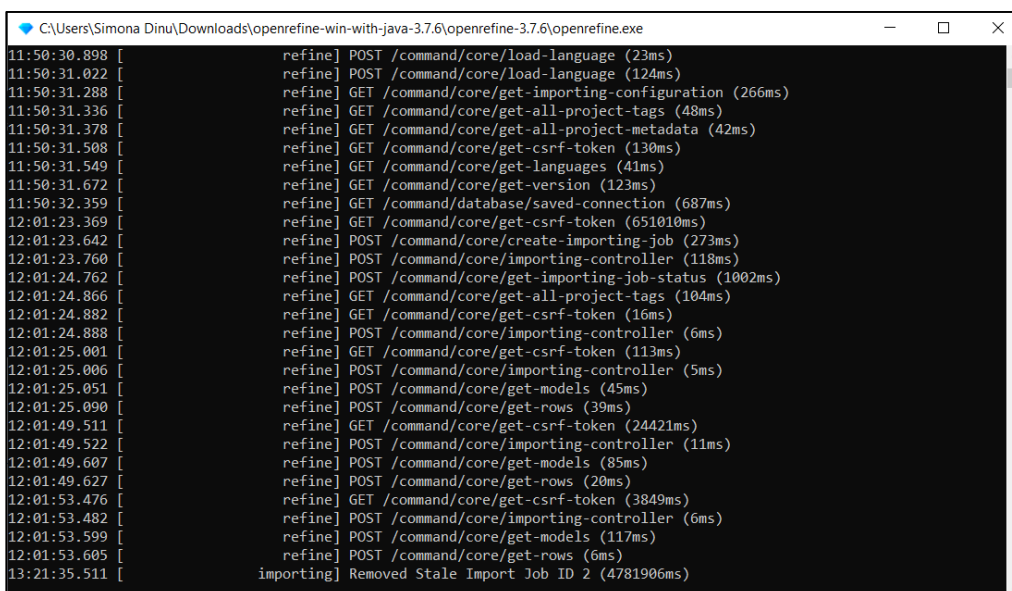
There are several commercial tools on the market dedicated to cleaning data, such as: OpenRefine, Oracle Enterprise Data Quality, RingLead, SAS Data Quality, Data Wrangler, Melissa Clean Suite, SCARE, Informatica, Cleanix, Tibco Clarity, NADEEF, Data Ladder, etc.

1.5.1 Data cleaning with OpenRefine

■ **OpenRefine** is a tool for cleaning and transforming data that works from the browser (it works locally, the data is processed locally and it does not have a cloud version, so the data is not sent to any online/cloud service), which ensures the safety of processing sensitive data through this tool. The application uses "Google Refine Expression Language" (GREL) to transform the data.

Originally known as Google Refine, it is a powerful tool that allows quick analysis of data contained in files. Detects data quality issues such as: duplicate data, missing data, variations in input data, or inconsistent data. It supports files of different formats (XLS, XSLX, XML, TSV, CSV, *SV, JSON, RDF or Google data documents). It is useful for exploring large data sets, allowing to clean and transform them from one format to another, but also to extend the data set through requests to external web services. It also allows UNDO/REDO actions in all operations.

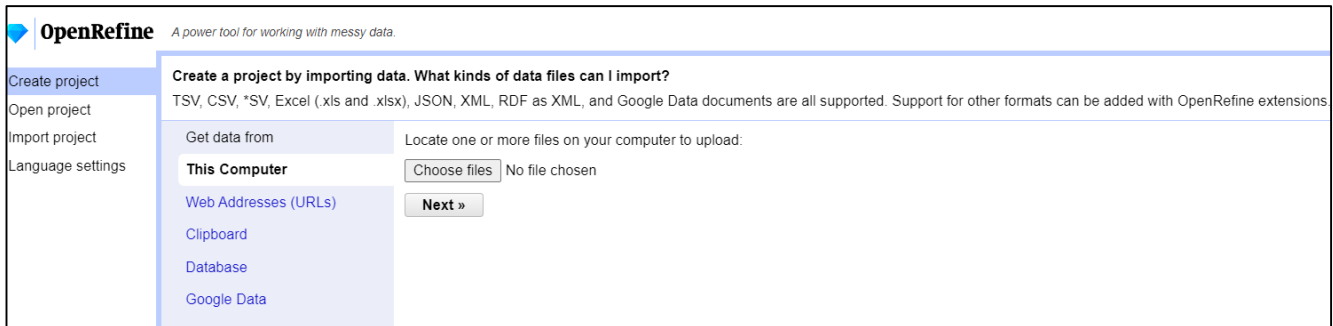
When running, a terminal will open that starts a local server that will ensure that the program works:



```
C:\Users\Simona Dinu\Downloads\openrefine-win-with-java-3.7.6\openrefine-3.7.6\openrefine.exe
11:50:30.898 [ refine] POST /command/core/load-language (23ms)
11:50:31.022 [ refine] POST /command/core/load-language (124ms)
11:50:31.288 [ refine] GET /command/core/get-importing-configuration (266ms)
11:50:31.336 [ refine] GET /command/core/get-all-project-tags (48ms)
11:50:31.378 [ refine] GET /command/core/get-all-project-metadata (42ms)
11:50:31.508 [ refine] GET /command/core/get-csrf-token (130ms)
11:50:31.549 [ refine] GET /command/core/get-languages (41ms)
11:50:31.672 [ refine] GET /command/core/get-version (123ms)
11:50:32.359 [ refine] GET /command/database/saved-connection (687ms)
12:01:23.369 [ refine] GET /command/core/get-csrf-token (651010ms)
12:01:23.642 [ refine] POST /command/core/create-importing-job (273ms)
12:01:23.760 [ refine] POST /command/core/importing-controller (118ms)
12:01:24.762 [ refine] POST /command/core/get-importing-job-status (1002ms)
12:01:24.866 [ refine] GET /command/core/get-all-project-tags (104ms)
12:01:24.882 [ refine] GET /command/core/get-csrf-token (16ms)
12:01:24.888 [ refine] POST /command/core/importing-controller (6ms)
12:01:25.001 [ refine] GET /command/core/get-csrf-token (113ms)
12:01:25.006 [ refine] POST /command/core/importing-controller (5ms)
12:01:25.051 [ refine] POST /command/core/get-models (45ms)
12:01:25.090 [ refine] POST /command/core/get-rows (39ms)
12:01:49.511 [ refine] GET /command/core/get-csrf-token (24421ms)
12:01:49.522 [ refine] POST /command/core/importing-controller (11ms)
12:01:49.607 [ refine] POST /command/core/get-models (85ms)
12:01:49.627 [ refine] POST /command/core/get-rows (20ms)
12:01:53.476 [ refine] GET /command/core/get-csrf-token (3849ms)
12:01:53.482 [ refine] POST /command/core/importing-controller (6ms)
12:01:53.599 [ refine] POST /command/core/get-models (117ms)
12:01:53.605 [ refine] POST /command/core/get-rows (6ms)
13:21:35.511 [ importing] Removed Stale Import Job ID 2 (4781906ms)
```

Closing the terminal window causes the server to stop, therefore OpenRefine suspends its functions, even if it remains open in the browser.

After installation on the computer, **the OpenRefine application is launched:**



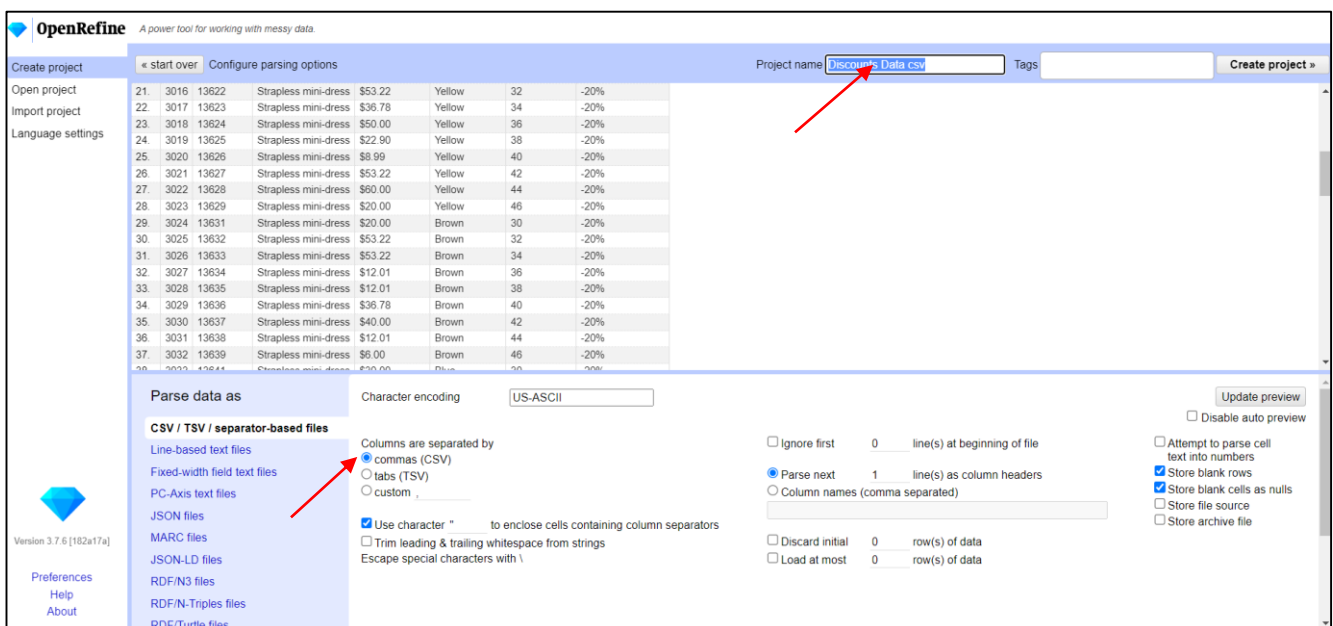
To create a new project:

1. **Create project** (from the left menu of the application) → **Choose files** → **Browse** → **select the data file** that will be used in the project → **Next**.


In the example below, the file data can be viewed and changes can be made, if necessary. For example, one can **change the name of the project** in the **Project name** field or **add a tag** in the **Tags** field for future reference to this project.

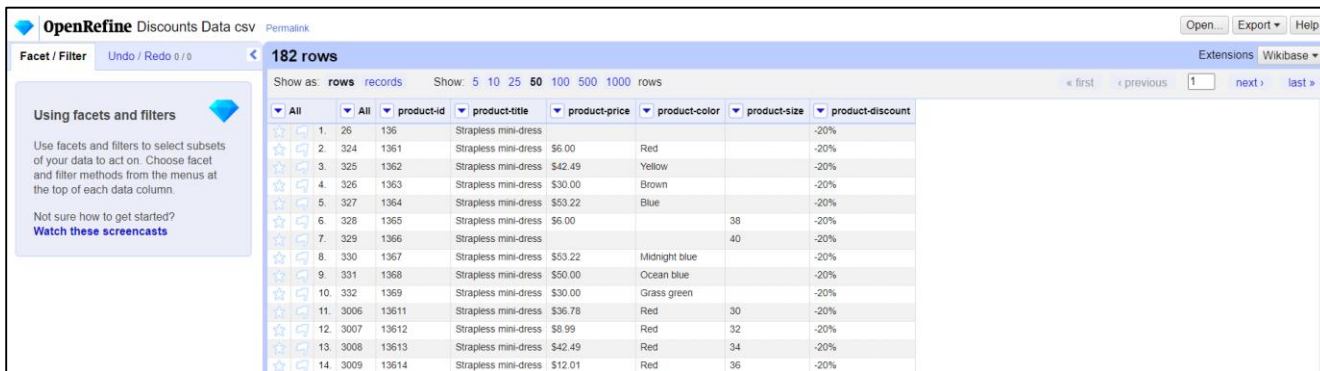
2. In the window that opens, the file data can be viewed and changes can be made, if necessary. For example, one can **change the name of the project** in the **Project name** field or **add a tag** in the **Tags** field for future reference to this project.

Note: In the imported .csv file, the **commas (CSV)** option must be ticked in the **Columns are separated by** section:



3. Click **Create project**: **Create project »**

4. In the project window, one can see that each column head has a button  that opens a menu of options with changes that can be applied to the respective column.



Also, in the menu on the left side of the window, there are two other options: **Facet/Filter** and respectively **Undo/Redo**.


Facet is an option that allows the user to group the existing values in a column for filtering and editing the values in the cells.

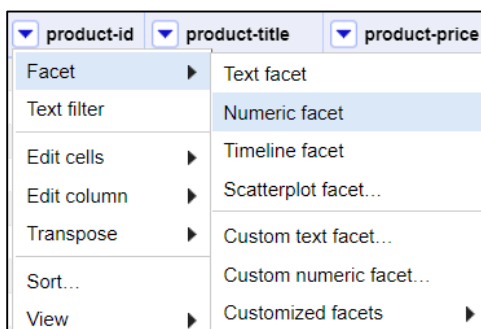
Undo/Redo allows the return to the existing data before the respective processing.

The **next** and **last** buttons  allow navigation through the pages with records.

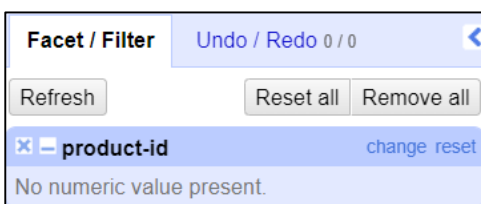
► **To check that there are no inconsistencies in the data:**

For example, in the **product-id** column (field that defines a unique identifier):

1. Click on the button  corresponding to the column → **Facet** → **Numeric facet**:

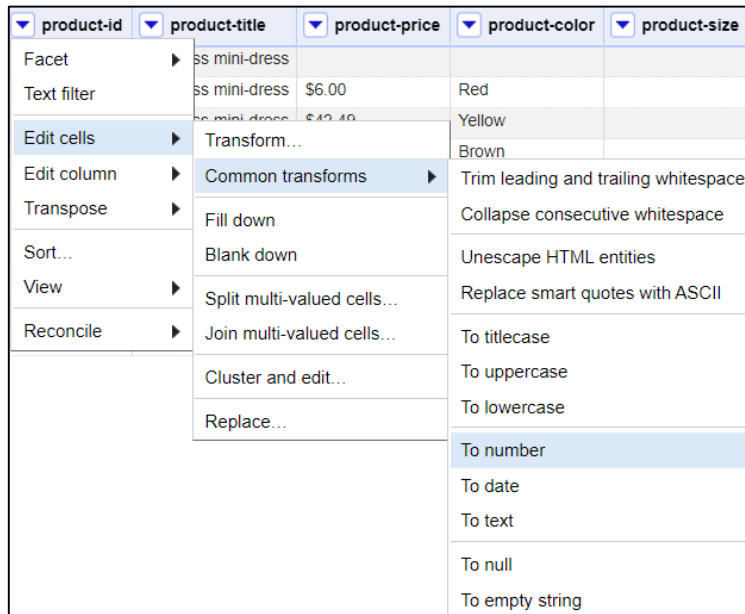


2. In the menu on the left of the page appears the name of the field: **product-id** and the information: **”No numeric value present”**:

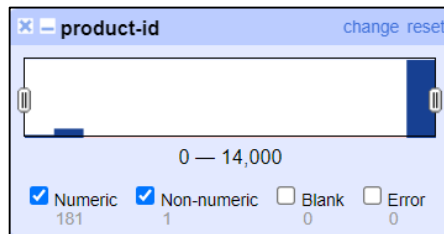


Note: When importing data, the OpenRefine application will save all data as text data by default.

3. To fix this issue: Click on the button  corresponding to the column product-id → **Edit cells** → **Common transforms** → **To number**:



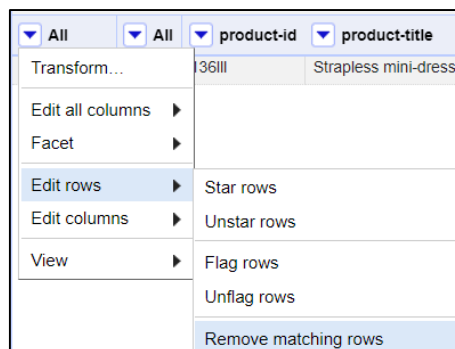
4. In the mini-window for viewing the result, it can be seen that a non-numeric value was identified in the field data. To see what this value is, **uncheck the Numeric button** in the mini-window:



5. In the window that appears, this non-numeric value is visualized:

1 matching rows (182 total)							
Show as: rows records		Show: 5 10 25 50 100 500 1000 rows					
All	All	product-id	product-title	product-price	product-color	product-size	product-discount
★	🗨	1. 26	136III	Strapless mini-dress			-20%

6. To delete this record: Click on the button  **All** → **Edit rows** → **Remove matching rows**:



7. One can note that this inconsistency problem has been resolved and that the data has been cleaned.

Press the **Remove this facet** button to return to viewing data fields:

Facet / Filter Undo / Redo 3 / 3

Refresh Reset all Remove all

0 matching rows (181 total)

Show as: rows records Show: 5 10 25 50 100 500 1000 rows

product-id change reset

0 — 14,000

Remove this facet

8. The application shows how many cells have been transformed. These cells are now marked with green color, which shows that the transformation has worked:

Permalink

Text transform on 182 cells in column product-id: value.toNumber() Undo

182 records

Show as: rows records Show: 5 10 25 50 100 500 1000 records

All	All	product-id	product-title	product-price	product-color	product-size	product-discount
1.	26	136	Strapless mini-dress				-20%
2.	324	1361	Strapless mini-dress	\$6.00	Red		-20%
3.	325	1362	Strapless mini-dress	\$42.49	Yellow		-20%
4.	326	1363	Strapless mini-dress	\$30.00	Brown		-20%
5.	327	1364	Strapless mini-dress	\$53.22	Blue		-20%
6.	328	1365	Strapless mini-dress	\$6.00		38	-20%
7.	329	1366	Strapless mini-dress			40	-20%
8.	330	1367	Strapless mini-dress	\$53.22	Midnight blue		-20%
9.	331	1368	Strapless mini-dress	\$50.00	Ocean blue		-20%

► **To remove duplicates from the data:**

For example, in the **product-id** column (field that defines a unique identifier):

1. Click on the button corresponding to the column → **Sort**.
2. In the window **Sort by product-id** that appears → check the options **numbers** and **smallest first** (to sort the data in ascending order) → **OK**:

Sort by product-id

Sort cell values as

text case-sensitive

numbers

dates

booleans

Position blanks and errors

Valid values

Errors

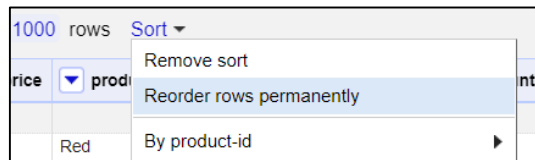
Blanks

Drag and drop to re-order


smallest first largest first

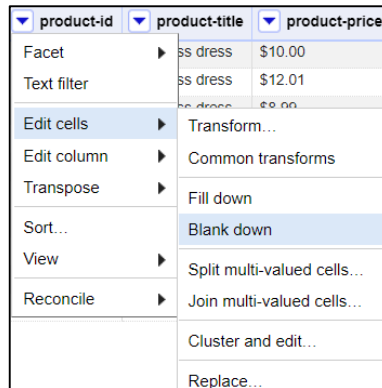
OK Cancel

3. To remove duplicates, select **Sort** → **Reorder rows permanently**:




If there is the same information in two rows, the information in the second row must be deleted. For this, proceed as follows:

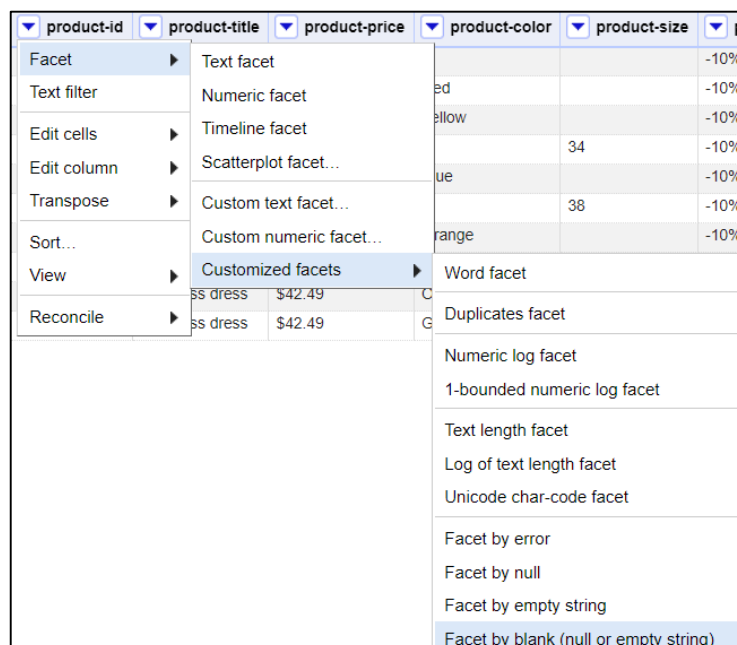
4. Click on the button  corresponding to the column → **Edit cells** → **Blank down**:



Show as: rows records Show: 5 10 25 50 100 500 1000 rows

	All	All	product-id	product-title	product-price	product-color	product-size	product-discount
11.	323	1359	Strapless dress	\$42.49	Grass green		-10%	
12.	324	1362	Strapless mini-dress	\$6.00	Red		-20%	
13.	325		Strapless mini-dress	\$42.49	Yellow		-20%	
14.	326	1363	Strapless mini-dress	\$30.00	Brown		-20%	
15.	327	1364	Strapless mini-dress	\$53.22	Blue		-20%	
16.	328	1365	Strapless mini-dress	\$6.00		38	-20%	
17.	329	1366	Strapless mini-dress			40	-20%	
18.	330	1367	Strapless mini-dress	\$53.22	Midnight blue		-20%	
19.	331	1368	Strapless mini-dress	\$50.00	Ocean blue		-20%	
20.	332	1369	Strapless mini-dress	\$30.00	Grass green		-20%	

5. Click on the button  corresponding to the column → **Facet** → **Customized facets** → **Facet by blank (null or empty string)**:



6. To delete duplicates: select **True**:

product-id		change	invert	reset
2 choices Sort by: name count				
false	181	exclude		
true	1			
Facet by choice counts				


7. Click on the button  **All** → **Edit rows** → **Remove matching rows**:

All	All	product-id	product-title
Transform...			Strapless mini-dress
Edit all columns ▶			
Facet ▶			
Edit rows ▶		Star rows	
Edit columns ▶		Unstar rows	
View ▶		Flag rows	
		Unflag rows	
		Remove matching rows	

8. Press the **Remove this facet** button to return to viewing data fields.

► **To analyze text type data in order to standardize them, one can use Faceting as an overview of text data:**

For example, in the **product-title** column:


1. Click on the button  corresponding to the column → **Facet** → **Text facet**:

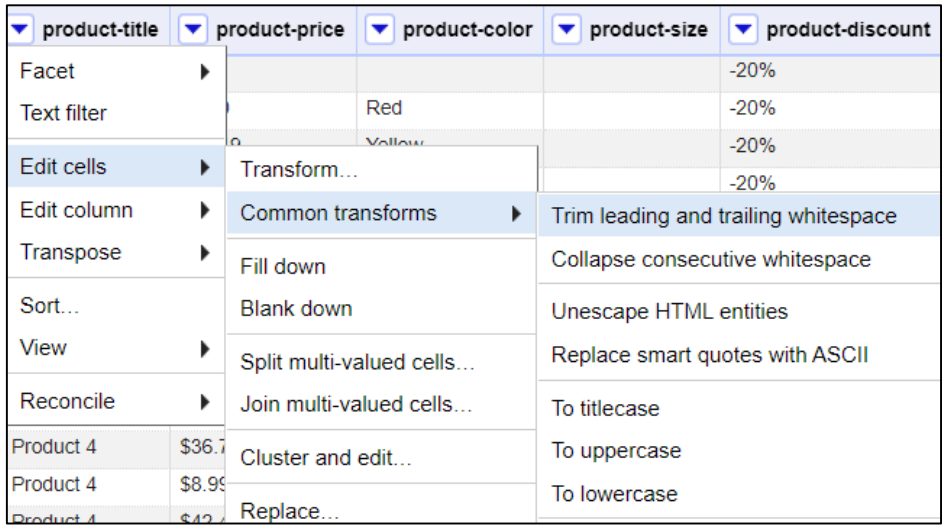
product-title	product-price	product-color
Facet ▶		Text facet
Text filter		Numeric facet
Edit cells ▶		Timeline facet
Edit column ▶		Scatterplot facet...
Transpose ▶		Custom text facet...
Sort...		Custom numeric facet...
View ▶		Customized facets ▶

2. One can notice that some records, although they are identical, are preceded by spaces, which makes it seem that they are different records. Then these data must be standardized:

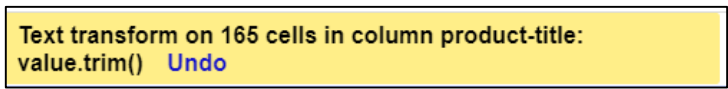
product-title		change	invert	reset
13 choices Sort by: name count Cluster				
Product 4	1	}		
Product 4	1			
Product 4	1			
Product 4	1			
Product 15	1			
Product 3	79			
Product 2	1			
Product 4	78			
Product 5	1			
Product 1	6			
Product 2	10			

► **To remove leading and trailing whitespace from text data records:**


1. Click on the button  corresponding to the column → **Edit cells** → **Common transforms** → **Trim leading and trailing whitespaces**:

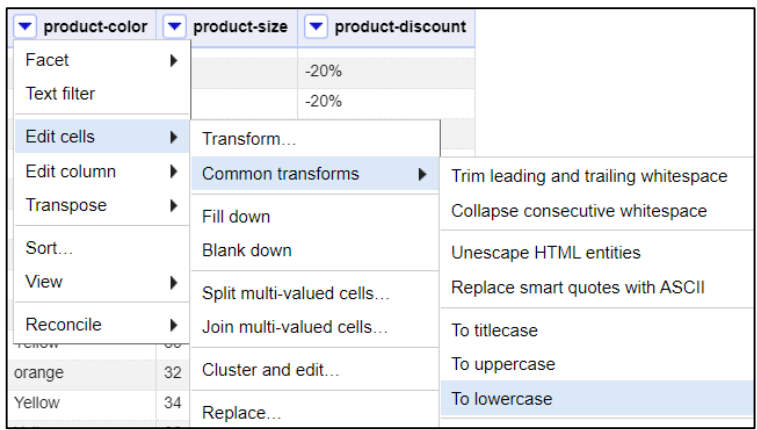


2. The application shows how many cells have been transformed:




► **To convert the text to be written in lowercase letters:**

1. Click on the button  corresponding to the column → **Edit cells** → **Common transforms** → **To lowercase**:




► **To collapse consecutive whitespace** (for example, when more than one space is left between words):

1. Click on the button  corresponding to the column → **Edit cells** → **Common transforms** → **Collapse consecutive whitespace**.

Another functionality offered by the application in order to standardize data is to find clusters of different cell values that could be other representations of the same thing.

► **To determine such groups of cells with close text values:**

1. Click on the button  corresponding to the column → **Edit cells** → **Cluster and edit...**:

product-color	product-size	product-disc
Facet		-20%
Text filter		-20%
Edit cells		20%
Edit column		
Transpose		
Sort...		
View		
Reconcile		
Red	30	

2. The application shows which values are in a cluster, how they look and if it is decided to merge them. Also, in the **New cell value** field, the application also suggests a common name for these similar data entries. But the user can enter another name under which all these records can be merged:

Cluster and edit column "product-color"

Find groups of different cell values that might be other representations of the same thing. For example, "New York" and "new york" likely refer to the same concept and just differ by capitalization, and "Gödel" and "Godel" probably refer to the same person. [Find out more...](#)

Method: Key collision Keying function: Fingerprint 3 clusters found

Cluster size	Row count	Values in cluster	Merge?	New cell value
3	19	<ul style="list-style-type: none"> • Red (16 rows) • red (2 rows) • Red 	<input type="checkbox"/>	<input type="text" value="Red"/>
2	20	<ul style="list-style-type: none"> • Orange (19 rows) • orange 	<input type="checkbox"/>	<input type="text" value="Orange"/>
2	20	<ul style="list-style-type: none"> • Brown (19 rows) • brown 	<input type="checkbox"/>	<input type="text" value="Brown"/>

Choices in cluster

2 — 3

Rows in cluster

19 — 20

Average length of choices

3.66 — 6

Length variance of choices

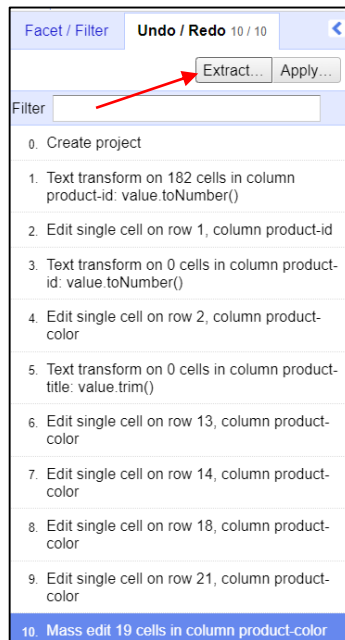
0 — 0.9430000000000001

Select all
Deselect all
Export clusters
Merge selected & re-cluster
Merge selected & Close
Close

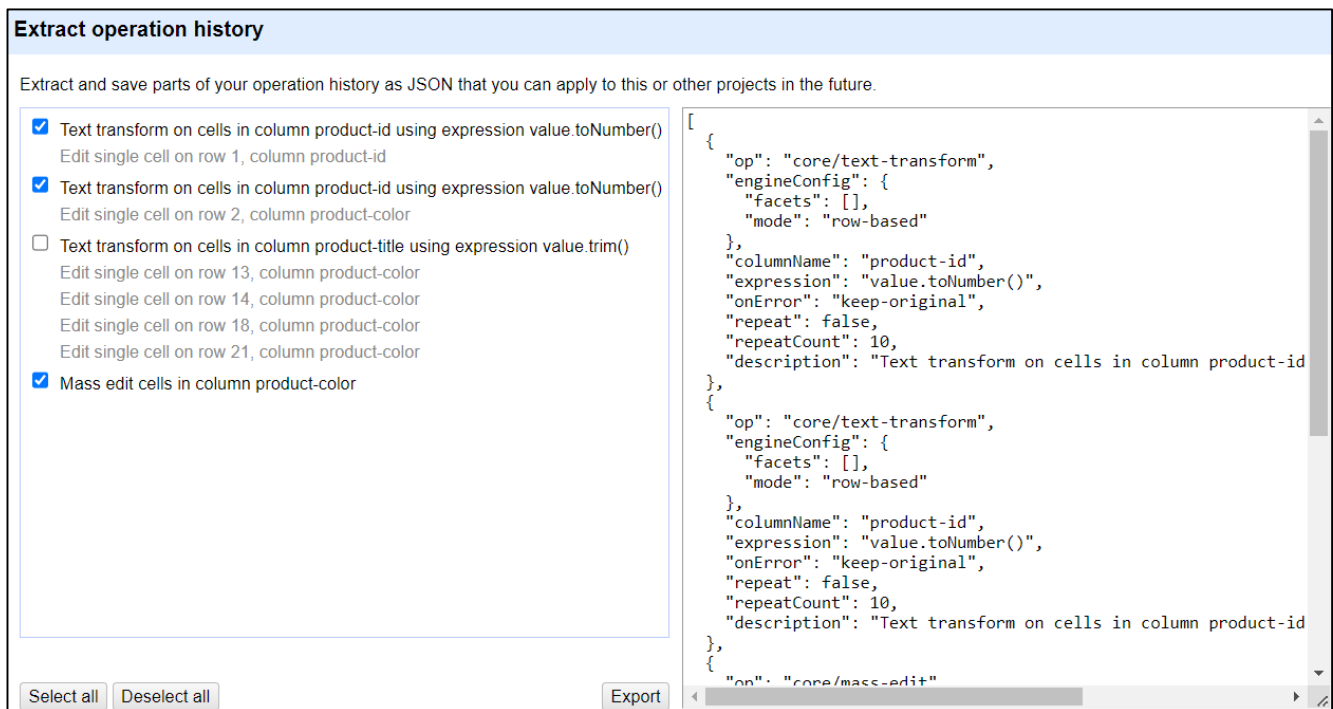
The application keeps track of all the operations performed on the data file so that in the future it will be known what actions were taken on the file. Moreover, these steps can be extracted to be applied to another similarly formatted project.

► **To extract operations history:**

1. Click on the button Extract...

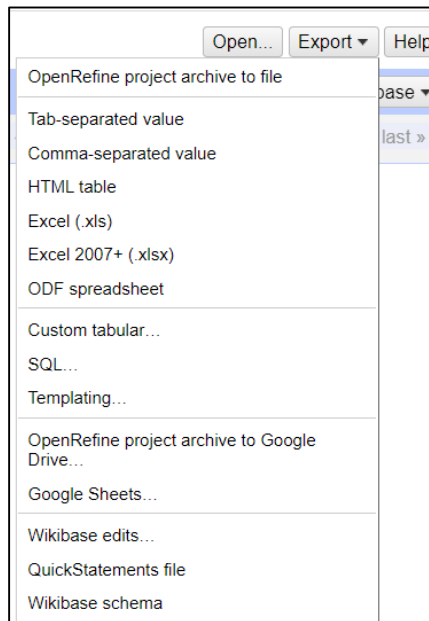


2. In the Extract operation history window that appears → select the steps to be applied → copy the script (which is in JSON) → Export:



Once all the data cleaning actions are finished, the updated content of the data file can be exported.

► **To export the file:** access the **Export** button at the top right of the application window.



The **OpenRefine project archive to file** option creates a project that can be shared with other users and that shows the steps that have been carried out; **a tar.gz type file will be created** which can be opened later only through this application.

Another frequently accessed option is **saving in open format** (non-proprietary format), such as **Tab-separated value** or **Comma-separated value**, where no specific software will be needed to open the saved file.

Another option would be to save as an **Excel** file or as **Google Sheets**, etc.

1.5.2 Operations and functions for cleaning data in Excel

MS Excel is a widely used application that allows the manipulation of large data sets, and therefore reviewing and correcting inaccurate, error-prone and inconsistent data is very important. Excel has various built-in techniques and functions for removing incorrect data, removing blank spaces, removing duplicates, changing the format of data, splitting or concatenating data, filling in missing information, etc. to obtain an updated database, which can later be subjected to analysis and processing, with the certainty of obtaining correct results for decision-makers.

In addition, since new sources of errors and inconsistencies may appear over time, the data cleaning process must be repeated at set time intervals.

Below are presented different data cleaning techniques and how they are done in Microsoft Excel.

► **Separating the information** from a column that contains a composite variable made up of two or more constituent parts, to enter it in two fields.

For example, in the Excel file below, to separate the information from the **project & deadline** column into two columns, one in which to record the **name of the project**, and in the adjacent column the **project deadline**:

1. Enter the first record in each column, respectively Project_1 and September:

A	B	C	D	E	F
No.	Department	Number of persons	project & deadline	project	deadline
1	department_1	5	project_1 september	project_1	september
2	department_2	2	project_2 october		
3	department_3	7	project_3 march		
4	department_4	4	project_4 january		
5	department_5	3	project_5 april		
6	department_6	6	project_6 september		
7	department_7	9	project_7 september		
8	department_8	4	project_8 october		
9	department_9	9	project_9 july		
10	department_10	8	project_10 november		
11	department_11	7	project_11 july		
12	department_12	12	project_12 september		
13	department_13	9	project_13 april		

2. Click in the cell where the project_1 record was entered → CTRL + E → the project column was automatically filled in with all the information.

Click in the cell where the September entry was entered → CTRL + E → the deadline column was automatically filled in with all the information:

A	B	C	D	E	F
No.	Department	Number of persons	project & deadline	project	deadline
1	department_1	5	project_1 september	project_1	september
2	department_2	2	project_2 october	project_2	october
3	department_3	7	project_3 march	project_3	march
4	department_4	4	project_4 january	project_4	january
5	department_5	3	project_5 april	project_5	april
6	department_6	6	project_6 september	project_6	september
7	department_7	9	project_7 september	project_7	september
8	department_8	4	project_8 october	project_8	october
9	department_9	9	project_9 july	project_9	july
10	department_10	8	project_10 november	project_10	november
11	department_11	7	project_11 july	project_11	july
12	department_12	12	project_12 september	project_12	september
13	department_13	9	project_13 april	project_13	april

► **Combining the information** from separate columns (join columns) for rearranging data, using the & operator.

Enter the formula for concatenating the information in a free cell of the spreadsheet → copy this formula in all the cells of the respective column:

M	N	O
project	deadline	Project and the corresponding deadline
project_1	september	=M2&" has the deadline in:"&N2
project_2	october	
project_3	march	
project_4	january	
project_5	april	
project_6	september	
project_7	september	
project_8	october	
project_9	july	
project_10	november	
project_11	july	
project_12	september	
project_13	april	

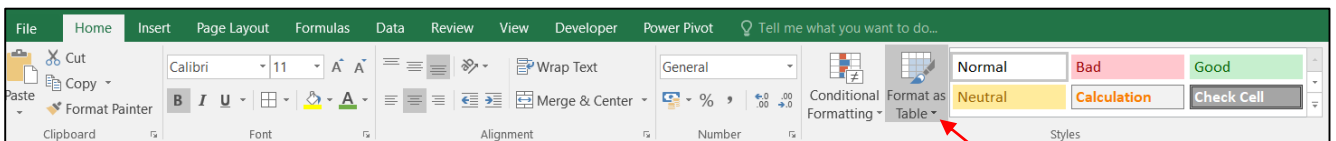
M	N	O
project	deadline	Project and the corresponding deadline
project_1	september	project_1 has the deadline in: september
project_2	october	project_2 has the deadline in: october
project_3	march	project_3 has the deadline in: march
project_4	january	project_4 has the deadline in: january
project_5	april	project_5 has the deadline in: april
project_6	september	project_6 has the deadline in: september
project_7	september	project_7 has the deadline in: september
project_8	october	project_8 has the deadline in: october
project_9	july	project_9 has the deadline in: july
project_10	november	project_10 has the deadline in: november
project_11	july	project_11 has the deadline in: july
project_12	september	project_12 has the deadline in: september
project_13	april	project_13 has the deadline in: april

► **Formatting the data in a worksheet as a table** will allow additional functions to be performed on each column in the resulting table:

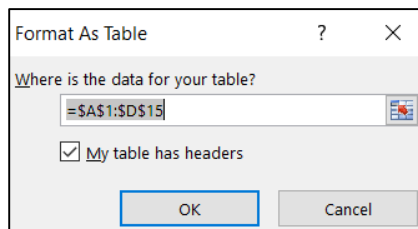
1. **Select the data columns** that must be formatted as a table:

No.	Department	Number of persons	project & deadline
1	department_1	5	project_1 september
2	department_2	2	project_2 october
3	department_3	7	project_3 march
4	department_4	4	project_4 january
5	department_5	3	project_5 april
6	department_6	6	project_6 september
7	department_7	9	project_7 september
8	department_8	4	project_8 october
9	department_9	9	project_9 july
10	department_10	8	project_10 november
11	department_11	7	project_11 july
12	department_12	12	project_12 september
13	department_13	9	project_13 april

2. Click **Format as Table**:

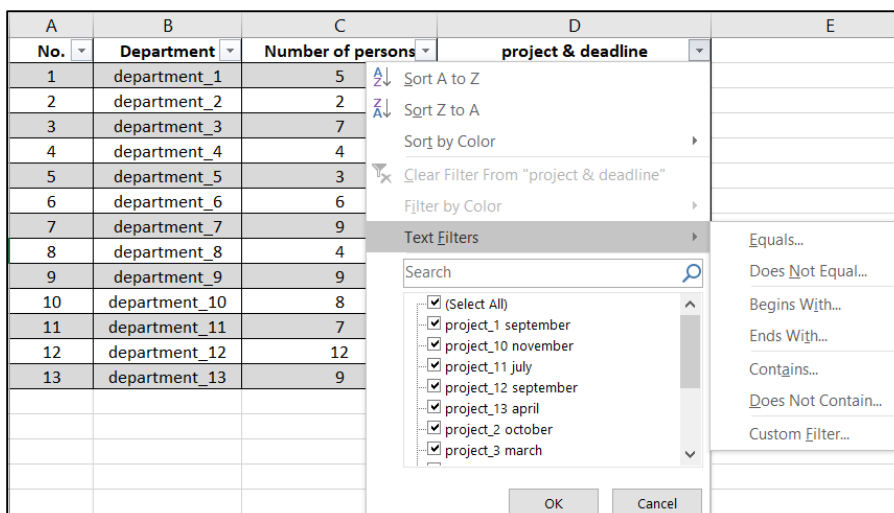


2. A dialog box will appear to confirm the date range to be formatted:



If the option **My table has headers** is checked, then the first row of the data set will be used as a header and not as part of the data set. If the selected data does not have a header, this option must be unchecked.

3. The **table created will contain headers and buttons that open additional functions** that allow selection of values, filtering, sorting and detection of abnormal values:

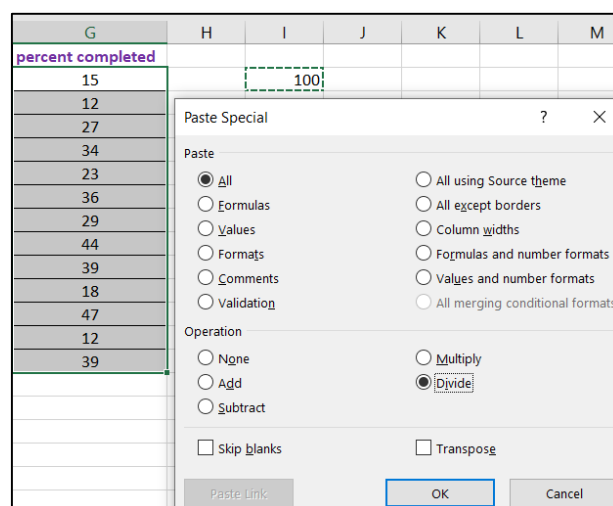


► **Transforming numerical values into percentage values:**

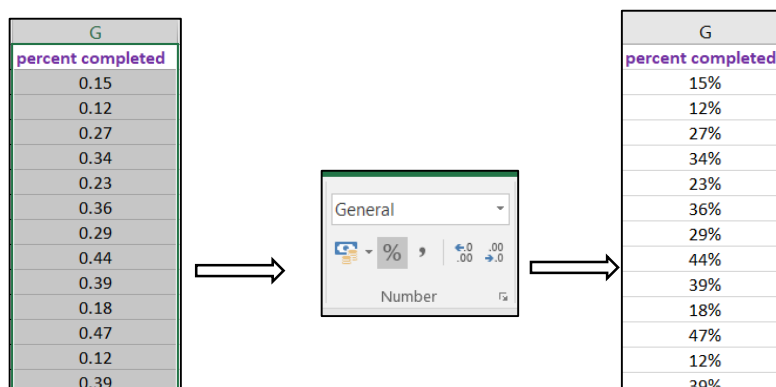
1. Type the value 100 in a free cell of the spreadsheet → copy this value:

A	B	C	D	E	F	G	H	I
No.	Department	Number of persons	project & deadline	project	deadline	percent completed		
1	department_1	5	project_1 september	project_1	september	15		100
2	department_2	2	project_2 october	project_2	october	12		
3	department_3	7	project_3 march	project_3	march	27		
4	department_4	4	project_4 january	project_4	january	34		
5	department_5	3	project_5 april	project_5	april	23		
6	department_6	6	project_6 september	project_6	september	36		
7	department_7	9	project_7 september	project_7	september	29		
8	department_8	4	project_8 october	project_8	october	44		
9	department_9	9	project_9 july	project_9	july	39		
10	department_10	8	project_10 november	project_10	november	18		
11	department_11	7	project_11 july	project_11	july	47		
12	department_12	12	project_12 september	project_12	september	12		
13	department_13	9	project_13 april	project_13	april	39		

2. Select the cells to be converted into percentage values → **Right click** → **Paste Special** → select the **Divide** option:



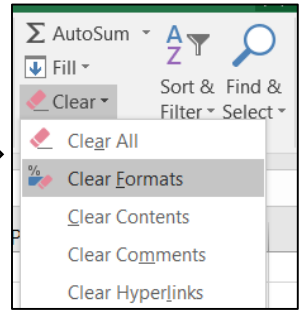
3. Select the cells converted to percentage values → **Home tab** → select the % option (from the Number section):



► **Data standardization by bringing them to the same format:**

1. Select the cells that must be brought to the same format → **Home tab** → select the **Clear** option (from the Editing section) → **Clear Formats**:

A	B	C	D	E	F	G
No.	Department	Number of persons	project & deadline	project	deadline	percent completed
1	department_1	5	project_1 september	project_1	september	15%
2	department_2	2	project_2 october	project_2	october	12%
3	department_3	7	project_3 march	project_3	march	27%
4	department_4	4	project_4 january	project_4	january	34%
5	department_5	3	project_5 april	project_5	april	23%
6	department_6	6	project_6 september	project_6	september	36%
7	department_7	9	project_7 september	project_7	september	29%
8	department_8	4	project_8 october	project_8	october	44%
9	department_9	9	project_9 july	project_9	july	39%
10	department_10	8	project_10 november	project_10	november	18%
11	department_11	7	project_11 july	project_11	july	47%
12	department_12	12	project_12 september	project_12	september	12%
13	department_13	9	project_13 april	project_13	april	39%



2. All data will be automatically brought to the same standard format:

A	B	C	D	E	F	G
No.	Department	Number of persons	project & deadline	project	deadline	percent completed
1	department_1	5	project_1 september	project_1	september	0.15
2	department_2	2	project_2 october	project_2	october	0.12
3	department_3	7	project_3 march	project_3	march	0.27
4	department_4	4	project_4 january	project_4	january	0.34
5	department_5	3	project_5 april	project_5	april	0.23
6	department_6	6	project_6 september	project_6	september	0.36
7	department_7	9	project_7 september	project_7	september	0.29
8	department_8	4	project_8 october	project_8	october	0.44
9	department_9	9	project_9 july	project_9	july	0.39
10	department_10	8	project_10 november	project_10	november	0.18
11	department_11	7	project_11 july	project_11	july	0.47
12	department_12	12	project_12 september	project_12	september	0.12
13	department_13	9	project_13 april	project_13	april	0.39

► Another source of data inconsistency is the extra spaces that usually appear when data from other systems are copied into Excel. **To remove these extra spaces:**

In an adjacent cell, enter the TRIM() function, which removes any spaces from a text, except for single spaces between words; **in the TRIM function, the address of a cell where this problem occurs is entered as an argument** → the formula is copied in all the cells below:

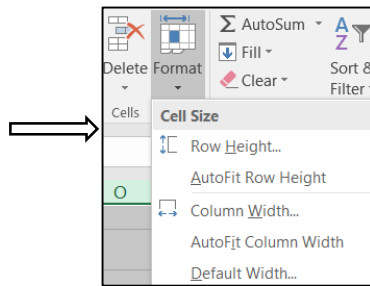
F	G
deadline	
september	=TRIM(F2)
october	
march	
january	
april	
september	
september	
october	
july	
november	
july	
september	
april	

F	G
deadline	
september	september
october	october
march	march
january	january
april	april
september	september
september	september
october	october
july	july
november	november
july	july
september	september
april	april

► **Automatic resizing of too long rows and/or too short columns** to improve data visualization. For automatic resizing:

Select the entire page (CTRL + A) → Format (from the Cells section) → Autofit Row Height and Autofit Column Width:

E	F	H
project	deadline	buget
project_1	september	125789.64
project_2	october	3578945.69
project_3	march	#####
project_4	january	564789.00
project_5	april	6547812.69
project_6	september	#####
project_7	september	98754.96
project_8	october	#####
project_9	july	457123.89
project_10	november	#####
project_11	july	897456.23
project_12	september	8971245.65
project_13	april	#####

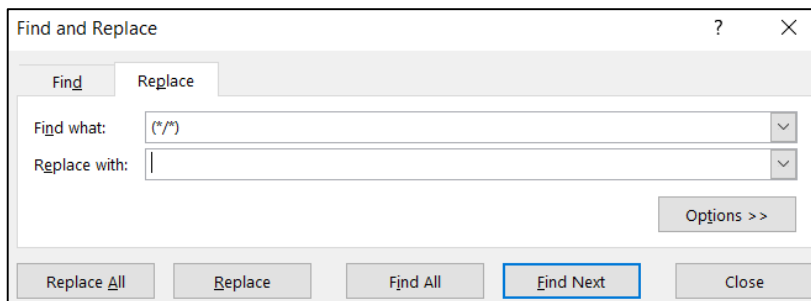


E	F	H
project	deadline	buget
project_1	september	125789.64
project_2	october	3578945.69
project_3	march	987895462.00
project_4	january	564789.00
project_5	april	6547812.69
project_6	september	68745554.98
project_7	september	98754.96
project_8	october	98712935.99
project_9	july	457123.89
project_10	november	89712547.87
project_11	july	897456.23
project_12	september	8971245.65
project_13	april	897471897.66

► **Removing parts of text that are found in several cells:**

For example, in the **Head of department column**, to **remove everything that is written between brackets** after the person's name:

1. **Select the column** → **Find & Select** (from the Editing section) → **Replace**.
2. In the **Find and Replace** window, in the **Find what** field, **any appearance of brackets is replaced by writing (*/*)** and the **Replace with** field is **left empty**, so as not to replace anything:



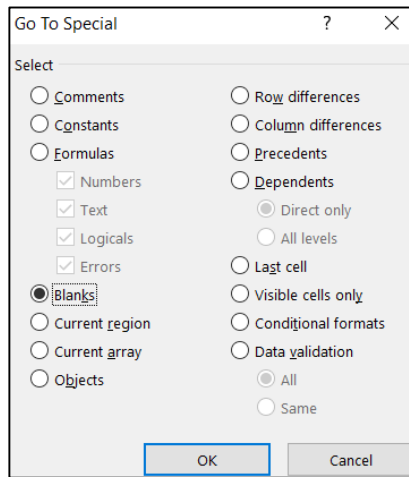
B	C
Department	Head of department
department_1	Adam Smith (xw/B1)
department_2	Carol Morgan (ab/C2)
department_3	Paul Johnson (w3/C1)
department_4	Martha Brown (b9/D1)
department_5	John Murphy (h6/E1)
department_6	Mary Cooper (af/F1)
department_7	Dona Raven (cd/H1)
department_8	Will Bart (rt/C8)
department_9	Paul Taylor (w9/D1)
department_10	Ana Barrel (kj/P1)
department_11	Beth Evans (w9/D1)
department_12	David Scott (ut/P7)
department_13	Mary Coampbell (as/B6)



B	C
Department	Head of department
department_1	Adam Smith
department_2	Carol Morgan
department_3	Paul Johnson
department_4	Martha Brown
department_5	John Murphy
department_6	Mary Cooper
department_7	Dona Raven
department_8	Will Bart
department_9	Paul Taylor
department_10	Ana Barrel
department_11	Beth Evans
department_12	David Scott
department_13	Mary Coampbell

► **Replacing empty cells with a predefined text, for example NA (Not Applicable):**

1. **Select the entire page** (CTRL + A) → **Find & Select** (from the Editing section) → **Go To Special**.
2. In the **Go To Special** window → **check the option Blanks** → **OK**:



3. At this moment, all empty cells have been automatically selected:

A	B	C	D	E	F	G	H
No.	Department	Head of department/department name	Number of persons	project & deadline	project	deadline	budget
1	department_1	Adam Smith/department_C3	5	project_1 september	project_1	september	125789.64
2	department_2	Carol Morgan/department_B2	3	project_2 october	project_2	october	3578945.69
3	department_3	Paul Johnson/department_D1	7		project_3	march	
4	department_4	Martha Brown/department_D1	4	project_4 january	project_4	january	564789.00
5	department_5	John Murphy/department_C1	3	project_5 april	project_5	april	6547812.69
6	department_6	Mary Cooper/department_B1	6	project_6 september	project_6	september	68745554.98
7	department_7	Dona Raven/department_D2	9	project_7 september	project_7	september	98754.96
8	department_8	Will Bart/department_D3	4	project_8 october	project_8	october	
9	department_9	Paul Taylor/department_D2	9	project_9 july	project_9	july	457123.89
10	department_10	Ana Barrel/department_B1	8		project_10	november	
11	department_11	Paul Johnson/department_D1	7	project_11 july	project_11	july	897456.23
12	department_12	David Scott/department_A2	12	project_12 september	project_12	september	8971245.65
13	department_13	Mary Coampbell/department_B3	9	project_13 april	project_13	april	897471897.66

4. Next, to replace these cells with the text NA:

In the Formula bar → enter the text "NA" → CTRL + Enter:


A	B	C	D	E	F	G	H
No.	Department	Head of department/department name	Number of persons	project & deadline	project	deadline	budget
1	department_1	Adam Smith/department_C3	5	project_1 september	project_1	september	125789.64
2	department_2	Carol Morgan/department_B2	3	project_2 october	project_2	october	3578945.69
3	department_3	Paul Johnson/department_D1	7	NA	project_3	march	NA
4	department_4	Martha Brown/department_D1	4	project_4 january	project_4	january	564789.00
5	department_5	John Murphy/department_C1	3	project_5 april	project_5	april	6547812.69
6	department_6	Mary Cooper/department_B1	6	project_6 september	project_6	september	68745554.98
7	department_7	Dona Raven/department_D2	9	project_7 september	project_7	september	98754.96
8	department_8	Will Bart/department_D3	4	project_8 october	project_8	october	NA
9	department_9	Paul Taylor/department_D2	9	project_9 july	project_9	july	457123.89
10	department_10	Ana Barrel/department_B1	8	NA	project_10	november	NA
11	department_11	Paul Johnson/department_D1	7	project_11 july	project_11	july	897456.23
12	department_12	David Scott/department_A2	12	project_12 september	project_12	september	8971245.65
13	department_13	Mary Coampbell/department_B3	9	project_13 april	project_13	april	897471897.66

It can be seen that this text was automatically added to all the empty cells.

► **Conversion of mixed written text type data (uppercase/ lowercase letters) into text in which the first letter of each word is capitalized, and the rest of the word is written in lowercase letters:**

1. Enter the PROPER() function in an adjacent cell, which has the respective cell as an argument → Copy the formula for the entire column:


D	E
Head of department	
Adam Smith	=PROPER(D2)
carol Morgan	
Paul Johnson	
martha brown	
john Murphy	
mary Cooper	
dona Raven	
will Bart	
Paul Taylor	
ana Barrel	
beth Evans	
david Scott	
Mary Coampbell	



D	E
Head of department	
Adam Smith	Adam Smith
carol Morgan	Carol Morgan
Paul Johnson	Paul Johnson
martha brown	Martha Brown
john Murphy	John Murphy
mary Cooper	Mary Cooper
dona Raven	Dona Raven
will Bart	Will Bart
Paul Taylor	Paul Taylor
ana Barrel	Ana Barrel
beth Evans	Beth Evans
david Scott	David Scott
Mary Coampbell	Mary Coampbell

Note: Through the combined use of the PROPER() and TRIM() functions, the combination of the two effects can be obtained:

D	E
Head of department	
Adam Smith	=PROPER(TRIM(D2))
carol Morgan	
Paul Johnson	
martha brown	
john Murphy	
mary Cooper	
dona Raven	
will Bart	
Paul Taylor	
ana Barrel	
beth Evans	
david Scott	
Mary Coampbell	




D	E
Head of department	
Adam Smith	Adam Smith
carol Morgan	Carol Morgan
Paul Johnson	Paul Johnson
martha brown	Martha Brown
john Murphy	John Murphy
mary Cooper	Mary Cooper
dona Raven	Dona Raven
will Bart	Will Bart
Paul Taylor	Paul Taylor
ana Barrel	Ana Barrel
beth Evans	Beth Evans
david Scott	David Scott
Mary Coampbell	Mary Coampbell

► **To find out the size of a character string** (text type data) one can use the LEN() function:

1. Enter the LEN() function in an adjacent cell, which has the respective cell as an argument → Copy the formula for the entire column:

H	I
Telephone	
+40744111111	=LEN(H2)
+40744222222	
+40744333333	
+40744444444	
+40744555555	
+40744666666	
+40744777777	
+40744888888	
+40744999999	
+40744232323	
+40744242424	
+40744252525	
+40744262626	




H	O
Telephone	
+40744111111	14
+40744222222	12
+40744333333	12
+40744444444	12
+40744555555	12
+40744666666	12
+40744777777	15
+40744888888	12
+40744999999	12
+40744232323	12
+40744242424	18
+40744252525	12
+40744262626	12

2. It is observed that, although apparently the data in each cell of the column have the same length, in fact it is not so, due to the existing additional spaces.

Therefore, to find out the exact size of the data, a combination of a TRIM() and LEN() functions is used:

H	O
Telephone	
+40744111111	=LEN(TRIM(H2))
+40744222222	
+40744333333	
+40744444444	
+40744555555	
+40744666666	
+40744777777	
+40744888888	
+40744999999	
+40744232323	
+40744242424	
+40744252525	
+40744262626	



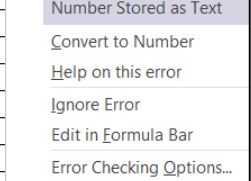

H	O
Telephone	
+40744111111	12
+40744222222	12
+40744333333	12
+40744444444	12
+40744555555	12
+40744666666	12
+40744777777	12
+40744888888	12
+40744999999	12
+40744232323	12
+40744242424	12
+40744252525	12
+40744262626	12

▶ Another inconsistency in data loaded into an Excel file is data that was loaded as text data instead of numeric data.


To check the numerical format of the data, the function ISNUMBER() is used, which results in the value TRUE if the respective date is numerical and FALSE, otherwise. Afterwards, the data recognized as text data is transformed into numerical data by the VALUE() function.

1. Enter the ISNUMBER() function in an adjacent cell, which has the respective cell as an argument → Copy the formula for the entire column:

G	H
Number of persons	
25	
13	
7	
4	
13	
6	
19	
24	
9	
18	
7	
13	
9	


G	H
Number of persons	
25	=ISNUMBER(G2)
13	
7	
4	
13	
6	
19	
24	
9	
18	
7	
13	
9	




G	H
Number of persons	
25	TRUE
13	FALSE
7	TRUE
4	TRUE
13	TRUE
6	TRUE
19	TRUE
24	FALSE
9	TRUE
18	TRUE
7	TRUE
13	FALSE
9	TRUE

2. Enter the VALUE() function in an adjacent cell, which has the respective cell as an argument → Copy the formula for the entire column → Check that the transformed data are of type number:

G	H
Number of persons	
25	=VALUE(G2)
13	
7	
4	
13	
6	
19	
24	
9	
18	
7	
13	
9	



G	H
Number of persons	
25	25
13	13
7	7
4	4
13	13
6	6
19	19
24	24
9	9
18	18
7	7
13	13
9	9




H	I
25	TRUE
13	TRUE
7	TRUE
4	TRUE
13	TRUE
6	TRUE
19	TRUE
24	TRUE
9	TRUE
18	TRUE
7	TRUE
13	TRUE
9	TRUE

► In the case of a data field of score-type information, this can be visually represented by using the REPT() function, which replaces the respective score with a number of instances of the selected symbol. Basically, the function repeats entering the symbol that number of times.

1. Enter the REPT() function in an adjacent cell, which has the respective cell as an argument → Copy the formula for the entire column:

O	P
Client satisfaction score	
10	=REPT("●",O2)
9	
8	
9	
7	
6	
9	
8	
9	
10	
9	
10	
7	




O	P
Client satisfaction score	
10	●●●●●●●●
9	●●●●●●●
8	●●●●●●
9	●●●●●●
7	●●●●●
6	●●●●●
9	●●●●●●●
8	●●●●●●
9	●●●●●●
10	●●●●●●●●
9	●●●●●●●
10	●●●●●●●●
7	●●●●●


► In the case of a data field in which numerical information with a given number of digits must be stored, if the number in a cell has fewer digits and then a corresponding number of zeros must be added in front of them, proceed as follows:

1. Determine for each cell how many zeros must be added. For example, in the field below, the numbers must be 11 digits long. First, the length of the numbers is determined by applying the LEN() function → then the required number of zeros is determined by applying the formula 11-LEN():

R	S
59612578964	=LEN(R2)
357894569	
98789546200	
56478900	
654781269	
16874555498	
9875496	
9871293599	
45712389	
8971254787	
89745623	
897124565	
89747189766	




R	S	T
59612578964	11	=11-LEN(R2)
357894569		
98789546200		
56478900		
654781269		
16874555498		
9875496		
9871293599		
45712389		
8971254787		
89745623		
897124565		
89747189766		



R	S	T
59612578964	11	
357894569	9 00	
98789546200	11	
56478900	8 000	
654781269	9 00	
16874555498	11	
9875496	7 0000	
9871293599	10 0	
45712389	8 000	
8971254787	10 0	
89745623	8 000	
897124565	9 00	
89747189766	11	

2. Enter in each cell the required number of zeros, previously determined, using REPT() function, with arguments the symbol "0" and number of zeros determined by the 11-LEN() function:


R	S	T
59612578964	11	=REPT("0",11-LEN(R2))
357894569	9	
98789546200	11	
56478900	8	
654781269	9	
16874555498	11	
9875496	7	
9871293599	10	
45712389	8	
8971254787	10	
89745623	8	
897124565	9	
89747189766	11	



R	S	T
59612578964	11	
357894569	9 00	
98789546200	11	
56478900	8 000	
654781269	9 00	
16874555498	11	
9875496	7 0000	
9871293599	10 0	
45712389	8 000	
8971254787	10 0	
89745623	8 000	
897124565	9 00	
89747189766	11	

3. The necessary zeros are concatenated with the initial data using the & operator:

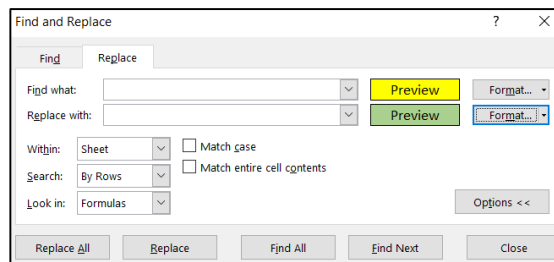
R	S	T	U
59612578964	11		=REPT("0",11-LEN(R2))&R2
357894569	9	00	
98789546200	11		
56478900	8	000	
654781269	9	00	
16874555498	11		
9875496	7	0000	
9871293599	10	0	
45712389	8	000	
8971254787	10	0	
89745623	8	000	
897124565	9	00	
89747189766	11		




R	S	T	U
59612578964	11		59612578964
357894569	9	00	00357894569
98789546200	11		98789546200
56478900	8	000	00056478900
654781269	9	00	00654781269
16874555498	11		16874555498
9875496	7	0000	00009875496
9871293599	10	0	09871293599
45712389	8	000	00045712389
8971254787	10	0	08971254787
89745623	8	000	00089745623
897124565	9	00	00897124565
89747189766	11		89747189766

► To replace a color format of some cells with another color format:

1. Select the respective cell range → Home → Find & Select → Replace.
2. In the window Find and Replace → Replace Format:



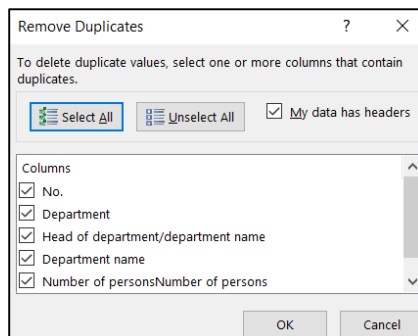
L	O	P
project & deadline	Project and the corresponding deadline	Client satisfaction score
project_1 september	project_1 has the deadline in: september	10
project_2 october	project_2 has the deadline in:october	9
project_3 march	project_3 has the deadline in:march	8
project_4 january	project_4 has the deadline in: january	9
project_5 april	project_5 has the deadline in:april	7
project_6 september	project_6 has the deadline in:september	6
project_7 september	project_7 has the deadline in:september	9
project_8 october	project_8 has the deadline in:october	8
project_9 july	project_9 has the deadline in:july	9
project_10 november	project_10 has the deadline in:november	10
project_11 july	project_11 has the deadline in:july	9
project_12 september	project_12 has the deadline in:september	10
project_13 april	project_13 has the deadline in:april	7



L	O	P
project & deadline	Project and the corresponding deadline	Client satisfaction score
project_1 september	project_1 has the deadline in: september	10
project_2 october	project_2 has the deadline in:october	9
project_3 march	project_3 has the deadline in:march	8
project_4 january	project_4 has the deadline in: january	9
project_5 april	project_5 has the deadline in:april	7
project_6 september	project_6 has the deadline in:september	6
project_7 september	project_7 has the deadline in:september	9
project_8 october	project_8 has the deadline in:october	8
project_9 july	project_9 has the deadline in:july	9
project_10 november	project_10 has the deadline in:november	10
project_11 july	project_11 has the deadline in:july	9
project_12 september	project_12 has the deadline in:september	10
project_13 april	project_13 has the deadline in:april	7

► Removing duplicate records:

1. Select the range of cells to be analyzed → Data → Remove Duplicate.
2. In the Remove Duplicates window → if identical registration lines must be removed, check all the columns:



1.6 Intelligent document processing

Business Intelligence focuses on how to capture, access, store, process, analyze and visualize the resulting information and knowledge by transforming one of a company's most valuable assets, raw data, with the goal of improving business performance. Processing information with appropriate analytical tools provides decision makers with competitive information that effectively differentiates them in their business environment.

Unlike the traditional method of document processing (based on manual data entry), which is laborious, requires a large amount of processing time, is also time-consuming and error-prone, intelligent document processing (IDP) is a technology-based approach that automates document processing and extracts valuable information. The main objective of this technology is to quickly extract information without compromising the accuracy of the process. With the introduction of this integrated technology, companies can automate the data entry and analysis process, leading to a significant reduction in time spent and error rates.

Within companies, many types of files (contracts, invoices, rental documents and utility bills, etc.) contain unstructured data that needs to be converted into a more usable, structured form. While managing structured data is simple, processing and analyzing unstructured data is laborious. This data needs to be captured, structured, cleaned, sorted, validated and loaded into a data warehouse for reporting and analysis (Figure 1.3):

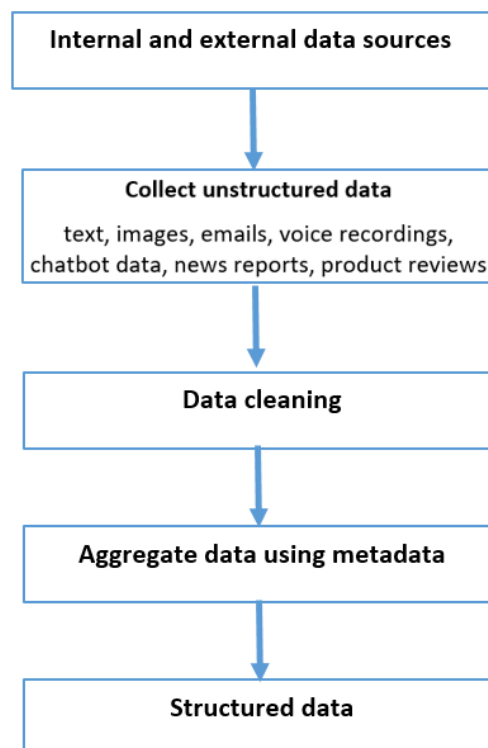


Figure 1.3: Transforming the unstructured data into structured data

But many companies only have opportunities to use structured data, thus ignoring a considerable volume of unstructured data. Therefore, the challenge facing the business world is to develop effective

methods for transforming the countless forms of unstructured or semi-structured data into structured data.

Intelligent document processing (IDP) emerged with the development of the first optical character recognition (OCR) solutions, the purpose of which was to transform character images into coded text. Later, the technology evolved by incorporating another technology: natural language processing (NLP), which allowed document processing to advance beyond simple character recognition and enabled some level of text interpretation and understanding. Gradually, these solutions were combined with other solutions, and today, IDP includes a wide set of capabilities, combining optical character recognition (OCR), artificial intelligence (AI), natural language processing (NLP) and machine learning (ML) algorithms to process documents. The collaboration between these technologies makes it possible to streamline many work processes.

IDP can handle a wide range of document formats and types, making it an extremely versatile tool. Whether it's a scanned image, a PDF file or a handwritten note, IDP can accurately extract and process the data, eliminating any possibility of human error during document processing. This capability is particularly advantageous for companies that handle a large amount of different documents every day. The speed with which large volumes of data are processed, compared to the time it takes to do it manually is obviously a big advantage of IDP technology.

For employees who, for various reasons, have to work remotely, intelligent document processing has allowed them to modify and manage the text reported in documents much more easily, avoiding the need to manually copy all the content and facilitating the search for information. Understanding the content, context and meaning of the data present in these documents, regardless of their format or structure, is fundamental as they form the basis of further analysis. By adopting IDP, a company will move from manual, paper-based processes to automated workflows. This shift allows the company to take advantage of emerging technologies and drive innovation across the business.

IDP implementation also leads to cost savings for companies: by automating document processing tasks, companies can significantly reduce labor costs. Automated processes enable work to be completed in a shorter period of time, which also means lower operating costs compared to manual tasks.

Processes that contain repetitive and mechanical work and that would normally take place over several working days become executable by machines, allowing workers to focus on tasks that require greater intellectual effort and creativity, something that machines are not equipped to do. In addition, the use of IDP has a profound impact on customer satisfaction: by automating tedious tasks, staff can focus more on customer service and less on red tape, further improving the customer experience.

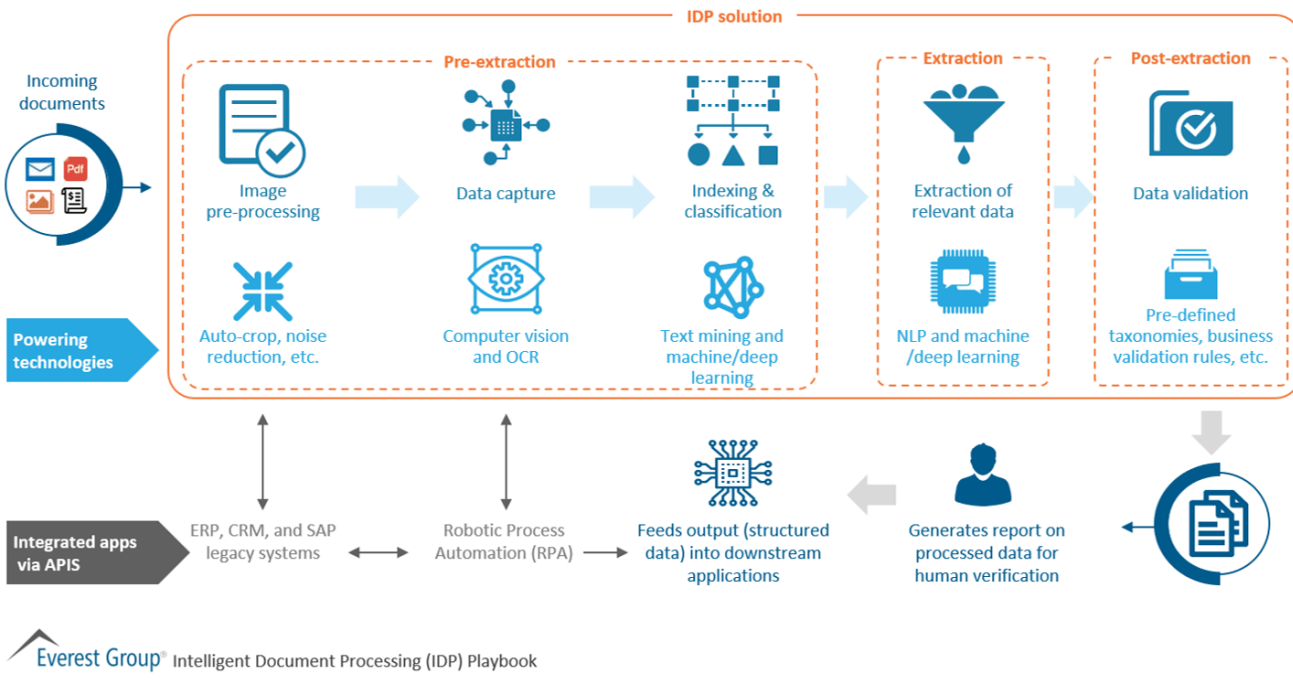
Therefore, it is of paramount importance to effectively evaluate and manage company information to turn data into a competitive resource and make strategic choices.

Here's an overview of how IDP technology works:

Understanding Enterprise Grade IDP Solutions

An enterprise-grade Intelligent Document Processing solution performs the following actions:

- **Pre-extraction:** Performs image pre-processing to increase the quality of the scanned document, captures data, and indexes & classifies the documents into categories
- **Extraction:** Extracts relevant data leveraging NLP and ML/DL capabilities for further processing
- **Post extraction:** Validates the extracted data with the help of pre-defined taxonomies, data dictionary, and business validation rules



www.everestgrp.com/2019-12-understanding-enterprise-grade-idp-solutions-market-insights-52033.html/

Depending on the business requirements, intelligent document processing contains several stages. In the general case, the key steps involved in this process are:

► **Collection of unstructured raw data:** paper documents (invoices, receipts, bank statements, application forms, reports, charts or any other handwritten documents) and digital (possibly using custom interfaces) from a variety of available content sources. Hardware and software integrations (such as scanners, cameras, or mobile apps) allow IDP tools to perform large-scale collection processes.

► **Preprocessing documents** (scanned or captured by camera) to meet certain quality standards. Given that documents exist at different quality levels, a variety of techniques are automatically applied at this stage to improve document quality and readability, such as:

- image binarization: the technique of converting a color image into black and white pixels;
- elimination of skew: when a document is scanned, it is possible that the image is not completely aligned and this situation is not desirable for the OCR technique and must be corrected by specific methods;
- noise reduction: eliminating scribbles, marks, so that OCR does not confuse these elements with characters;

- standardization of image resolution;
- data cleaning: analysis of redundant, incomplete or inaccurate data;

► **Data classification and extraction** - the document is analyzed and its components are identified, separated and automatically classified into different categories by type, content, structure (structured, semi-structured or unstructured document) and/or format (if the file is a PDF, JPF, PNG document or any other format). To classify these inputs, Natural Language Processing algorithms are used, based on textual content, which can divide documents according to the type of content presented, or Computer Vision algorithms to classify images or document scans.

Once the information contained in the document has been correctly classified, the textual data must be extracted and digitized to be directed to the next destination.

► **Data validation** - involves detecting inaccuracies in the extracted data: to check the accuracy and integrity of the extracted information (for example by correcting spelling errors) and that the data extracted from documents corresponds to the expected formats (by adjusting the data to standard formats), the data is authenticated using a set of validation rules, external databases and preconfigured vocabularies, but also artificial intelligence techniques. Also, databases or external services are used that allow adding information to those extracted from the document to have more detailed and higher quality data.

In this stage the data is therefore validated with different criteria or techniques with the ultimate goal of improving the accuracy of the IDP system. In this way, any correction acts as an input to the system so that the accuracy improves in future extractions. This feedback loop will increase the accuracy of the system.

Since no verification model has maximum accuracy, a human intervention step will need to be included in this IDP stage. Based on checks by human authenticators, the validation model continuously learns and improves its accuracy over time, because the more documents are processed and reviewed, the better the accuracy of the data mining model.

► **Integrating validated data** (structured data) into the systems that will use them. This information (often presented as graphs, charts, movies, images, plain text, etc.) will be available for immediate consumption in a format that seamlessly integrates with other systems such as databases or business tools intelligence. Thus, they can be used by the company for analysis and reporting, to take quick action and provide an efficient service to its customers.

1.7 Key technologies behind Intelligent Document Processing

► **Optical Character Recognition (OCR)**

OCR ("Optical Character Recognition") is a technology that identifies and recognizes printed or handwritten text characters from scanned or image-based text, from various files such as scanned documents, structured forms, photos, pdf files, etc., to make them available for further processing: the

document can be edited and stored on the computer or smartphone. Thus, the computer can interact with this document in the same way as with any other digital document: the document can be processed with functions for search, with tools for analysis and comparison, for editing and sharing of the resulting information.

OCR technologies typically have two components: the hardware to digitize the document and the software to convert the documents into machine-readable text. The software OCR is a fundamental tool in Computer Vision systems – an interdisciplinary scientific field that, from an engineering perspective, aims to train computers to be able to acquire, to process, to analyze, to understand the visual world and to extract meaningful information from digital images. Computer Vision technologies work similarly to human vision, and once trained, these systems can inspect or analyze thousands of images, spotting issues imperceptible to the human eye.

Without an OCR software, a computer perceives a scanned document as an image: after scanning, an analog sheet is just a graphic on the computer, made up of pixels with various color values. In this format, the computer cannot recognize individual letters/words/sentences. But OCR software, through several steps, recognizes known patterns, which are then identified as individual letters and translated from image to text, in the form of sentences.

As for the accuracy of the identification, it differs depending on the OCR software: there are many OCR programs available that identify the respective texts with varying degrees of accuracy and convert them into an editable format:

- *Simple OCR software*: works by storing different fonts and text image patterns as templates. In this case, the software uses pattern matching algorithms to compare the text images, character by character, with the internal database.
- *Intelligent Character Recognition (ICR) software*: enables the capture and conversion of handwritten text into digital files using advanced methods that train machines to behave like humans through the use of machine learning and artificial intelligence software. A machine learning system called a neural network analyzes text at multiple levels by repeatedly processing the image. It looks for different image attributes like curves, lines, intersections, circles and combines the results of all these different levels of analysis to get the final result.
- *Intelligent Word Recognition (IWR) software*: works on the same principles as an ICR software, but processes whole word images instead of preprocessing the images character by character: the software compares the image of the written or handwritten word with a reference database to identify the word. The software analyzes various attributes of the word image: the general shape, height, width and other specific characteristics of its constituent characters. This type of software is primarily needed by companies that manage large volumes of various documents.
- *Optical Mark Recognition (OMR) software*: identifies logos, watermarks, trademarks and other images or text symbols in a document, being useful for documents that include elements that do not represent standard text: drawings, images, graphics, etc.

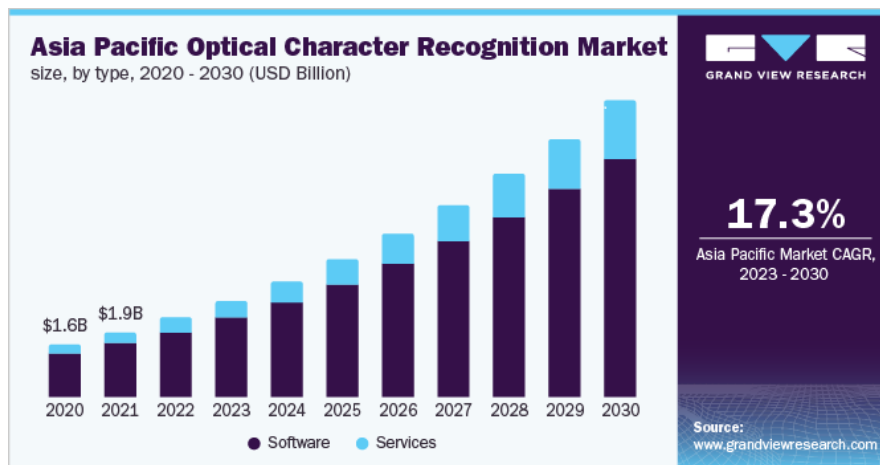
But thanks to advanced research in this field, all these variants of software solutions are continuously improved and augmented with technologies in the field of Artificial Intelligence to provide accurate results and to overcome limitations due to the quality of documents (for example old, stained or yellowed letters or documents).

There are many other OCR-based solutions, such as automating the processing of forms when a large amount of data is available in printed form, or translation applications that read texts through a smartphone camera. There are also vehicles that also use this technology: they automatically recognize traffic signs and inform the driver. The same goes for tools that capture credit card data via camera. Or the use of text-to-speech programs and OCR technology to support visually impaired people. With the help of this technology, authorities and companies automatically read addresses, personal data or license plates, validate passports, etc.

As options for use, the OCR software is available in three variants: it can be used online directly in the browser (if such software is needed less often), it can be used offline, by download, or can be a combination of both options. For example, cloud-based solutions are available in Microsoft OneNote or Evernote applications. And the OCRSpace app offers a free online OCR service that doesn't require registration.

Although considered a mature technology, OCR technology continues to evolve towards integration with other technologies in the field of Artificial Intelligence and Machine Learning to increase the productivity of retrieving unstructured data from audio documents, videos or photos. Today there is a trend towards the development of software solutions that combine OCR technology with intelligent applications, which allow the extraction of data from forms (available in archived form) in order to organize this data in digital format.

According to a report by Grand View Research, the integration of these technologies has allowed OCR software to become more accurate, faster, and more versatile; if the global optical character recognition market size was valued at USD 10.62 billion in 2022, it is expected to grow at a compound annual growth rate (CAGR) of 14.8% from 2023 to 2030:



<https://www.grandviewresearch.com/industry-analysis/optical-character-recognition-market>

Another area of integration of OCR technology is with cloud-based technologies (eg virtual collaboration platforms) to make necessary information available to partners. Also, because it offers the possibility of processing large volumes of documents, OCR technology continues to expand in many industries (telecommunications, tourism, logistics and transport, retail) and in areas such as healthcare or other health sectors, in the financial banking or legal field. Equally, this technology makes an important contribution in the field of education, by streamlining the workflow of physical documents and records on paper (diplomas, transcripts, school records, various records), simplifying the process of their digitization.

► **Artificial Intelligence (AI)** is an advanced technology implemented on computers that aims to replicate human intelligence and decision-making processes. In the context of IDP, AI is the general technology that automates the processing of information contained in documents; trained on extensive and representative data, AI models can analyze, interpret and extract relevant information from unstructured data, and can make their own decisions and predictions.

► **Robotic Process Automation (RPA)** is an advanced technology that uses software robots or "bots" to automate repetitive, rules-based tasks. While these tasks would be laborious and time-consuming to perform manually, by a manual operator, an RPA bot can perform them quickly and accurately, thereby simplifying data entry and eliminating errors.

Many RPA solutions are pre-built, easy-to-install software tools that work seamlessly with existing systems to access applications, enter data, calculate and complete tasks, or copy data between applications or workflows.

► **Machine Learning (ML)** is a subfield of AI that deals with the design and study of algorithms that can detect patterns in large volumes of data and interpret their meaning. In the context of IDP, Machine Learning Algorithms can recognize patterns and automatically extract relevant information or specific fields from invoices, forms or contracts, etc., to improve the accuracy and efficiency of document processing.

Although IDP technology has already undergone strong evolution, recent years have seen new trends and IDP is developing in various aspects. Intelligent data processing seems more and more oriented towards expanding its possible applications in various fields and its technology is evolving at the same pace. Today, IDP systems have numerous applications, both in the financial-banking sector and in other economic fields.

Developers are particularly focused on increasing the accuracy rate of IDP, which will cause systems to recognize an ever wider range of even complex documents such as graphs or tables. In this preparation phase, human input - that of experts in the sector of interest - is therefore fundamental. Finally, systems that involve learning based in part on user feedback could also be effective.

1.8 OCR tools to extract text from images or PDFs

1.8.1. Microsoft OneNote 2016

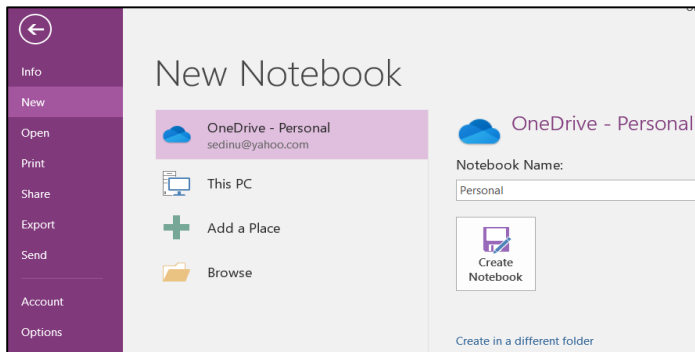
OneNote is a software component of the Microsoft Office package, dedicated to storing personal information in digital format, fulfilling the function of a digital notebook. Many of the functions contained in this application are made to make daily work easier: with one click, all necessary information is immediately accessible, information in documents can be updated, highlighted, annotated for changes, files are easy to retrieve, and connecting to the Internet facilitates online communication and collaboration between various users. OneNote is thus a software for cooperation and exchange of ideas and information between several users in the same data network, through which they have access to notes, drawings, sketches, audio comments, screenshots, etc. of others, for real-time collaboration.

An important functionality of OneNote is the included OCR solution, which allows users to recognize text from images, captures or documents. The OCR technology included in OneNote is limited to basic fonts (such as sans serif fonts such as Arial or Helvetica) and works better with higher resolution images, so it is only recommended for scanning and extracting a limited amount of text.

The included OCR application allows:

a) copying text from images:

1. Open the application OneNote → File → New:

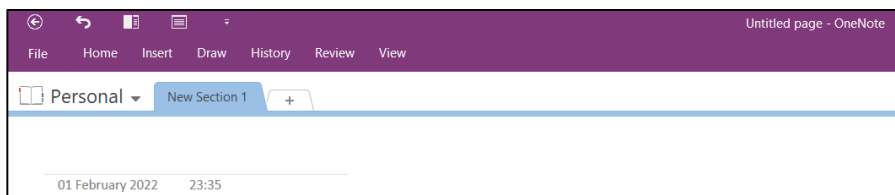


2. Choose the **place where the new notebook will be created** (for example in the cloud on OneDrive or locally on the computer).

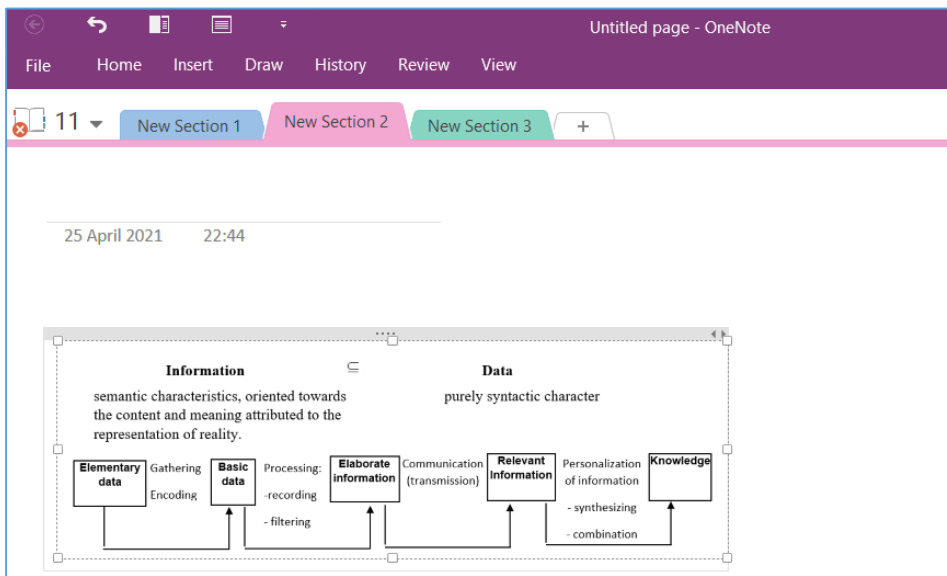
3. Enter a **name for the notebook** in the Notebook Name box. For example, type Personal.

4. Click the **Create Notebook** button to add the new notebook to the chosen location.

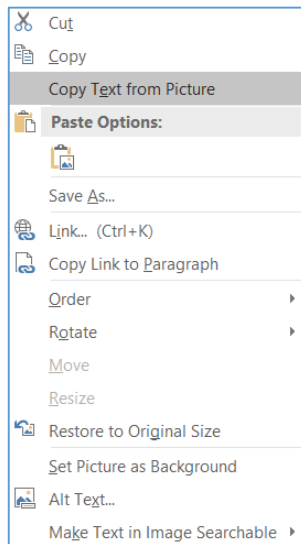
5. The new blank notebook opened in OneNote on the computer is visualized, with the chosen name:



6. Import the image file into the notebook:



7. **Right-click** on the image → from the list of options select the **Copy Text from Picture** option:



8. The text bits will be copied to the clipboard:

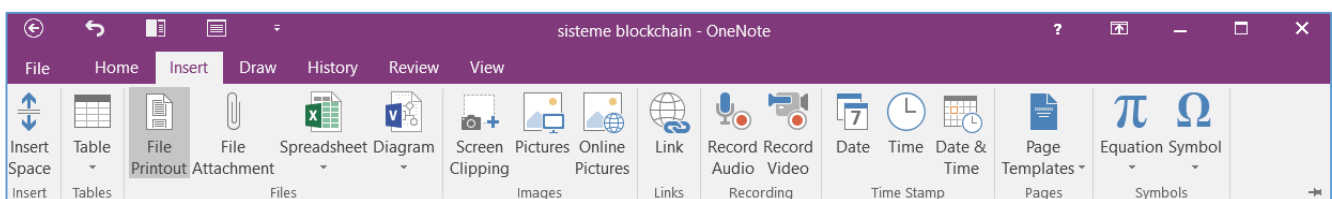
Information
semantic characteristics, oriented towards
the content and meaning attributed to the
Data
purely syntactic character
Knowledge
Communication
(transmission)
Relevant
Information
Personalization
of information
- synthesizing
- combination
representation of reality.
Elementary Gathering Basic Processing:
Elaborate
information
data
Encoding data
-recording
-filtering

9. The resulting text can be pasted into another application (Word, Notepad, etc.).

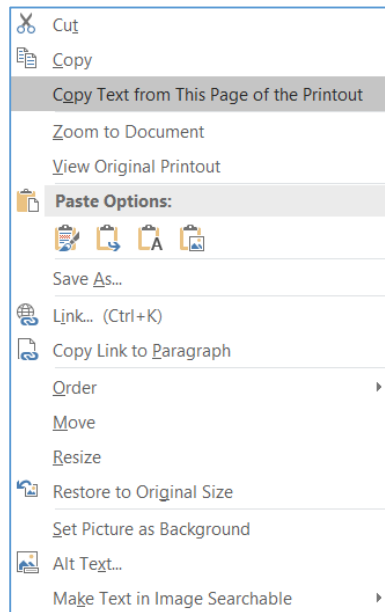
b) copy text from pdf files:

The first five steps are identical to those performed in case a).

6. **Insert** → **File Printout** → **import the PDF file** into the OneNote application:



7. **Right-click** in the pdf file → from the list of options select the option **Copy Text from This Page of the Printout**:



8. The text bits will be copied to the clipboard and **the resulting text can be pasted into another application** (Word, Notepad, etc.).

The OneNote application also has other useful facilities for data management in order to solve various problems. Another very important feature is the intelligent integration of notes with the rest of the programs in the Microsoft Office suite. For example, one can edit a document in Word or a spreadsheet in Excel, which can be embedded and then edited in OneNote.

There are some other important features, such as handwriting that is later converted to text or screenshots of online maps that are automatically saved in OneNote. Captures can later be edited, with specific instructions, to make them even more useful.

1.8.2. Cisdem PDF Converter OCR

Another useful application for migrating data between various formats, in the digital context, is the Cisdem PDF Converter OCR application, which allows converting files from and to PDF format, in a wide range of output file types: Word, Excel, PowerPoint, ePub, text, Keynote, RTF, HTML or other popular document formats. The application recognizes 17 widely used languages: English, French, Italian, German, Russian, Arabic and many others.

Application allows extraction of text, images, tables, etc. from images and PDFs that can be copied and pasted into other files.

It also has a built-in OCR function that allows users to recognize text from scanned or image-based PDFs; can convert PDFs and scanned images into searchable and editable formats.

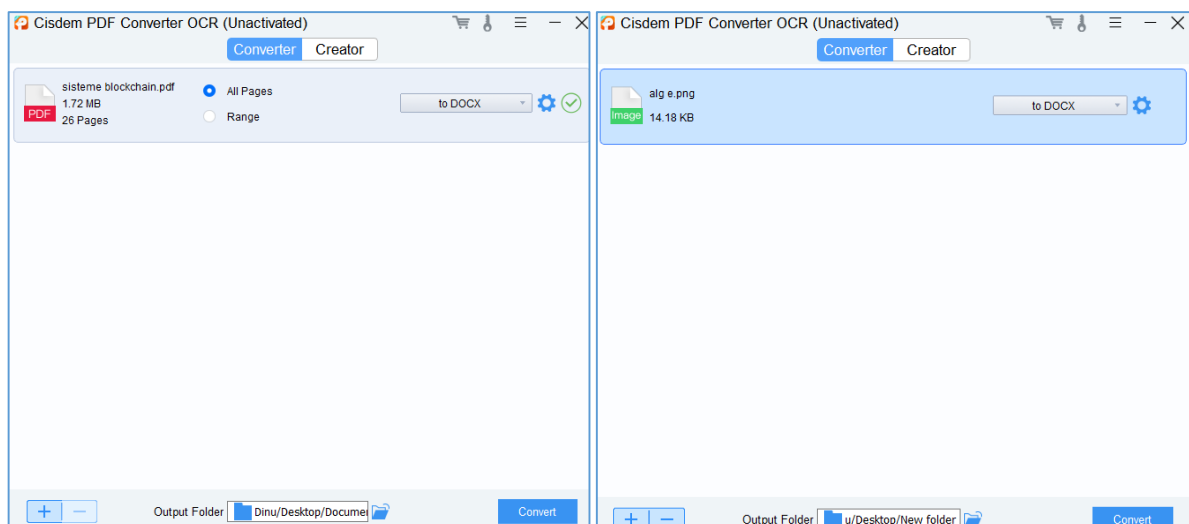
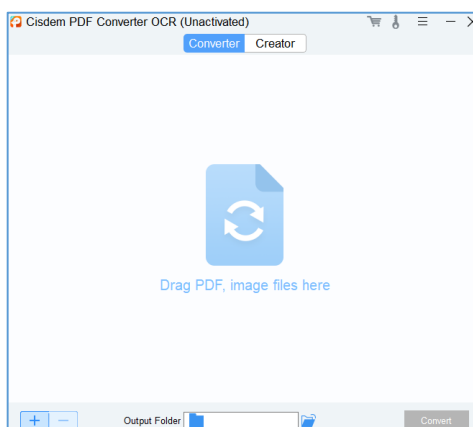
Note: the OCR module is not pre-installed, but it can be downloaded and installed.

A major advantage is that it allows simultaneous processing of several files, which is very advantageous when working with large volumes of data.

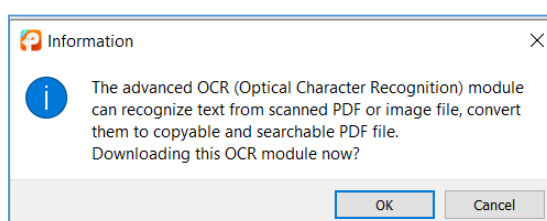
Step 1: The trial version of the application is downloaded and installed from the address: <https://www.cisdem.com/pdf-converter-ocr.html>



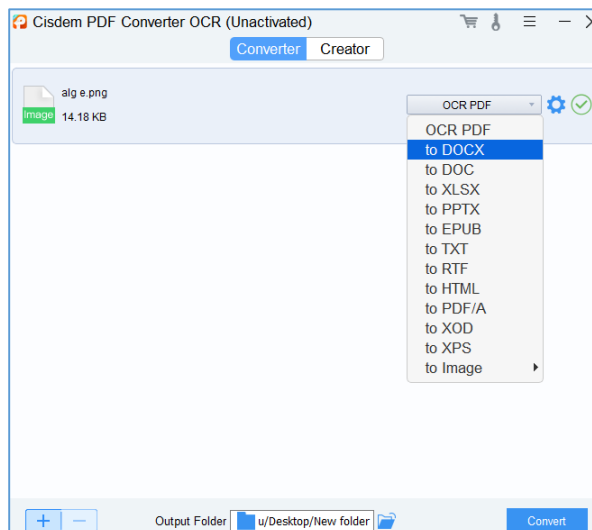
Step 2: **Add one or more PDFs (native or scanned) or images (BMP, PNG, JPG, TIFF or other common image formats) by dragging and dropping:**



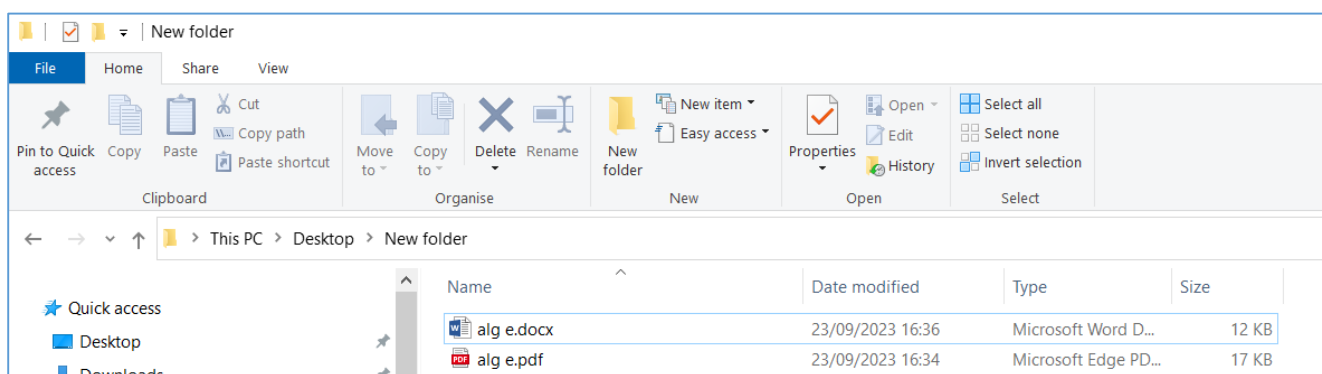
Step 3: Download the OCR module:



Step 4: From the output format dropdown list, select the desired output format, and from the Output Folder section select the destination folder → Convert:



Step 5: Open the file(s) where the save was made:



1.8.3. i2OCR.com (Online)

i2OCR is a free online, reliable and simple optical character recognition (OCR) software that extracts text from images and can be edited, formatted, indexed, searched or translated. This app does not require any email or registration to login and use the i2OCR converter.

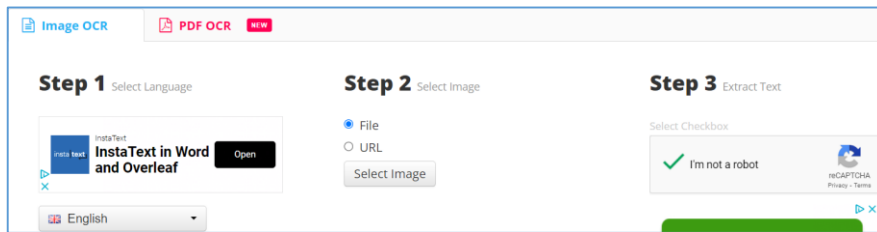
The application supports input image files of type TIF, JPEG, PNG, BMP, GIF, PBM, PGM and PPM. It supports over 60 recognition languages, major image formats, multi-column document analysis and free unlimited uploads.

Note: The i2OCR app is only capable of scanning images and PDF OCR.

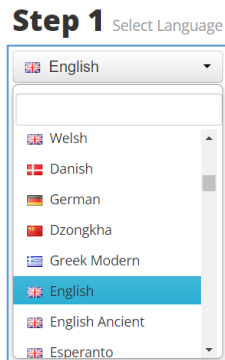
a) copying text from images:

Step 1: Access the application at: www.i2ocr.com

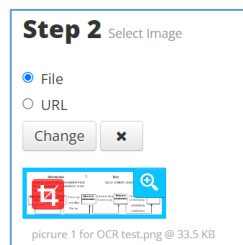
The interface contains a guide to use i2OCR to get quick results:



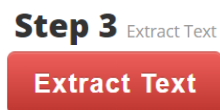
Step 2: From the list of options, select the desired OCR language:



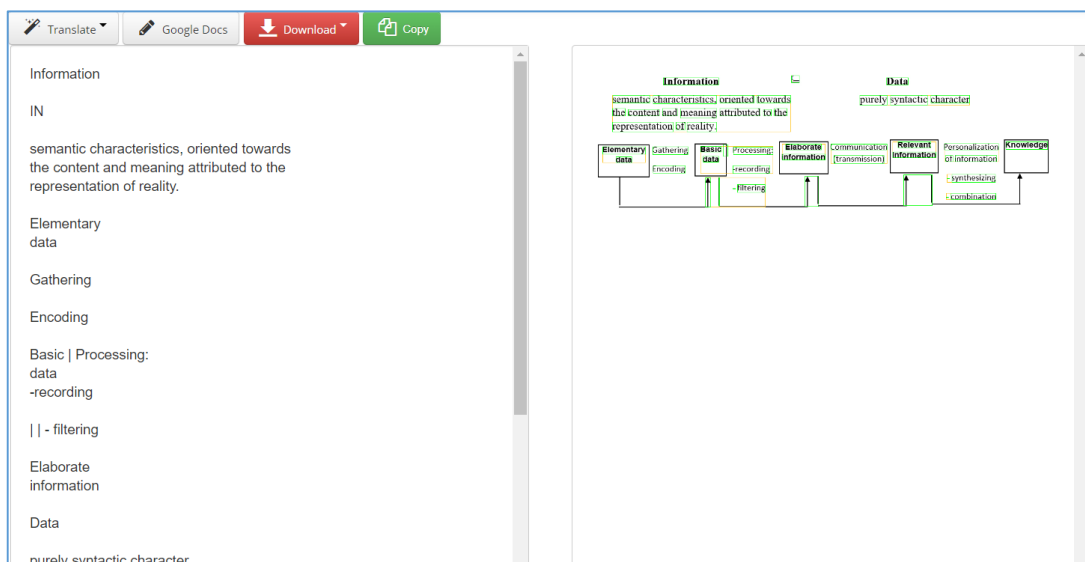
Step 3: Select the image file or the pdf file. The app allows to either upload a file directly or access it via an url (enter the URL of an image or PDF) to extract an image or PDF from it → Enter:



Step 4: Press the button to extract text from image:



Step 5: The i2OCR text editor opens where one can view the extracted text, which can then be translated, copied or edited:



b) copy text from pdf files:

The first three steps are the same as in case a)

Step 4: Click **PDF to image** which first converts a PDF to an image file:

Step 3 PDF to Images

PDF to Images

Step 5: Now there are two options:

- either one can **download the i2OCR text** found in the PDF (clicking on the button **Download Text**),
- or if the OCR result is not satisfactory, click **Extract Page Text** to retrieve text from the image of a PDF file:

Download the text we found in the PDF. It can be the original text or a previous OCR text, which may not be accurate. If you are not satisfied with the results, you can select a page below to OCR using i2OCR

Step 4 Extract Text of Selected Page

page 1 page 2 page 3 page 4 page 5 page 6 page 7 page 8

Extract Page Text

1.8.4 OnlineOCR.net (Online)

OnlineOCR.net is a free optical character recognition service that can recognize and convert scanned documents into various types of editable text files: Microsoft Word, Excel, RTF, HTML and TXT. As input formats, the application supports PDF, JPG, PNG, BMP, TIFF, PCX, GIF and ZIP. The app supports 46 different languages and does not require account registration to use the service. Only up to 15 PDFs/images can be converted per hour and no more than 15 pages per file, and the imported file is limited to 15MB.

To upgrade to a larger file size or to scan more than 15 files per hour, the user must register with an account and pay for advanced features.

Step 1: Access the application at: <https://www.onlineocr.net/>

The interface contains a guide to use OnlineOCR.net to get quick results:

1 STEP - Upload file
SELECT FILE...

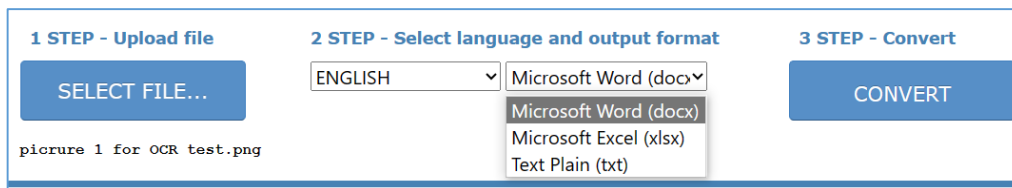
2 STEP - Select language and output format
ENGLISH Microsoft Word (docx)

3 STEP - Convert
CONVERT

Max file size 15 mb.

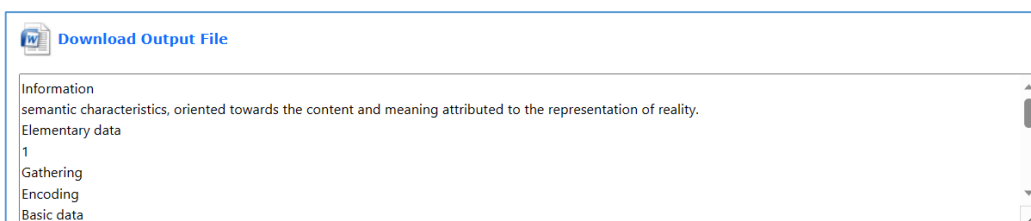
Step 2: Click **Select file** button to upload a PDF file or a photo in JPG, BMP, GIF, and TIFF format. Multiple files can also be uploaded at once.

Step 3: From the list of options, **select the desired OCR language** of the uploaded file and **choose an output format** from Word, Excel, and TXT:



Step 4: Press the button **Convert** to start recognizing and converting the files.

Step 5: The OnlineOCR.net text editor opens where one can view the extracted text, (which can then be translated, copied or edited), along with a download link of the output file.



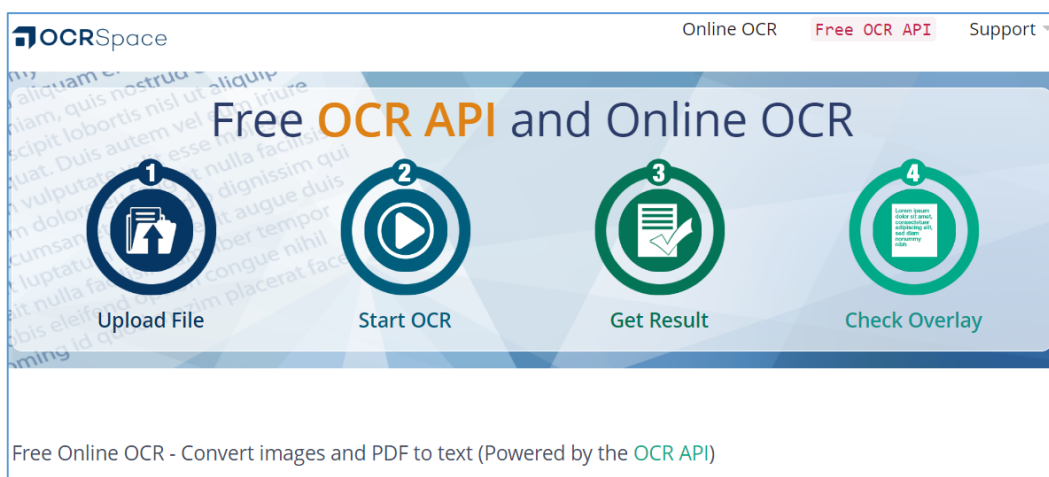
1.8.5 OCR.Space

OCR.Space is an online OCR program designed for Windows, Mac and Linux platforms that supports PNG, JPG and PDF image to text conversion. In addition to uploading the file from the local hard drive, the app also allows to upload the URL of the file for conversion.

It allows to choose a specific language for that document: it supports up to 20 languages, including Arabic, Simplified Chinese, Traditional Chinese, Croatian, Czech, Danish and so on. This application also does not require any email or registration.

Step 1: Access the application at: <https://ocr.space/>

The interface contains a guide to use OnlineOCR.net to get quick results:



Step 2: On the search bar **enter the URL** or select **Choose file** to upload the file:

Step 3: From the list of options, **select the desired OCR language** of the uploaded file and **select the recognition or orientation** if needed. Besides, users can customize the menus according to the preferred result, for example, they are enabled to create searchable PDFs by changing how the PDF is displayed in **Create Searchable PDF**;

Step 4: Press the button **Start OCR!** to start recognizing and converting the files.

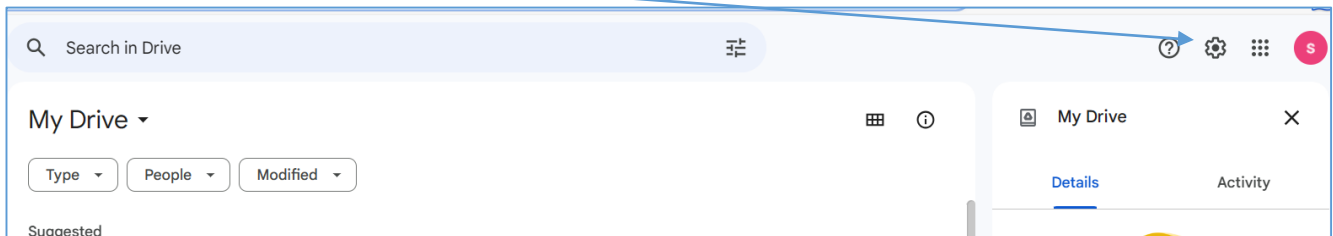
Step 5: The application will then process the document and provide the converted text in the selected format. In the right column, the recognized texts will be presented in the output format "Text" and "Json". One can click "Download" to access the text PDF. This converter offers a preview of the OCR results so that one can check whether the text PDF meets the standard:

1.8.6 Google Docs

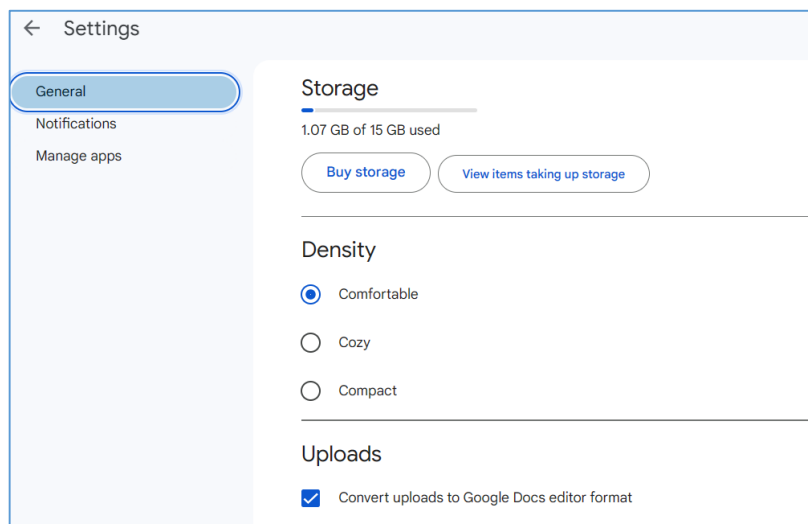
Google Docs allows using OCR recognition to make PDF files and images editable to copy the text inside. In addition to the ability to upload files of any type and size, Google Drive has a very interesting feature, the OCR function, which allows to extract text from a PDF or an image.

To activate the OCR function on Google Docs:

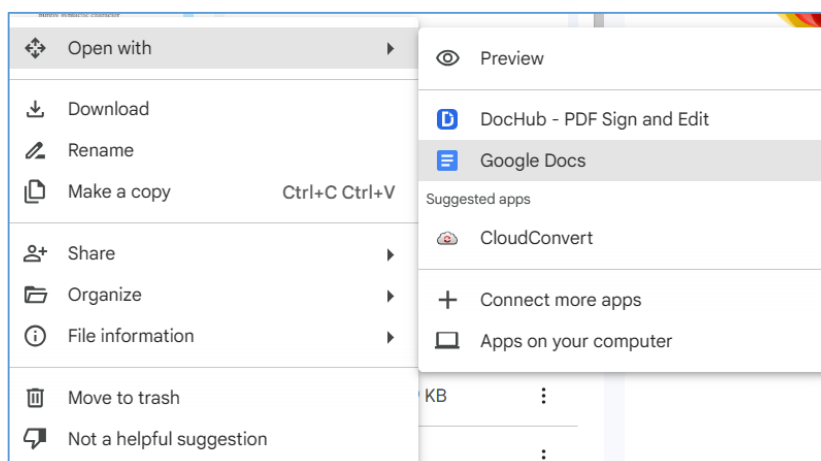
1. Open the Google Drive page: <https://drive.google.com/>
2. Press on the gear icon at the top right → Settings:



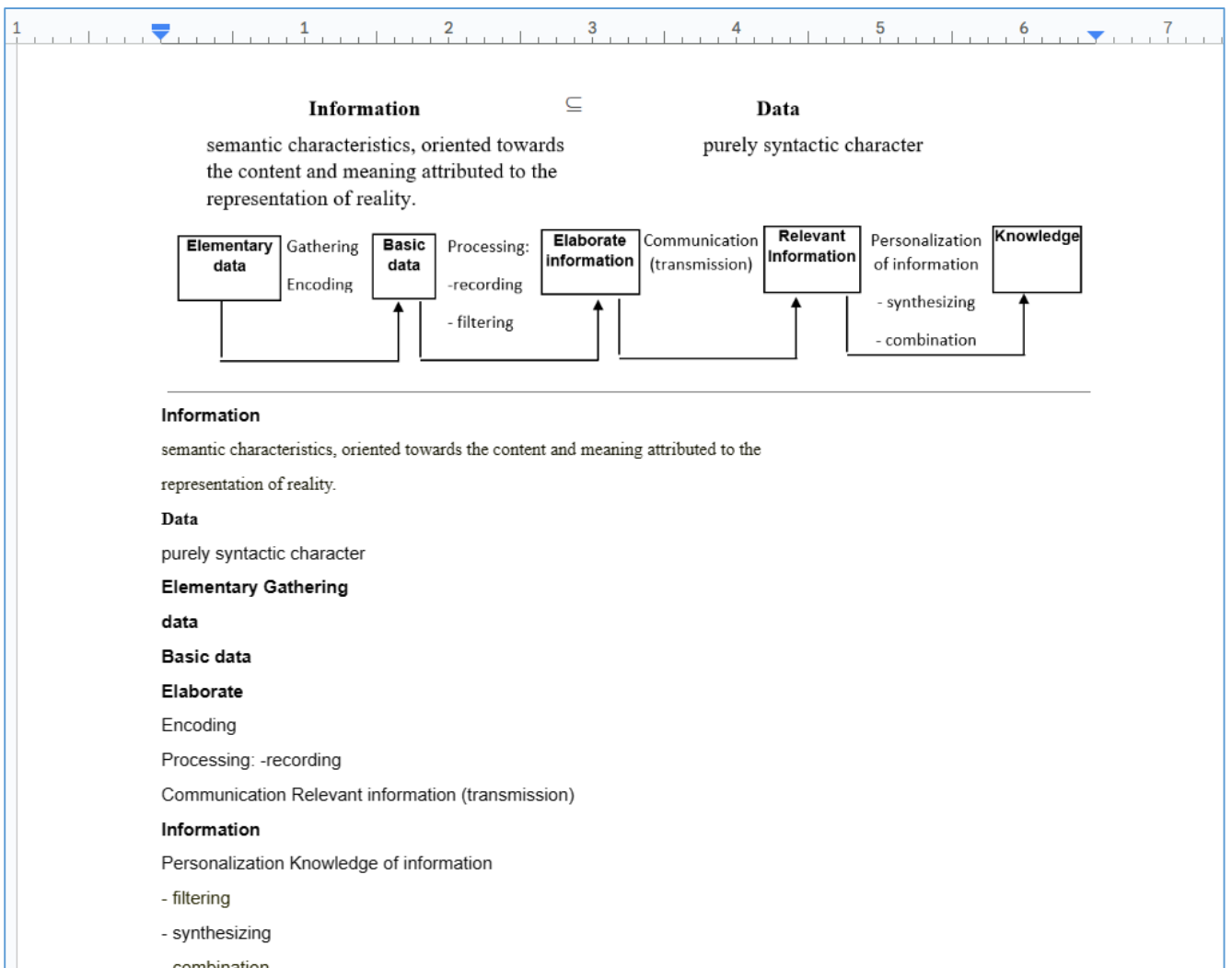
3. In the window that opens, check the option **Convert uploaded files to the Google Docs editor format**:



4. Upload a PDF or image with text to Google Drive → **right-click on the file** just uploaded → use the **Open with - Google Docs** option:



5. From the uploaded PDF or image, a text sheet will be obtained that can be edited directly with Google Docs tools. The text file can then be saved back to PDF on the computer or to a file:



Chapter II. Business process modeling for Business Intelligence

2.1 Business processes - the key element in the analysis of any business

Since a large volume of information to be processed by means of business intelligence applications comes from business processes, conceptual clarifications on the notion of economic process are needed.

Every business involves processes, which help to identify the tasks that are important for the respective business; processes effectively represent how things are done within an organization. According to the Rubinian Legal Dictionary: [www.rubinian.com], business processes include the set of activities that can achieve company-specific objectives and are specific to economic life: production, distribution, consumption, investment processes, etc., in which the combination of production factors takes place for obtaining goods or services, the entire system of activities in which all economic agents participate. The business process helps the efficient use of resources but also the efficient communication between people, functions, departments, to fulfill specific tasks.

Every process has inputs (human resources, technological resources, materials or other types of resources) for the development of the activities that comprise it, as well as also exits: products, services, information, financial assets or other types of assets are expected. And, according to the ISO 9000:2000 standard, a process is “a set of activities mutually related or interacting, which transform entry elements into results”. Based on this definition, the process-based approach highlights how the desired results can be achieved more efficiently if activities are grouped, given that these activities, in turn, must allow a transformation of some inputs into outputs.

The notion of process can be interpreted within a hierarchy of abstractions that identify principles, processes, procedures, resources, rules, activities, tools and methods (Fig. 2.1).

A *process* can be carried out within one or more *procedures* whose implementation is based on specific *tools* and *methods*. A procedure is the specific way of carrying out an activity, what should be done, who should do it, when, where, how it should be carried out, what materials, equipment, documents should be used and how it should be controlled and checked. So, if a process defines what is done, and a procedure defines how to do it. Procedures group sets of *activities* performed by a wide range of participants.

Activities describe sequences of *operations* applied to *resources* under certain conditions (*rules*). Not all activities carried out in a company can be interpreted as processes. To determine whether the activity carried out by a company is a process or a sub-process, it must meet the following criteria [Ángel, 2010]:

- The activity contains inputs and outputs, and customers, suppliers and final products can be identified.
- The activity has a clear mission or purpose.
- The activity must be able to be broken down into well-defined tasks.
- The activity can be characterized by applying the process management methodology (time, resources, costs).

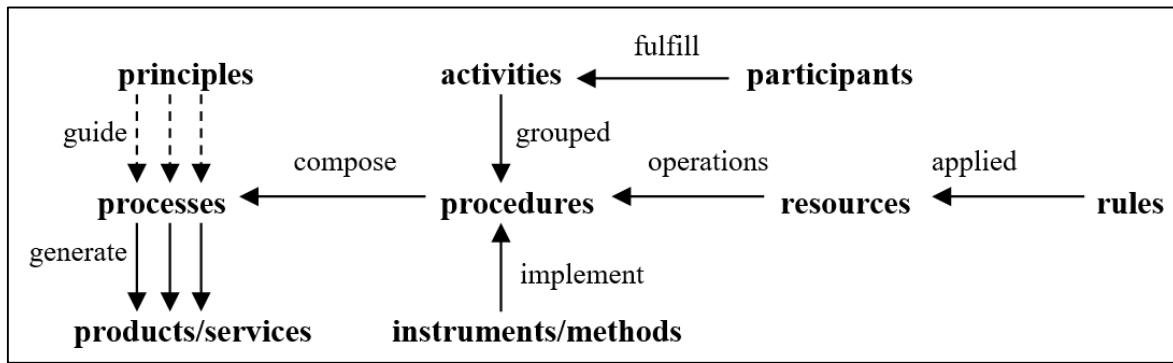


Figure 2.1: The notion of process within the abstraction hierarchy

The processes represent the framework for carrying out the organization's activities: its members fulfill multiple roles within several processes, which in turn are carried out in one or more organizational subunits or within several distinct organizations.

The current taxonomy uses several criteria for classifying business processes. Within an economic organization, economic processes can be broken down into four groups of important activities (Fig. 2.2) that capture the development in time and space of an economic phenomenon [Muller, 2010]:

- ▶ **Customer-oriented processes:** repetitively performs a well-defined set of activities regarding direct interactions with customers;
- ▶ **Processes carried out in the stage of creating a product (group of products):** it represents the joint effort of different functional areas and departments (marketing, engineering, production, sales, services) for the transformation of work objects into finished products.

This group of processes can be decomposed into three subgroups, strongly related, according to the following levels of competence:

- *Commercial:* defines how to obtain a product by defining product specifications based on requirements and feedback from customers; this will allow the orientation of processes towards quality, towards satisfying the needs of customers and stakeholders;
- *Project management:* concerns the realization of the product, according to the agreed specifications, the proposed completion time and the availability of the allocated resources;
- *Design:* technical design process further associated with architectural design, which aims to translate the requirements into appropriate architectural models;

▶ **Processes specific to technological and human management:**

- *human resource management*, as the main resource of the organization, which develops information, knowledge and skills;
- *the management of the processes of conception, assimilation and adaptation of new application results*, embodied in high-performance technologies and of training employees in these new technologies;

- ▶ **Planning processes:** transform external constraints into internal constraints, defining the organization's guidelines, forecasts and plans broken down by periods.

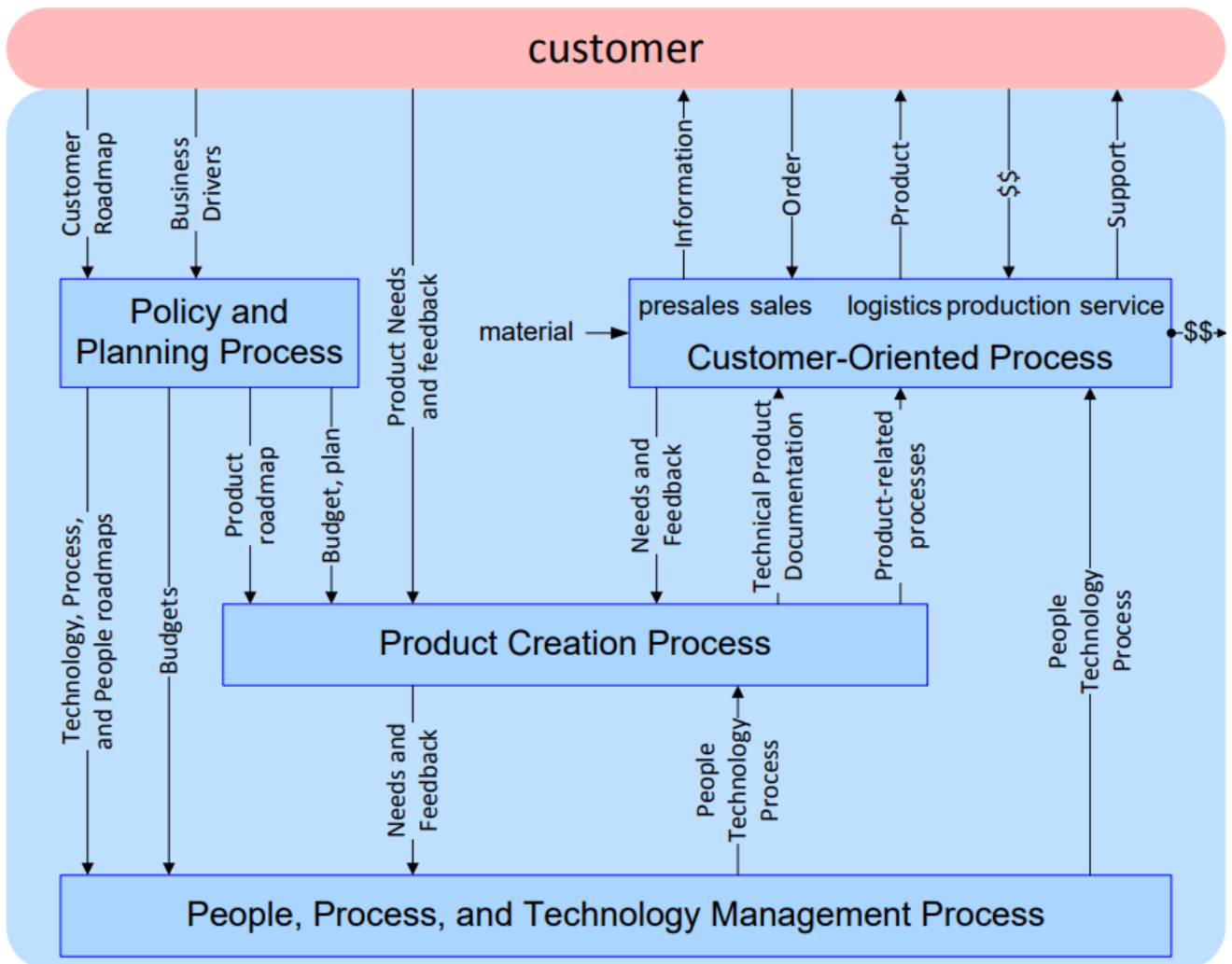


Figure 2.2: Simplified decomposition of an economic process [Muller, 2010]

According to the areas involved, the processes are divided into:

- ▶ **Macro processes:** global processes, which go beyond the boundaries of a department;
- ▶ **Micro-processes:** A more specific process, composed of a series of detailed steps and activities and that can be performed by a single person; a micro process can become a sub-process of a macro process.

Weske [Weske, 2007] differentiates the processes according to the objectives pursued:

- *fundamental objectives:* identifies the main goals of the organization, related to a long time horizon;
- *general objectives:* deduced from the fundamental objectives (departmental objectives or objectives by functions);
- *derived objectives:* they are part of the general objectives, have a concrete definition, and employees who perform restricted work processes participate in their achievement;
- *specific objectives:* works and actions that contribute to the achievement of derived objectives;
- *primary objectives:* describe the tasks (operations) to be performed by each operator.

The performance of an economic organization can be appreciated depending on the extent to which the various tasks and activities are carried out or not in accordance with these objectives. To achieve the organization's objectives, activities and tasks are identified and distributed in a way that establishes individual roles and contributions to the organization's performance. This process of designing activities is called task allocation.

2.2 Process approach - important element in business modeling

The process approach involves the association of processes with the organization's strategy and planning. Thus, all activities must be treated as processes, and it must be established how the outputs of one process represent inputs to another process. This new orientation represents a systemic vision of the activities; the approach as a system of related processes of the organization's activities generates a common image, an integrated vision of the organization, available to all employees.

Highlighting the intercorrelations and mutual effects between the processes allows the transparency of the process structure and the continuous improvement of the organization's activity.

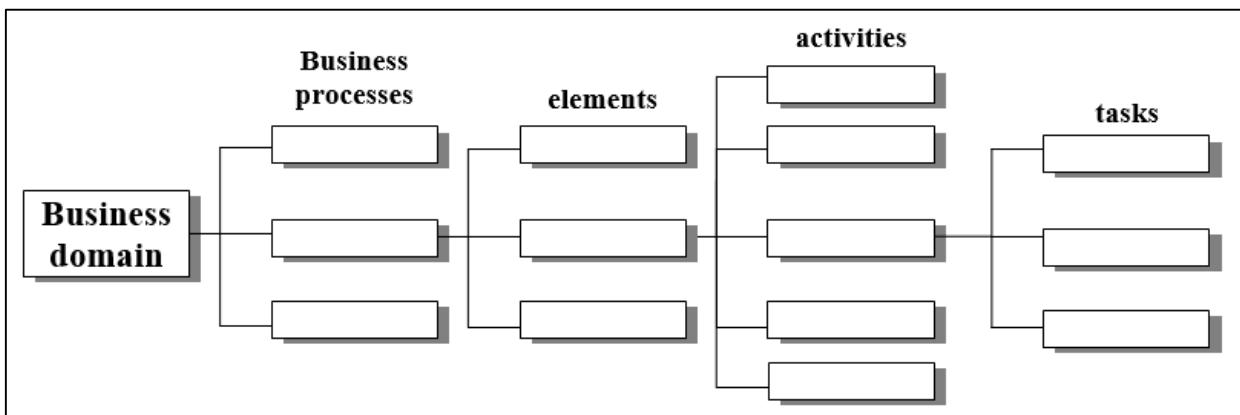


Figure 2.3: The process described by hierarchies of component elements

From a structural point of view, in [Carley & Gasser, 1999] and [Ferrer et al., 2010] the processes are described by hierarchies of elements that facilitate the analysis and description of relationships, influences and causalities within the entire network of processes (Fig. 2.3). For example, the customer service process can be structured in four elements: marketing, sales, orders and logistics. Furthermore, sales can be defined by three activities: invoicing, customer support and complaint management. Finally, the activities are defined at the level of specific assignments and tasks.

The task is the smallest unit of individual work and defines: the primary objective that must be achieved, the time, duration and deadline (start/finish), price, quality requirements.

The assignment defines a lot of tasks necessary to carry out a certain process; tasks are executed periodically or continuously, consume inputs and allocate time and resources to produce outputs.

The activity consists of homogeneous attributions, of a certain nature (technical, organizational, economic, administrative) that compete to achieve a certain objective.

This organization by processes, activities, attributions, up to the level of tasks, is part of the description of the organization. The set of organization tasks can be considered the task environment (or problem space) for that organization [Carley & Gasser, 1999]. Thus, the task representation model is an important step in the organization modeling process.

Between the component tasks there can be different dependencies that generate a temporal sequencing of them at the level of the organization. In [Bell & Kozlowsky, 2002], four types of dependencies are identified at the level of process flows, in increasing order of the degree of complexity:

- **correlated flow**: the structure in which tasks and activities are executed separately, and their results, taken together, are necessary to execute a different task (Fig. 2.4 (a));

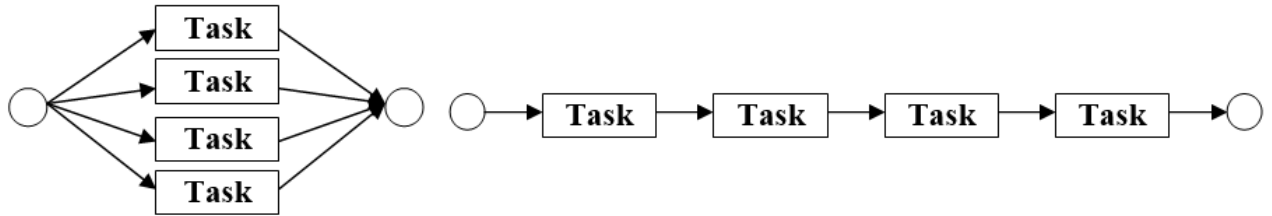


Figure 2.4: (a) Structure of a correlated flow

(b) Structure of a sequential flow

- **sequential flow**: tasks and activities must be executed sequentially, in a certain order (Fig. 2.4(b));

- **reciprocal flow**: between tasks there is a mutual dependence if the tasks are mutually dependent and must be executed at the same time (Fig. 2.5);

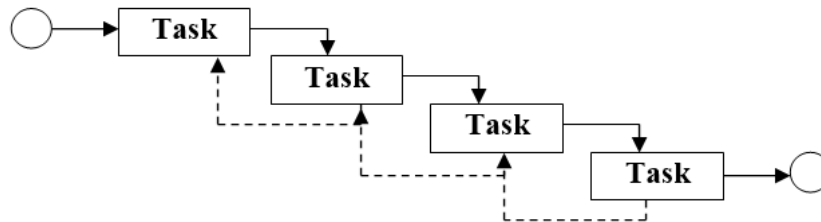


Figure 2.5: Structure of a reciprocal flow

- **intensive flow**: it is the most complex and interdependent flow and describes the situation in which the tasks are executed simultaneously and collectively; it is the case of a project in which there is an overlap between the design and realization stages, the realization of the first stages of the project being started while the final stages are still in the design phase.

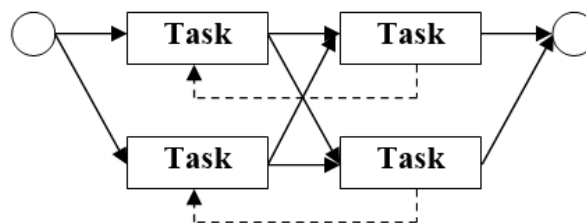


Figure 2.6: Structure of an intensive flow

Apart from dependencies, the set of tasks can be characterized based on the *degree of repetition, volatility, confusion and complexity* [Carley & Gasser, 1999]. Depending on the degree of repetition, tasks can be repetitive, quasi-repetitive (same type of tasks, but there are different details depending on the instances) or non-repetitive (each task is unique). Enterprise transformation can be represented as a sequence of instantiations that describe the states of the processes during the different stages of development. These instantiations describe process operations, resources, relationships, capabilities, etc., include comparative analyzes and the introduction of performance metrics. Volatility reflects the rate of task change, confusion expresses the extent to which different tasks have the same result, and complexity measures the amount of processing at the task level.

2.3 Representation and modeling of business processes

Although the evolution of the corporate environment and globalization increasingly require the alignment of technology with organizational strategy, this is a desirability that is still a challenge for many organizations. The corporate way of working of increasingly complex and comprehensive businesses requires that the terms, processes and technical activities within the company are introduced to employees and other members who are not used to these aspects. To do this, business processes are approached using standard methodologies that facilitate their understanding.

The basis of business process management is their explicit representation, with activities and execution restrictions. In this sense, process modeling is a tool that provides insight and understanding of existing processes by graphically visualizing (as a diagram, map or model) the processes of an organization to logically represent the structure and functionality of the business, showing the relationship between its processes, subprocesses and activities, according to the natural flow of activity execution, built from observation and study of the real world.

Modeling allows the representation of these elements that make up a process, through a standardized formalism, in order to understand it adequately.

This stage of representation is followed by the validation of the model, testing its reactions under different conditions to ensure that its operation will meet the global requirements set in terms of quality, performance, cost, reliability and so on. The ultimate goal is to document and improve the process flow, to understand and analyze the work done, to transform and, when possible, to automate business operational processes and initiate further improvement actions.

Methodologies focused on business processes, such as Business Process Modeling (BPM) define, enable and manage the exchange of information in organizations, providing a view of the entire business process, which includes employees, customers, partners, applications and databases. For the design, representation, analysis and control of operational business processes, BPM uses a set of methods, tools and business software tools: Business Process Management Systems (BPMS). BPMS represents a category of information systems that includes a set of software tools used to define, implement and improve business processes and that respect a group of technical characteristics necessary to apply the BPM concept.

Traditionally, business processes are done manually, guided by the knowledge of the company's personnel and assisted by the regulations and procedures established by the organization. But to

coordinate the activities involved in the business process, companies can gain certain benefits by using a standard language or notation to represent and model them.

As the development of technology has advanced, several BPM notations and standards have emerged used to represent a workflow, but the most well-known representation standard is Business Process Modeling Notation - BPMN.

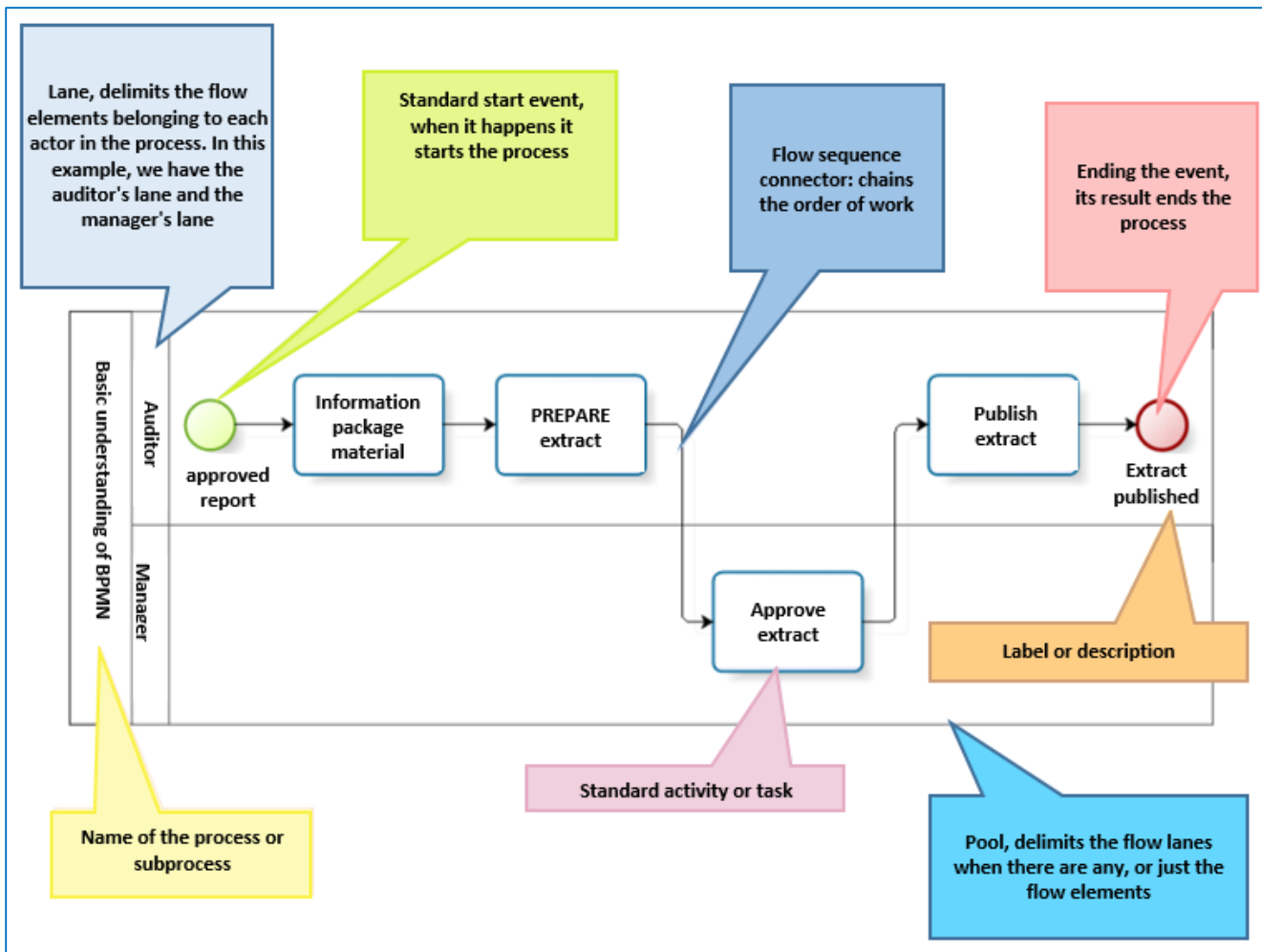


Figure 2.7: Expanded Business Process Flow represented in BPMN

BPMN is a tool used worldwide to support the use of BPM, being a standard for modeling business processes, with the objective of mapping the business processes of organizations in a workflow format, through graphic representations. BPMN enhances organizational BPM efforts by providing a common graphical language, facilitating communication and better understanding of business processes in both business and IT. Although it is mainly used in the "business world", the BPMN language can also be used in the technical, industrial field, in the modeling of medical or agricultural work flows, etc.

It was developed by the Business Process Management Initiative and is currently maintained by the Object Management Group - OMG (which is an open, non-profit international consortium dedicated to the standardization of technologies), after the merger of the two organizations. BPMN provides a graphical notation for specifying business processes in a business process diagram -BPD, based on a flow diagram technique very similar to Unified Modeling Language -UML activity diagrams. BPMN

offers resources that aim to respond to the most diverse types of processes, from the most generic to the most specific, such as: administrative, financial, operational, quality assurance, software development, product or service development, etc.

BPMN is the first international standard for modeling business processes, being adopted in practice by important companies in the IT industry (IBM, SAP, Oracle.)

2.3.1 Diagrams and design elements of the BPMN language

Depending on the need for documentation, three levels of BPMN notation usage can be accessed:

- *the descriptive level*: it is used to document the processes of the value chain;
- *the analytical level*: it is used in projects where the objective is to analyze and improve processes;
- *the executable level*: it is used in the modeling processes that will be automated by any BPMS tool.

The latest version of the BPMN standard, BPMN 2.0, is a standardized graphical language based on an XML structure, which uses a set of specialized graphical elements to describe a process as a whole and how a sequence of activities in a process can be detailed in order to achieve an objective. The language allows simple diagrams to be drawn, representing the progress and sequence of tasks: for example, activities are represented only by simple rectangles and decisions like diamonds.

In BPMN modeling, diagrams are important because they graphically exemplify the current process and allow knowing the time in which each activity is carried out. Also, through the functionalities that allow the description of the attributes that accompany the graphic elements, the diagrams allow the definition of the responsible persons and their activity within the process. In addition, they contain process simulation tools that allow the identification of unnecessary activities and questionable situations (repetition of tasks, downtimes, deadlocks, etc.).

Typically, a BPMN modeling tool consists of:

- a central drawing space allowing modeling.
- a palette of graphic elements that can be moved in the drawing space by “drag & drop”; these elements can sometimes be colored in a personalized way via a graphic color palette.

There are five main categories of BPMN elements:

1. **Flow objects** represent the basic elements of process diagram. In turn, they can fall into one of the following categories: Event, Activity, Gate (or Gateway).
2. **Connecting objects** have the role of connecting flow objects to each other or to other types of objects. The three types of connection objects are: Sequential flow, Message flow and Association.
3. **Partitioning objects (swimlanes)** establish subgraphs in the process flow, with the aim of logically separating certain portions of it, depending on the entities participating in the implementation of the process. They can be of two types: Container (pool) and Lane.
4. The **data** are necessary to highlight the data that the activities need or that are produced by them. Data can fall into four categories: data object, input data, output data and stored data.

5. **Artifacts** are created to provide additional information within a diagram. There are two standard artifact types: the group and respectively textual annotations, but both the language and the modeling tools provide the ability to add any other custom user artifacts needed to understand the model.

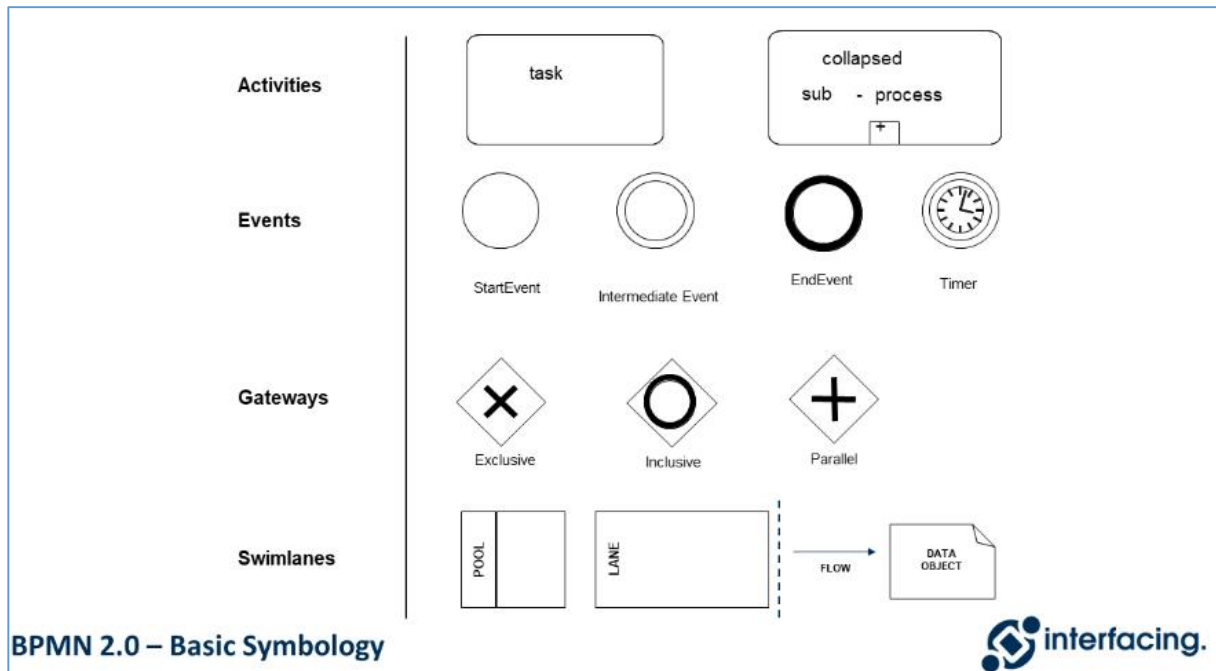


Figure 2.8: Main categories of BPMN elements

<https://www.interfacing.com/bpmn-2-0-symbology>

► **Activity:** is a generic term for the work performed in a company. It is represented by a rectangle with rounded edges. Activities represent a minor part of a process and can be of two types: atomic (Tasks) and non-atomic (Sub-Processes).

- a **Task** represents one activity (the lowest level of detail shown in the diagram).

- a **Sub-Process** represents an activity that consists of one or more activities and is represented by a rectangle, so with a plus sign (“+”) in the lower central part.

An activity can have multiple input sequential flows. Each input sequential flow is independent of the other input sequential flows.

A sub-process is an activity whose internal details have been modeled using Activities, Gateways, Events and Sequential Flow. It is a graphical object within a process, but can also be "opened" to display a process at a higher level of detail. So, subprocesses can be nested (collapsed or extended) or reusable:

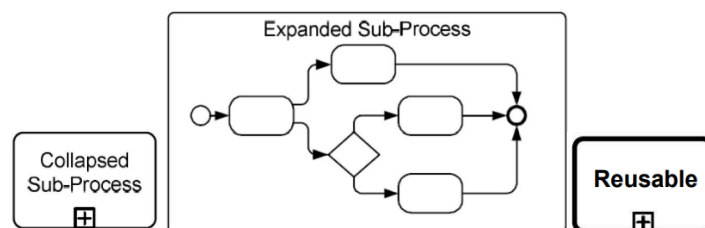





Figure 2.9: Types of subprocesses

► **Event:** it is something that happens during the course of the process and affects the flow of the process. Events usually have a cause (trigger) or a result. They are divided into three types: start, intermediate and end, according to the moment in which they affect the flow.

 Start event: It represents the starting point of a process.

 Intermediate event: Stops the flow until a condition occurs or triggers exception actions.

 End event: Indicates when a process ends.

► **Gateway:** control the flow of a process (the divergence or convergence of the sequential flow); are represented by a diamond and determine branches, bifurcations, combinations and mergers of the process. Gateways can be data-based or event-based and do not represent work.

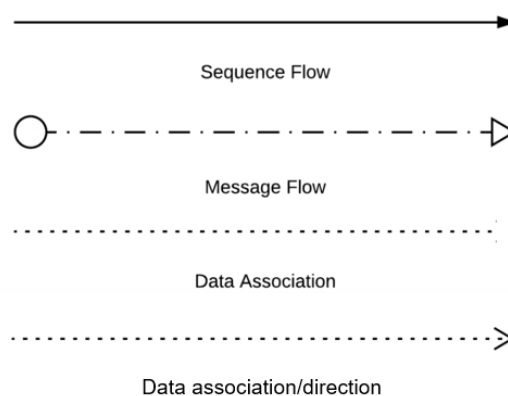


► **Connecting objects:** are responsible for connecting the flow objects of a process, defining the order of execution of the activities, that is, they create the basic structure of the business process.

- **Sequential flow:** shows the order of events, activities and decisions that are made within the process.

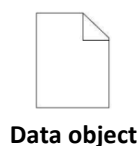
- **Message flow:** indicates the flow of communication between the different entities of the processes (e.g., an organization and its suppliers).

- **Association flow:** its function is to associate artifacts (data and information) with different objects in the diagram, or with with flow objects (activities, events, gateways).



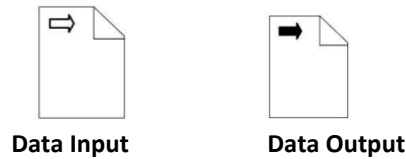
► **Data** provide information about what the Activities need or what they produce within the process. The basic data notations are:

- **Data object:** provide information about what the Activities produce.



Data Objects usually define the inputs and outputs of Activities. Although Data Objects do not affect the structure and flow of the Process, they are closely linked to the execution of the Activities. The Associations indicate their direction (input or output).

- **Data Input/Output:** are data that are specifically used as input or output parameters in some activities. They can also be defined as collections.



- **Data store:** provides a mechanism for activities to retrieve or update previously stored information so that it exists beyond the scope of the process and is available to other processes.



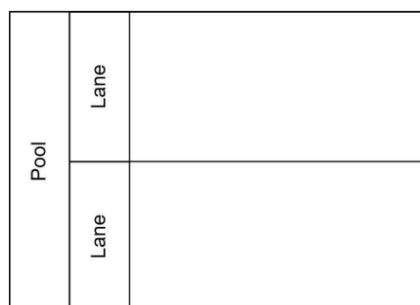
► **Swimlanes:** The term "Pool" was obtained by expanding the swimlane analogy: a swimming pool has lanes. BPMN uses swimlanes to break down and organize activities on a diagram. There are two main types of swimlanes:

- **Pools:** act as containers for processes, each representing a participant in an interactive or collaborative Business Process Diagram. A BPMN diagram can describe the processes of different participants, so each pool represents a different process and each participant has their own pool. A participant is defined as a general business role, such as a buyer, seller, shipper, or supplier.

A pool contains only one process and its name can be considered as the process name.

Note: Sequential flows cannot cross pool boundaries.

- **Lanes:** often used to represent internal business roles within a process, lanes actually provide a generic mechanism for partitioning objects within a pool, based on the characteristics of the process or elements. Typically, a pool defines a process, while the lane defines the participants that will perform that process.



Note:

- 1) Sequential flows can cross lane boundaries.
- 2) In the process that is being diagrammed, there can be no elements that are not placed in a lane or pool.
- 3) Lanes can be nested:

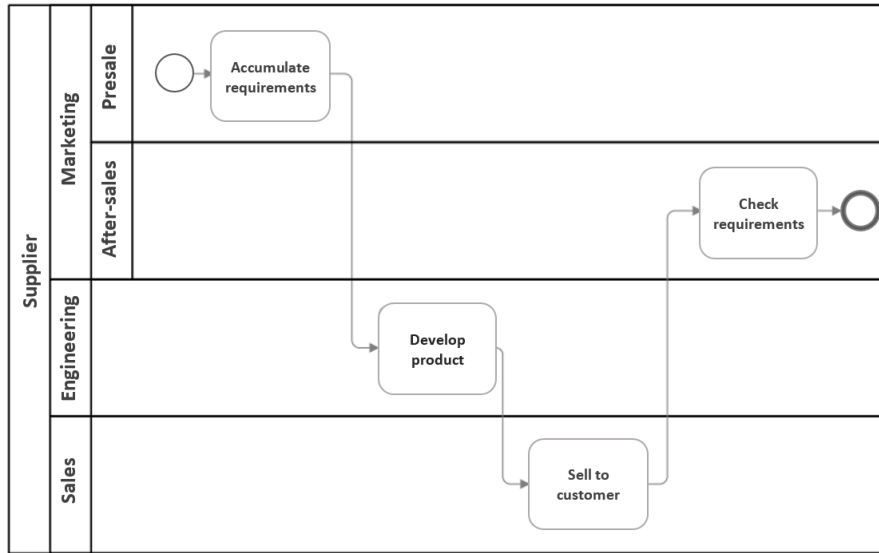


Figure 2.10: Nested lanes within a pool

- **Artifacts:** are graphic objects that provide support information for the elements of the process, without directly affecting its flow.
- **Groups:** allow grouping of elements. A dashed line with dots and dashes is used to surround a particular group of elements to indicate their association.

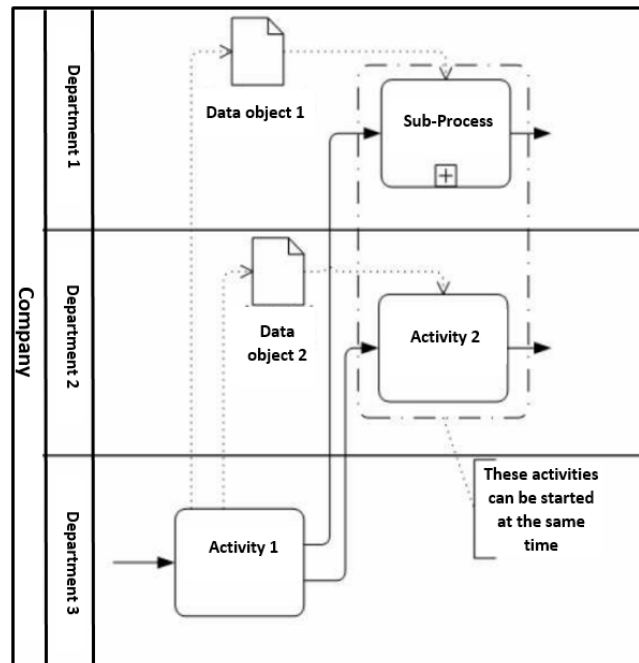


Figure 2.11: Group example

Groups do not affect process flow or add restrictions. They just group activities and other elements together to make important blocks of operations visible.

A group is simply a useful graphical mechanism for categorizing objects. Sequential flow and message flow move across group boundaries transparently.

Groups are simply positional and therefore a group is allowed to cross two lanes, as in the following example:

- **Text annotations:** text notes that can be associated with any element.

Text annotations provide the modeler with the ability to add more notes or descriptive information about a process or its elements. Text annotations can be attached to any object in the diagram or can float freely anywhere in a diagram. The text of a text annotation is accompanied by an open box that can appear on either side of the text:

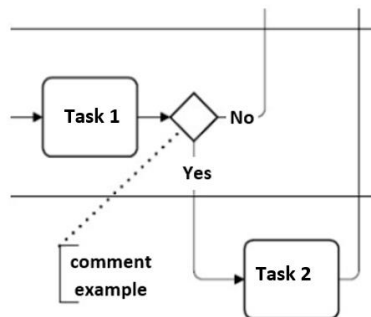


Figure 2.12: Text annotation example

2.3.2 BPMN Best Practice rules

1) The diagram must be carried out within the pools. If several processes are described in a diagram, they must be differentiated, using separate "pools". At least one pool must exist and there must be as many pools as processes.

The order of pools and lanes must occur according to process flow, therefore the first pool must always contain the start point of the process

2) All activities, events and Gateways must be connected through sequential flows.

3) Although in BPMN modeling, start and end events are optional, to avoid misinterpretation, these events should be used in every process and subprocess. In addition, if these events are used, they must be used together (so one cannot use only the start event or only the end event).

4) If there are flows in the diagram that end in the same end state, then they must be merged to the same end event. Conversely, if there are different end states (success and failure end states) in a process or sub-process, then they should be distinguished with separate end events.

Since a process can have one or more final events, it is recommended to use different names, corresponding to their final states;

5) The branching of flows is done through Gateways. Flows should not be branched using tasks.

Gateways must not be used to separate and join at the same time. But the same type of gateway must be used to split and join the flow.

6) One not add a completion event immediately after an outgoing sequential flow leaving a Gateway. In this case, a task must be inserted before leading to completion.

7) Process names should clearly describe their main purpose. In general, short names or abbreviations should not be used.

Event tags are useful when there are multiple start and end events. They should be labeled so that the diagram can be clearly understood.

Gateways must have names that indicate the condition that is being evaluated in the decision.

8) Diagram Orientation: the appearance of a diagram is an important factor that determines its understanding by readers. The standard for arranging a diagram requires that the processes shall be modeled from left to right, so that it extends downward and to the right:

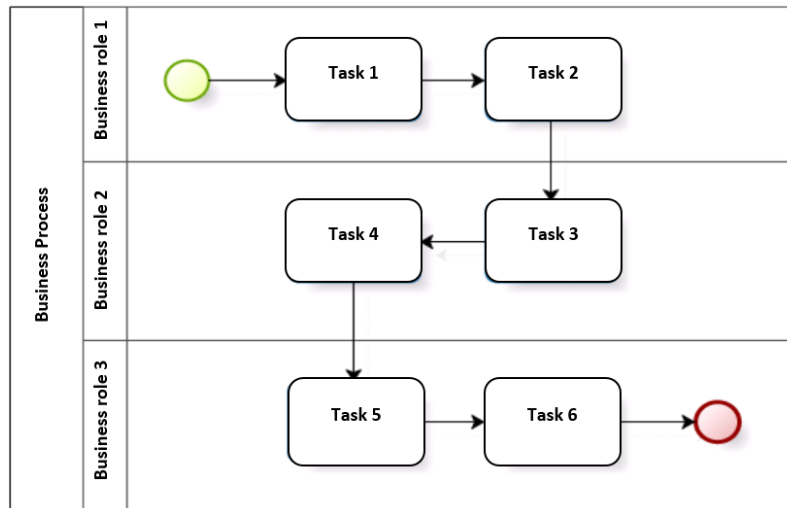


Figure 2.13: Incorrect orientation

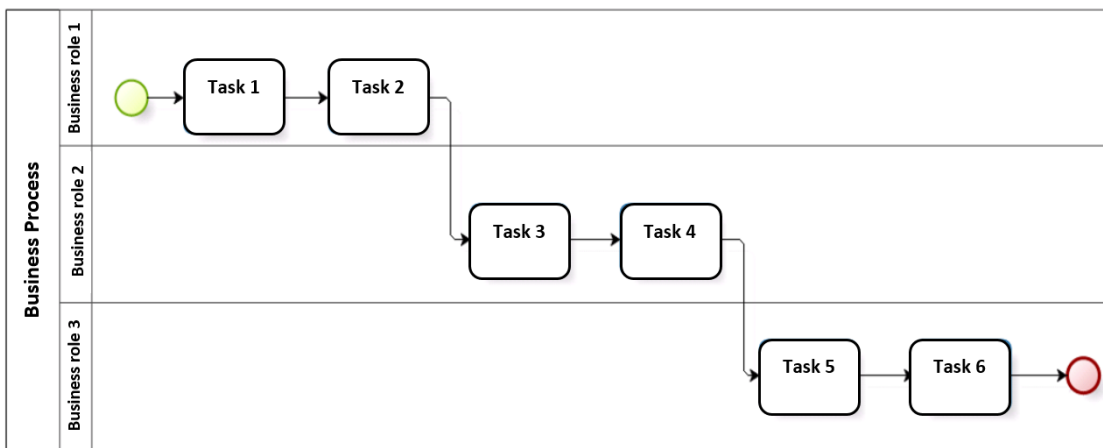


Figure 2.14: Correct orientation

9) For good readability and understanding, the diagram should have a uniform appearance, keeping a single format of font, colors, sizes, labels, etc.

It also needs to make efficient use of the chart space within the pools, as otherwise the flow chart can be difficult to appreciate in one view. In addition, the flow of a process without the layout of zigzag movements are indicated, but horizontally aligned and distributed in similar proportions.

Chapter III. Bizagi software - a tool based on the standardized set of BPMN symbols used for modeling and automating business processes

BPM projects have methods, techniques and software to design, control, analyze, document and improve operational processes, taking into account human, organizational and technological factors.

Companies gain tangible benefits by using BPMS software systems to coordinate the activities involved in Business Process Modeling. The BPM strategy is supported by BPMS tools which allow the modeling, execution and optimization of business processes. By definition, BPMS is a set of software tools that facilitate the definition, implementation and improvement of business processes, using for this purpose a rich set of graphic elements to represent a series of practical situations that can be encountered in process flows.

Its main functions are:

- Modeling and automation of business processes.
- Generation of process documentation.
- It allows the simulation of the processes carried out to measure their performance and be able to correct errors.
- Information integration.
- Deployment of applications that support the process.
- Graphically representation of the relationship between the different stages or phases of the process.

There are many BPMS tools with different types of free, open source or proprietary licenses. An electronic resource for modeling business processes is the Bizagi Modeler software. Designed by the Bizagi company, it is free and uses BPMN notation to represent processes. Bizagi has a very active community where there is a forum for each module and provides very complete documentation (provides a very complete user guide), video tutorials, version updates are very active adding new features and making product improvements.

Designed for descriptive, analytical and execution modeling of business processes, it enables modeling of business flows, supports the processing of sufficient documentation related to the process and allows publication of all this documentation in various formats, including Web format, which allows for greater publicity of the activities practiced by a company.

The BIZAGI platform in the cloud has the following components:



Figure 3.1: BIZAGI components in the cloud

https://help.bizagi.com/bpm-suite/en/index.html?get_started.htm

► **Bizagi Modeler:** the tool that allows the modeling of business processes. The processes are documented and diagrammed following the BPMN notation standard.

It also offers the possibility to import previously created diagrams in other tools such as Visio

► **Bizagi Studio:** the component that allows to automate processes that are built with Bizagi Modeler. It includes the definition of the data model, the user interface, validation rules and other elements that allows to shape business behavior in a system.

► **Bizagi Automation:** is the component that allows to execute automated business processes.

3.1 Bizagi Modeler: application interface and structure

Bizagi Modeler allows the simulation of the business flow to facilitate the analysis of events both in relation to time and in relation to the specific activities involved. For this purpose, it uses an intuitive graphic environment to organize the different processes and the relationships that exist in each stage.

Figure 3.2 shows the interface of this software:

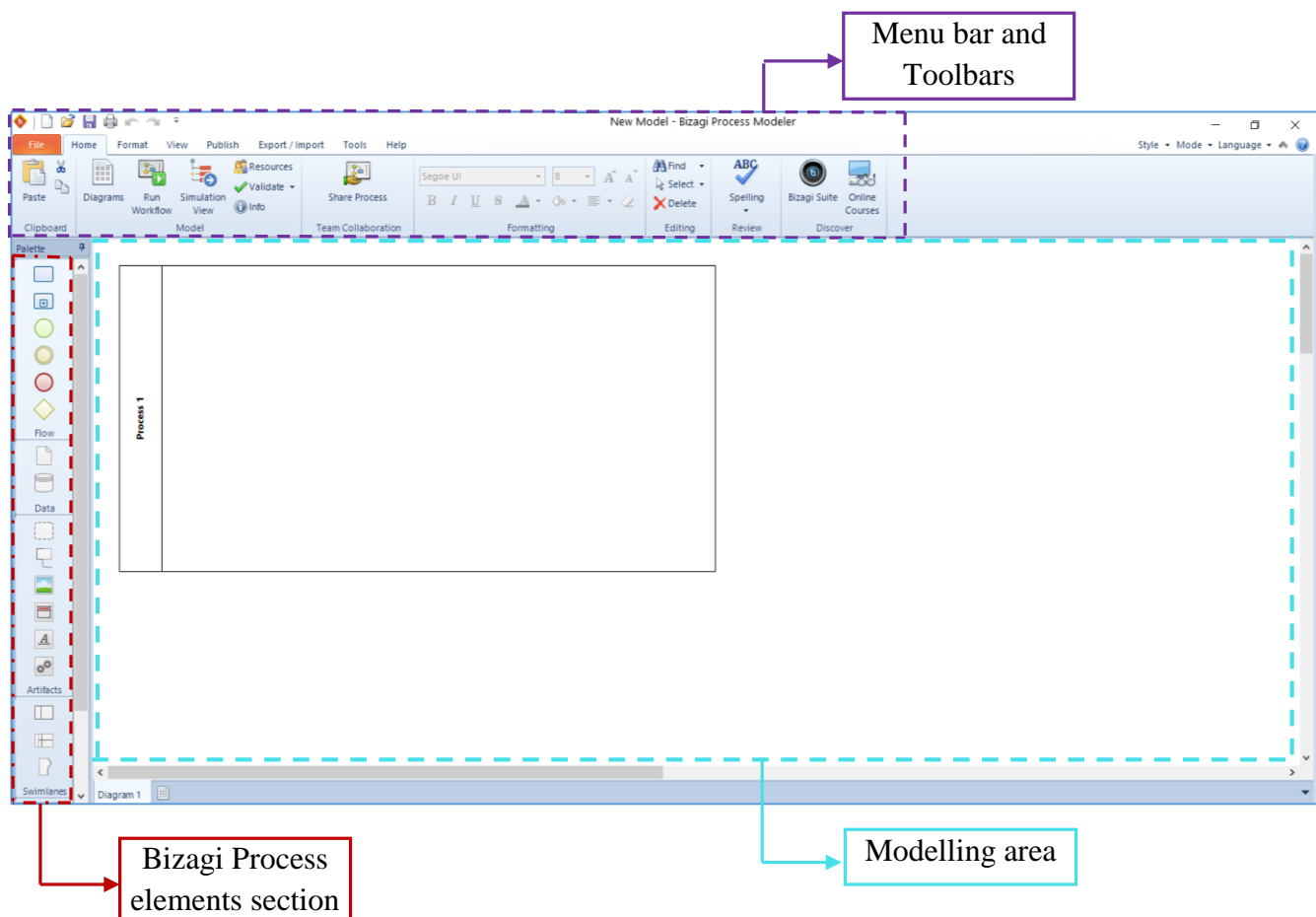


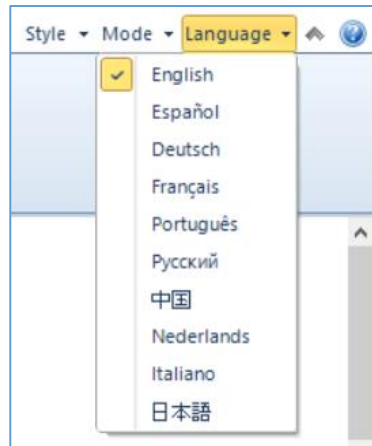
Figure 3.2: Bizagi software interface

The intuitive interface of the program allows a quick understanding of its operation mode, in congruence with the BPMN modeling idea. Bizagi allows designed graphics to be exported to image files, Word files, Portable Document Format (PDF)files, XML, etc.

■ **Setting up the application:**

► **Language** – to change the tool’s language:

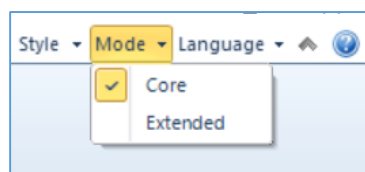
1. Access the “**Language**” menu located in the top right corner of the application → **select the desired language:**



2. **Close all Bizagi windows and open the application again** (for the selection to take effect).

► **Mode** – to change the operating mode:

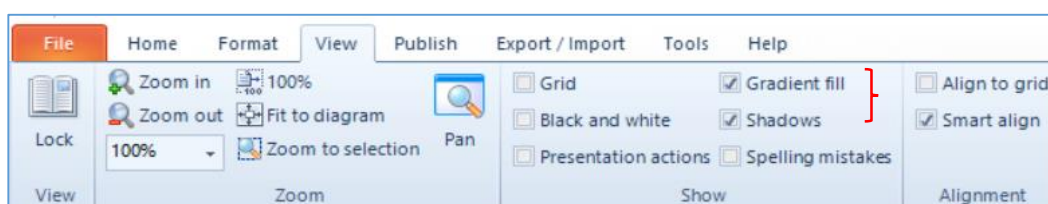
Access the “**Mode**” menu located in the top right corner of the application → select the “**Extended**” operating mode so that all flow elements, data, artifacts, swimlanes and connectors to become visible in the palette:



If this selection is not made, only the main elements will be visible in the selection palette.

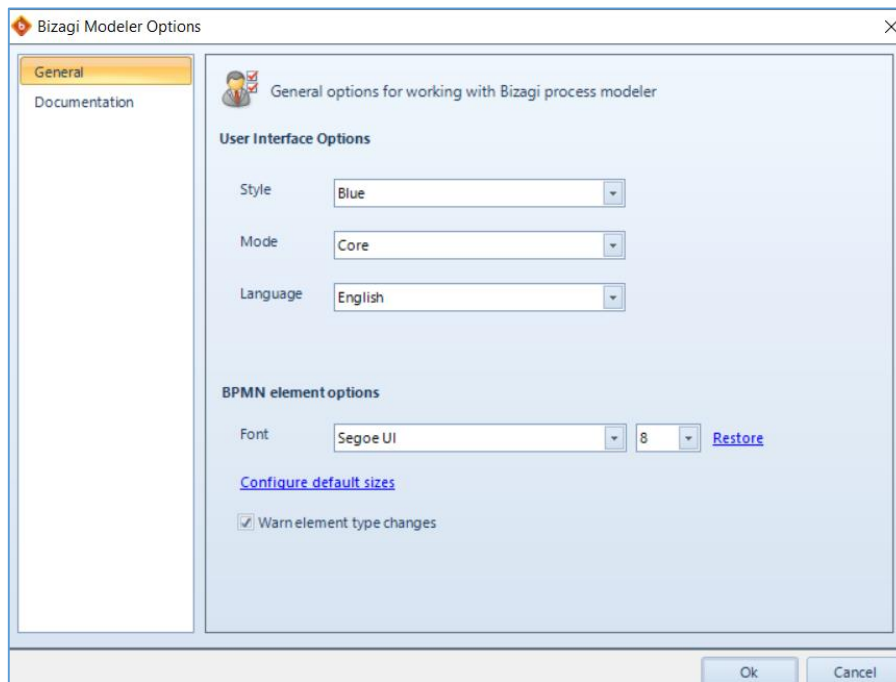
► **Gradient and Shadow** – to add a shading or gradient fill effect for the elements and background color:

View → check the **Gradient fill** and **Shadow** (in the **Show** group):

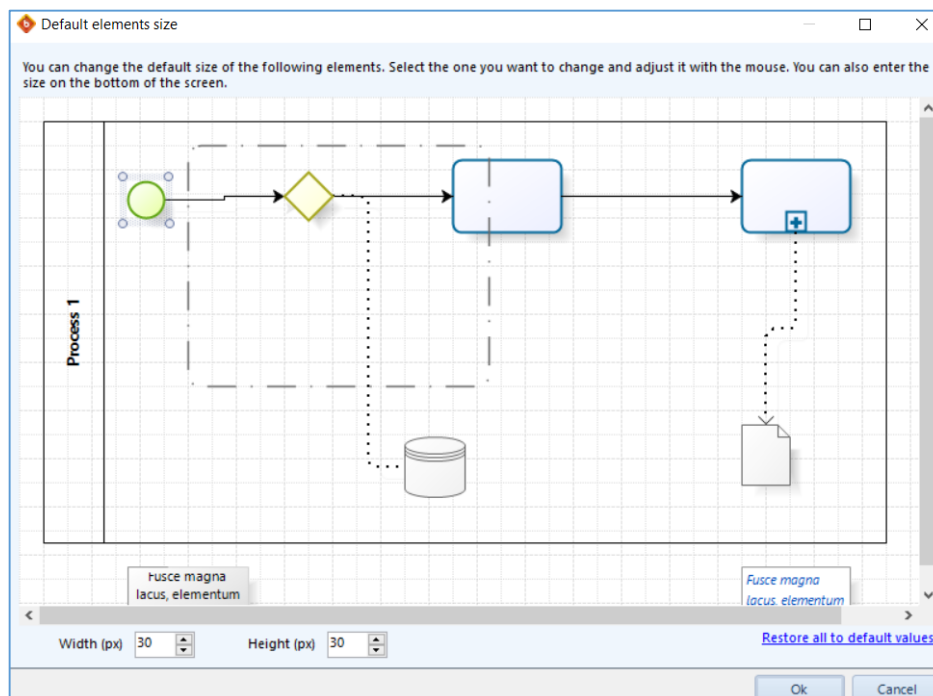


► **Size of elements** – to define a standard size for the diagram elements:

1. **File** → **Options** → the box **Bizagi Modeler Options** that opens allows **choosing the font**:



2. Pressing **Configure default sizes** allows to **configure the default dimensions for the elements of the diagram**:



3. **Select** each type of **diagram element** to **configure its default dimensions**.

One can choose, for example, the following dimensions for the diagram elements:

Event – width (40 px) and height (40 px);

Gateway – width (60 px) and height (60 px);

Activity – width (90 px) and height (60 px);

Sub-Process – width (90 px) and height (60 px);

Data Object – width (40 px) and height (50 px);

In modeling business processes with the help of Bizagi software, the following process-related considerations come into play [Benedict et al., 2013]:

► **Types of business processes:**

- **Primary Processes:** represent the essential activities that an organization carries out to fulfill its mission. These are the processes that primarily represent the raison d'être of an institution.

- **Support Processes:** designed to provide support to primary processes, other support processes or management processes. They do not directly generate value for the external customer, but they are fundamental, as they increase the ability to execute the processes they support.

- **Management Processes:** are those related to measurement, monitoring and control activities, ensuring that the organization operates in accordance with its objectives and performance goals. They also measure the performance of other processes.

► **The process structure or process architecture:** is the set of all the organization's business processes structured hierarchically, in levels, and organized in an effective way, so that they can be understood, communicated and optimized.

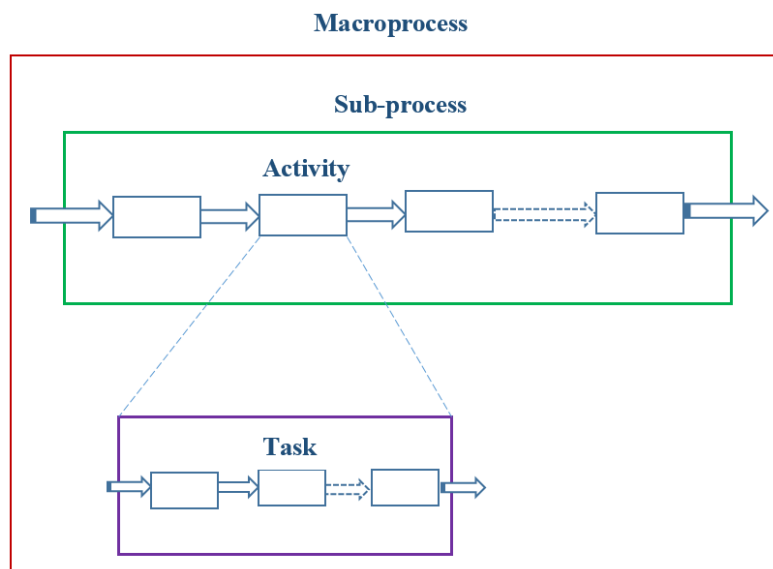


Figure 3.3: The process hierarchical structure

- **Macroprocess:** macroprocesses are actually large sets of activities through which the organization fulfills its organizational mission.

In general, a process is subdivided into sub-processes and these, in turn, are divided into activities. In this context of use (subdivided into other subprocesses), the process can be interpreted as a macroprocess. The figure above represents the hierarchy of this division:

The macroprocess is the highest level of process within an organization, and analyzing the process from the top down, i.e. starting from the perspective of the macro, provides a summary view of the complete flow diagram. Subsequently, representations can be made through increasingly detailed levels, from the overall picture of the process (macroprocess), which will be broken down into hierarchically organized levels (subprocesses) up to the activity level. In addition, each identified activity can be represented in detail, up to the level considered sufficient for analysis.

Regarding the implementation rules, a macroprocess must comply with the following requirements:

- must be contained in a single pool without lanes and this pool will be titled with the process name.
- will contain start and end events.
- must contain a maximum of eight descriptive sub-processes, and each sub-process contained in the macroprocess will represent a step (a stage). Each stage should contain activities from the same context, to facilitate the sequential and logical understanding of the process as a whole.
- all subprocesses of the macro process must be configured for the reusable type, "Reusable Subprocess", because with this configuration, it is possible to further design the analytical sub-process with pools and lanes.

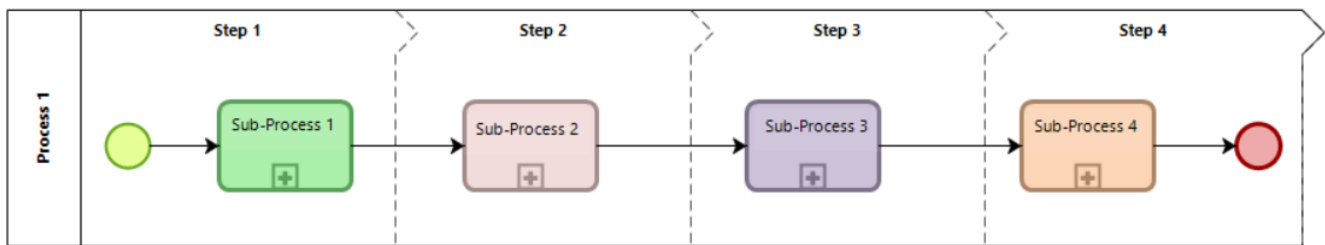


Figure 3.4 – Steps 1 to 4 of a macroprocess

Note: Whenever possible, it is suggested to start and end each step only with activities, avoiding that it starts or ends in a decision process (gateway).

- **Process:** is the set of interconnected or interactive resources and activities that receive inputs, add value and transform them into results (services/products).

According to BPM CBOOK 3.0 (2013), a process is the aggregation of activities to achieve results, such activities being performed by humans or machines.

Processes have a well-defined beginning and end, in a clear and logical sequence of interdependent actions that generate results.

- **Sub-process:** is a greater level of detail in the process. It is a process included in another process, that is, a group of complex operations that achieve a specific objective in support of another process.

Sub-processes are very important because they provide the possibility of hierarchically diagramming a process, detailing it at different levels.

Regarding the implementation rules, a sub-process must comply with the following requirements:

- all activities in a sub-process detail will have the same color as that sub-process, but gateways, events and connection objects will remain in the default color defined by the Bizagi application.

- the name of the sub-process will always start with a noun.

- all sub-process diagrams will have start and end events.

- **Activity:** is a set of actions that describe step by step the stages of a process or sub-process, generally carried out by a certain organizational unit and that produces a certain result.

- **Task:** it is the most operational level, it represents an action in the process that can be performed by a person or a system. It is the lowest level of detail in a work.

- **Levels of detail in the representation of the processes:** they are differentiated according to what is expected to be obtained through modeling. Consequently, depending on the requirements of the job, the processes might be described at various levels of detail and complexity.

In the hierarchy of process representation complexity, there are three different levels: the process diagram, the process map and finally the process model, the diagram being the most synthetic level, the map being the intermediate level of detail and the model being the most analytical level of detail.

- **Diagram:** is an initial and simplified representation of the process to be modeled. It demonstrates the basic flow focusing on the main activities, which are placed in order. It does not handle exceptions or failures in the process.

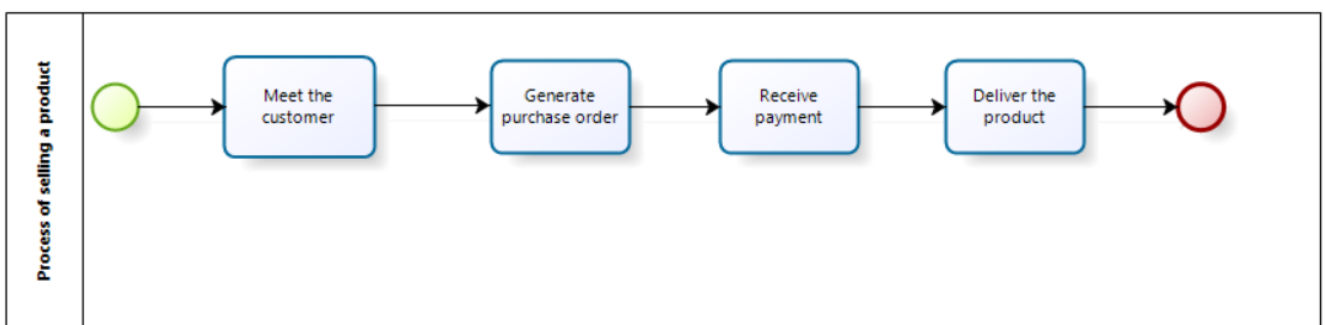


Figure 3.5: A diagram of the process of selling a product

- **Map:** is an evolution of the diagram, adding actors (employees), events, business rules, results and other detail elements of the process. Zoomed in for a more detailed view, the map provides more accurate process design information.

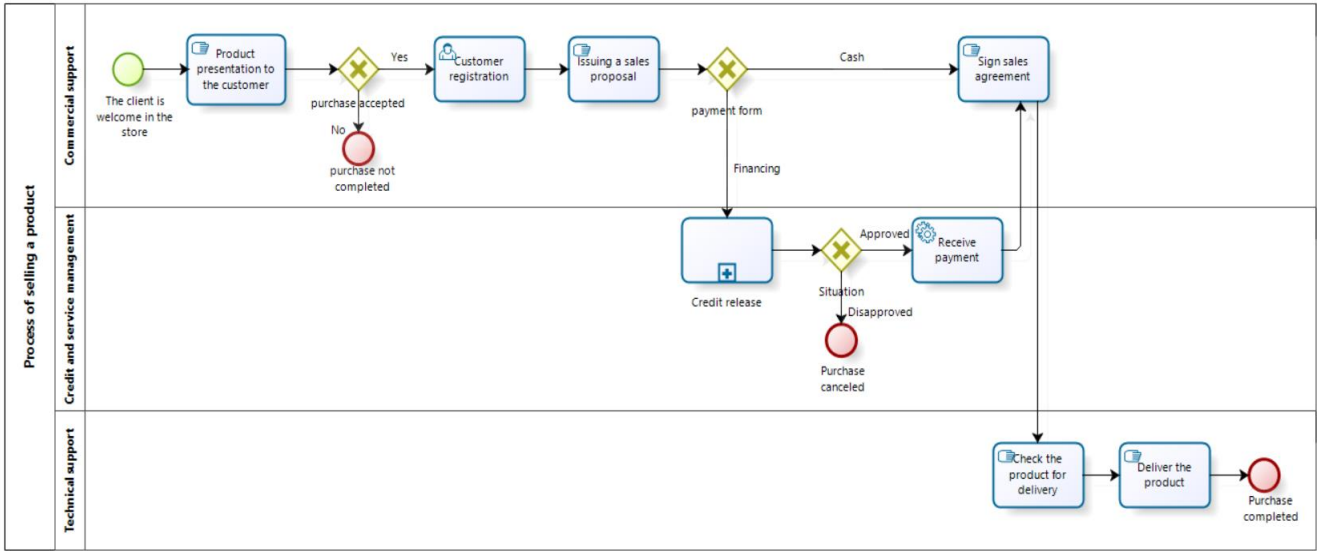


Figure 3.6: The Process improvement map

- **Model:** is the final version of the process evolution.

This representation brings a high degree of precision to the process by including new details, such as formulas, descriptions, systems, services or any other elements so that the process can be analyzed, simulated and executed.

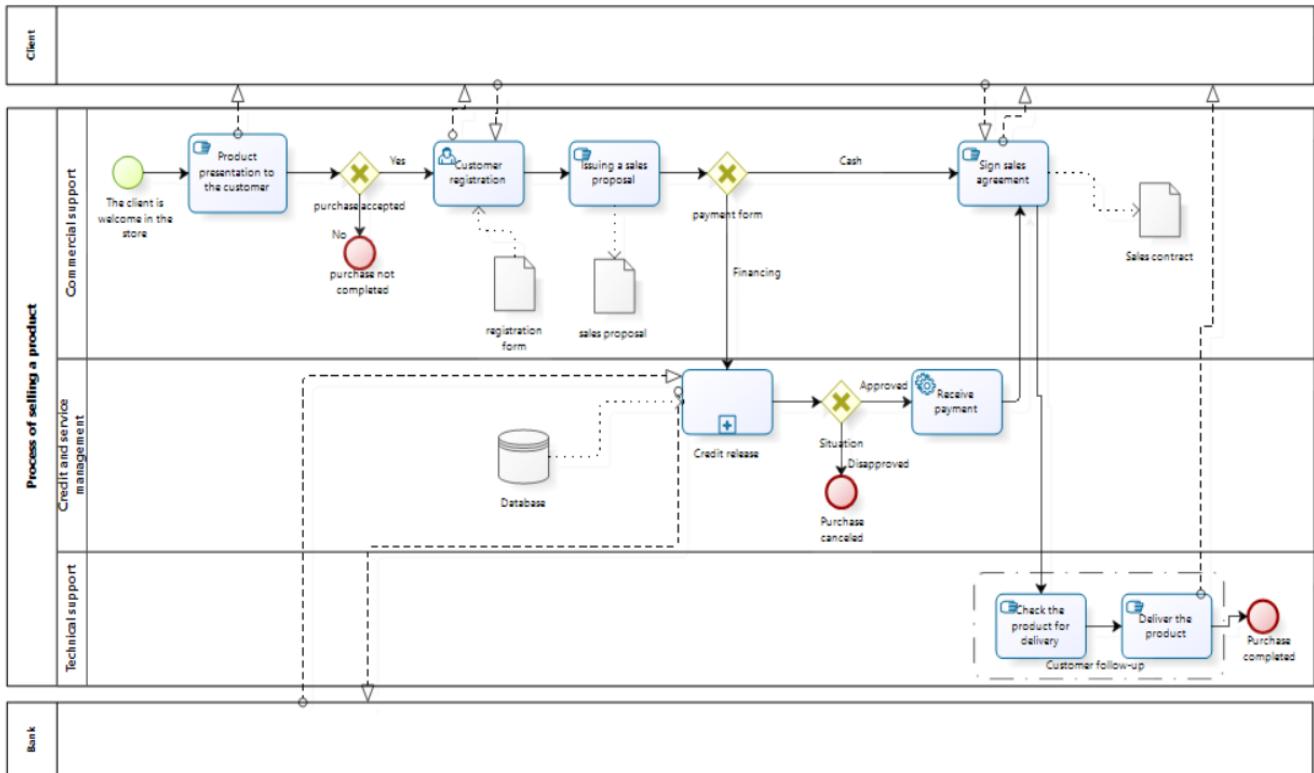
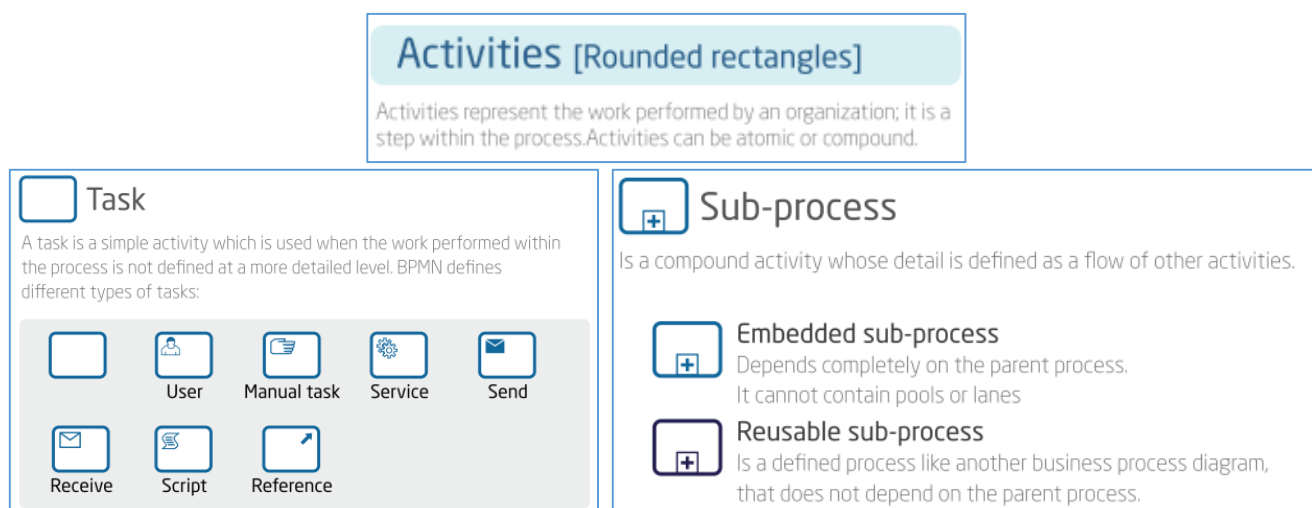



Figure 3.7: The Process model


3.2 Modeling elements used in the Bizagi Modeler application

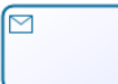



https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf


- **Tasks:** the following types of Tasks are used in Bizagi standardization:


 **Abstract Task (None):** task that has no specificity. It is the type of activity most often used during the early stages of process development.


 **Service Execution Task:** tasks that use some type of service, is carried out by a system, without the need for human intervention.

 **Receive Task:** is an external message receiving activity (waits for a message that will arrive from an external participant) and when the message is received, the task is complete.

 **Send Task:** is an activity of sending a message to an external participant; when the message is sent, the task is complete.

 **User Task:** task performed by a person with the support of a system (software).

 **Script Execution Task:** task where the modeler defines a script in a language that can be interpreted. When the task is achieved, the script starts executing and when execution is complete, the task will be complete.

 **Manual Task:** task performed by a person, without any intervention (without the help of any mechanism or application)



Business Rule Task: task that provides an input mechanism for a Business Rules Engine, in order to obtain a result or make a decision.

- **Sub-processes:** the following types of sub-processes are used in Bizagi standardization:



Embedded sub-process: the subprocess is part of a parent process (concatenation of work contained in this type of subprocess is exclusive to a given process - the parent process) and cannot be used in another process. Cannot contain pools or lanes.

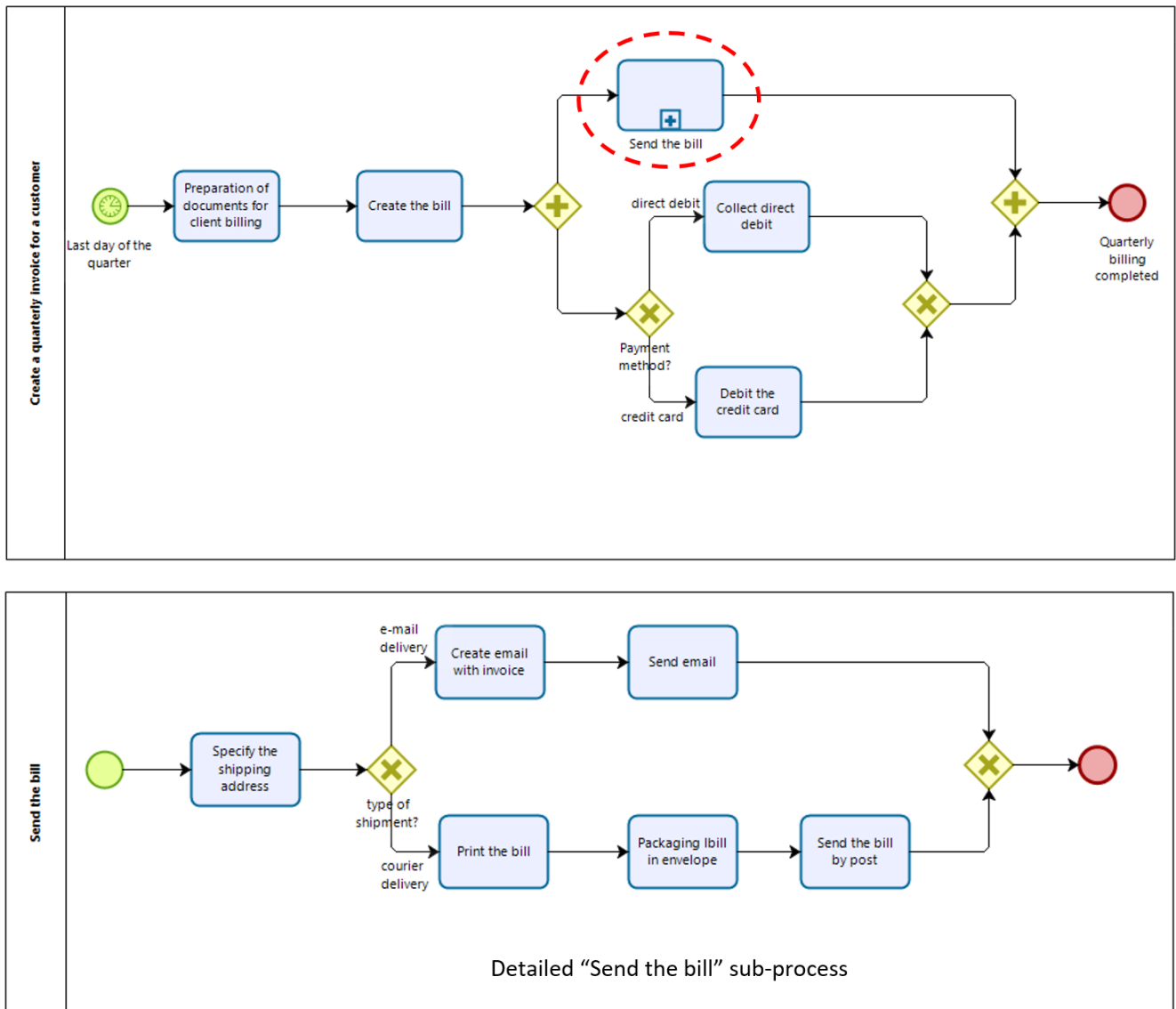
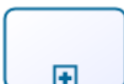


Figure 3.8: Example of an embedded sub-process



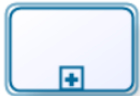
Reusable sub-process: known as Call Activity, it has a work sequence considered standard that can be replicated in several other processes and different contexts. It has a bold border.



Event sub-process: is a specialized sub-process, used within a process or another sub-process: is initiated through an event that occurs during the process, such as an error, for example. It has a dashed border.



Multi-instance sub-process: only apply to non-embedded processes. This property of the sub-process allows the creation of multiple instances: sub-processes can be repeated sequentially and behave like a cycle.



Transaction sub-process: is a type of subprocess that contains a set of activities, logically related, and can follow a specific transactional protocol (its behavior is controlled through a transaction protocol.) Widely used in automated processes. It has a double edge.

Loop, compensation and ad-hoc characteristics of tasks and sub-processes

Some situations require a task or sub-process to be performed repeatedly. This behavior is called a loop and the BPMN notation specifies standard markers to represent this situations:

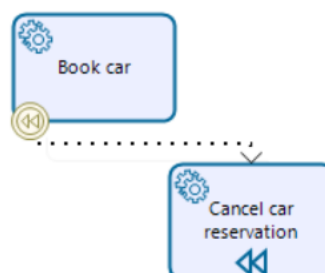
- **Loop Marker:** The task/the subprocess is executed repetitively until a condition is met or an event occurs.



- **Multi-Instance marker:** Another way of representing the loop is the execution of several instances. In this case, the activity or sub-process is executed several times with different sets of data. These instances can be executed in parallel or sequentially. Three vertical lines indicate **parallel execution** and three horizontal lines indicate **sequential** execution.



- **Compensation marker:** An activity (task or sub-process) can also have a compensation behavior. Compensation is a type of activity focused on returning the state of a process to a previous state, so that performing this activity will cancel or reverse the effect of another activity. For example, a compensation loop can be used to cancel a reservation if a payment fails.

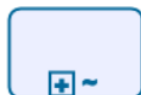


A compensation sub-process behaves the same as a compensation task, the difference being that the defined sub-process is executed instead of the task

Note: A compensation task is applied exclusively in the context of the existence of a compensation event and is executed only when a compensation event is launched.

A compensation task is integrated into the process diagram only through associations, never through sequence flows.

- **Ad-Hoc marker:** A sub-process with the Ad-Hoc marker indicates that it contains tasks that do not have a specific flow (or specific order of occurrence). Such a sub-process is used when there is no clarity regarding the sequence of tasks to be performed, but the operational flow must be preserved.



Note: There can also be combined subprocesses, such as an ad-hoc loop subprocess. Such a subprocess would repeat itself several times, but with an undefined order of steps in each iteration of the subprocess.

Gateways [diamonds]

Gateways are elements used to control divergence and convergence of the flow. (Split and Merge)

<p> Data-Based Exclusive Gateway Divergence: the Exclusive Decision has two or more outgoing Sequence Flows, but only one of them can be taken and the decision will be taken after evaluating a business condition. Convergence: is used to merge alternative paths.</p> <p> Event-Based Exclusive Gateway Is used as a Divergence element. This gateway represents a point in the process where only one of many paths of the process can be selected but based on an event, not a data expression condition.</p> <p> Parallel Gateway Divergence: is used to create parallel flow. Convergence: is used to synchronize multiple parallel paths into one. The flow continuous when all the incoming sequence flows have reached the gateway.</p>	<p> Inclusive Gateway Divergence: indicates that one or more routes can be activated from many available, and the decision is based on process data. Convergence: indicates that many outgoing routes of an Inclusive gateway, used as an element of divergence, can be synchronized into just one.</p> <p> Complex Gateway Divergence: is used to control complex decision points that are not easy to manage with other types of gateways. Convergence: When the Gateway is used as a Merge then there will be an expression that will determine which of the incoming Sequence Flow will be required for the Process to continue.</p>
--	--

https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf

Data-Based Exclusive Gateway:

- **Divergence:** As a divergence point, continues the flow through an exclusive condition (with two or more outgoing Sequential Flows), in which only one of the paths must be followed, according to the information to be tested.

- **Convergence:** As a convergence point, it is used to merge exclusive paths.

In the example below, the flow continues through an exclusive condition, where only one of the paths must be followed, according to the information to be tested (report analysis and evaluation):

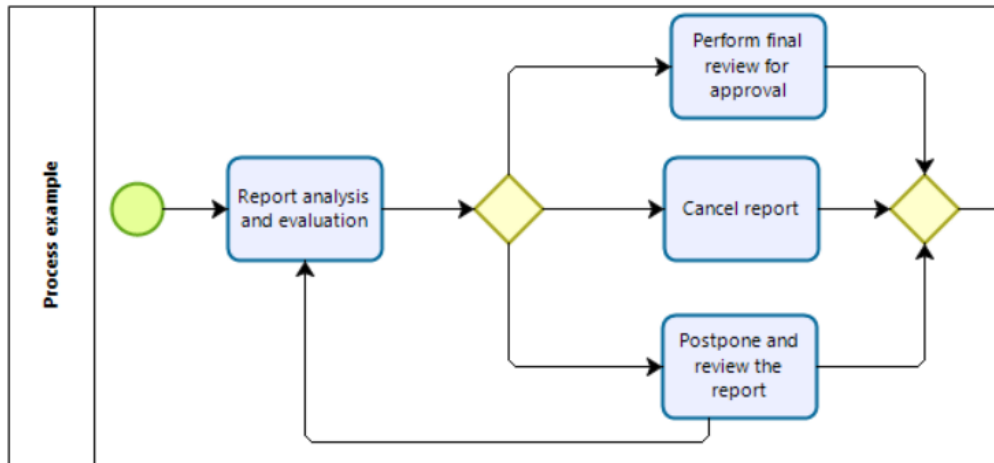


Figure 3.9: Data-Based Exclusive Gateway

Note: It should be avoided that nested gateways that refer to the same question, such as the one shown in the previous image, are used. It would be correct to choose only one gateway where the question has three ways of answering. Once the total number of gateways is reduced, the simplified structure provides a better understanding of the process.

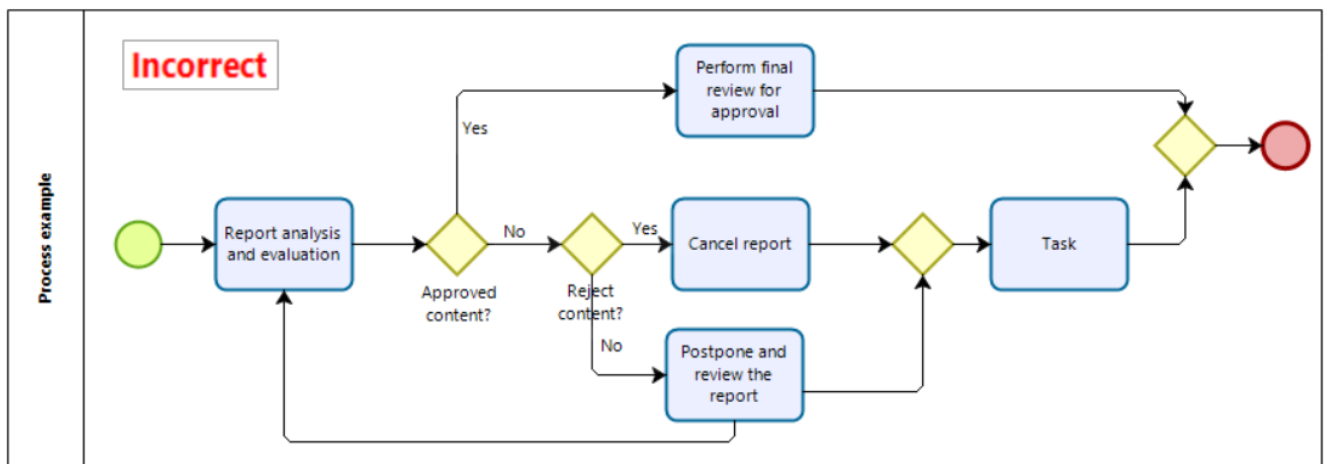


Figure 3.10: Example of incorrect use of a Data-Based Exclusive Gateway



Event-Based Exclusive Gateway: represents branching point alternatives where the decision is based on two or more events (not on a data expression condition) and only one alternative is chosen.

As seen in the modeling below, the Event-Based Exclusive Gateway precedes the intermediate events that contain the messages for the different response types.

Using an Event-Based Exclusive Gateway, the first event triggered cancels the other events. In the example above, either the process receives the final documents and further the credit analysis takes place, or the client is contacted for not sending the documents within 7 days.

Note: This gateway type does not have an associated name.

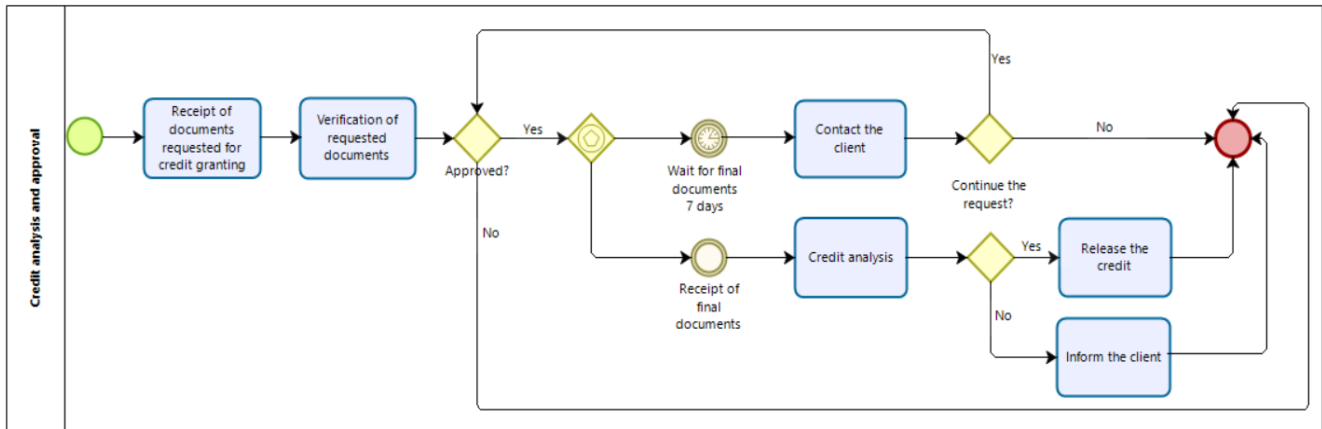


Figure 3.11: Event-Based Exclusive Gateway

+ **Parallel Gateway:** has no associated logical conditions. They must be defined in pairs, one as a divergent element to activate several parallel paths, and another as a convergent element, to synchronize the previously activated paths.

- **Divergence:** when divergent indicates the division of a flow into two or more paths, which will be run in parallel.

- **Convergence:** at convergence, it waits for all the parallel paths to be completed before continuing with the flow.

Using a parallel gateway reduces the execution time of the process, because instead of waiting for the completion of the activity, it can be carried out in parallel, together with another activity/activities.

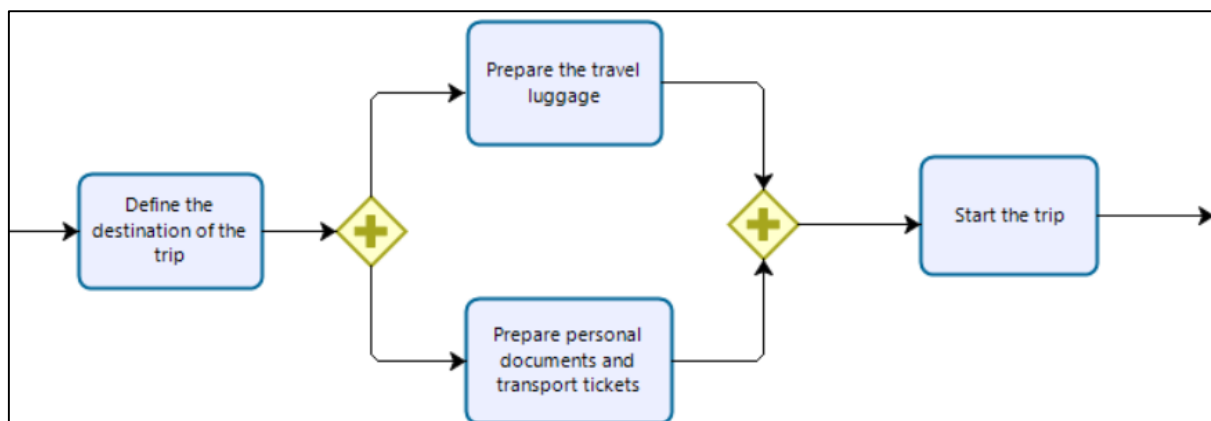


Figure 3.12: Parallel Gateway

◇ **Inclusive Gateway:**

- **Divergence:** represents a branch point, where the flow continues through an inclusive condition, where the alternatives are based on conditional expressions (there may be a combination of paths to follow, i.e. one or more alternatives may be true, according to the information to be verified).

- **Convergence:** Used to join a combination of alternative parallel paths.

These gateways work in pairs, so paths exiting an inclusive gateway must terminate with another gateway of the same type.

In the example below, the first Inclusive Gateway checks the output of the "Analysis and evaluation of documentation" activity; If, during the course of this activity, the need for approval by one of the two departments is identified, the respective flow will be executed. If the documentation presented is ok, the flow will follow the standard path. The following Inclusive Gateway was used to unify the resulting flows. This ensures that the "Notify Stakeholders" activity is only started after all the flows that were started by the previous gateway are executed.

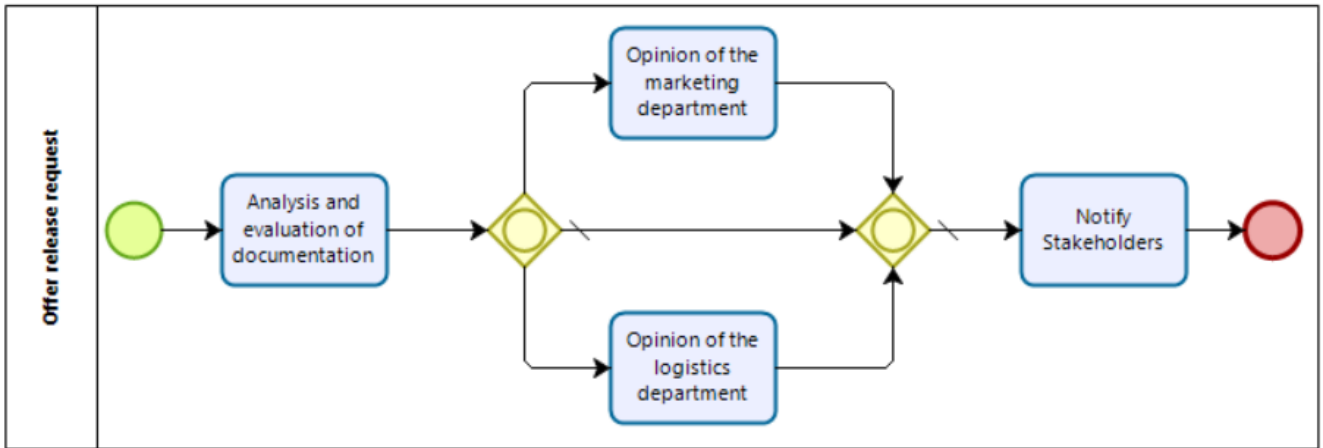


Figure 3.13: Inclusive Gateway

 **Complex Gateway:**

- **Divergence:** Used to control complex decision points in processes, in complex conditions that other types of gateway cannot accommodate.

- **Convergence:** Allows to continue at the next point in the process when a business condition is met.

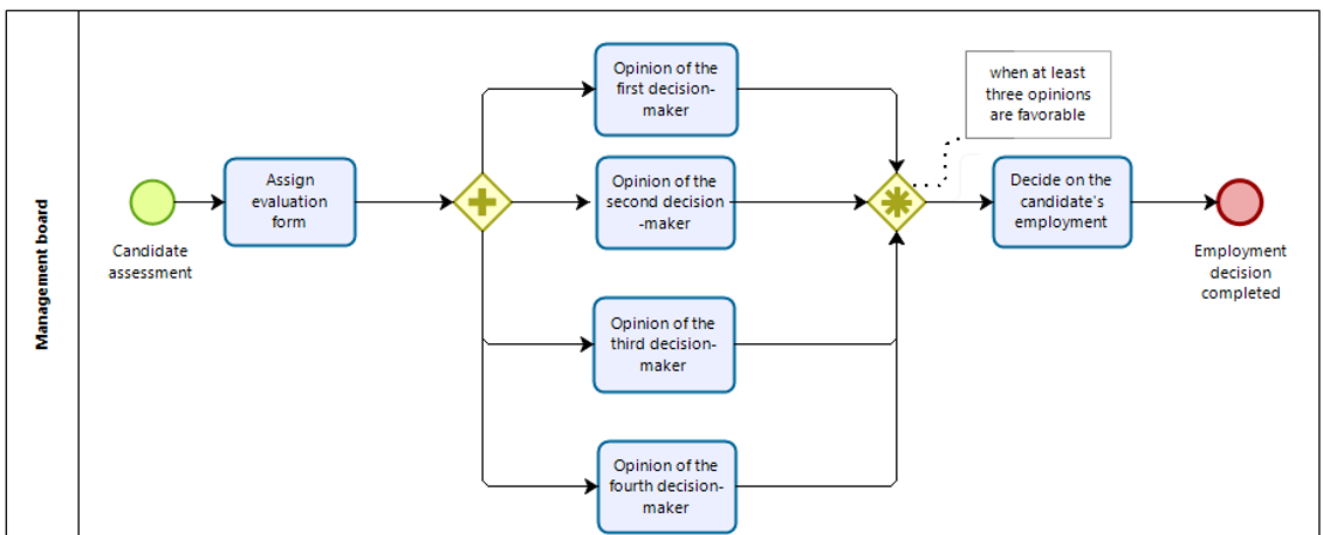


Figure 3.14: Complex Gateway



Parallel Event-Based Gateway: used to divide a path into several parallel paths and combine several paths into a single one. The paths run in parallel. So, it represents a branching point of the process where the output flows are based on the production of events and not on the evaluation of expressions using data, as happens in the case of exclusive and inclusive gates.

In the following example, if there is not necessarily a received photo and identification documents, the document processing process does not begin. So, the gateway was used to highlight in the diagram which events, when they occur together, will start the process.

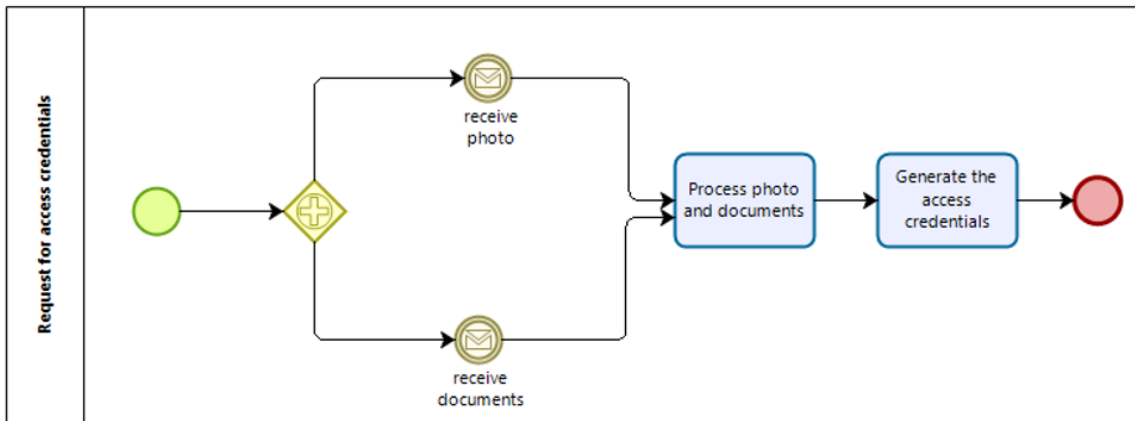



Figure 3.15: Parallel Event-Based Gateway


Events [circles]


Events represent something that happens or may happen during the course of a process. These Events affect the flow of the Process and usually have a cause or an impact and there are 3 types of events based on how the process flow is affected.

Start Events

- Indicate the instance or initiation of a process
- These do not have any incoming Sequence Flow


 **None Start Event**
Does not specify any particular behavior. It is also used for a Sub-Process.

 **Message Start Event**
A process starts when a message is received from another participant.


 **Timer Start Event**
Indicates that a process starts at certain time or on a specified date


 **Conditional Start Event**
A process starts when a business condition becomes true.


 **Signal Start Event**
A process starts when a signal coming from another process is captured. Note that the signal is not a message; messages have clearly defined who sent them and who receives them.

 **Multiple Start Event**
Indicates that there are many ways to start the process. Only one of them will be required to start the process.


https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf


 **None Start Event:** indicates that a process has started. There are no flows of sequence that precede it. No specific behavior is specified.


 **Message Start Event:** is used when at the beginning of a process it is possible to receive a message from an external participant.


 **Timer Start Event:** is used control time or set dates for carrying out activities. A timer start event indicates the start of execution of a flow at predetermined periods of time.


Can indicate a specific date, a specific time or can indicate a date and time.

 **Conditional Start Event:** triggers the beginning of a process when a condition is met.











 **Signal Start Event:** serves to model communication between different processes: indicates that the process will start when it receives a signal from another process(es).

 **Multiple Start Event:** indicates that there are multiple ways to start the process. Such as, for example, when a certain demand is received or when a certain deadline is reached.


 **Parallel Multiple:** indicates that there are multiple ways to trigger a process, namely when all of the respective requirements, events, or restrictions are met.

 **Intermediate Events**

- Intermediate Events indicate something that occurs or may occur during the course of the process, between Start and End.
- These can be used within the sequence flow or attached to the boundary of an activity. Intermediate Events can be used to catch or to throw the event trigger.
- When the event is used to catch the Event marker will be unfilled, and when the event is used to throw the Event marker will be filled.

<p> None Intermediate Event Indicates that something that occurs or can occur within the process. It can only be used within the sequential flow of the process.</p> <p> Message Intermediate Event Indicates that a message can be sent or received. If the event is of reception, it indicates that the process has to wait until the message has been received. This type of event can be used within the sequential flow or attached to boundary of an activity to indicate an exception flow.</p> <p> Timer Intermediate Event Indicates a waiting time within the process. This type of event can be used within the sequential flow indicating a waiting time between the activities or attached to boundary of an activity to indicate an exception flow when a time-out occurs.</p> <p> Conditional Intermediate Event Is used when the flow needs to wait for a business condition to be fulfilled. It can be used within the sequential flow indicating that it should wait until a business condition has been fulfilled or attached to boundary of an activity indicating an exception flow that is activated when the condition is met.</p> <p> Signal Intermediate Event Is used to send or receive signals. If it is diagrammed within the sequential flow of a process it can send or receive signals. If it is diagrammed attached to boundary of an activity, it can only receive signals and indicating an exception flow that is activated when the signal is captured.</p>	<p> Multiple Intermediate Event This means that there are multiple triggers assigned to the Event.</p> <p> Cancel Intermediate Event Is only used in Transaction Sub-Process. This event is always diagrammed attached to boundary of the transactional sub-process and indicates an alternative flow that can be made when the transaction sub-process is cancelled.</p> <p> Error Intermediate Event Is used to capture errors and to handle them. This event can only be attached to the boundary of an activity.</p> <p> Compensation Intermediate Event The Compensation Intermediate Event enables you to handle compensations. When used within the sequential flow of a process they indicate that a compensation is necessary (throwing). When used on the borders of an activity it indicates that this activity will be compensated when the event is triggered (catching).</p> <p> Link Intermediate Event Is used to connect two sections of the process.</p>
--	--

https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf

 **None Intermediate Event:** indicates that something is happening during the process. Can be used within the sequential flow to indicate something that occurs or may occur within the process (cancellation of a request, receipt of a request, etc.).



Conditional Intermediate Event: indicates that the continuation of the flow depends on a certain condition to be fulfilled.

In the example below, the preparation process of the two menus continues only if the two conditions are met: the oven temperature reaches a certain value and respectively only after the respective dish is ready is it taken out of the oven and portioned to be eaten:

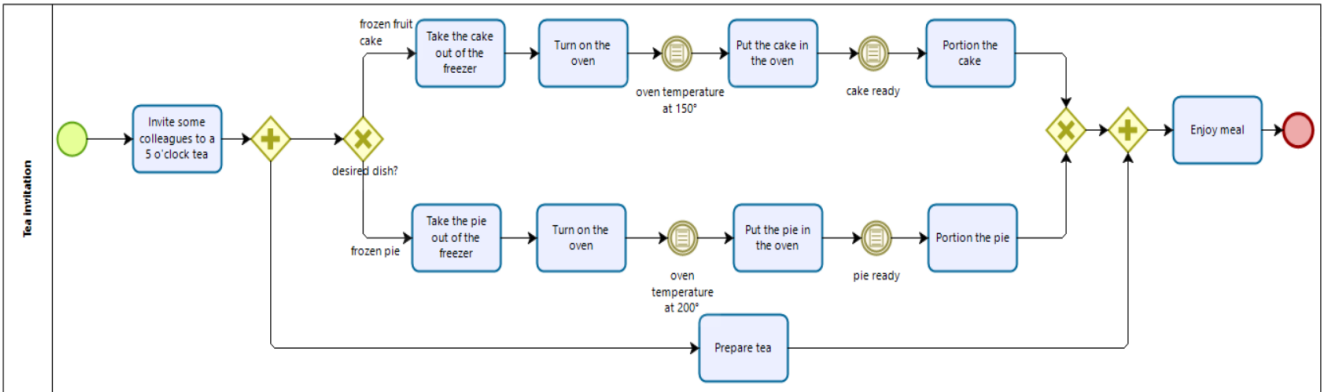


Figure 3.16: Conditional Intermediate Event



Message Intermediate Event: can be used to send (dark envelope) and receive (white envelope) messages.

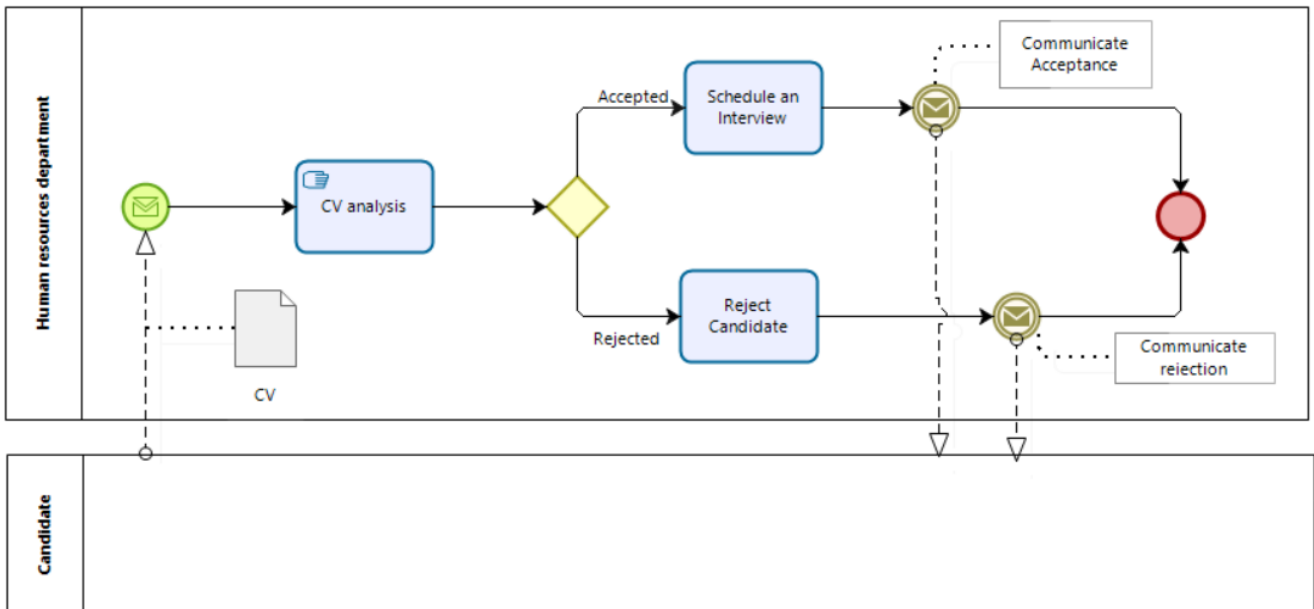


Figure 3.17: Message Intermediate Event



Timer Intermediate Event: used to introduce deadlines into the sequential flow (e.g. when there are deadlines for completing work) or can act as a delay mechanism. Can be used in sequential flow, indicating a wait between tasks before they are executed.

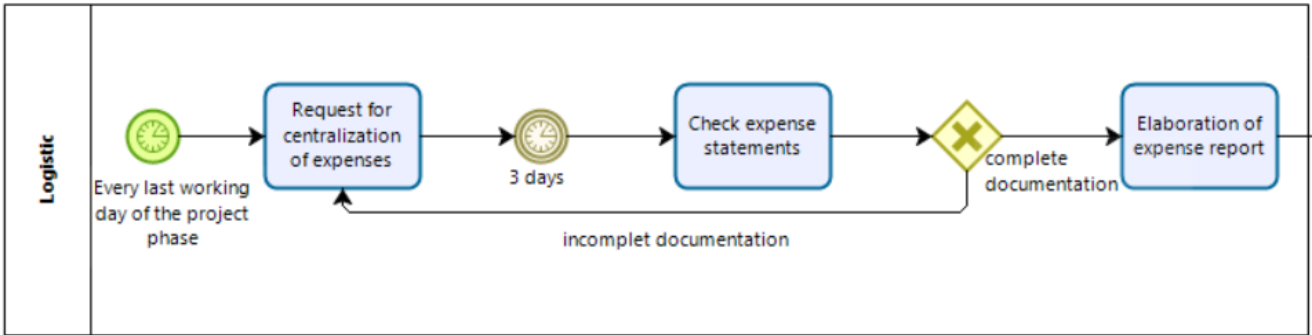



Figure 3.18: Timer Intermediate Event

 **Signal Intermediate Event:** indicates the sending (dark triangle) or receipt (white triangle) of some signal (a notification, a report, etc.) to continue the process flow.

- **Send (throw):** launches a signal that continues the flow of the process.
- **Reception (catch):** wait for a signal to continue with the flow of the process.

The following example shows a situation where there are two sub-processes that are designed to be reused within other processes as well. Thus, in order to work well with each other and with their parent process, they must send signals at the right times.

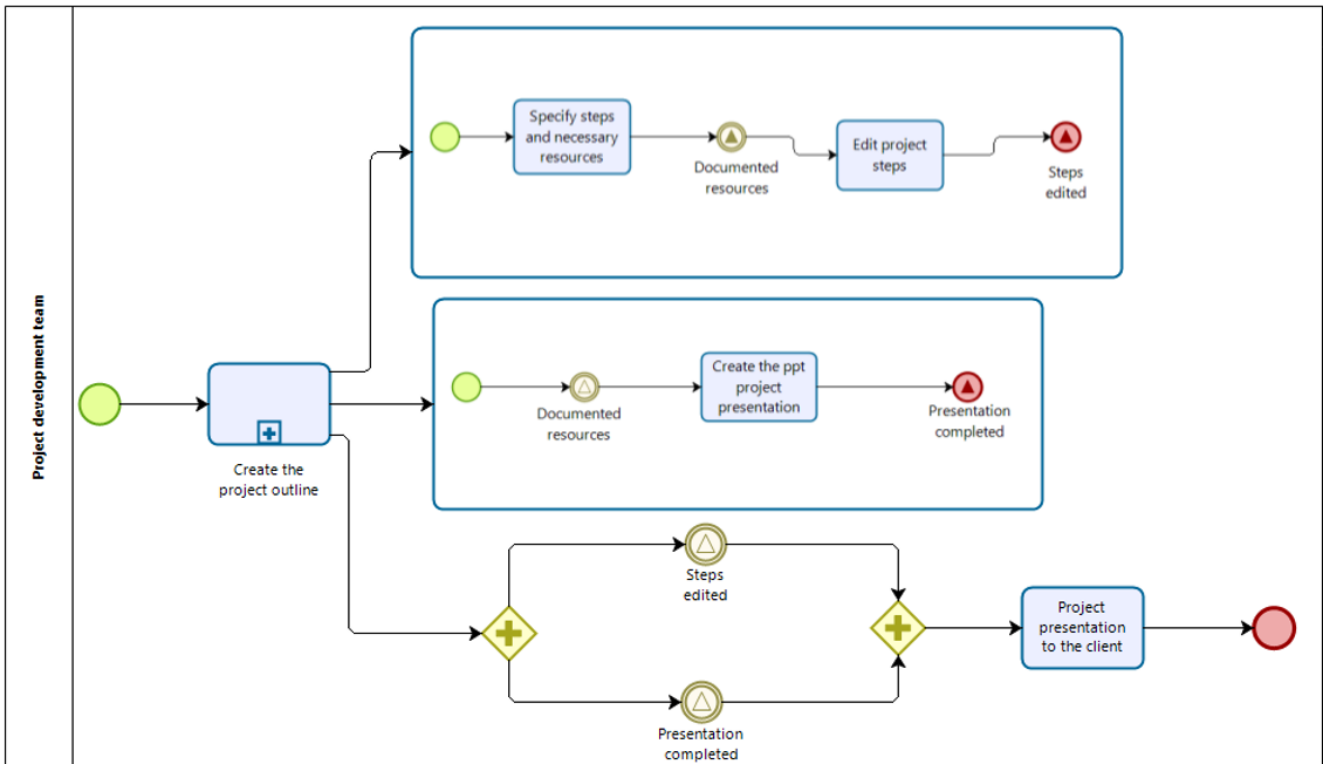


Figure 3.19: Signal Intermediate Event

The two sub-processes send signals to the parent process, and the first sub-process also sends a signal to the second sub-process.

The "Project presentation to the client" activity in the parent process must wait until the "Edit project steps" activity in the first sub-process from above and activity "Create the ppt project presentation" in the middle sub-process have been completed.

To do this, the parent process detects the signal from each of these two sub-processes.



Multiple Intermediate Event: indicates that there are multiple triggers attached to the event - the event can be triggered for multiple reasons.

Multiple event can include capturing multiple alternative events with one symbol.

As a trigger event, it reacts like a multiple trigger. Triggers can be any combination of messages, timers, conditions, and/or signals. For any of these triggers, as soon as the trigger is activated, a new process instance is spawned and the flow continues from that start event.

In the example below, each Start Event is independent of the other Start Events in the Process. This means that the Process will start when any of the Start Events are fired. The process will not wait for all startup events.

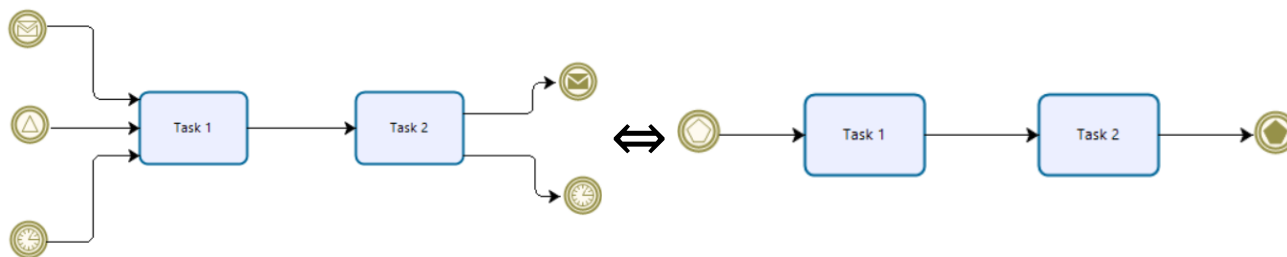


Figure 3.20: Multiple Intermediate Event



Error Intermediate Event: can be used to capture or insert pre-identified errors. Such an event can only be attached to activity/process boundaries (it may not be used in normal flow) and always interrupts the activity to which it is attached.

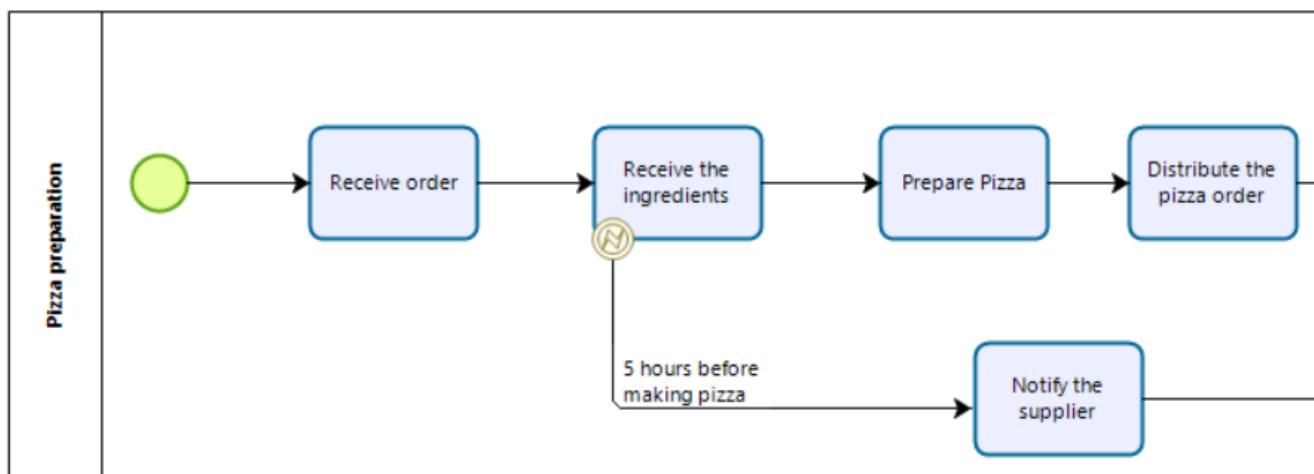


Figure 3.21: Error Intermediate Event

In the example above, if the "Receive the ingredients" activity is completed before 5 hours, the exception route is not activated, however if the ingredients are not received before the 5 hours required for the "Prepare Pizza" activity, then the exception route is activated and the "Notify the supplier" activity continues.

Compensation Intermediate Event: When used in a normal sequential flow, it indicates that compensation is required. When attached to the boundary of an activity, this event indicates that this activity will be compensated when the event is triggered.

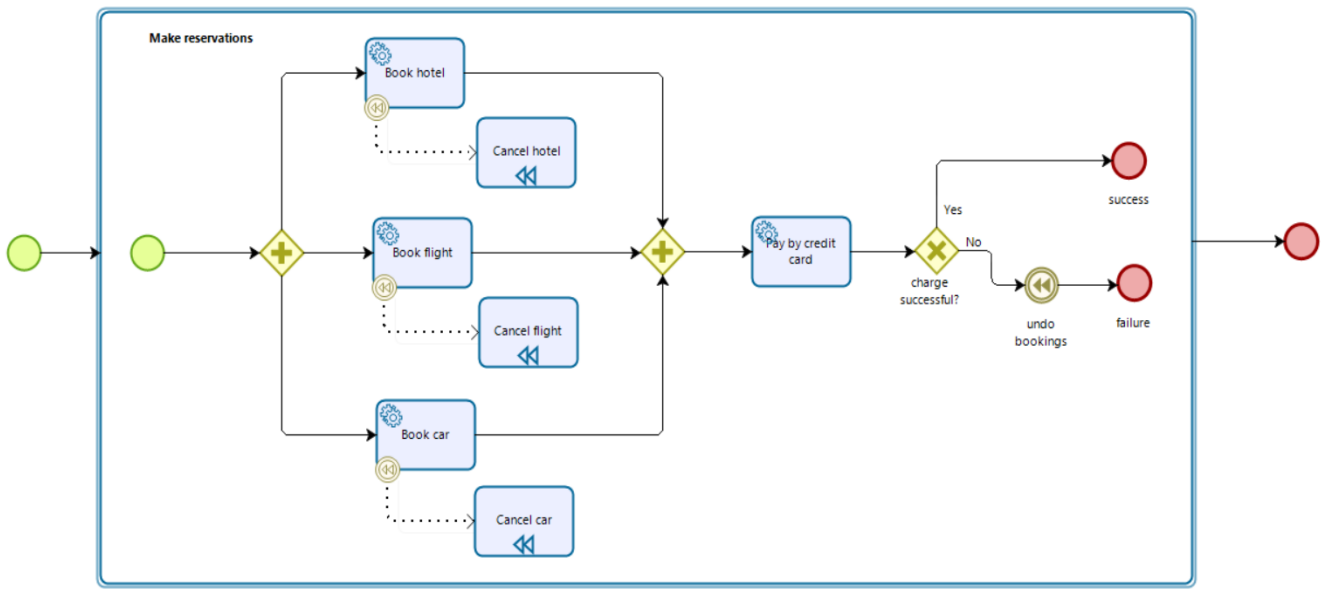


Figure 3.22: Compensation Intermediate Event

Adapted from: https://help.bizagi.com/process-modeler/es/index.html?long_lasting_transactions.htm

Link Intermediate Event: is used in normal sequential flow and connects two parts of the process; generally used when there are many activities or the process has visually distant points. The dark arrow indicates the origin and the white arrow the destination of the connection.

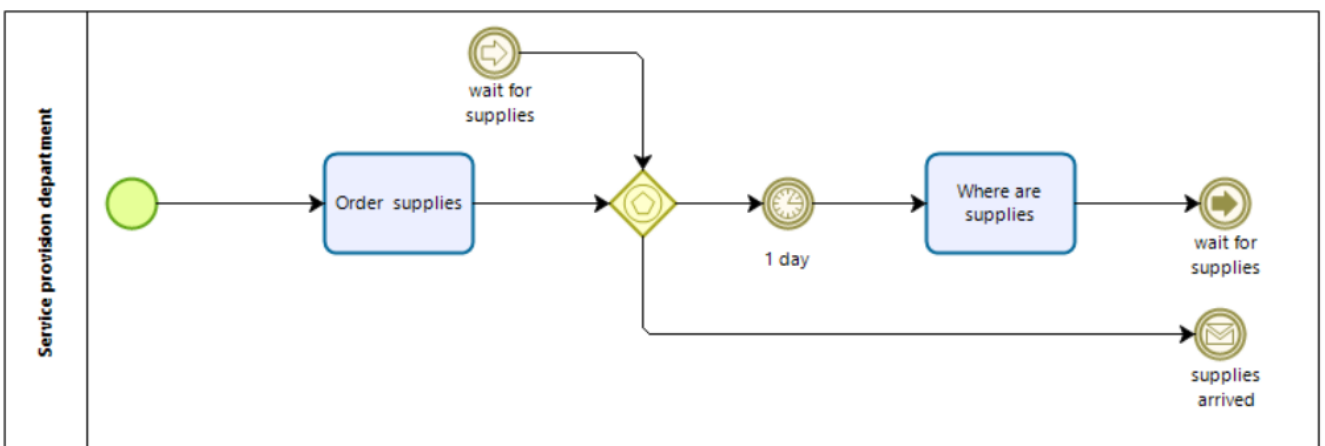


Figure 3.23: Link Intermediate Event



Parallel Multiple Intermediate Event: indicates that the event can only be activated when all requirements, events or restrictions are met.

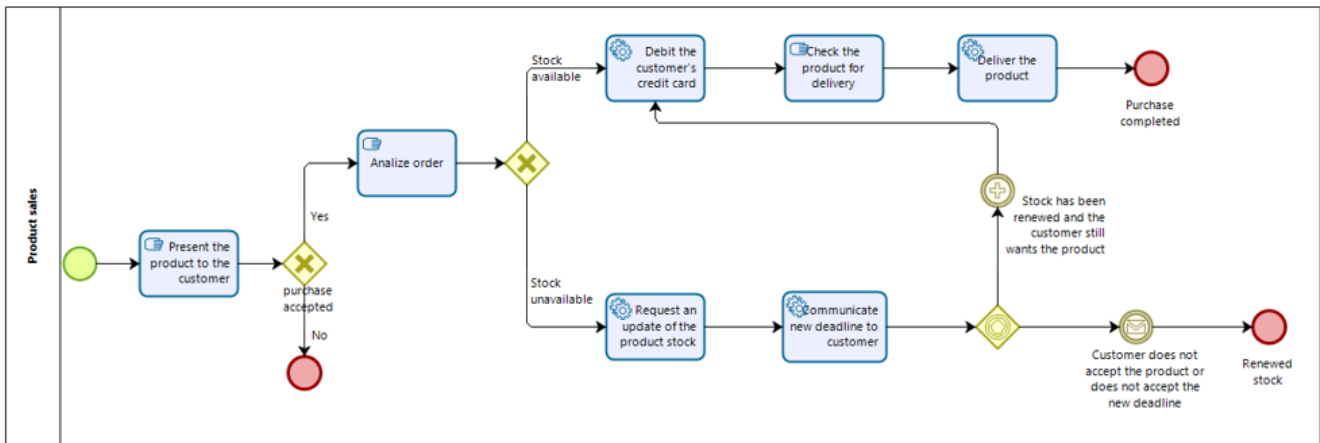


Figure 3.24: Parallel Multiple Intermediate Event



Escalation Intermediate Event: is used to handle an Escalation. If the activity is attached to the edge, it captures an Escalation. Unlike an Error, by default it does not abort the activity to which it is attached. However, the modeler may decide to change this definition:

- If the activity stops, it has a solid edge;
- If it does not interrupt the activity, it has a dotted border;

The following diagram shows different Escalation Event types:

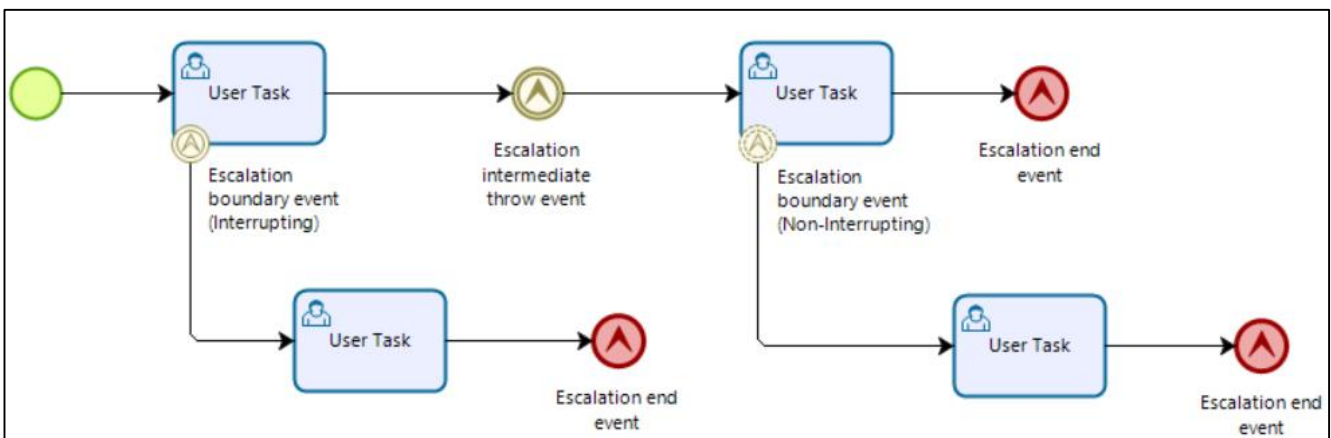


Figure 3.25: Example of BPMN Escalation Event

<http://www.javanibble.com/bpmn-escalation-event>



Cancel Intermediate Event: is used within a Transaction SubProcess (is attached to the boundary of a SubProcess). Is designed to handle a situation in which a transaction is canceled.

The figure below refers to a transactional process, which presents the perspective of a business trip. After the planning activity, the trip preparation follows, which is described in a transactional process.

It is noted that if the transaction is cancelled, then all the necessary compensation events are automatically triggered.

The possibility of a canceled transaction brings back to travel planning and another alternative must be sought.

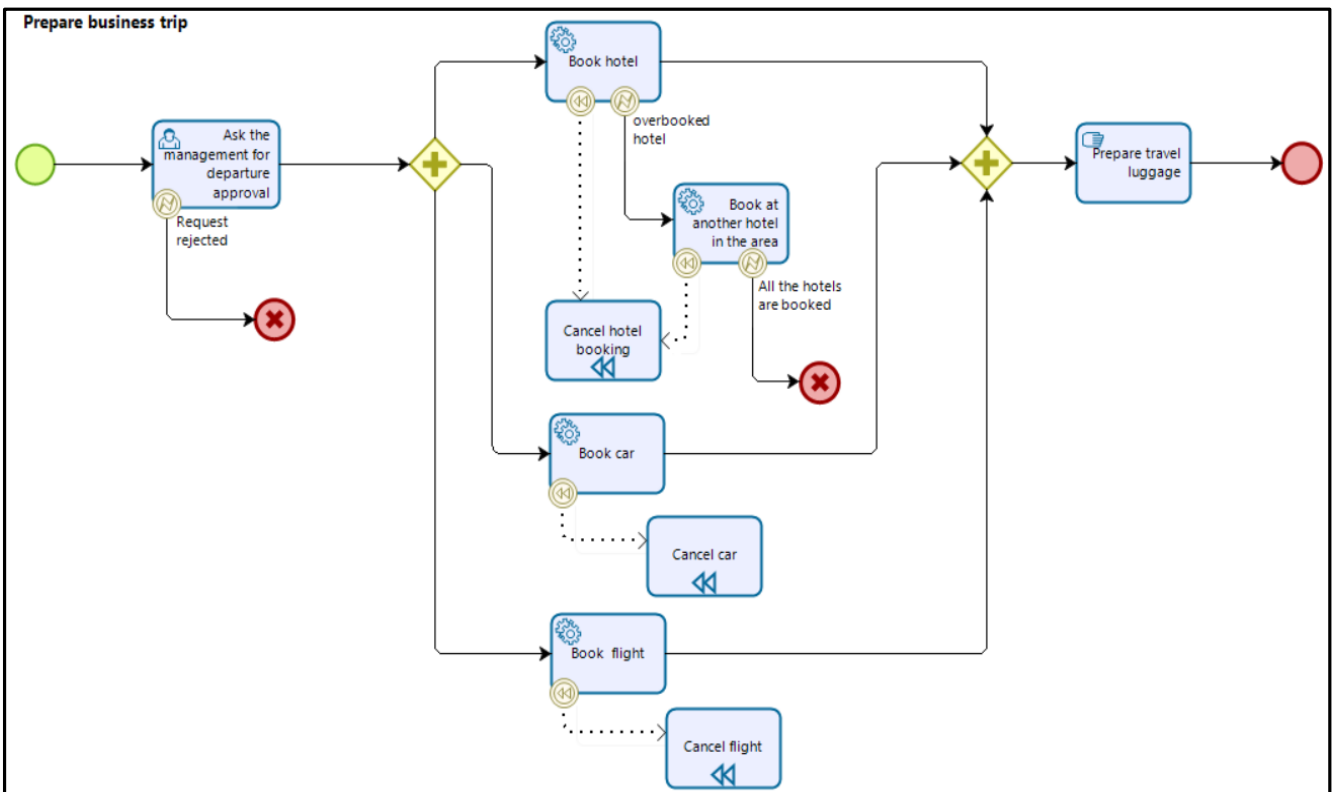
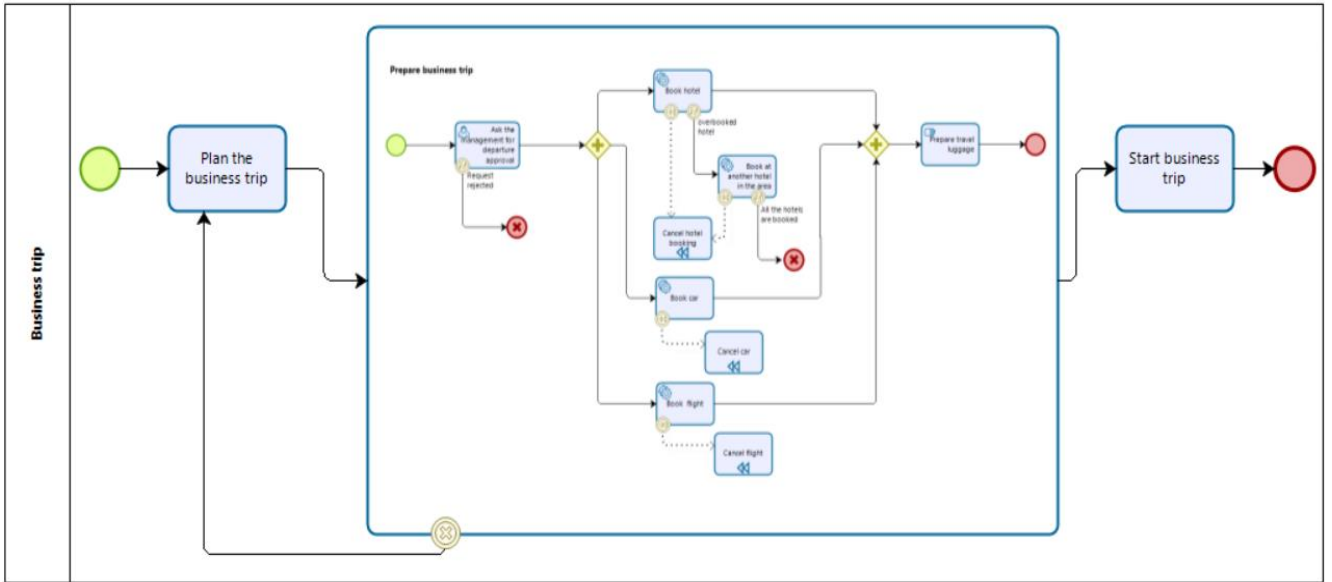


Figure 3.26: Example of using a Cancel Intermediate Event within a transactional subprocess

○ End Events

- End Event indicates where a process will end.
- A process can have more than one end. It does not have outgoing sequence flows.

<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>○ None End Event Indicates that a route of the process has reached its end. A process can only finish when all the routes of the flow arrive at an end</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>✉ Message End Event Indicates that a message is sent to another process when the process arrives at the end.</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>▲ Signal End Event Indicates that a signal is generated when the process ends.</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p>🏠 Multiple End Event Indicates that many results can be given at the end of the process. All the results should occur.</p> </div>	<div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>✖ Cancel End Event Is only used in Transaction Sub-Process and indicates that the Transaction should be cancelled.</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>⚡ Error End Event Indicates that a named Error is generated when the process ends.</p> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;"> <p>⏪ Compensation End Event Indicates that the process has finished and that a compensation is necessary.</p> </div> <div style="border: 1px solid #ccc; padding: 5px;"> <p>🛑 Terminate End Event This event ends the process immediately. When one of the routes of the flow arrives at its end, indicating that the process has completely finished.</p> </div>
---	---

https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf

○ None End Event: indicates that a path of the process reaches its end.

✉ Message End Event: indicates that the process ends with the sending of a message to a specific participant in the process.

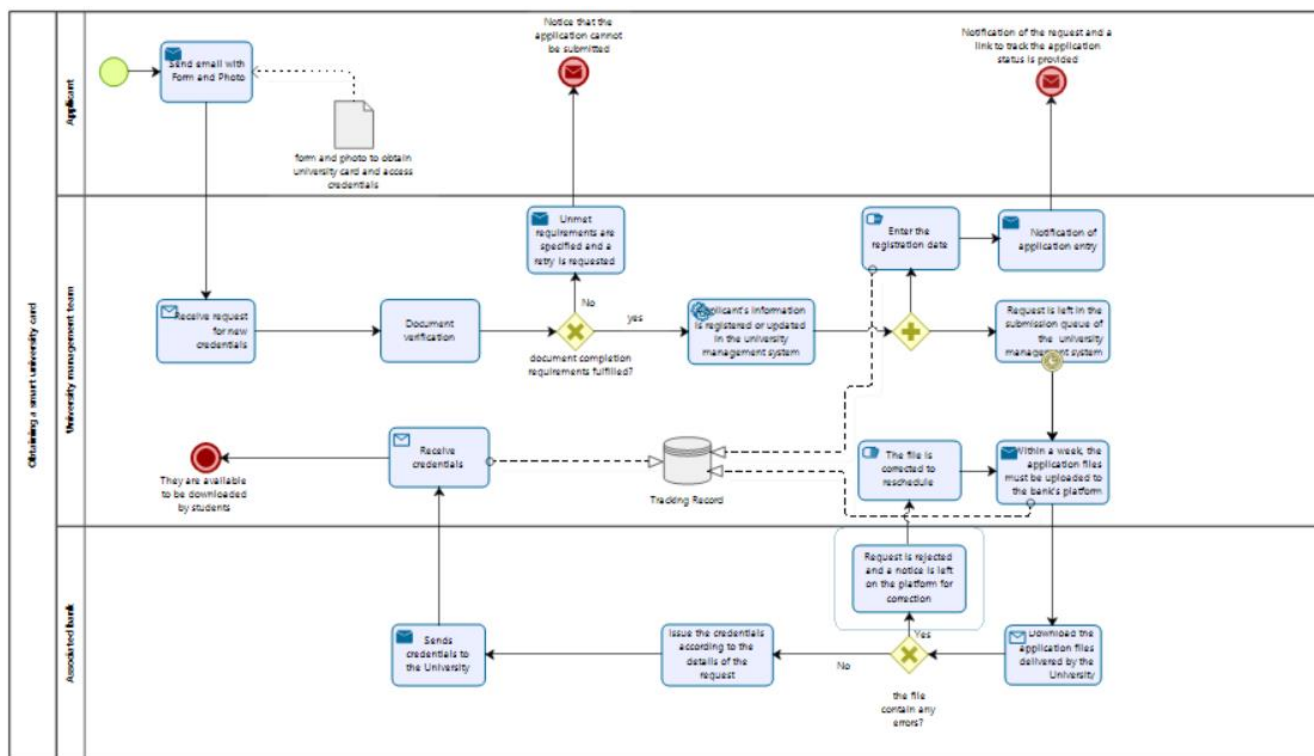


Figure 3.27: Message End Event

https://tui.uv.cl/images/BPMN/OBTENCIN_TUI_UV.pdf

Signal End Event: indicates that a signal will be transmitted when the end has been reached. Noting that signal, which is broadcast to any is not a message (that has a specific source and target).

Multiple End Event: indicates that there will be multiple consequences of completing the process, i.e. there are several results at the end of the process and all of them must occur.

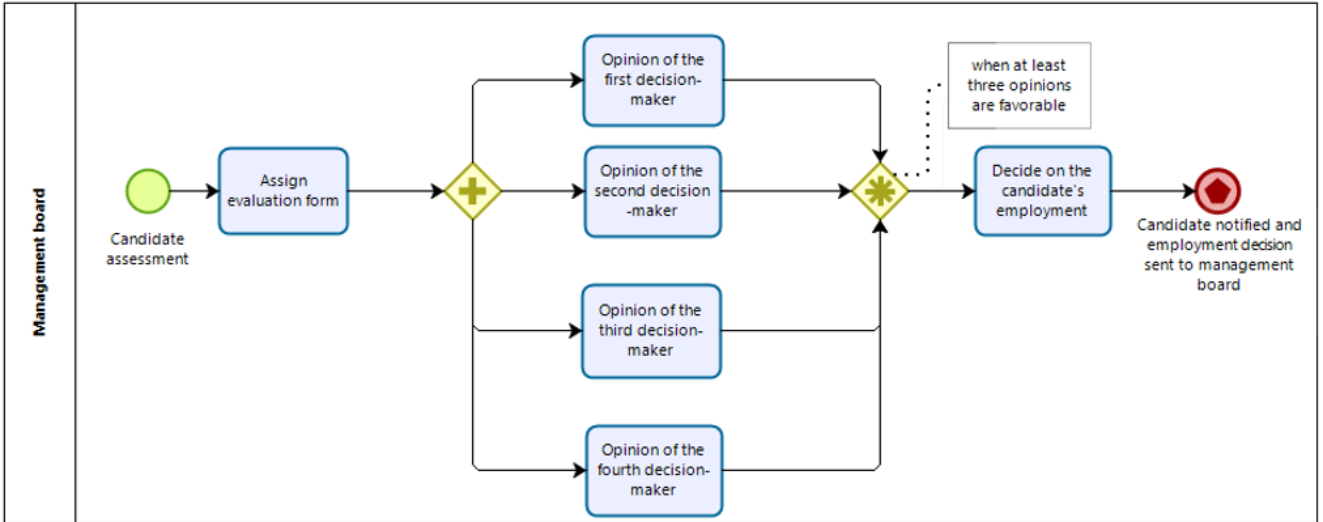


Figure 3.28: Multiple End Event

Cancel End Event: is used within a Transaction Sub-Process indicating that the transaction should be canceled. Additionally, indicate that a Transaction protocol cancel message should be sent to any entities involved in the transaction.

Error End Event: indicates that an Error should be generated, aborting the task or process, that is, that the process was terminated with an error; all threads in the process or sub-process will be terminated;

The exception in example below is the card that is rejected by the bank; the payment cannot be carried out normally, consequently the exception route is activated:

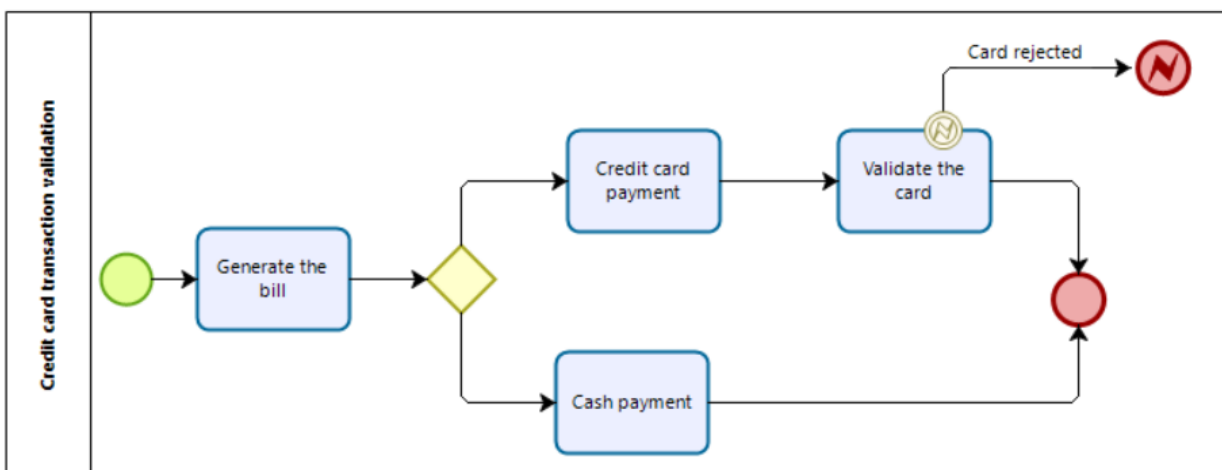


Figure 3.29: Error End Event



Compensation End Event: indicates that the process has been completed and that compensation will be required. If it is an activity that has been successfully completed, then that activity will be compensated. For this, the activity must have a boundary Compensation Event or contain a Compensation Event Sub-Process. When the activity to compensate is not defined, all activities that were successfully completed and have an attached compensation event will be compensated.

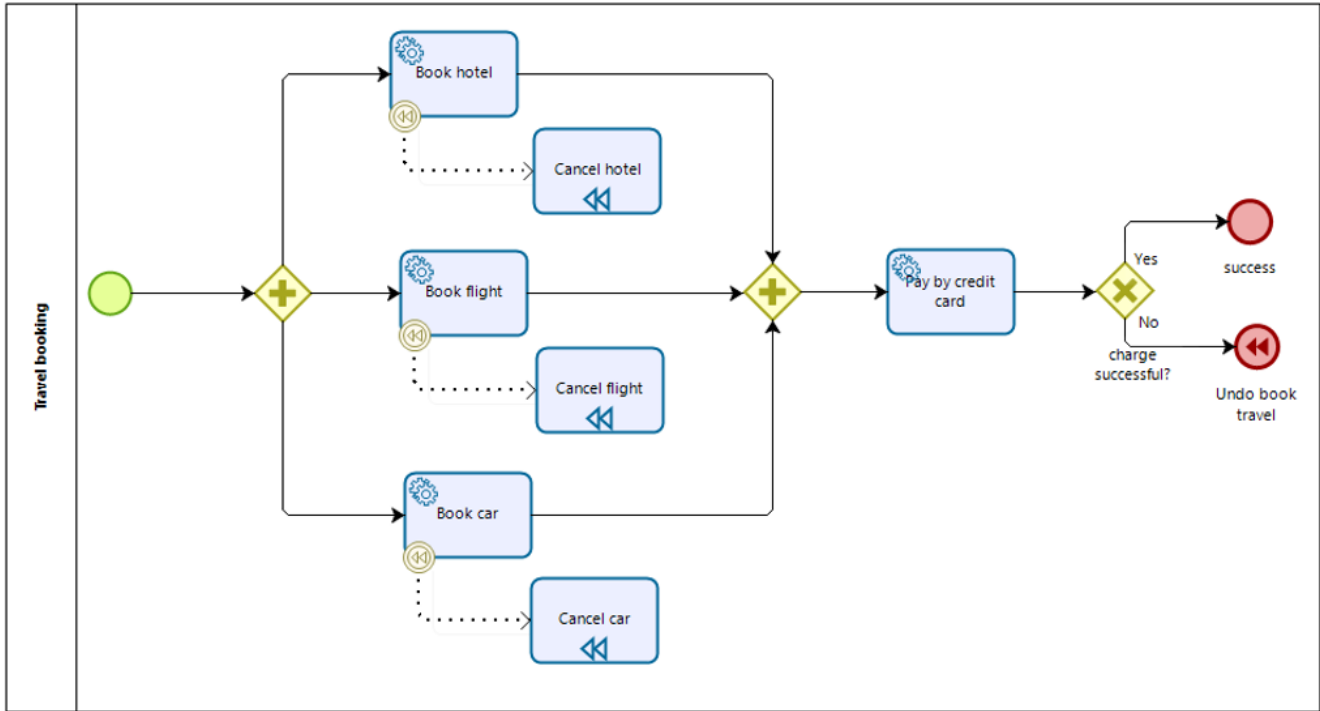



Figure 3.30: Compensation End Event




Terminate End Event: indicates that the process and all its activities have been completed, regardless of whether there were parallel flows running, which will be cancelled.

Swimlanes

 **Pool**

- A pool is a container of a single process.
- The name of the pool can be considered as the name of the process.
- There is always at least one Pool.

 **Lane**

- A lane is a subdivision of a pool
- Represents a role or an organizational area.

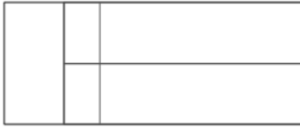
https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf



Pool: following the general BPMN rules and notations, in Bizagi software a business process is contained within a pool, so each diagrammed process must have a pool (a container).

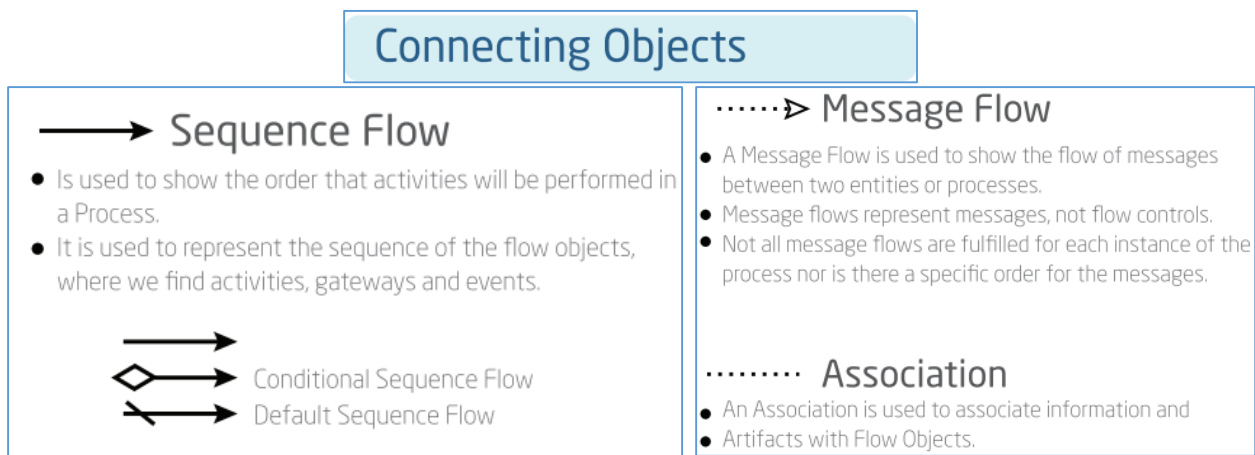
A pool contains a single process, its name can be considered as the name of the process and sequential flows cannot cross the pool boundaries.

In certain circumstances, a pool can represent a black box, that is, the representation of a participating/collaborating process of another process, whose modeling is not represented.



Lane: is a sub-partition within a process used to organize and categorize activities within it. Lanes are used to identify actors that participate in the process such as units, sectors, systems, internal departments, etc.

In the process that is being diagrammed, there cannot be elements that are not located in a lane or in a pool.



[https://resources.bizagi.com/docs/BPMN Quick Reference Guide ENG.pdf](https://resources.bizagi.com/docs/BPMN_Quick_Reference_Guide_ENG.pdf)

→ Sequence Flow: connects activities, subprocesses, events and gateways to each other and shows the order in which activities are carried out. It is required when there is a dependency on two tasks, in such a way that one task cannot start before the other is finished (serial activities).

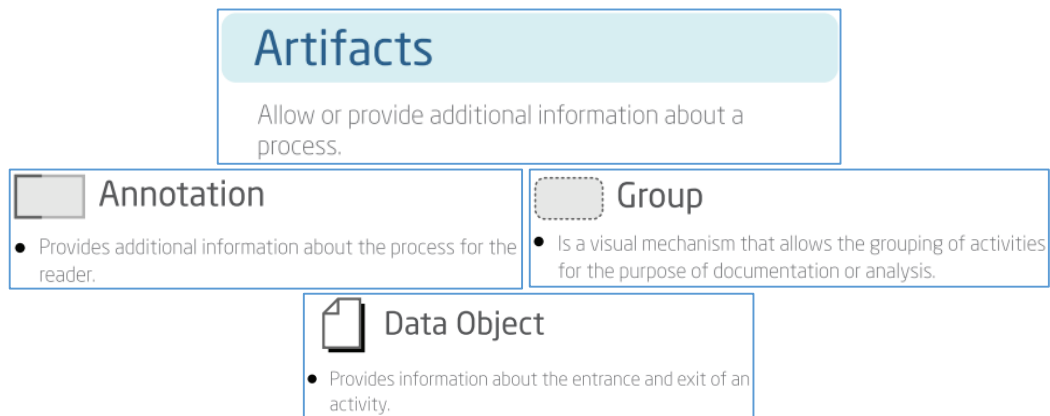
Each flow will have only one source and one destination and the sequential lines are represented by a solid arrow ending in a solid triangle.


◇ → Conditional Sequence Flow: in this type of flow, there is a condition that will be evaluated at run time to decide whether the path should be followed or not.


↘ → Default Sequence Flow: the situation where there are 2 or more output streams, one of which may be an implicit path, which will only be executed if all other streams are false at runtime is represented by an implicit sequential stream (with a diagonal bar).

○-⋯→ Message Flow: is used to show a message flow (a flow of information) between participants of the process.


⋯ Association: used to relate any information with BPMN graphical elements. Annotations and other artifacts can be associated with a graphic element through this connector.





 **Annotation:** is used to provide additional information that makes the diagram easier to read.

 **Group:** group activities and other elements together to highlight important blocks of operations.

It does not affect the flow of the process and does not add restrictions. Pools are positional elements, so they can cross multiple pools. For example, pools can be used to identify activities for a transaction distributed across multiple pools.

 **Data Object:** is an information or informational asset (form or document), which does not influence the process flow and is used to carry out activities, as input or output of an activity.

 **Data Object Collection:** represents a set of Data Objects.

 **Data Store:** system used to store, retrieve or update information for an indefinite period of time (it is a permanent repository of information).

3.3 Bizagi Modeler operating mode

3.3.1 BPMN process documentation

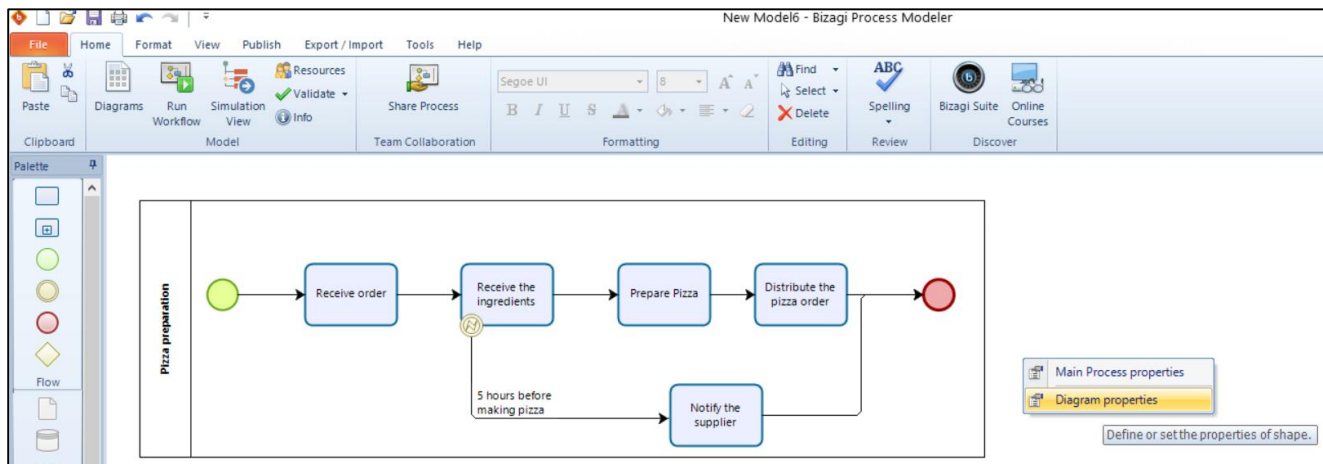
To describe all the details related to the sequence of elements that make up a process, to model a diagram, the Bizagi application offers various useful resources for documenting the process. Documentation is one of the most important steps in creating a model in Bizagi Modeler.

Through documentation, all process-relevant information will be included in the model, from process-level information to detailed element-level information within the diagram. Thus, in modeling the workflow, it will be possible to represent different resources used in documentation: various documents involved in activities, tables of business rules, control of execution time, etc.

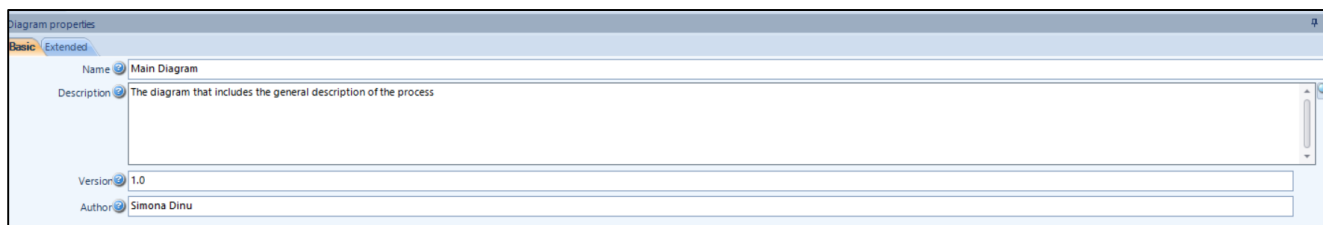
To accomplish this additional documentation, at the bottom of the Bizagi Modeler interface is the Element Properties panel, used to optimize the work of documenting all elements contained in a diagram, as well as the flow as a whole.

► **To add process level information:**

Right-click outside the pool boundaries → Diagram Properties:



The diagram properties panel, which opens at the bottom of the screen, allows the following information to be inserted: the name, description, version and author of the process:



Next, any element is selected and detailed information about each element contained in the process flow can be entered.

► **To access the properties of an element within the diagram:**

Right-click on the element → Properties.

The panel that allows accessing the element's properties is divided into the following tabs:

1) Basic properties: Contains basic information about the diagram element:



Name field: refers to the explicit text in each element.

Description field: allows you to enter detailed information and other considerations.

For example:

- in the case of the description of the properties of a diagram, a short description of up to 125 characters, of the scope of the respective modelling, can be made.

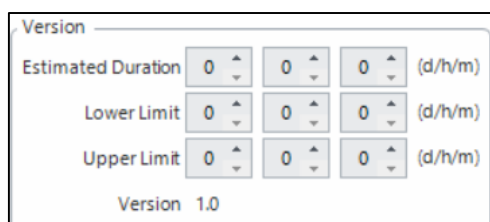
- in the case of describing the properties of a subprocess, it is recommended to use the description field to detail the scope of application.

- in the case of describing the properties of an activity, a more succinct description can be made, which includes the details that must be taken into account.

Category field: allows selecting the category to which the respective element belongs

Process field: property available only for sub-processes and allows to configure which diagram represents that sub-process;

Version field: property available only for diagrams and allows control of the version of the evolution of diagrams within a model:



Version			
Estimated Duration	0	0	0 (d/h/m)
Lower Limit	0	0	0 (d/h/m)
Upper Limit	0	0	0 (d/h/m)
Version 1.0			

Author field: property available only for diagrams and allows you to reference the author of the diagram within a model;

Timer Event Duration field: property available only for timer events and allows you to configure the moment(s) when the event will be triggered;

2) Extended Documentation: allows the creation of additional attributes for each process element to produce more detailed documentation:

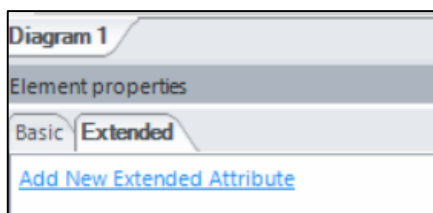
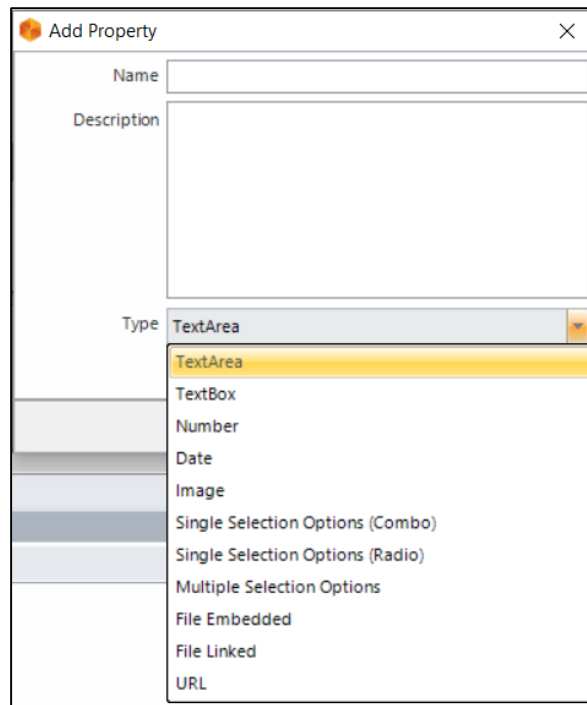


Diagram 1	
Element properties	
Basic	Extended
Add New Extended Attribute	

Note: If an extended attribute has been created for an element, that attribute will be available to all elements of the same type. For example, if a property was entered for an activity in the diagram, all other activities of the same type will have the same property.

At the diagram level, this information will not appear in the web publication, but only in DOC or PDF files.

Types of extended properties that can be assigned to elements:

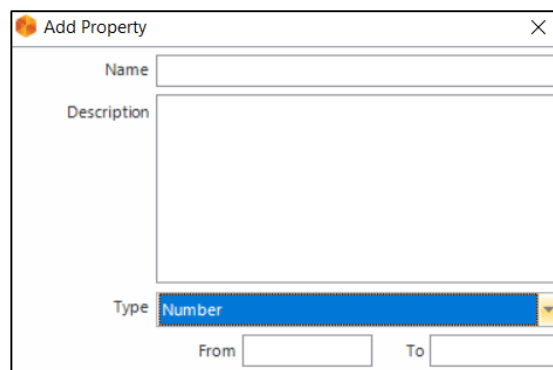


The screenshot shows a dialog box titled "Add Property" with a close button (X) in the top right corner. It contains three main sections: "Name" with a text input field, "Description" with a large text area, and "Type" with a dropdown menu. The dropdown menu is open, displaying a list of property types: TextArea (highlighted in yellow), TextBox, Number, Date, Image, Single Selection Options (Combo), Single Selection Options (Radio), Multiple Selection Options, File Embedded, File Linked, and URL.

-**TextArea:** it stores about 32,000 characters and displays long texts with line breaks. It is used to insert remarks, other considerations relevant to an item, rules, checklists, work products, and skills needed to perform an activity.

-**TextBox:** it stores about 32,000 characters and presents short texts without line breaks.

-**Number:** used to indicate the duration of an activity: a minimum and maximum allowable interval must be defined:



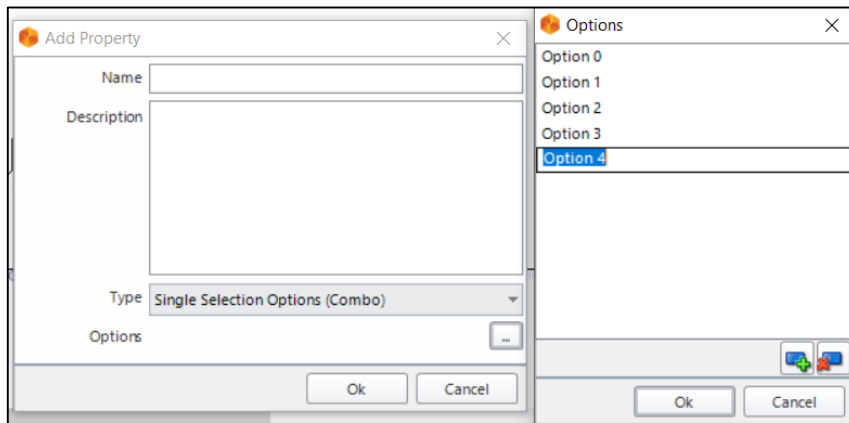
The screenshot shows the same "Add Property" dialog box. The "Type" dropdown menu is now set to "Number" (highlighted in blue). Below the dropdown menu, there are two input fields labeled "From" and "To" for defining the interval.

- **Date:** storage date - this attribute can be used to record the last update made to an activity.

- **Image:** allows the storage of images with jpg, bmp, png and gif extensions;

- **Single Selections Options (radio):** allows to define multiple options to be chosen, but only one of the check options can be selected;

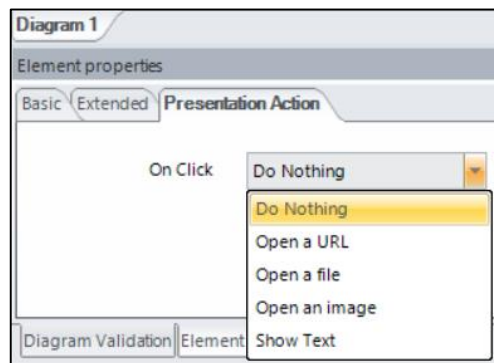
- **Single Selections Options (combo):** allows to define multiple options to be chosen, but only one of the options in the list can be selected:



- **Multiple Selection Options:** allows to define multiple selection options and allows to select one or more boxes from these options;
- **File Embedded:** allows to attach files to elements so that they are available in the model. The file is copied to a template directory;
- **File Linked:** allows to include a link or path to a file. The template stores a link to the file, not the file itself. Both relative path and absolute path can be used for linked file;
- **URL:** stores a link to an internet resource; attribute can be used to create a hyperlink to a website that is closely related to the execution of the activity being documented.

3) **Presentation Action:** defines what will be displayed in Bizagi Presentation mode.

Recommended to be used only at the activity level within a process or sub-process, to open a file, image, or URL that has some relationship to that activity.

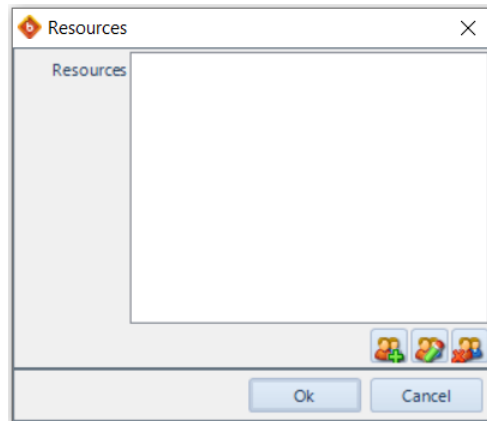


Defining model resources:

According to the specifications of the Bizagi application, by resource is meant the Business Entity or Role that controls or is responsible for a certain task.

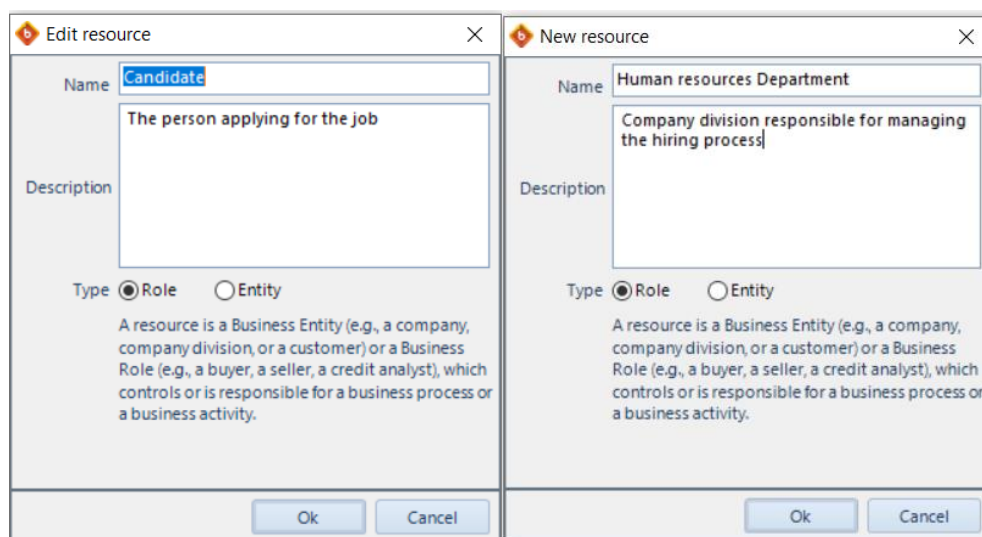
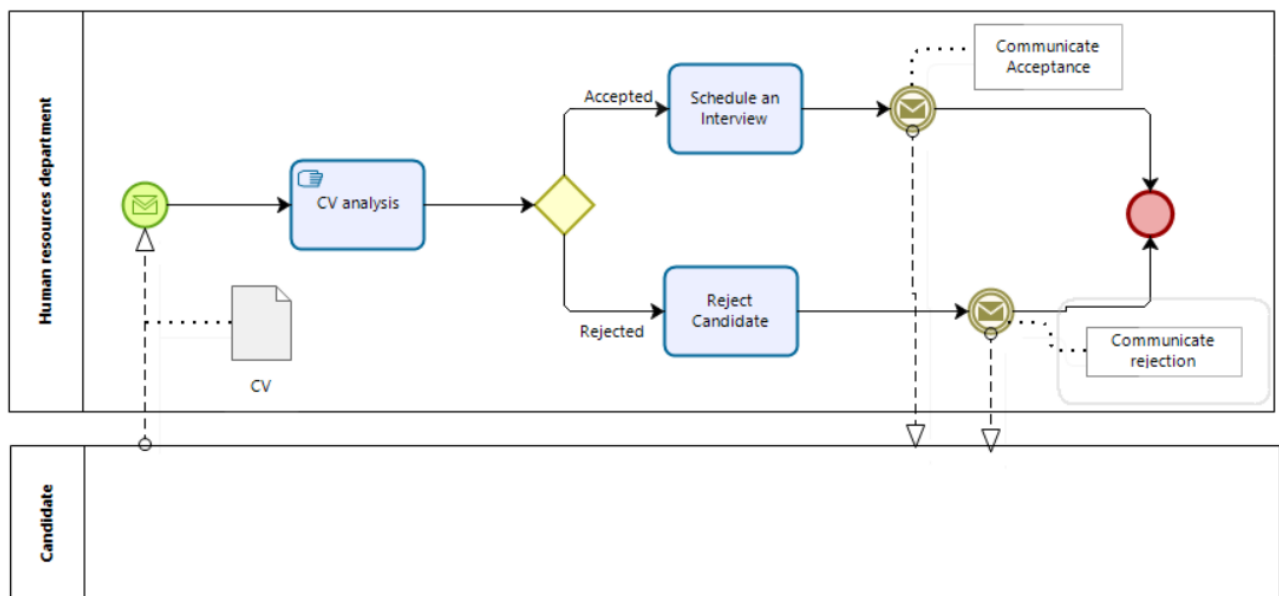
► **To define model resources:**

1. **Home tab → Resources:**



2. The buttons at the bottom right of the window that opens allow to **add, edit or delete any resource**.

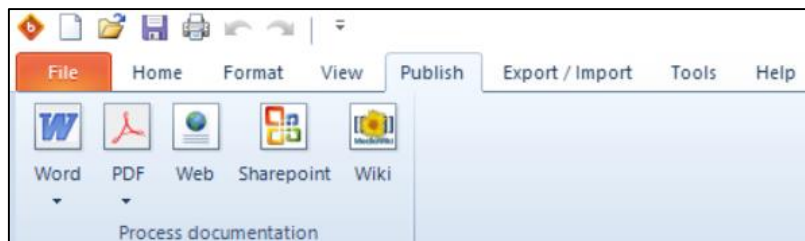
For example, in the model below regarding hiring a candidate, the resources of the two main participants of the process will be created: Human resources Department and Candidate:



Once these **resources** have been created, they **can be associated with any activity role**.

Publishing a process that has been modeled in Bizagi Modeler:

This option should be used when there is an interest in sharing the workflow, as well as all the documentation inherent to the elements that make up such a workflow, with the entire organization and external customers.

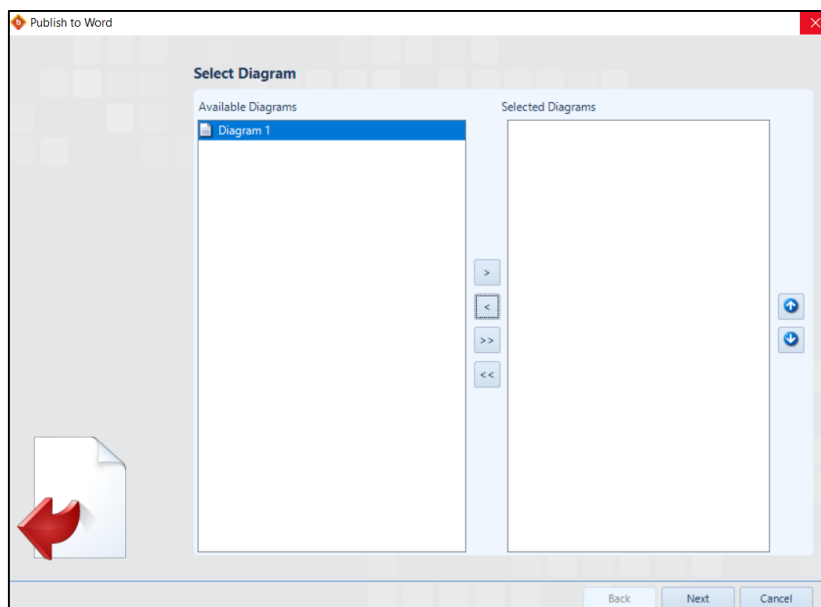


After designing the process workflow with Bizagi Modeler solution and after the process documentation has been completed, these documents can be published in editable text format (DOC), in non-editable text format (PDF), in collaboration tool format (Wiki), in Web format, or send it to a Sharepoint or Mediawiki server - information sharing tools.

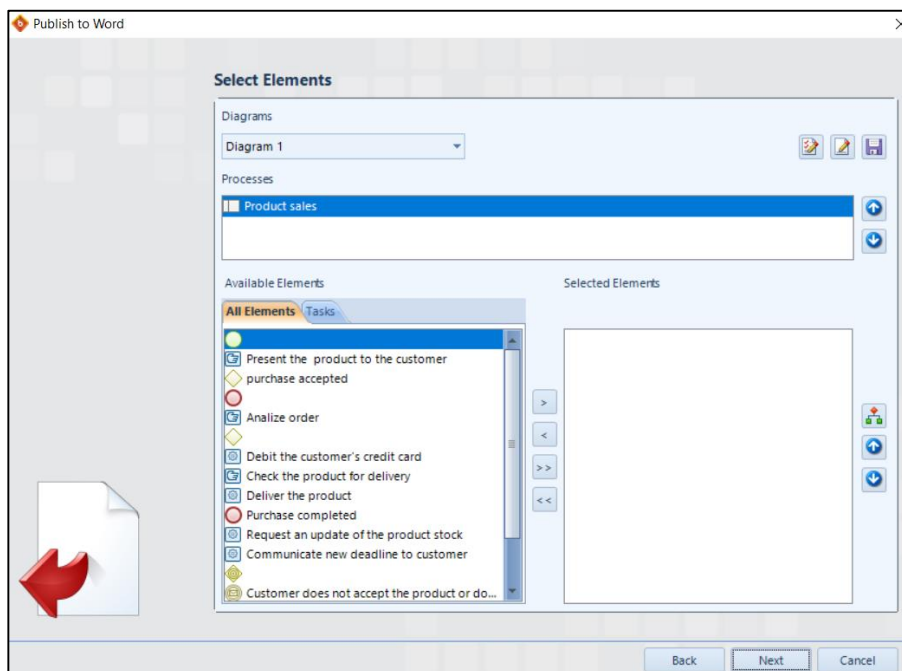
Note: In the DOC and PDF formats, a single file will be generated in the Word and Acrobat Reader formats, respectively, while for the Web format a folder will be generated that contains all the file structure necessary for an execution in the Web environment, and this structure cannot be modified.

► Publishing in Doc/Pdf format:

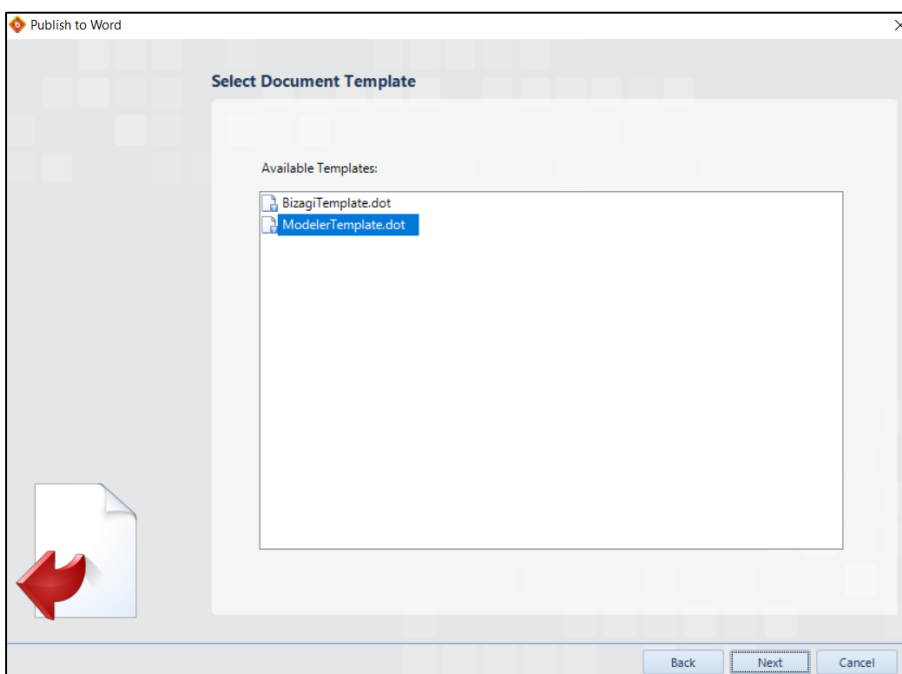
1. **Publish** tab → **select the button corresponding to the format in which the publication is to be generated.**
2. In the **window that opens** → **first send all the diagrams you want to publish in the box on the right.**
3. Using the vertical arrows, you will **organize the charts so that they are published in the order you want** → **Next:**



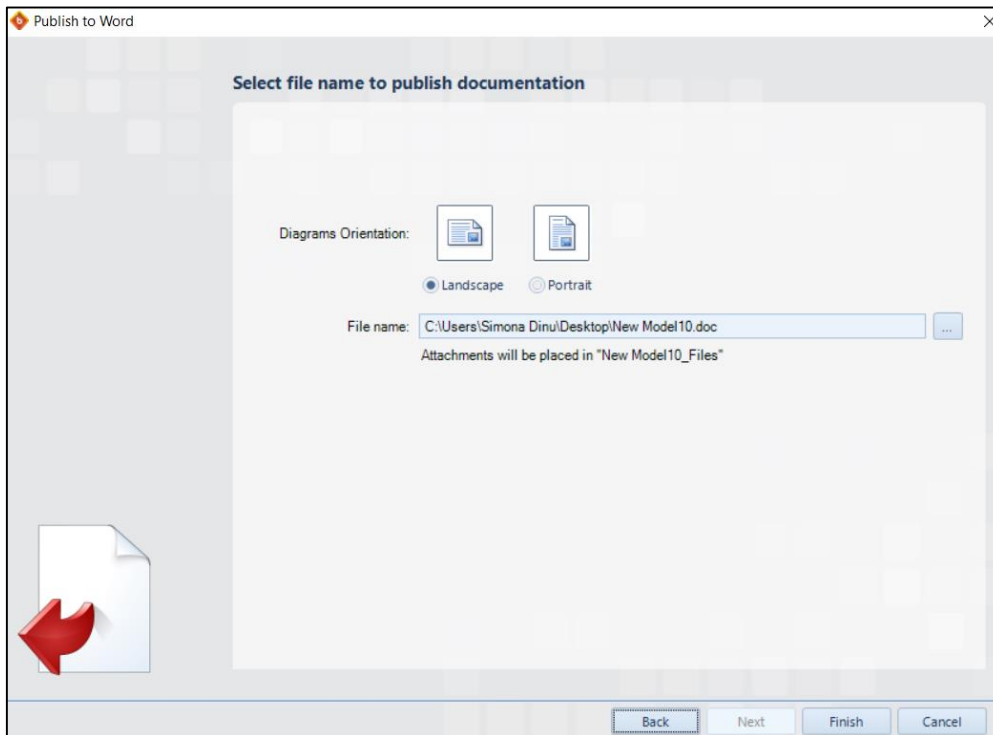
4. In the next window that opens, the **Diagrams** field contains all the diagrams that will be published. **Selected diagrams** one by one → at the bottom of the window, send all the elements you want to publish from the diagram in question in the box on the right → **Next**:



5. In the next window that opens, **select the document template that will be generated**: the **BizagiTemplate.doc** or **ModelerTemplate.dot** option → **Next**:



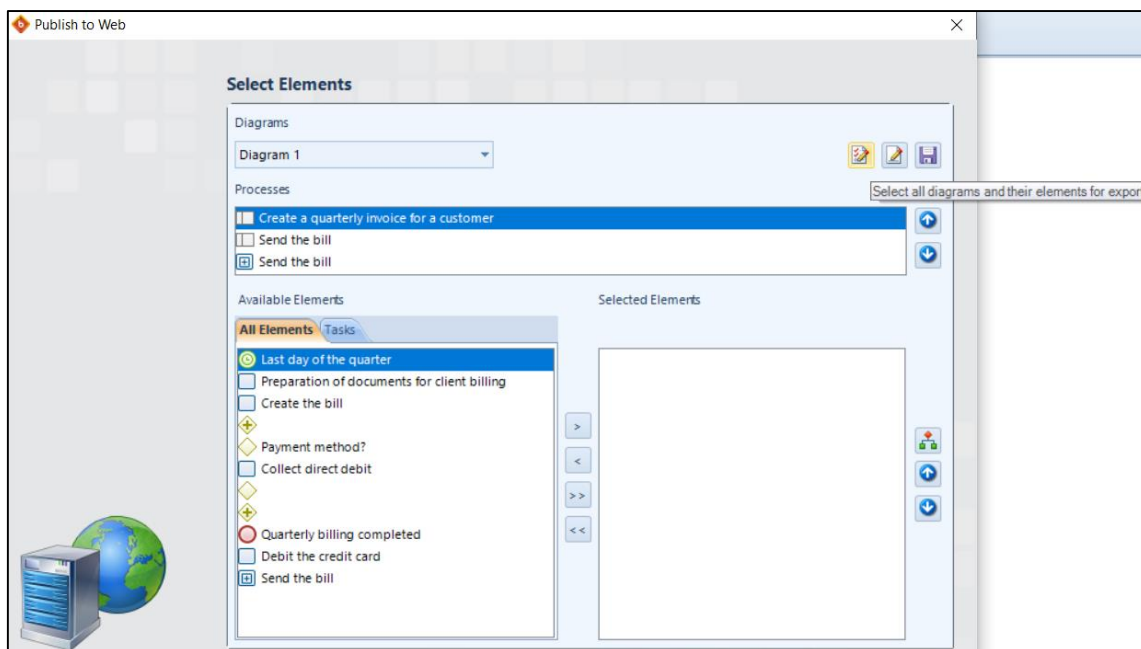
6. In the last window of the step-by-step publication guide, **select the orientation of the paper** in which the diagrams will be generated → **select the directory** where the publication document will be generated → **Finish**:



Note: The publishing result will be the same for both DOC and PDF format, only the file format differs.

► **Publishing on the Web:**

Follow the first three steps above, and in step 4, in the window that opens → click on the **Select all diagrams and their elements for export** button, because all the elements of all diagrams will be published. Using the vertical arrows, one can arrange the diagrams, so that they are published in the desired order:



Note: The directory generated after the Web publication is complete cannot be modified because there are a number of dependencies between the objects contained in it.

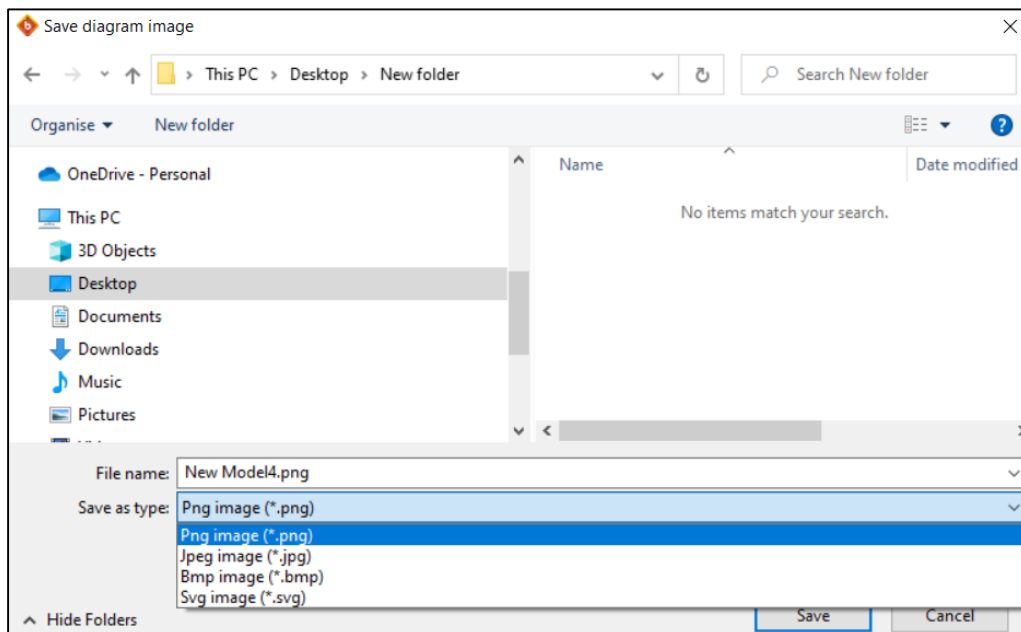
► **Exporting a process that has been modeled in Bizagi Modeler:**

The export is carried out when the flow of a process must be transferred to another modeling tool, such as Visio, or when certain elements must be reused in another process modeled through Bizagi Modeler.

One can also export the flow in an image format, in which case, unlike publishing, only the design of the flow is generated, so no documentation will be associated with the image file.

To export in an image type format:

Select the button corresponding to the format in which you want to generate the export → **select the directory** where the image file will be generated → **give the file a name** → **select the type of image** that will be generated, which can be PNG , JPG, BMP and SGV → **Save**:



3.3.2 Simulation in Bizagi Modeler

Simulation is a tool that allows the improvement of business process models built in BPMN by evaluating the execution of the models, under different instances and over different time periods. Thus, the aim is to optimize the performance of the process, by eliminating unforeseen blockages or overuse of material and human resources, correcting specifications, etc. To ensure that the results are valid, it is necessary that the simulations are run long enough to produce random behavior with minimal possibility of error.

Simulation in Bizagi allows rigorous "What-If" analysis methods; a simple simulation run can provide a variety of insights into the execution of a given scenario. In addition, the simulation of different scenarios and the possibility to compare the results represent an important support for decision-making.

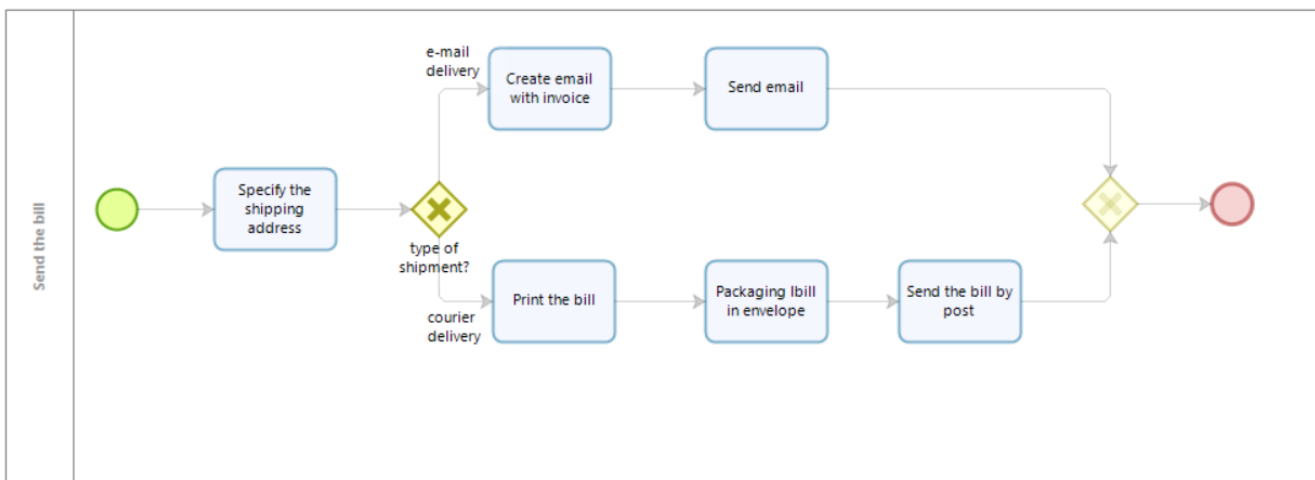
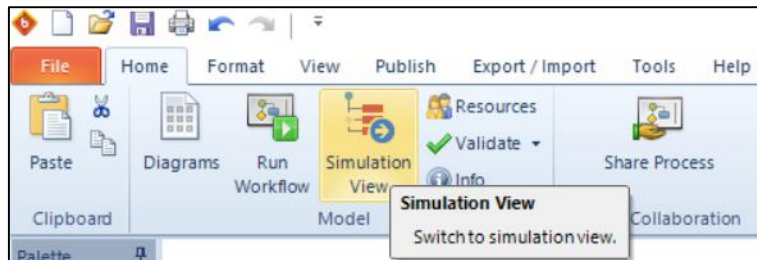
To create a simulation scenario, the process diagram must first be completed. Afterwards, the four simulation levels are run one after the other. Each subsequent level incorporates additional information that will add more complexity, providing a coherent analysis of the process.

For each simulation level, the following steps are performed:

- Collection of process data for simulation;
- Adding data to the diagram;
- Running the simulation;
- Interpretation and presentation of results.

► **To define the specifications of a scenario:**

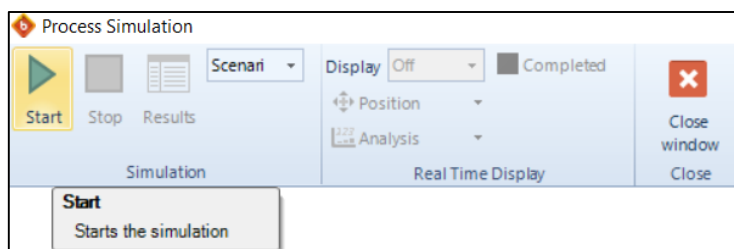
In the **Home** tab → click on the **Simulation View** button → the diagram will be presented in read-only mode:



For each simulation level, the elements that require information will be highlighted.

The first step is to define the properties of the scenario. The main information to be defined are: name, duration, time unit, currency unit and number of replications. The number of 30 replications is recommended to ensure that the simulation will reach a steady state. After entering the required data, save the simulation information and then simply run the simulation:

Click the **Run** button to open the simulation view → Click the **Start** button to run the simulation:



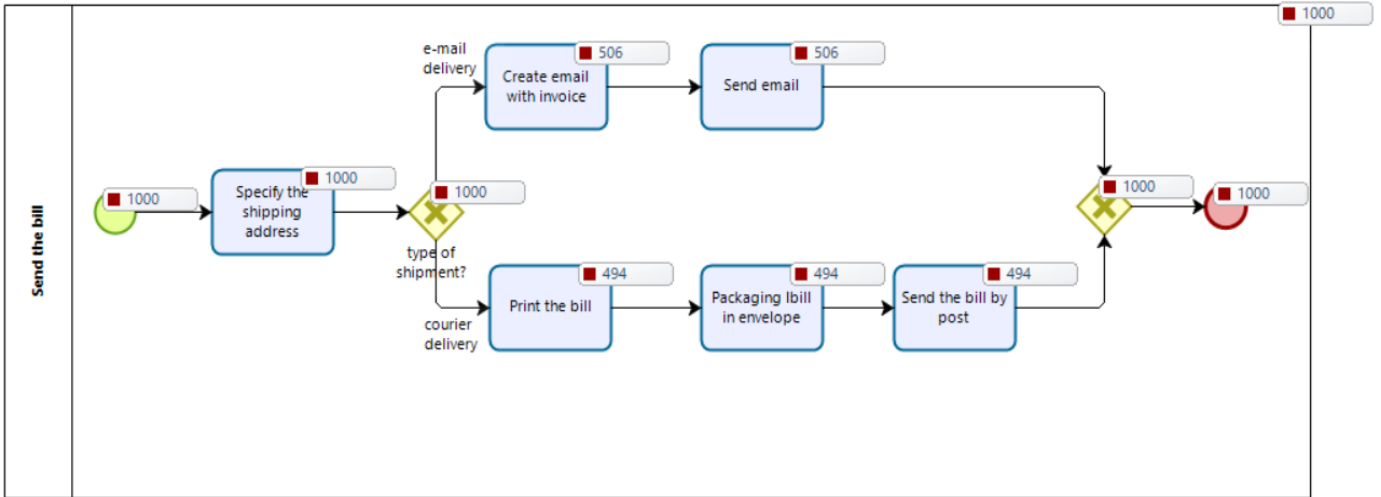
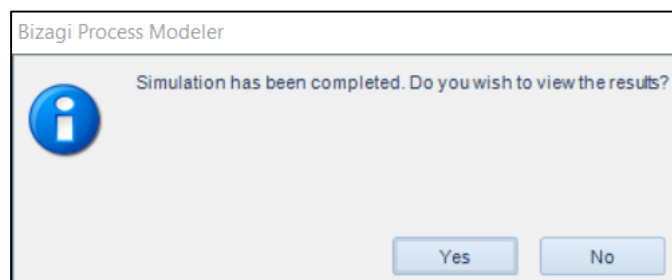


Table 1 presents the results generated when executing the simulation that allowed the validation of the process. In Table 1 it can be seen that the operation is as expected given that the number of instances created (1000) is equal to the number of instances completed (506 + 494), so no inefficiencies are generated in the workflow from the definition of the process.

Table 3.1: Simulation results on process validation

Name	Type	Completed instances
Start	Start event	1000
Specify the shipping address	Task	1000
Create email with invoice	Task	506
Send email	Task	506
type of shipment?	Gateway	1000
Print the bill	Task	494
Packaging bill in envelope	Task	494
Send the bill by post	Task	494



Note: Converting trial to a paid subscription allows visualization and analysis of simulation results, as well as access to various other advanced functionalities of the application.

3.4 Automation of Business Processes: Process automation Wizard of Bizagi Studio software

The automation of business processes is the use of technology (different software tools and application integrations) to carry out repetitive tasks or activities in a firm when manual labor may be substituted. Cost reduction, increased efficiency, and process optimization are the goals.

The benefits of automation are obvious:

- The more automated the business, the more employees can focus on the work that matters. Automating processes will allow employees to stop focusing on unnecessary or low-value tasks. They will be able to focus on the work that creates the most value for that business.
- Increased employee satisfaction: light and robotic tasks are extremely bad for motivation and work enjoyment. If such things can be automated (for example: distributing mail through internal office means), employees will be happier doing more meaningful work.
- Minimizing human error: No matter how much attention to detail a company's employees have, there is always a small chance that someone will forget something or make a mistake.

The consequences here can range from minor to catastrophic. The right software will remind them of their tasks regularly.

- Greater control by the system administrator of past and current tasks: within the automated process, the administrator has the ability to observe all interactions between users, which provides the necessary information to take corrective or preventive actions as appropriate.
- Cloud-based business process automation tools store company data in a central database, which will make it easy to access it from any location or device, whenever needed. Thus, business processes will be much more transparent, processes can be tracked and monitored as they run, which can improve accountability and visibility.

Also, the ability to monitor processes on the fly will also help detect and correct errors as they occur, and performance reports will provide insights to take preventative action against recurring errors.

- From a long-term perspective, faster response times and reduced costs due to fewer manual interventions will be observed.
- Another benefit relates to improving labor allocation, as the application will take care of all recurring trivial tasks. In this way, employees can be redirected to tasks that require human effort and judgment.

The business process automation system will ultimately enable business efficiency to increase.

Depending on the type of processes that want to be automated, one can opt for different software. In the Bizagi application, the automation stage seeks to convert all the activities of the process flow into a technological application, and in this sense it provides a Process automation Wizard that guides through all the necessary steps for the automation and execution of business processes.

This **Process automation Wizard** is designed in the following seven steps:

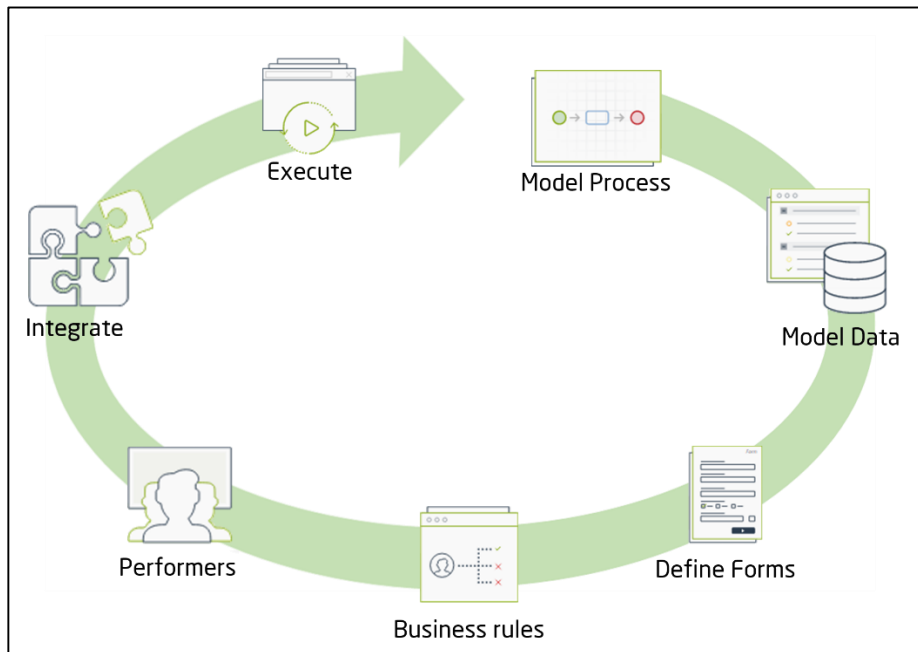


Figure 3.31: The seven steps of the Process automation Wizard

[https://help.bizagi.com/bpm-suite/en/11.2.4.2x/index.html?modeling_data.htm]

- **Model Process:** allows to diagram and design the process flow in a complete way
- **Model Data:** allows the design of a data model that organizes and stores the information used in the different activities of the process.
- **Define Forms:** allows the design of user interfaces that will be displayed throughout the process.
- **Business Rules:** allows to define the flow conditions of the gateways, i.e. to define the forms that will be displayed according to the course of action of the process.

This phase allows defining the expressions needed to model the behavior of the business situation.

- **Performers:** defines and designates the users responsible for executing the various activities of the process.

- **Integrate:** allows the configuration of connections with external systems by using web services.

This is an optional process within the methodology.

- **Execute:** is the final phase that allows processes to be brought into test and production environments.

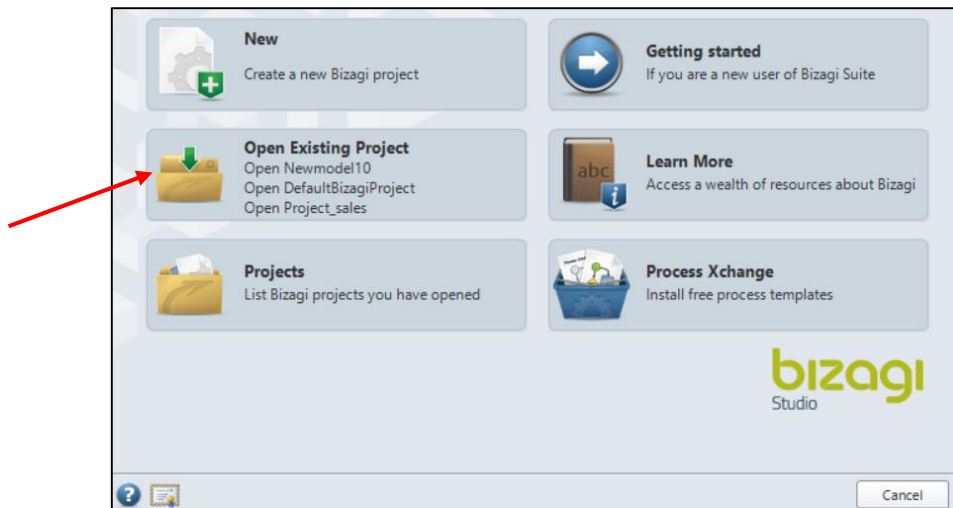
3.4.1 Data model creation

Once the first modeling stage is completed, by generating the process diagram, you can move on to creating the data model that will include all the information requested by the Process.

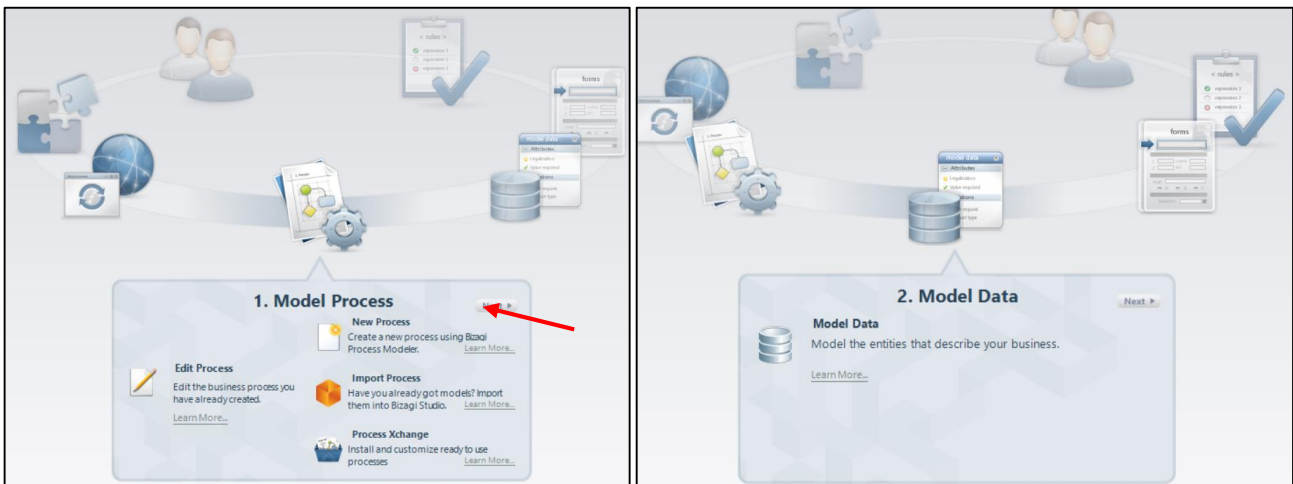
The data model in Bizagi defines how data should be stored and accessed.

To perform this step:

1. Access the **Bizagi Studio component** → open one of the previously saved projects:



2. In the window that opens: press the **Next** button → **Model Data** component:



3. In the window that opens: enter the **name for the Process Entity** - the entity that gives access to the rest of the data model → **Ok**:



4. The new window that opens allows **the design of the data model**, which contains the Process Entity: entities, attributes and relationships can be created.

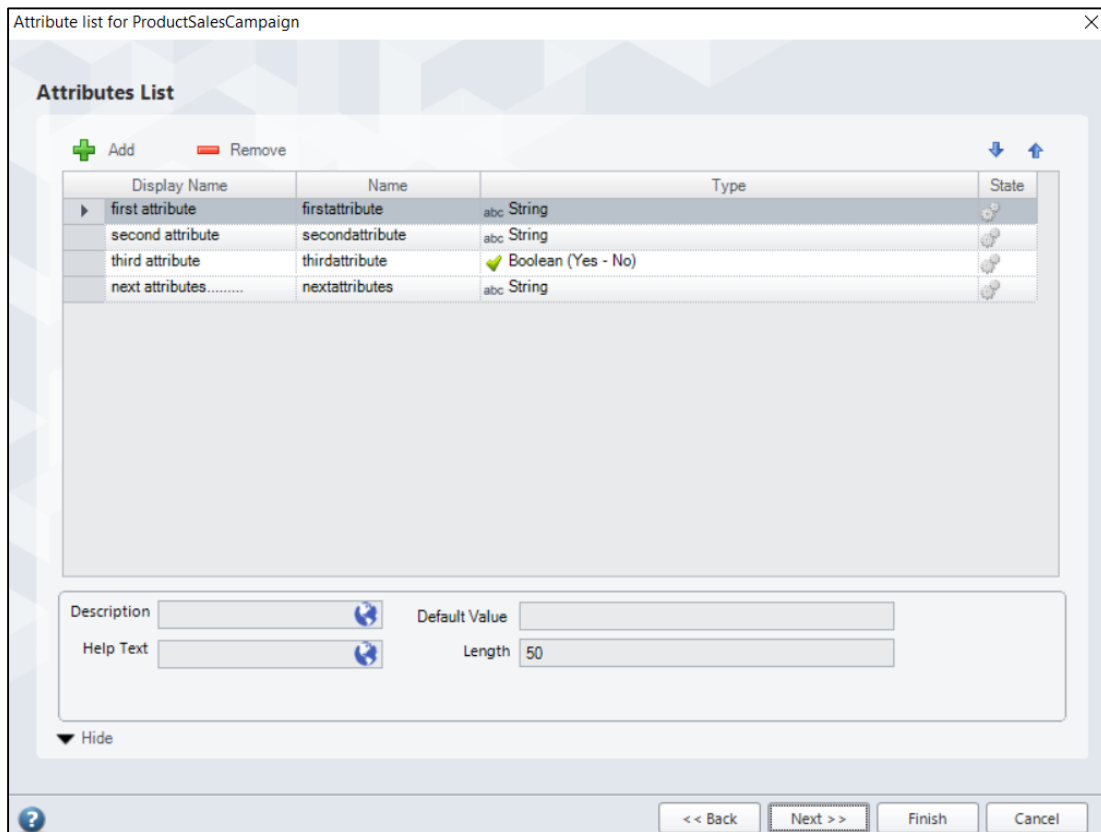
Bizagi provides four types of Entities and four types of Relationships to build the data model.

Entities: represent abstract or real objects, such as people, physical resources, buildings, etc., that can be uniquely identified and that have information of interest to the business.

Entities are characterized by attributes which are the properties that describe each entity. For example, a person has name, age, salary, department, identification number, etc.

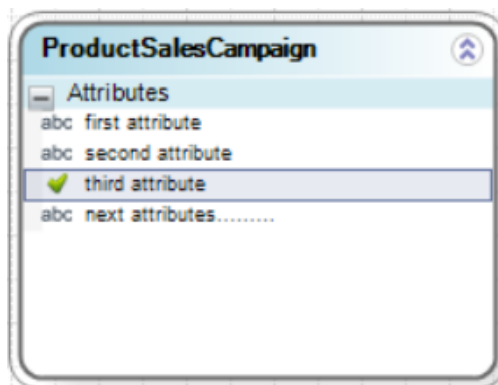
For an entity instance, Bizagi automatically generates a consecutive number that identifies each entity record. This identifier is called a Surrogate Key and uniquely identifies each row in the entity, unrelated to the attribute data.

5. Right-click on the respective entity → Edit Attributes List:



6. After entering all the desired attributes: **Next** → **Finish**.

7. The entered data is presented graphically in the **diagram screen**:

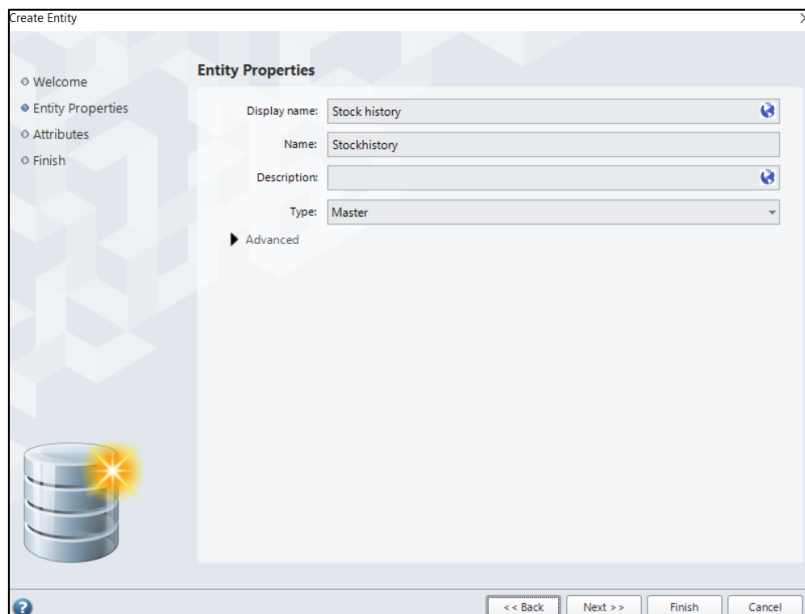


Note: according to the specifications of the Bizagi platform (https://help.bizagi.com/bpm-suite/en/index.html?attribute_types.htm), the following types of attributes can be defined:

ATTRIBUTE TYPE	DESCRIPTION
Boolean	Stores one of only two Boolean values, true or false.
Currency	Stores a numeric value with decimals using the currency format and decimals defined in the Business Configuration . Due to the database's engine restrictions, this value limits decimals up to 4 digits. For further information, refer to Money topic.
Date - time	Stores an attribute that can be a date, or a date and time.
Integer	Stores an integer in the following range: -2,147,483,648 to 2,147,483,647.
String	Stores a string. Its length can be defined in the additional properties found in the Advanced option of the Entity Wizard. Please see image below.
Big integer	Stores an integer in the following range: -999,999,999,999,999 to 999,999,999,999,999
Extended text	Stores a string with no character limit.
File	Stores and attaches uploaded files . Further options are set in the Environment Configuration (such as maximum filesize), in the Advanced tab under Uploads options. It also creates and stores Document Templates . It also offers ECM integration possibility.
Float	Store floating point numbers in 8 byte binary format with up to 38 decimal digits and 15 significant digits of precision.
Image	Stores uploaded images and displays them in the Work Portal as thumbnails.
Real	Store floating point numbers in 4 byte binary format with up to 38 decimal digits and 7 significant digits of precision.
Small integer	Stores an integer in the following range: -32,768 to 32,767.
Tiny integer	Stores an integer in the following range: 0 to 255

8. Next, more entities can be added to the diagram:

Right click anywhere on the Diagram screen → select New Entity:



9. Enter the **name of the entity** in the **Display name** field → select **Master** as entity type → **Next**.

In Bizagi, six different types of entities can be selected: Master, Parameter, System, Application, Stakeholder and Runtime, each with a specific purpose.

- **Master entities** are business entities that store information that is directly and exclusively related to each process. Each process in Bizagi has a Master process entity. This process entity is a starting point to access the rest of the data model, in other words, it is the main entity through which users access the rest of the entities in the data model.

- **Parameter entities** store predefined values or parametric values, which are independent of the execution of the process. For example, the Gender entity contains values such as Male and Female.

In Bizagi one can include as many parametric entities as the project requires.

It is possible to establish a relationship between master entities and parametric entities.

- **System entities** are entities that belong to Bizagi's internal data model. These entities contain information related to the work environment such as current user, area, location, roles, positions, etc.

These system entities are created by default in each project and cannot be modified or deleted. Relationships can be established with other entities to include information from the system entities within the data model.

- **Application entities** are entities related to the data model of the entire project and are used to centralize the information of each application. These entities are created by default to allow structural organization of the process when the application is created and cannot be modified or deleted by users.

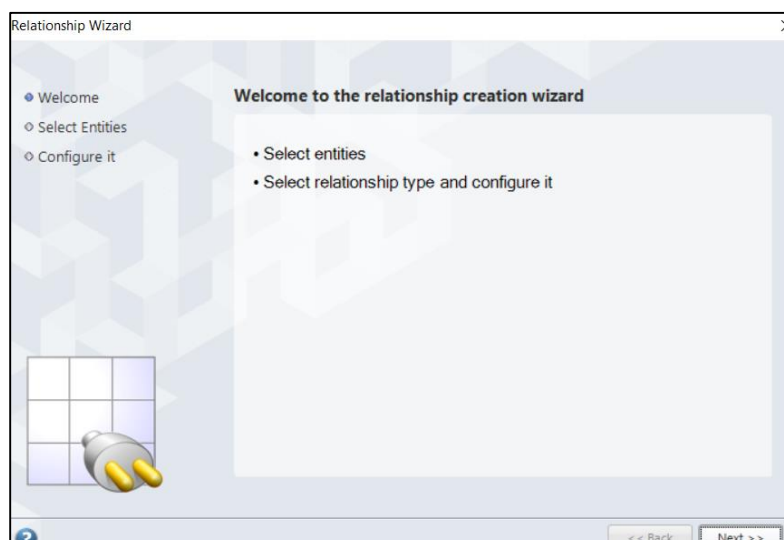
- **Stakeholder entities** represent those who have an interest in the project: Employee, Customer, Sponsor, etc.

- **Runtime entities** are created implicitly in each project and contain read-only information relating Bizagi's process and cases internal data.

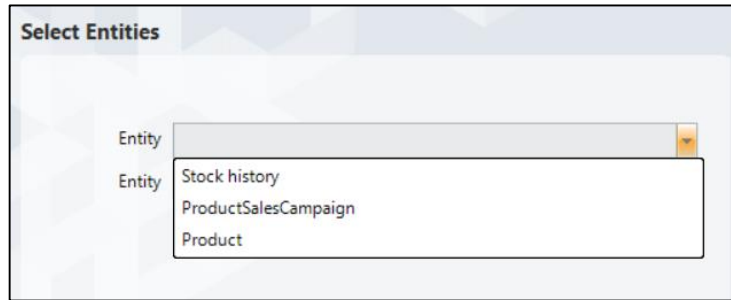
10. Next, attributes are assigned to the entity as was done for the ProductSalesCampaign Entity.

11. **To establish relationships between entities** added to the diagram:

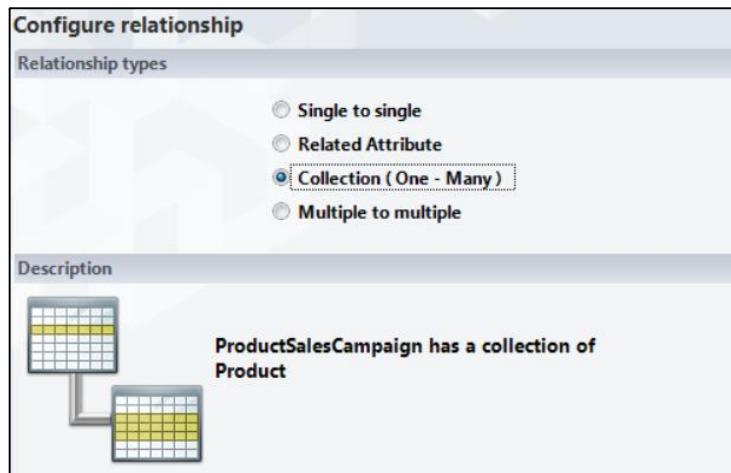
Home → Relationship → the Relationship Wizard will open → Next:



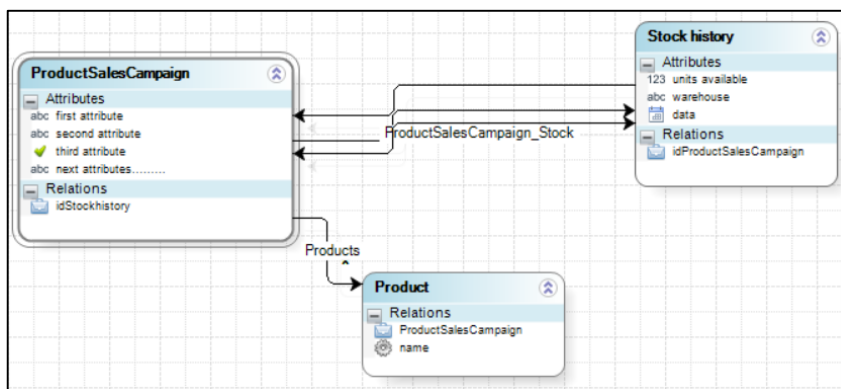
12. In the **Select Entities window** that opens → **select the two relationships** between which the relationship will be defined → **Next**:



13. In the **Configure relationship window** that opens → **select the relationship type** → **Finish**:



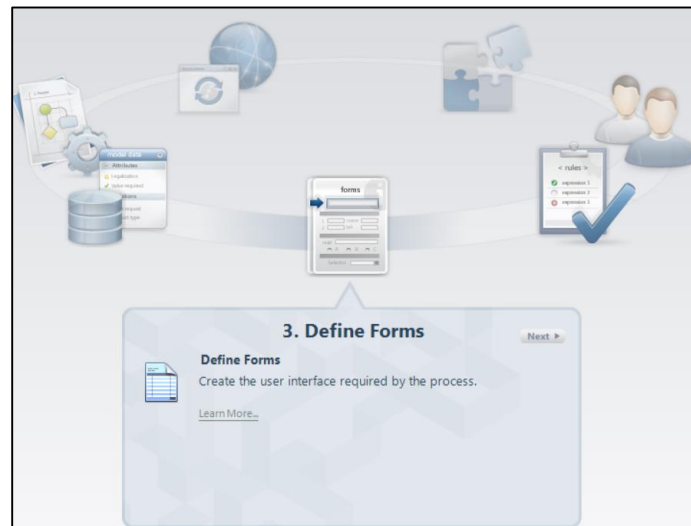
14. Next, the other entities will be connected to the ProductSalesCampaign entity:



15. **Save the data model** → **close the Diagram window** to return to the Bizagi Studio application window.

3.4.2 Forms design

Once the Process Diagram and the Data Model are completed, it's on to a new stage: creating the forms (user screens) associated with each of the human activities of the process. In this step, Bizagi allows to design the different forms that will be used by the participants of the process:

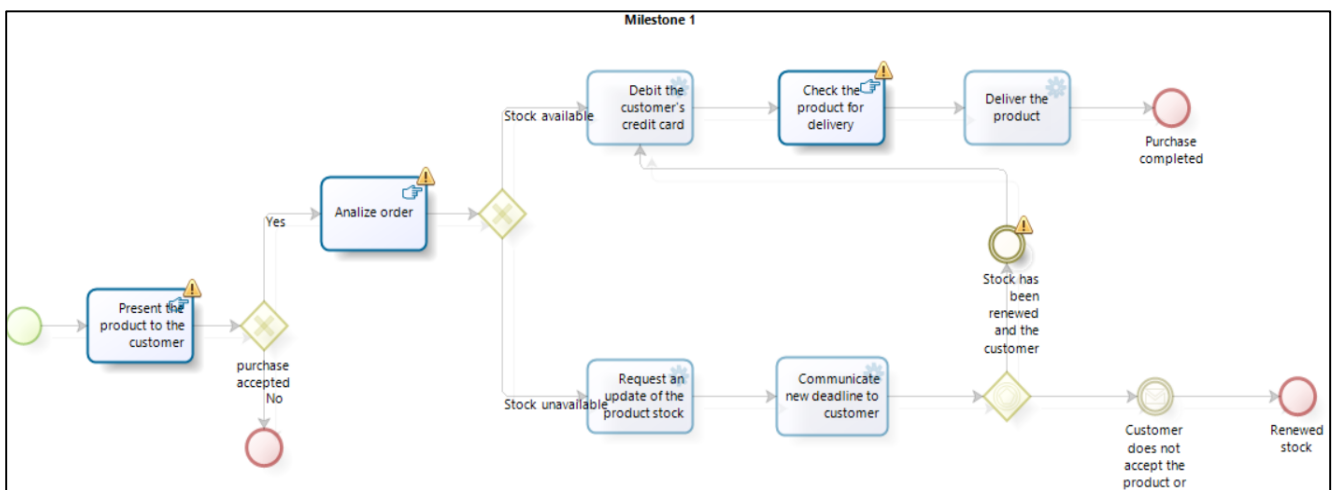


The form designer implemented in the Define Forms stage, manages all user interfaces for human activities.

Building forms for users is a very intuitive process, because for each task a form is created by inserting the necessary data using the drag and drop feature of the data already recorded in the previous step. Bizagi incorporates a set of controls and tools, which can be dragged and placed within the forms, graphically.

Forms can be configured to enter data, multiple-choice lists or images; data can be retrieved from the data model, and graphics displayed.

In addition, this interface creation module allows defining both simple and complex validations and executing actions on the information entered by the user to ensure that all the information recorded in the process is correct and appropriate for processing.



In this diagram view, only user tasks are available to create Forms in; those user tasks that have no forms associated are highlighted with an exclamation mark in order to alert the user that they must define a form for said activity. Elements in the process that do not need forms will be displayed in read-only mode.

The specifications of the BizAgi application offer the following specifications related to the creation of process forms:

Each item or element included in a Form is known as a Control in Bizagi.

Elements are added onto the form individually by means of drag-and-drop from the Left panel of the Forms Designer. They can be added in one of two ways:

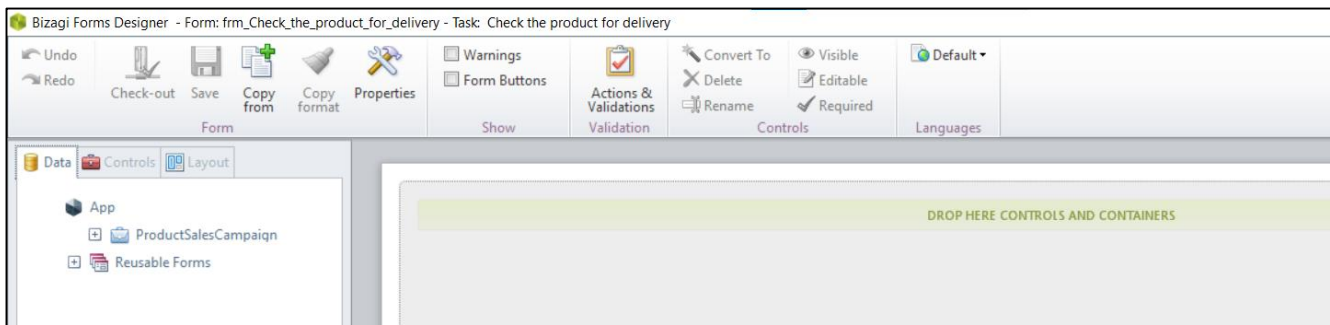
- Add the attributes directly from the data model located in the **Data** tab.
- Select a control type element from the **Control** tab and then associate the attribute.

When an attribute or control type element is added to a Form, it is interpreted by Bizagi and becomes a **Control**.

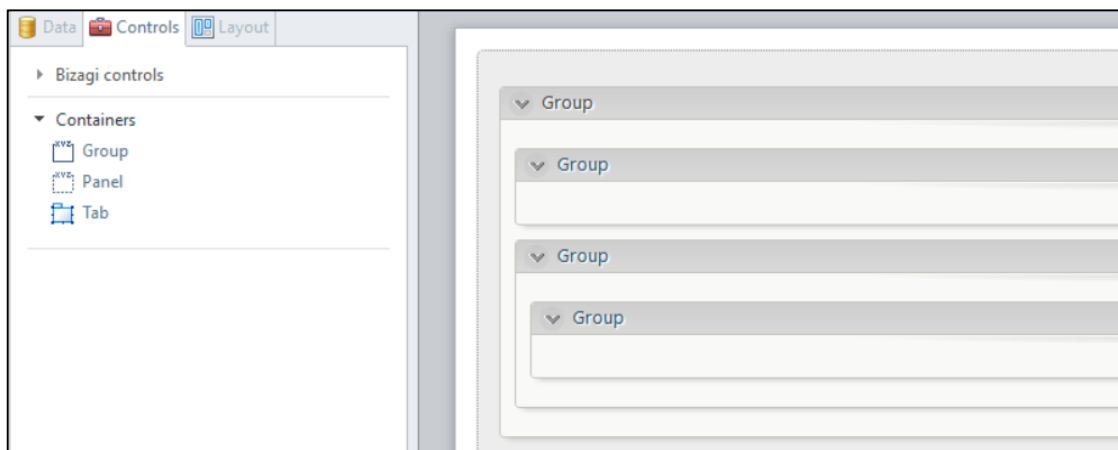
Each Control originating from an attribute element has a type: text, Date, Combo (drop-down list), Yes/No, Number, etc, based on the attribute type. Controls originating from control type elements (Control tab) will merely inherit the control type. Examples include Money, Text Box and Search control types.

https://help.bizagi.com/bpm-suite/en/index.html?forms_controls.htm

1. Click on such a task to create its form. For example, if the "Check the product for delivery" task is selected, the **Forms Designer window** appears:



2. From the **Containers** menu located in the **Controls** tab → **drag and drop onto the Display Area Groups** to place the attributes in the form:



3. After configuring all the forms → **Save** the form → **close it** to return in the Bizagi Studio application window.

3.4.3 Definition of business rules that control the Process

To automate a process, some rules must be defined by which certain expressions are specified that control, for example, the flow sequence, to define which path a process should take according to its conditions. These business rules can be applied to exclusive gates within the process (where only one of the business rules must be valid for the flow to continue one-way), but also to Script Tasks, etc.



Business rules for routing or operations

A business rule is a regulation that must be respected in the business field. According to Business Rules Group (2001), a business rule is a statement that defines or restricts some aspect of business in such a way that it is never possible to perform invalid actions. Business processes are governed by rules that guarantee proper execution in accordance with the organization's strategies, objectives and philosophy. Business rules establish the procedures to be executed and the conditions to be evaluated and controlled in the process flow.

Business rules apply to well-defined points in an application and are intended to perform an action that contributes to the organization's goals and prevents invalid actions from being implemented.

BizAgi includes a powerful Business Rules Engine through which business rules can be defined for:

- Process flow transitions: flow control to define the path the Process should follow according to specific business conditions. For example, if a travel request has been approved, the respective bookings must be proceeded with, otherwise the rejection must be notified.
- Executing procedures necessary to complete a task, such as validations and calculations. For example, when an employee reports the number of hours allocated to a project, the total time should be calculated automatically.
- Manipulation of user interfaces to avoid errors in the information entered in the process. For example, if an expense claim is rejected, the rejection comment control must be shown, otherwise it must be hidden.
- Defining the conditions that a user must meet in order to be assigned a task.
- Automatic notifications by sending emails.
- Defining the editing, visibility or mandatory conditions of the form fields.

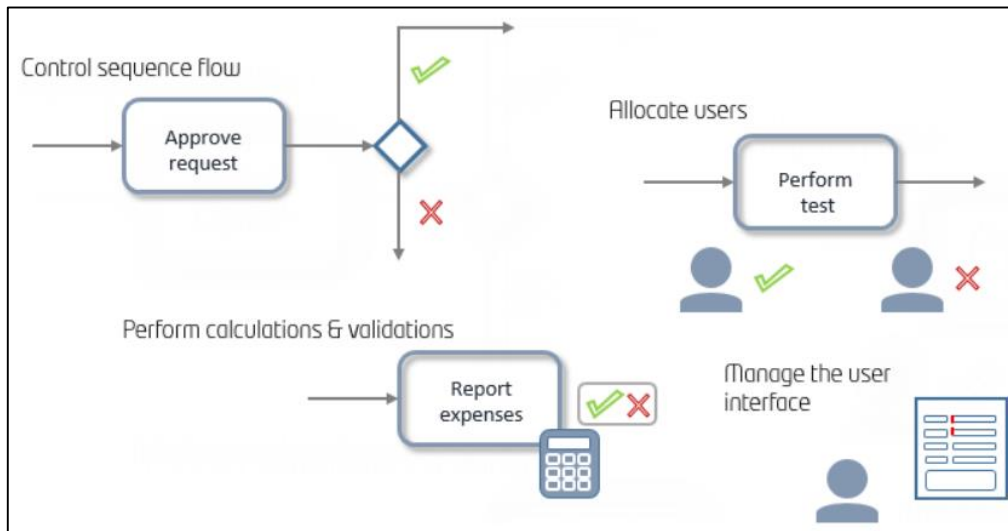


Figure 3.32: Where to use business rules

[\[https://help.bizagi.com/bpm-suite/en/index.html?defining_business_rules.html\]](https://help.bizagi.com/bpm-suite/en/index.html?defining_business_rules.html)

As rules solve different business situations, Bizagi helps modeling in an organized way by classifying each of the rules according to their usage. Thus, when a rule is associated with a specific situation, Bizagi will list only those that correspond to that category.

At this stage, in the Bizagi application there are two options:

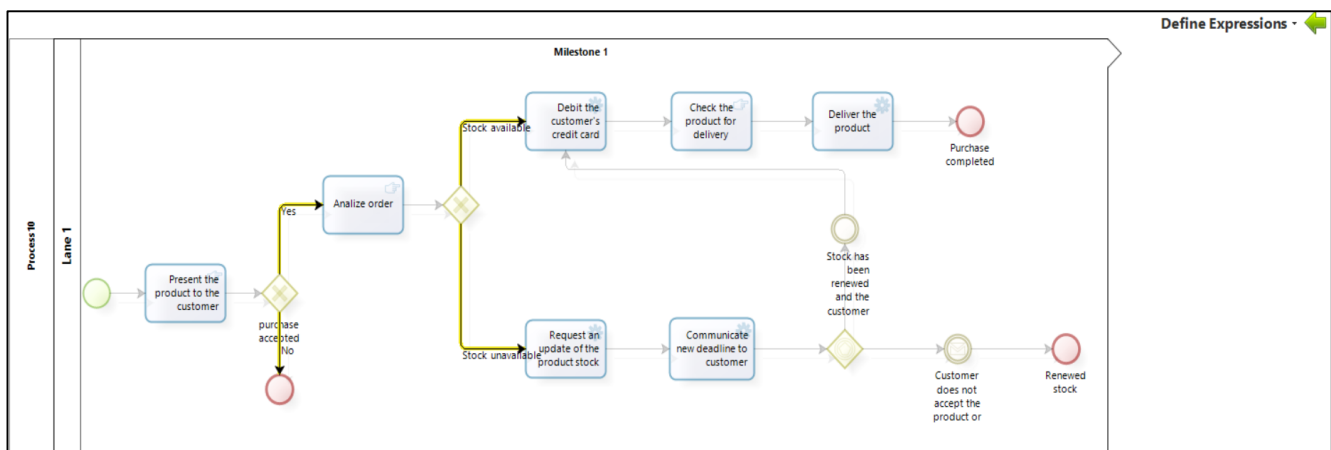
► **Define expressions** necessary to model the behavior of the business situations. These are expressions associated with sequence flows (Transition conditions).

Business rules will help determine the path to be followed by the process flow when reaching the two divergent Gateways that require an expression: Exclusive and Inclusive Gateways or a conditional task.

These expressions direct the process flow: the path that a process must follow is defined according to the condition defined in the process.

To define rules that evaluate conditions and decide where the process flow should continue:

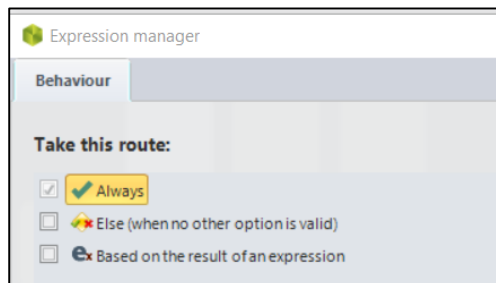
1. Select **Define Expressions** → Bizagi shows the process flow **highlighting the transitions that do not have associated rules**:



2. Clicking on a transition will present three options for following a route:

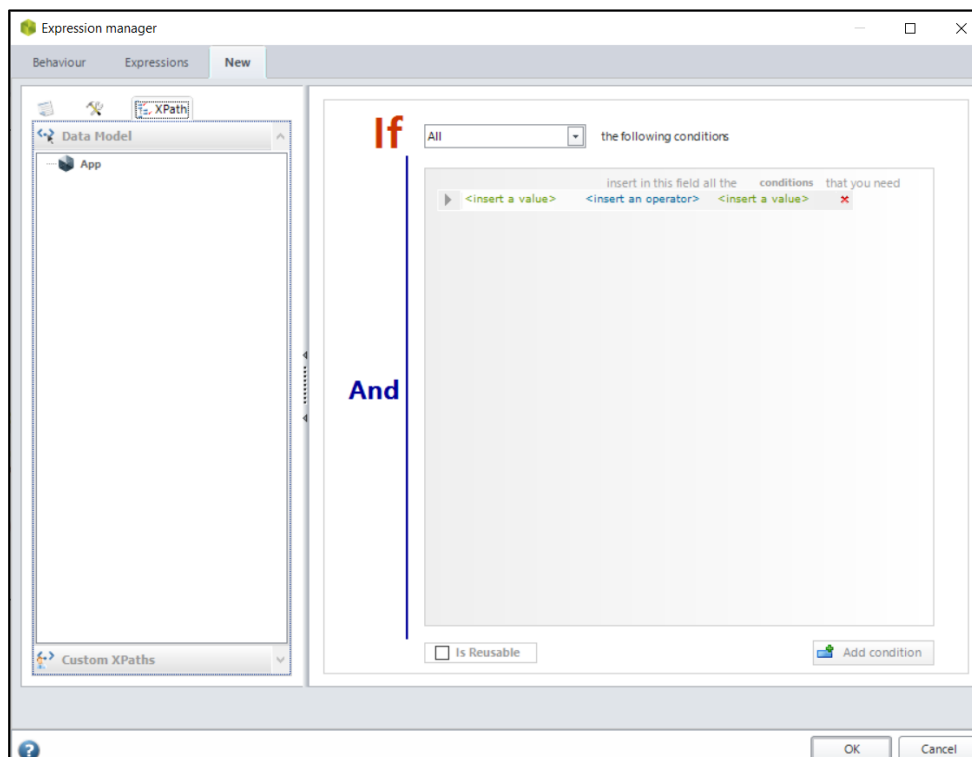
- **Always:** when selected, Bizagi will continue on this route ignoring the others sequence flows.
- **Else:** when selected, Bizagi will continue on this route when no other route is valid.
- **Based on the result of an expression:** when selecting it, Bizagi will evaluate an expression to know whether or not to continue on that route.

For example, selecting the transition named Yes that emerges from the gateway "purchase accepted ?", three options will be presented to continue on a route:



3. Selecting the option **Based on the result of an expression** will open the **Expression manager** window that will display the list of system expressions and previously created expressions.

Since there are no expressions created, click **New** → The New tab will display to **enter the Boolean expression**:



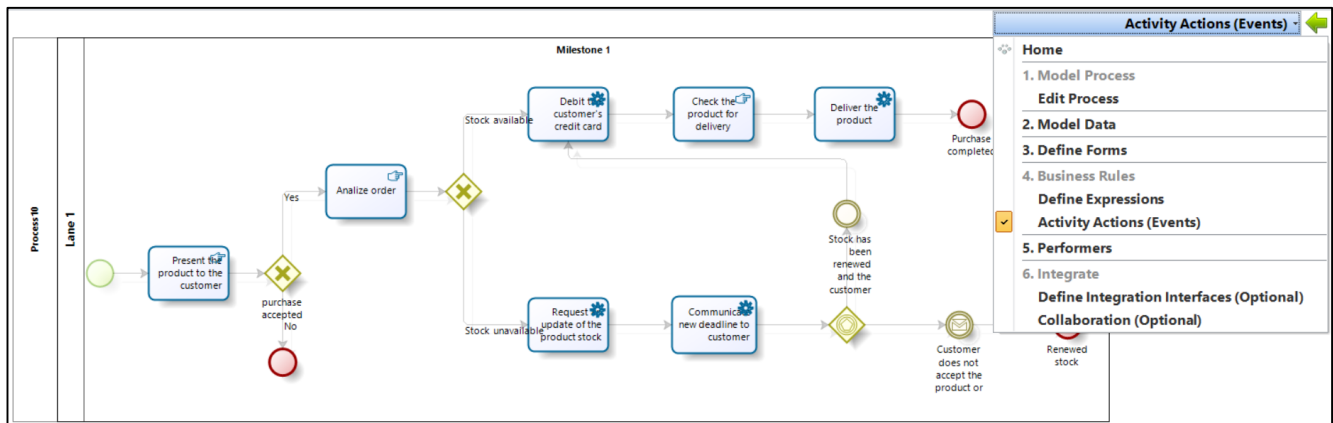
The Data Model is displayed on the left of the window, and on the right is the space to create the condition that will be evaluated.

4. **Drag and drop from the data model the attribute to evaluate** → **Choose the evaluation condition** from the drop-down list → **Select all elements of the condition** to be evaluated → **OK** to save the business rule.

► **Activity Actions (Events)** offers the possibility to perform actions on the activities at a given time (event).

To perform actions (calculations and validations) in the process activities:

1. Select **Activity Actions (Events)** → A new window displays activities for which one can add actions:

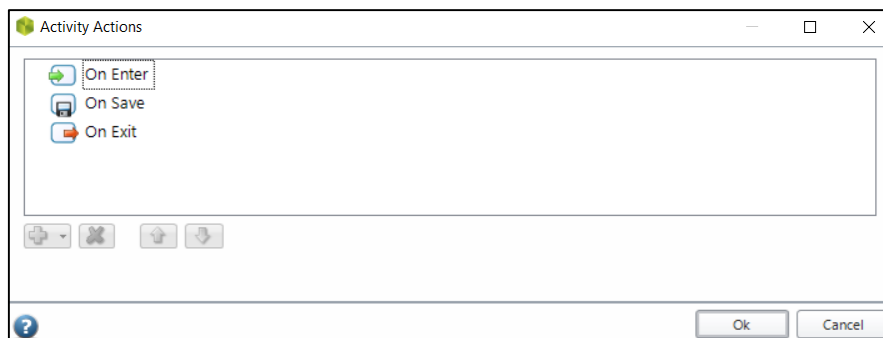


2. To add an action: **click on an activity** → **indicate at what point in the execution of the task that action will be performed:**

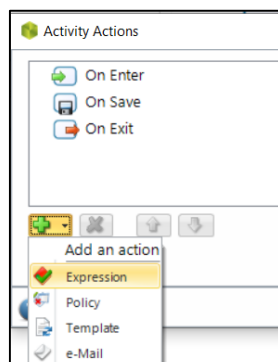
Actions can be created:

- **On Enter:** only once, namely as soon as the activity is created
- **On Save:** every time when the users clicks on Save or when the task is refreshed.
- **On Exit:** only once, namely as soon as the activity ends.

For example, selecting the "Analyze order" activity, the Activity Actions window opens, indicating at what point in the execution of the task the action will be performed:

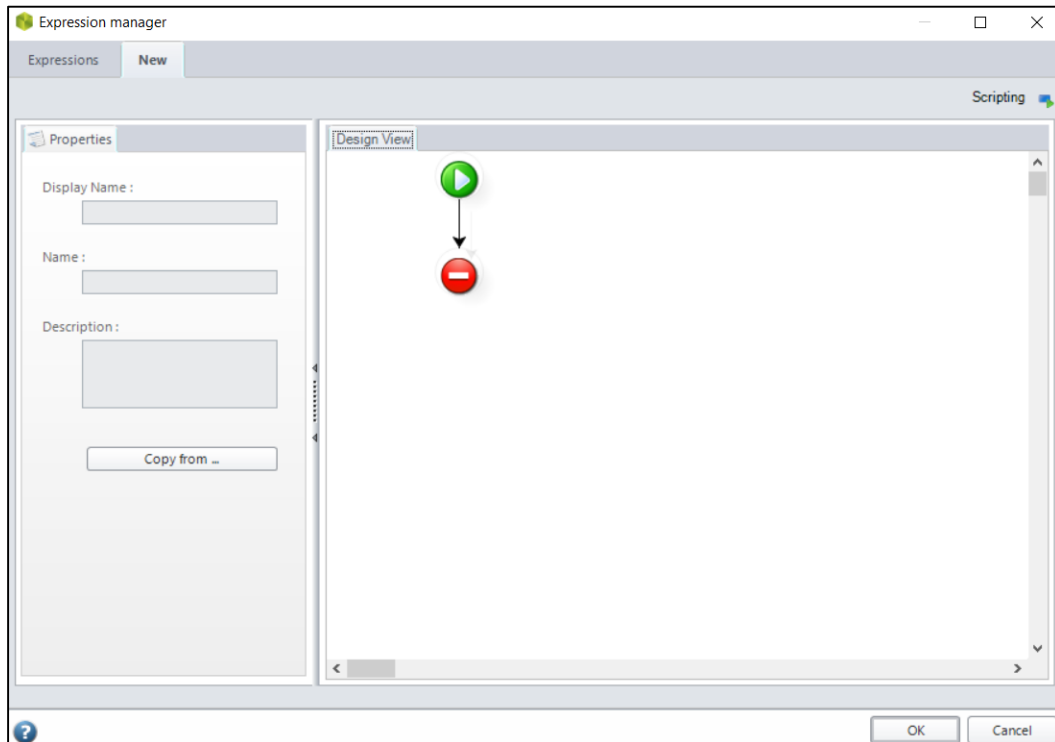


3. Selecting the **option On enter** → **Click on the Plus icon to add an Action** → **select Expression:**



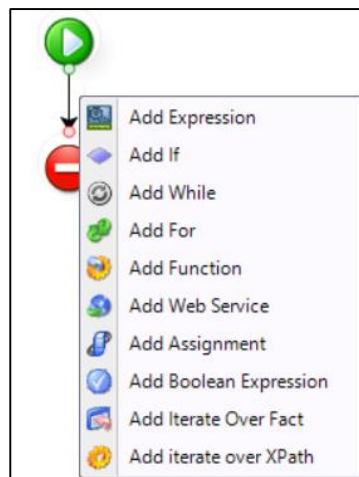
4. In the **Expression manager** window that opens → Click **New** to create a new expression, or select and existing expression → click **Edit** to edit it.

When selecting **New** → the Expression editor will display:

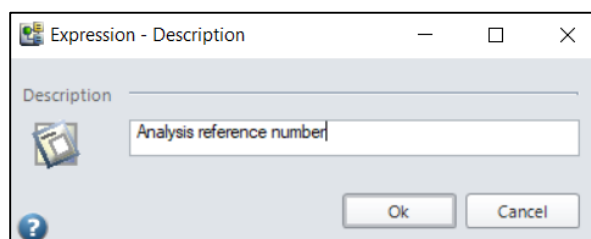


Fields **Display Name**, **Name** and **Description** will be completed, as suggestive as possible. This information will allow the expression to be identified for possible future use.

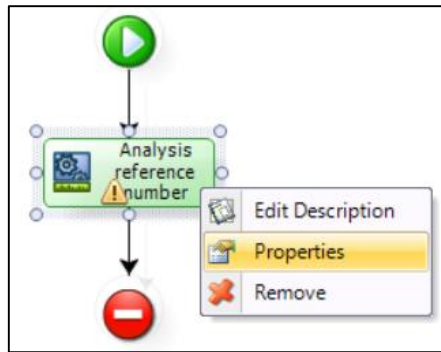
5. **Right-click the black arrow to include an Expression Element:**



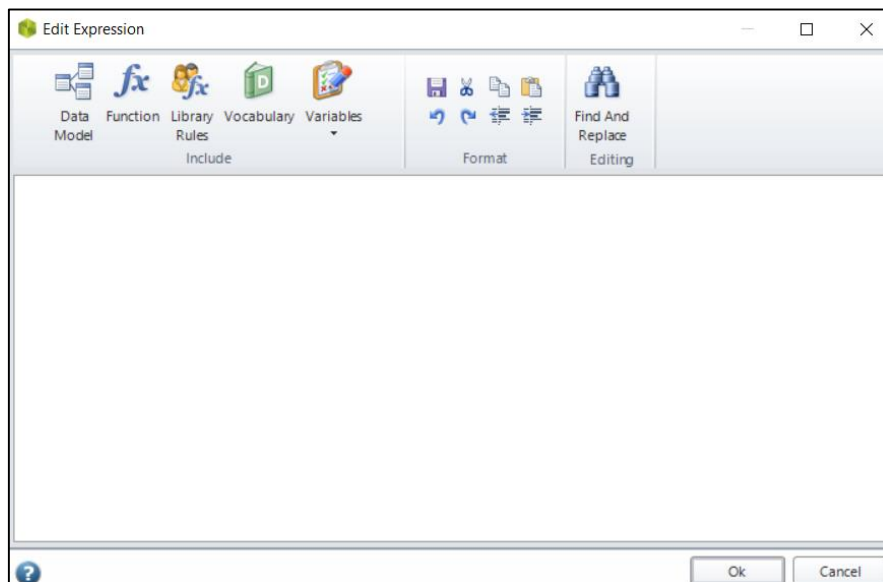
6. For example, select **Add Expression** → type **Analysis reference number** → **OK:**



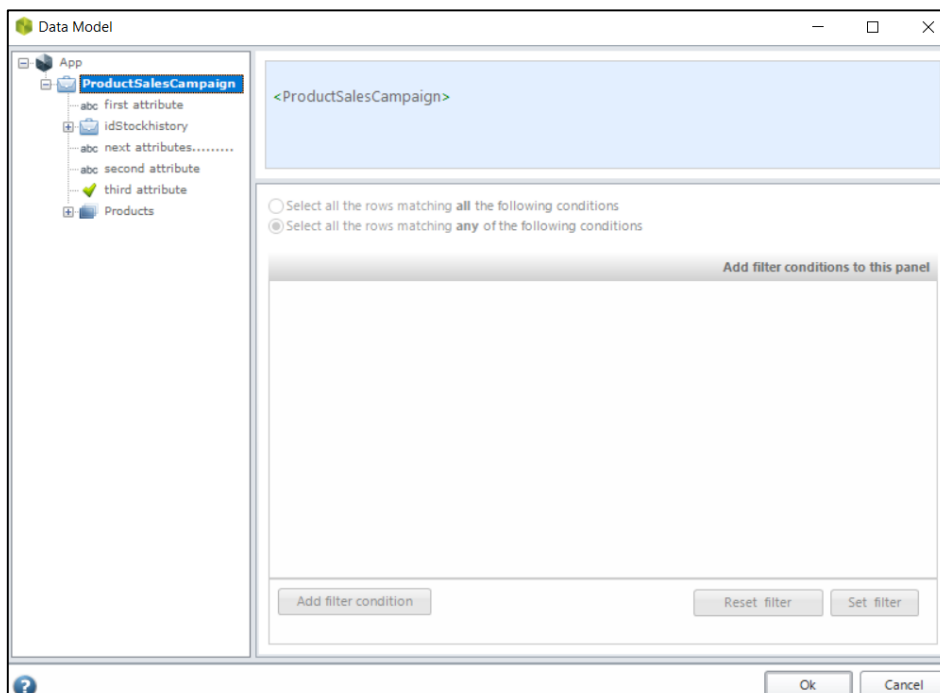
7. Right-click on the module "Analysis reference number" just included → select Properties:



8. In the **Edit Expression** window that opens → Click on **Data Model**:

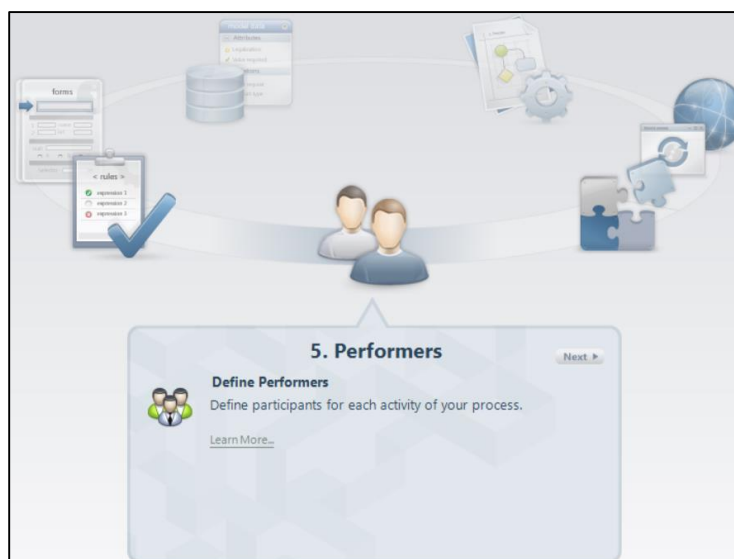


9. In the **Data Model** window that opens → select the attributes and operators to compose the expression → OK:



3.4.4 Defining the participants for each activity

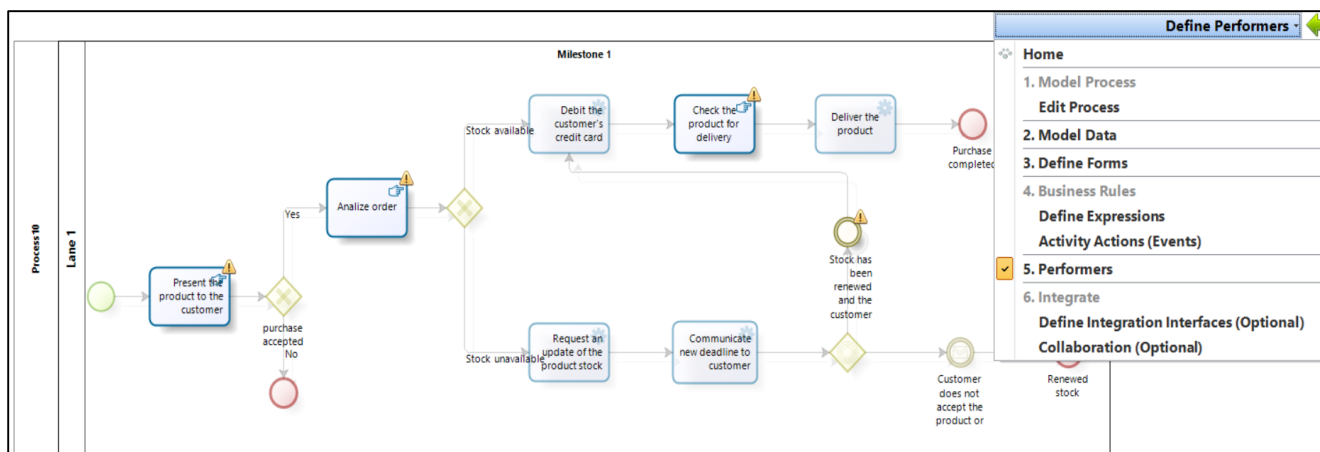
The next step completed by means of the Process automation Wizard is the allocation of users (performers) who meet the requirements and qualities to be assigned to the tasks (the allocation rules defined for each task). They will have access to work on the activity assigned to them.



Note: Before assigning performers, it is necessary that all employees have a user account created. It is important that each user account is correctly configured to ensure that Bizagi performs the assignment properly.

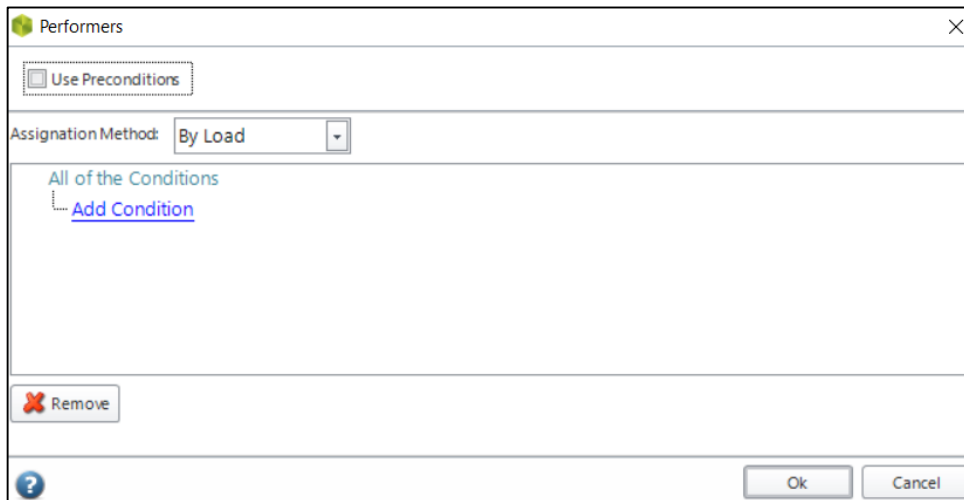
To define the work allocation for each process activity:

1. Select **Define Performers** → Bizagi opens the process flow and tasks that are available for assignment will be highlighted and activities in which participants have not been defined have an exclamation mark:



Note: Available for allocation are only those activities and events that interact with end users.

2. Select an activity or an event to define its performers → the **Performers allocation window** that opens allow to configure the assignment, where three types of conditions can be configured: **assignment rules, assignment method and preconditions:**

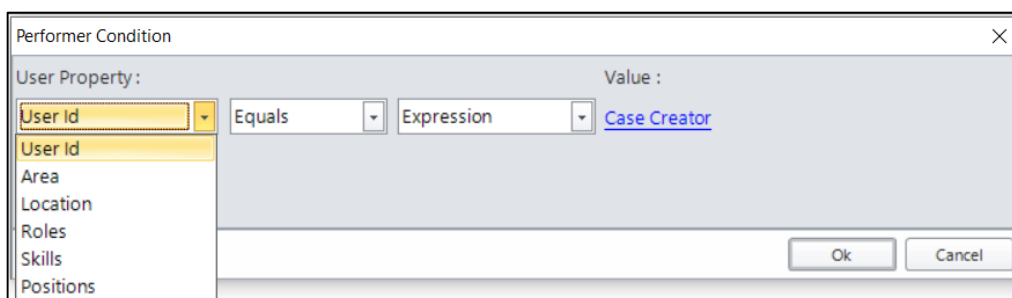


Assignment rules allow to define a set of conditions that must be met when Bizagi performs the assignment.

3. Click on **Add Condition** to **include a condition where to select the properties** that a user must meet in order to be allocated.

The properties that are predefined by default are:

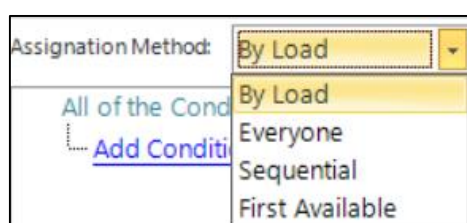
- **User ID:** identification number registered in the data model and that uniquely represents a specific user.
- **Area:** area or department where the user works within an organization.
- **Role:** role carried out by a person in the organization. A user can have one or more roles.
- **Ability:** ability or capability for performing an action.
- **Position:** organizational structure that indicates a user's position in his chain of command.



The comparison operator (**Equals** or **Not equals** to a condition) allows a User Property to be compared to a specified value.

Expression field allows the definition of a business rule to set the value of a condition.

4. **Assignment methods** provide a set of functions for choosing how a task is assigned to a user.



The types of assignment are:

- **By load:** The task is assigned to the user with the lowest workload. First of all, the system checks if any of the users in the user group has already worked on the case before. If so, the respective task is assigned to him, regardless of the user's workload.
- **Everyone:** Assignments are made to all users who meet the recommended characteristics. The first who takes ownership of the case will carry out the task. Therefore, it will no longer be shown to other assignees.
- **Sequential:** Each task is assigned equally and sequentially among users who meet the assignment criteria, without taking into account their workload.
- **First available:** Tasks are assigned to the user who is first available depending on the related time zone. As there may be more than one user available, assignment will be made through a workload assessment of each available user.

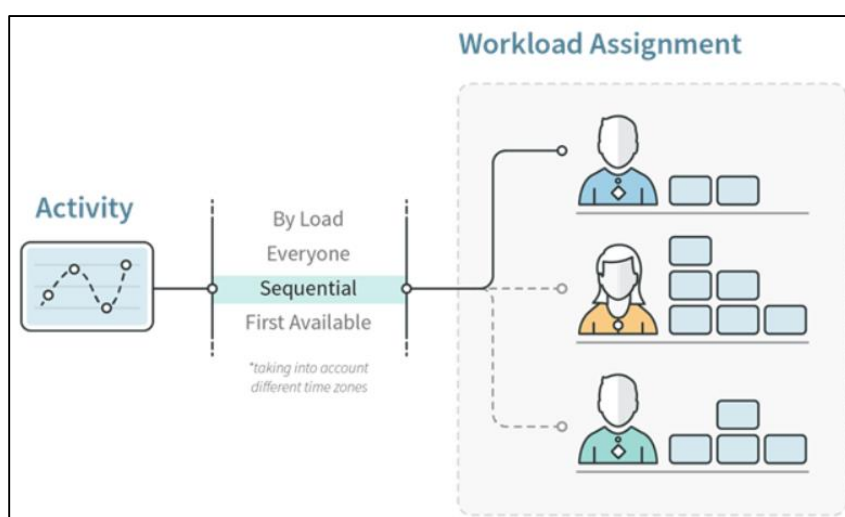


Figure 3.33: How Bizagi allocates a Task to the available users

[https://help.bizagi.com/bpm-suite/en/index.html?work_allocation.htm]

5. **Use Preconditions** allow assignments based on business rules that evaluate a condition and return true or false, indicating whether the condition applies to the defined profile or not. So, a precondition allows a user to set rules in order to decide which assignment rules should be followed.

3.4.5 Integration Stage

In this step of the BIZAGI process wizard, one can configure the existing connections between the process and external systems or entities. In order to cover this integration functionality, Bizagi provides a powerful integration layer that follows a service-oriented architecture.

To be able to integrate external services within Bizagi processes, it is recommended to define activities as service tasks in the first step of the process flow model wizard.

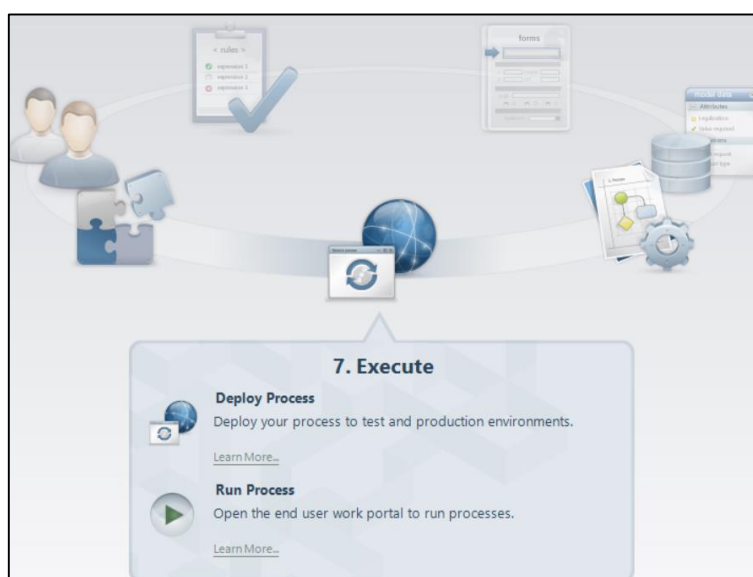


As described in the specifications of the Bizagi application, there are two integration options:

- **Define integration interfaces (Optional):** integrate Bizagi with the rest of your organization and ecosystem
- **Collaboration (Optional):** implement message exchange between processes.

3.4.6 Process execution

Execution is the final stage, which allows processes to be transferred to test and production environments.



Note: The free versions of the components of the Bizagi solution offer access to the basic functionalities of the application, but certain functions require subscription to one of the paid services before being able to use them.

Chapter IV: Business Intelligence architectures and tools

4.1 Architecture of a Business Intelligence solution

In a broader sense, the architecture of a BI platform includes the following main elements, structured on three levels:

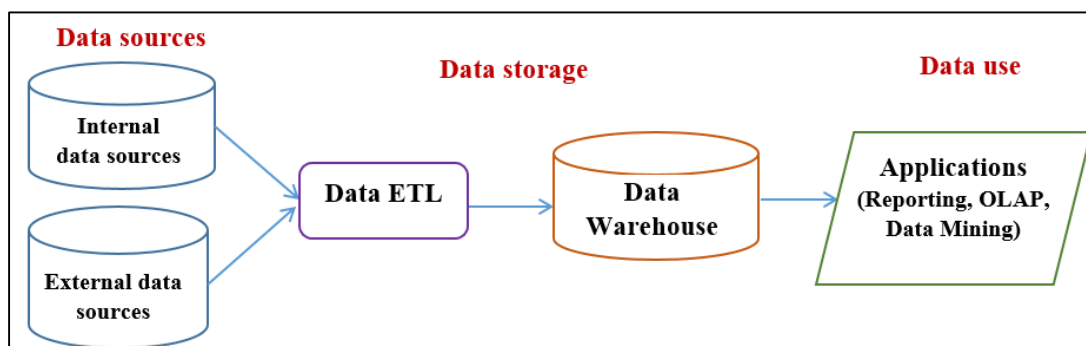


Figure 4.1: The architecture of a BI platform

► **At the first level** are the systems that produce and contain elementary data. Data structures, their representation and their storage (from simple databases to data marts or data warehouses), the principles of organization and control over data (essential inputs for the production of information and knowledge in companies) should be considered a key element in Business Intelligence analysis.

In the first level, it is necessary to collect and integrate the data present in the various primary and secondary sources, heterogeneous in origin and type, which come from both inside and outside the company.

Data sources:

- *Data from operational systems within organizations:* business operations - data about customers, products, contracts, sales, accounting records, etc. These operational systems are called online transactional systems because they process large amounts of transactions in real time.
- *Data from departmental IT systems:* analyses, reports, forecasts, budgets, spreadsheets, etc.

Capturing this information can be done through direct access to the respective database, through dedicated services or APIs, or by extracting data from spreadsheets or text files. This is structured data and generally easier to obtain.

- *Data from various external sources* and which are mostly unstructured, untabulated data. These can come from e-mails, documents or various data from sources external to the company (partners, social networks or financial information sites, agencies and economic institutions, various other organizations). As we are dealing with large volumes of unstructured data, i.e. with textual information without a predefined positioning, the use of these sources requires a specific collection and storage technique.

► The intermediate level is composed of semi-finished data integration and archiving systems (Data Warehouse systems). The Data Warehouse is one of the most important components of the architecture

of a Business Intelligence system. A Data Warehouse is a database that stores historical and current data from various dispersed sources in a way that maximizes its flexibility, ease of access and administration. Storing data of potential interest for decision-making in the company represents, from a technical point of view, the first step for the development of a Business Intelligence solution.

The data acquired at the first level must undergo an extraction, cleaning and transformation process before being loaded into the Data Warehouse (DW), through a process called ETL (extract, transform, load). ETL involves efficient mechanisms for extracting data, standardizing data, cleaning, aggregating and transforming it into a consistent form. These steps are performed automatically by specific software tools.

- **Extract:** is the process of periodically extracting data from various sources, where they are stored in various formats. Because the loading process runs repeatedly, it is necessary to handle errors and evaluate the possible temporary unavailability or non-update of this data.

Data extraction is mainly done with specialized ETL tools, designed to perform this function and which allow visualization of the process and detection of errors during loading.

- **Transform:** it is the most complex stage of the ETL process, through which the data is processed (cleaned and transformed) and placed in a specific format, is validated with the help of business rules and is aggregated in consistent formats for reporting and analysis.

Through cleaning and transformation, we try to improve the quality of data extracted from various sources: through format changes, code replacement, through additions or correction of possible inconsistencies, inaccuracies and deficiencies, and later through data conversions that guarantee their homogeneity, as far as it concerns the integration of different sources.

There are several types of preprogrammed transformation tools, which are configured for each type of task, from simple scripts, made in scripting languages or SQL (Structured Query Language), to advanced data processing techniques.

- **Load:** represents the effective inclusion of integrated and cleaned data in the BI platform, in the destination Data Warehouse or in departmental Data Marts.

In this stage it is validated that the values of the records loaded in the Data Warehouse are consistent with the definitions and formats of the Data Warehouse.

The analysis of the data stored in a Data Warehouse is carried out through OLAP technologies (or other technological alternatives), which offer high computing capacity, queries, planning functions, forecasting and scenario analysis in large volumes of data.

► **The last level** contains systems for accessing data and producing information: techniques for extracting, analyzing and exploring data to identify hidden patterns and correlations and making the results available for use by end users and visualization techniques data - designed to provide information in a transparent and easy to understand form.

Regarding end users, there are generally three levels of users to be served, and for each of them there is an information structure model:

- *Corporate level*: data structures grouped into information that are used at the level of directors and managers.
- *Departmental level*: data structures in the form of reports and detailed views, used by coordinators and analysts who operate day by day.
- *Personal level*: data structures grouped in dynamic spreadsheets, which generate a precise data analysis. They are used by specialists (regardless of hierarchy) looking for new analyzes and relationships at the data level.

The intelligent search of data, the production and analysis of information for support in control and decision-making activities is carried out through reporting tools, data exploitation, through simulations and forecasts or online analytical processing (OLAP). These tools enable direct access to data in the Data Warehouse, enable the display of data through multidimensional views, and enable ad hoc or predefined queries.

4.1.1 Data Warehouse Technology

The continuous increase in the volume of data recorded at the level of a company, the need for the proactive use of information and the need for instant access to any corporate information required the transition from traditional database systems - fragmented databases, to a new type of informational infrastructure, which to provide easy access to the collected, integrated and certified data of the company and a set of tools for consultation, analysis and presentation of information. Apart from the requirement to integrate the data distributed in different database structures, for a global analysis, the need to separate the data by major subjects was imposed.

These desires were found in the Data Warehouse technology, a concept formalized by Bill Inmon in 1990, through the following definition: "a subject-oriented, nonvolatile, integrated, time-variant collection of data in support of management's decisions."

Based on this definition, the main characteristics of this technology can be distinguished:

- **Integrated**: As they are loaded into a data warehouse, the data produced from different sources must be integrated and homogenized into a consistent structure (common codes, common formats or common units of measure), which will eliminate any type of inconsistency that exists between the different representations of data sources.

For example, there are situations where different departments have different rules for what constitutes the same information (elements from different sources have different names, different coding structures, different units of measure, etc.), which means that the data representing the same information to be represented in different ways within the systems used by the organization over time.

- **Subject-oriented**: Data is organized and grouped by business themes and topics of greatest interest to the organization (customers, vendors, products, contracts, prices, regions), which bring together descriptive, qualitative and quantitative values relevant to solving a business problem, which was defined by the decision makers. It thus provides decision makers with an integrated view of the business problem, ignoring all information that is not needed for decision making.

And there are certain advantages:

- allows a multidimensional view of the data;
 - facilitates the evaluation of the company's performance and the detection of sources of inefficiency at the company level.
 - facilitates access and understanding by end users.
- **Non-volatile:** After data has been entered into a Data Warehouse, i.e. after initial and incremental data loading, users cannot modify or update the data. Data can be accessed in read (consult) mode, which ensures a stable, unchanging database for analysis and decision making.
 - **Time-variant:** The data stored in a Data Warehouse is historical data, recorded over a long time horizon: it is stored in relation to certain time units, such as hours, days, weeks, months, quarters or years. Thus, these data can be used to identify patterns, hidden relationships and evaluate the trends of the analyzed phenomenon. They are therefore very useful for data-mining analyses.

When access to all information is not necessary for all users within a company (because they only need certain specialized information), it is necessary to create smaller Data Warehouses, which are decentralized and support the specific needs and requirements of a department or of a division of the company (sales, marketing, human resources, etc.). These departmental Data Warehouses (Data Marts) not only allow a better control of the information in the respective field, but also allow the acceleration of queries, by reducing the volume of data to be used in the data warehouse [Inmon, 2005].

According to Inmon [Inmon, 2005] and Kimball & Ross [Kimball & Ross, 2013], a Data Mart can be a part of a Data Warehouse and directly depends on the Data Warehouse (Data Mart dependent) or a small Data Warehouse that serves a single department and has nothing to do with the rest of the company (independent Data Mart).

4.1.2 The multidimensional data model

"A data model is a collection of conceptual tools for describing data, relationships, semantics, and consistency constraints" [Silberschatz et al., 2011]. In a classic data model - the tabular form of data representation, each row of the data table represents the record of a database object and each column represents its characteristic attribute.

For example, in the tabular representation in Table 4.1 below, the quantities of each type of product transported from several warehouses in each time period are described.

But, in the case of a Data Warehouse, in order for the data from the various sources to be as uniformly structured and aggregated within the warehouse, they follow a multidimensional conceptual design.

A multidimensional data model enables the analysis of information in the Data Warehouse from different perspectives through user-friendly visualization forms. In the multidimensional model, the data is visualized in an n-dimensional space, a typical representation of this structure being the data cube (hypercube). The hypercube is an extension of the notion of a 3-dimensional cube, allowing the modeling and visualization of data in multiple dimensions.

Table 4.1: Number of transported products, corresponding to each combination of City, Product Type and Warehouse

Product Type	Warehouse	Time interval	Quantity
Product I	Warehouse I	Trim I	25
Product I	Warehouse I	Trim II	40
Product II	Warehouse III	Trim I	26
Product IV	Warehouse II	Trim III	29
Product III	Warehouse III	Trim I	43
Product II	Warehouse IV	Trim I	45
Product III	Warehouse II	Trim II	19
Product I	Warehouse I	Trim IV	22
Product II	Warehouse III	Trim I	26
Product IV	Warehouse IV	Trim I	35
.....
Product II	Warehouse III	Trim III	45
Product IV	Warehouse III	Trim I	35
Product II	Warehouse I	Trim I	23

In a hypercube, the axes represent dimensions or categories of information (the identifying attributes of the objects represented in the database). Figure 4.2 represents a data cube, the three dimensions being: time interval, warehouse and product type, the accumulated value stored in each cell of the cube being the quantity of transported products. For example, for that company, the total of product III transported to warehouse II, in the first quarter, was 24 units.

This model was imposed due to the multidimensionality of the data needed to facilitate decision-making. For example, through a query like how many units of a certain product X were transported to the warehouse Y in the time interval Z, data from three dimensions are aggregated: the shipments must be known by product, warehouse and time dimensions. Thus, one can imagine a cube with three dimensions, product, warehouse and time interval, where each dimension has four attributes that describe the respective dimension:

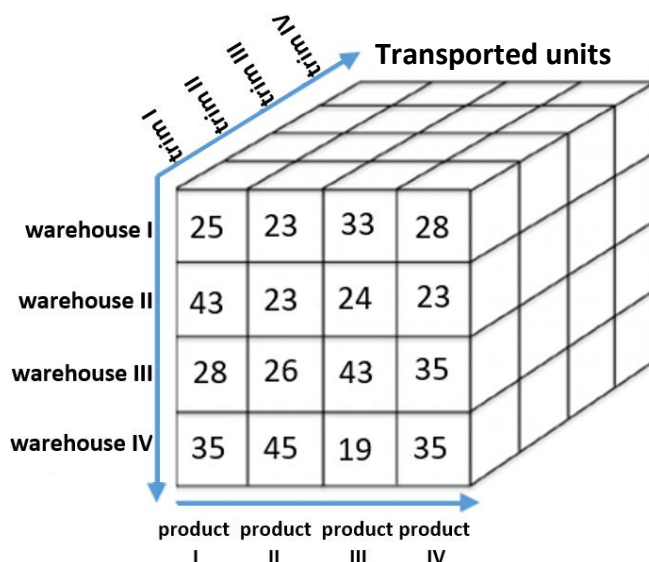


Figure 4.2: A multidimensional cube with three dimensions

The multidimensional model is characterized by three basic concepts: facts, dimensions and measures.

● **Dimension:** A definition formulated by the OLAP council considers dimension to be a "*structural attribute of a cube that consists of a list of members, perceived to be of similar type (e.g. all months, quarters, years form the time dimension)*" [<http://olapcouncil.org/research/resrchly.htm>].

Dimensions describe key aspects of the business, such as products, time, customers, personnel, resources, distribution channels, etc.

Visualizing data in such a cube has the advantage that it can be represented in any size.

Depending on the selection made, the data selection can represent:

- *a subcube:* when performing a selection on two or more dimensions.
- *a page:* if all but two dimensions have a single member selected, the other two dimensions define a spreadsheet.
- *a single cell:* if all dimensions have only one member selected.

● **Facts:** represent the information that will be analyzed; they are analytical elements from the database, measurable elements - numerical data that can be summed up and analyzed on different levels. Facts allow the representation of events and business elements, such as a commercial transaction, or any event that can be used in the analysis of business processes.

Facts are inserted into what is called a fact table and can be presented in simple or aggregated form.

Facts are implicitly defined by combining dimension values. Thus, the facts can be:

- *Additives:* When the values can be summed according to all dimensions.
- *Semi-additive:* When the sum can be performed in relation to some dimensions or exclusively with one dimension; minimum, maximum and average values can be calculated.
- *Non-additive:* When the value cannot be summed in any dimension or cannot produce any valid meaningless value.

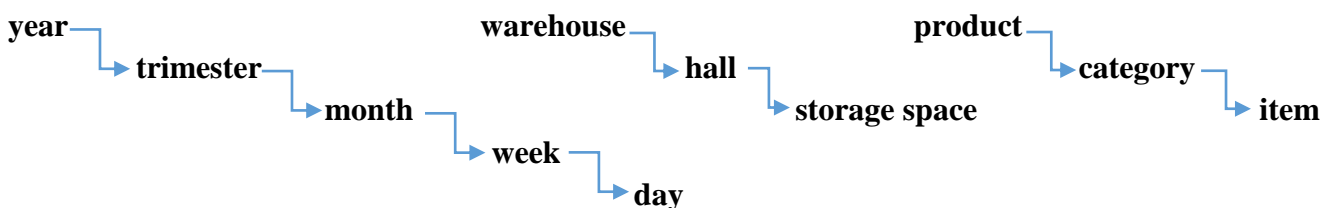
A collection of data for which there is a tabular representation can generate a multidimensional representation by following the steps:

- first of all, the categorical (qualitative) attributes that will represent the dimensions of the multidimensional representation must be identified;
- the attribute associated with the purpose of the analysis must be identified in order to generate aggregated data (total or average quantities);
- each row of the table will be mapped into a cell of the multidimensional representation, where the value of the cell is the value of the target attribute;
- the cell indices are specified by the values of the selected attributes as dimensions.

● **Measures:** Relevant information about facts is represented by a set of indicators (measures or attributes). Thus, a fact can be characterized by qualitative or quantitative measures (for example the

price of a product, income, number of articles, number of complaints, turnover, etc.) that quantify the fact and allow the comparison of different facts.

Another defining element of a Data Warehouse is the level of granularity: the level of detail proposed for the design of the Data Warehouse. For example, for each of the previously exemplified dimensions, time interval, warehouse and product type, a hierarchical structure can be defined:



Further analysis can generate aggregates applied to all data related to each dimension.

4.1.3 Solutions for analyzing data stored in a Data Warehouse

Data Warehouses provide access to data for complex analysis and respond to a company's requirements for efficient and relevant information through various types of applications.

Today there are various technologies that allow the processing and visualization of information stored in a Data Warehouse, but the most widespread are OLAP technologies (Online Analytical Processing) or data mining applications.

OLAP is a software technology used in the field of Business Intelligence that offers data analysts a multidimensional analysis of information, allowing the query of large amounts of data and offering various types of visualizations of information generated from a company's records. This is the usual way that decision makers apply to analyze information because business models are normally multidimensional.

Used for management business reports, for marketing, sales reports, etc., OLAP technology generates an aggregate of synthesized and summarized data, covering a period of time large enough to be able to detect time-related interesting patterns and insights, to detect relationships between different variables or factors or some other useful knowledge.

OLAP tools allow queries, selecting different attributes from the multidimensional schema; apart from simple aggregations, such as sum, average, minimum, maximum, etc., it offers the possibility of performing complex analyses, based on complex algorithms. Users can perform these analyzes at the highest level of aggregation or at the highest level of detail.

Regarding the functionalities that an OLAP tool must offer, there is an analysis implemented in the form of 12 requirements, formulated by Edgar Frank Codd [Codd et al., 1993], a British computer scientist, pioneer of the relational database model:

- *Multidimensional view*: users can manipulate multidimensional data models easily and intuitively through operations in which levels and hierarchies can be chosen. Thus, by defining the degree of detail

of the dimensions, they allow in-depth data analysis (for example, from the time interval dimension you can choose trimester, month, week or day).

- *Transparency*: OLAP tools should be integrated in the context of an open systems architecture and should support different data sources (homogeneous or heterogeneous database environments); also, the user need not have knowledge or care about the specific details of data access or transformations.

- *Accessibility*: OLAP tools must serve as an access interface to query data, regardless of its origin and structure, serving as an intermediate step for obtaining information. The OLAP system should access only those data that are necessary to perform the respective analysis and perform the necessary conversions to generate a consistent and meaningful visualization for the user.

- *Performance in reporting*: to maintain ease of use, when the size of the database increases or the number of dimensions included in the analysis increases, the performance of reporting should not be altered.

- *Client/server architecture*: OLAP products should be able to work in a client-server environment and should provide maximum performance regardless of the number of clients connecting to the server, which should be able to integrate databases from different sources .

- *Generic dimensionality*: OLAP tools must allow flexible definition of dimensions without imposing restrictions on the number of dimensions. Moreover, for each dimension of data, the same operations, formulas and reporting formats can be applied.

- *Dynamic manipulation of sparse matrices*: the physical schema of OLAP tools must optimally handle sparse matrices, i.e. identify null or empty values.

An important problem of OLAP technology refers to the waste of memory depending on the so-called "data sparsity", terminology that means the percentage of cells containing null values, given that they correspond to events that never occurred. For example, a hypercube with 20% sparsity means that 20% of its cells will contain no values. The problem of data sparsity can be partially solved by dividing the n-dimensional cube into "chunks" of multidimensional sub-cubes.

- *Multi-user support*: it must allow simultaneous access for several users (in terms of integrity and security), who can apply operations to individual cubes or views of the same database.

- *Unrestricted Cross-Dimensional Operations*: dimensions are by definition created equal, so computing and manipulating data must be possible for all dimensions; also, regardless of the number of data attributes a cell contains, relationships between data cells should not be restricted.

- *Intuitive Data Manipulation*: for all operations and data manipulations, users should be able to use simple functions, such as drag and drop functions, instead of complex operations or menus.

- *Flexible Reporting*: OLAP tools must have the possibility of reporting information both in full (up to the maximum number of dimensions), but also synthesized, as the user wants or exclusively the information he needs, through customization options.

- *Unlimited Dimensions and Aggregation Levels*: a powerful OLAP tool should be able to support at least fifteen and preferably twenty dimensions of data in a common analytical model. Additionally, these generic dimensions should allow for a virtually unlimited number of user-defined aggregation levels within a specific consolidation path.

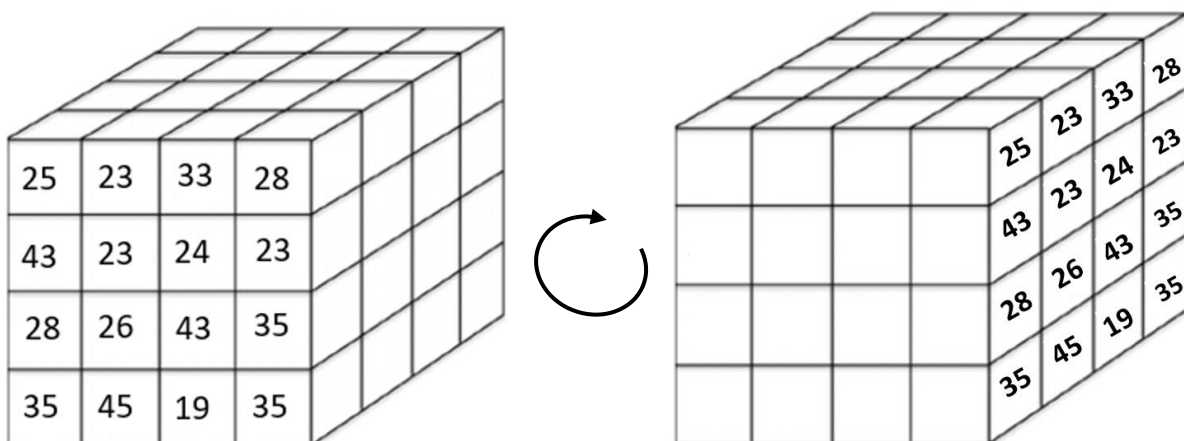
Later, following Nigel Pendse's report [<http://www.olapreport.com/FASMI.HTM>], the OLAP Council adopted a more precise alternative definition of the essential properties of the OLAP concept, the Fast Analysis of Shared Multidimensional Information (FASMI) test :

- *Fast*: this requirement refers to the fact that the system must be able to respond quickly and nimbly to queries launched by a user, who must not wait more than about five seconds to solve simple requests and no more than twenty seconds in complex requests.
- *Analysis*: it refers to the fact that the system must be able to support any business logic in order to answer the specific questions and needs of the business: relevant and necessary analyzes for various categories of end users.
- *Shared*: aspect that refers to the fact that the system must offer tools to guarantee data confidentiality and multiple access security, at an adequate level.
- *Multidimensional*: it is perhaps the most important requirement and refers to the fact that the OLAP tool must offer a multidimensional conceptual view of the data.
- *Information*: the system must be able to manage all relevant information and derived information, which are important for the analysis performed.

OLAP tools have manipulation operators that perform operations such as drill-down, roll-up, slice-and-dice, pivot, rotation, etc. These operations are intended to offer analysts various perspectives on the data, starting from a higher degree of detail, to a lower degree of detail (through aggregation) or by extracting a certain subset of data.

Also, OLAP tools have pre-calculated operations (technique called consolidation), for example totals or other operators that calculate various statistics.

■ **Rotation (pivot)**: is an operation that rotates the data axes to provide an alternative presentation of the data.

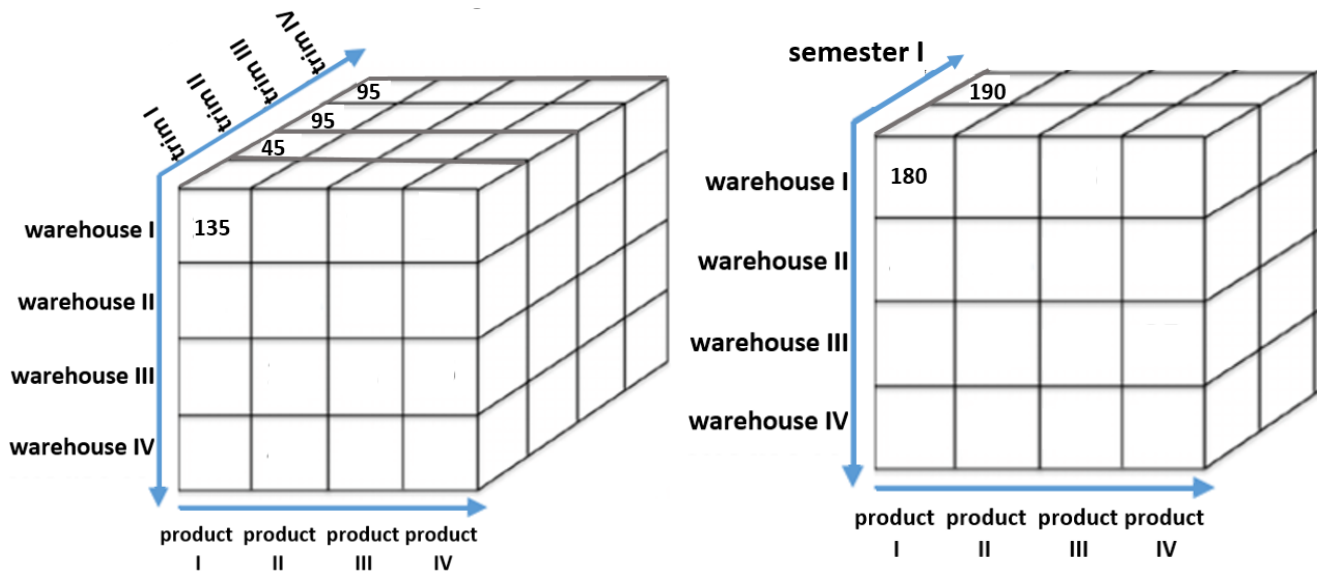


For an n-dimensional cube there is $P_n = n! = 1 \cdot 2 \cdot \dots \cdot n$ rotation possibilities.

■ **Roll-up**: involves aggregating data in one of the following ways:

- by "climbing the hierarchy" on a certain dimension
- by eliminating one or more dimensions of the multidimensional cube

For example, in the roll-up operation in the aggregation process below, the location hierarchy moves up from trimester to semester:

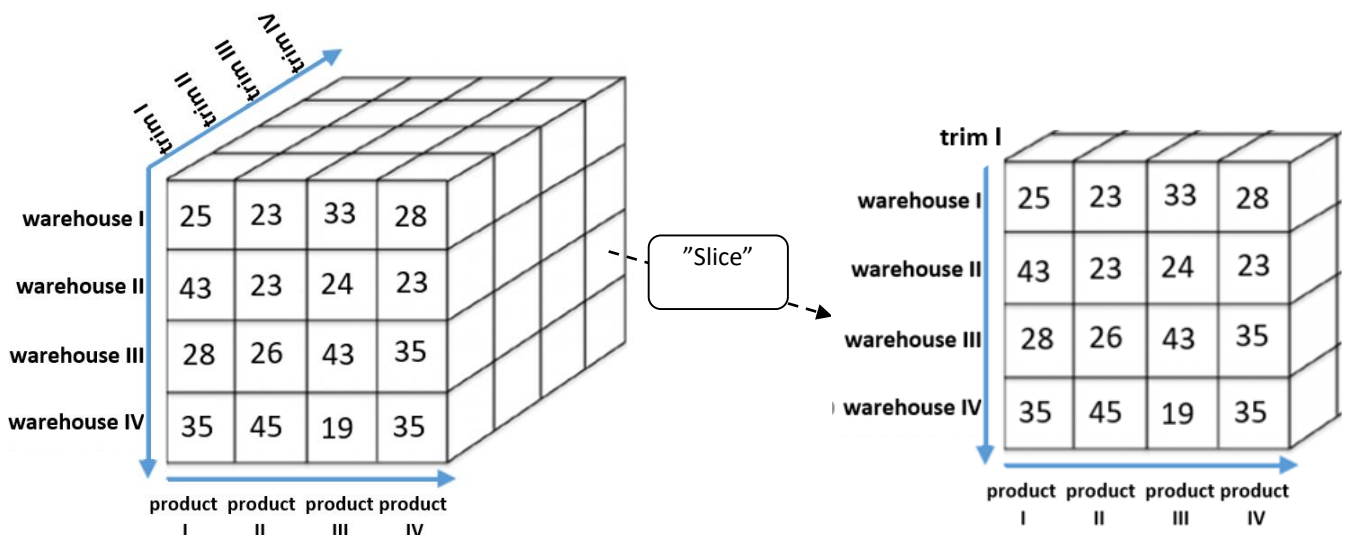


■ **Drill-down:** is an operation opposite to the Roll-up operation, which is carried out in one of the following ways:

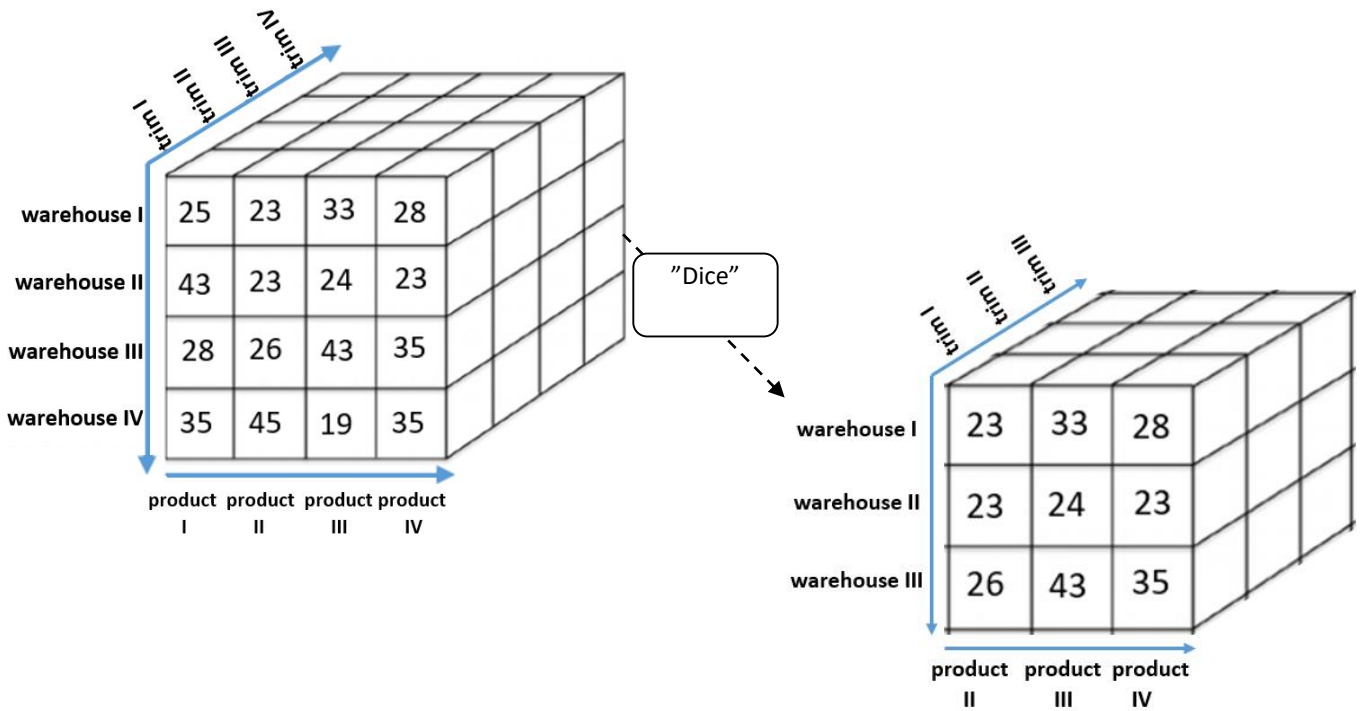
- "descending in the hierarchy" on a certain dimension;
- by adding a new dimension.

In the example above, it would mean obtaining more detailed data, for example going down in the hierarchy from "product" to "item".

■ **Slice:** Selecting and projecting a single dimension of the cuboid, thus creating a subcube of the original one. This reduces the amount of data.



■ **Dice:** is a similar operation to Slice, where it selects multiple values from one or more dimensions, forming a smaller sub-cube.



4.1.4 Implementation architectures for OLAP techniques

There are mainly three ways of implementing an OLAP analysis tool, with different architectures. Accordingly, OLAP tools are classified into three broad categories: MOLAP, ROLAP and HOLAP; Other OLAP tools have also emerged, such as DOLAP (desktop OLAP systems).

■ **MOLAP** (Multidimensional On-Line Analytical Processing): are the models in which the physical organization of data is carried out in multidimensional structures, having specific storage structures, tools for handling dimensional data and data compression techniques that improve the Data Warehouse performance: once the data is loaded, a series of precalculations and aggregations are performed at all levels of the dimension hierarchy, and indexes and algorithms are generated to improve query access time.

Generally, MOLAP systems are used for small Data Warehouses and whose multidimensional model does not change much.

■ **ROLAP** (Relational On-Line Analytical Processing): is an OLAP version that is based on the structures of classical relational models: the physical organization is implemented on relational technology, the data being organized in relational databases.

Being in fact a system that uses a relational database with specific adaptations for OLAP, the ROLAP architecture has both the advantage that it can be easily integrated into another existing relational database system, but also the fact that relational data can be stored more efficient than multidimensional data.

■ **HOLAP** (Hybrid On-Line Analytical Processing): OLAPs that have a hybrid structure that combines ROLAP and MOLAP technologies (relational databases and multidimensional databases), relational structures being usually used for data with greater granularity, and multidimensional structures being dedicated to storing aggregates (lower granularity).

■ **DOLAP** (Desktop On-Line Analytical Processing): it is oriented so that the user, who triggers an SQL statement to the database server, receives the microcube of information to be analyzed on his own workstation, from where he will perform the queries he needs. Although the functionalities and analysis possibilities are more limited than in the case of the other solutions, the advantage is that the database server is not overloaded and there is less traffic in the network, so the information traffic between the client and server environment is minimized.

4.2 Business Intelligence tools

In recent years, thanks to the spread of the Internet and mobile devices that circulate information, there is an increasing influx of data in companies, which has generated an increased demand for Business Intelligence solutions. These products have developed a lot and are provided both in proprietary code and as free software (open source). At the moment, there are a large number of technologies and tools that allow the construction of Business Intelligence solutions, provided as software solutions, and their choice depends mainly on the needs of the companies that want to implement them. If in the past, only large companies with sophisticated applications, which needed the assistance of experts in this sector, were targeted by Business Intelligence technologies, today, to the same extent, the implementation of Business Intelligence solutions is frequently encountered in small and medium-sized enterprises and organizations. This technology allows the creation of dynamic reports based on a variety of requirements according to the needs of managers, and the corresponding software solutions are available at reasonable prices, which allow any organization to use its data for decision-making.

Among the most well-known proprietary software providers are Oracle, SAP, IBM, SAS, Microsoft, Qlik, MicroStrategy, Clear Analytics, Tableau, which for years have been offering solutions to large, small and medium-sized companies. And among the providers of open software solutions (free licenses) are Pentaho, CloverETL, ClicData, Jaspersoft, Birt, Helical Insight, Jedox, KNIME, Zoho Analytics.

The Business Intelligence solutions offered can be classified according to the useful tools that provide technological support for the implementation of this process: data management tools (Data Warehousing), study applications, analysis and extraction of information from data (Data Analytics) and visualization tools, interpretation and reporting (Data Visualization). The rapid and constant evolution of technology has allowed companies dedicated to the development of Business Intelligence solutions to include the three types of solutions in a single tool. This integrative vision is motivated by a complex analysis, which shows that without the Data Warehouse storage infrastructure, data cannot be captured and stored effectively; without advanced data management tools, data analysis techniques cannot be used, and without information distribution to decision makers, analyzes are not relevant [Rikhardsson & Yigitbasioglu, 2018].

An analysis between free software and private use is presented in the gnu.org study, where some of these aspects are explained.

Free Software

Free software must allow its users to run, copy, distribute, study, change, and improve the software, aspects that are summarized in the following four basic freedoms [<https://www.gnu.org/philosophy/po/free-sw.ml-en.html#four-freedoms>]:

- *Freedom 0*: freedom to run the program for any purpose.
- *Freedom 1*: the freedom to have access to the source code of the software application to study the program code and modify it to perform other functions or remove those that are not needed.
- *Freedom 2*: the freedom to distribute the program to help other users.
- *Freedom 3*: Freedom to distribute copies of modified versions to third parties.

To the extent that a software application offers users all these freedoms, then it can be appreciated that it represents free software.

■ Advantages of free software:

- Allows saving or even complete elimination of the costs related to the purchase of licenses; a free software generates a community that supports the development of the software, which leads to a decrease in the cost of production, ultimately having a positive impact on the user.
- It is accessible (a computer is needed to access the software) and available for all operating systems.
- It is an effective way to combat illegal software copying, but at the same time it prevents proprietary software sellers from inspecting the contents of the user's hard drive without warning (according to the concluded license agreement).
- It promotes technological growth and generates collaborative technological innovation, but also social solidarity: through sharing and cooperation it allows the provision of social and technological benefits to communities.
- The programs are made available to the public (users and programmers) for testing, detecting and correcting faults.
- In the case of free software, security bugs can be identified more easily, and security vulnerabilities can be reduced, unlike a proprietary software, where only the company that owns the software can fix these aspects.

■ Disadvantages of free software:

- There is no clear definition of the guarantee, the software being free does not have a guarantee from the author, it is used at your own risk
- There is no single company that supports the technology.
- The variety of versions and user licenses can cause confusion for users.
- May have less hardware compatibility.

Proprietary software

Proprietary software is a computer program in which the programmer, the company, the corporation or the foundation that develops it has all the rights to it (the source code is not available or access to it is restricted), thereby causing users or the general public to have limitations when it is about use, modification or redistribution (with or without modifications).

Unlike free software, which can be used and modified openly, in proprietary software modifications are prohibited for copyright reasons; the software remains unchanged until the developers or owners make updates, new versions or changes.

■ Advantages of proprietary software:

- Proprietary companies usually have quality control departments that perform several tests before releasing the software and thus can provide users with a good user experience and a high level of quality in terms of customer support or services.
- Being a widely used software, there are capable and experienced people in its use, so that if needed, it is relatively easy to find someone who knows how to use it.
- In general, developer companies conclude user agreements with universities (with free or significant discounts in the purchase of licenses), so that students and graduates are familiar with the use of that software.
- Developer companies have the budget for research and to hire the best professionals - capable and experienced programmers for software development and there are numerous documentations, tutorials, specialized literature for understanding the use of the software.

■ Disadvantages of proprietary software:

- This type of program incorporates a number of restrictions on its free use: users cannot modify, copy and distribute the software; in addition, it must be specified the users who can access it, the number of computers on which it can be installed (since there are licenses that are limited to a certain number of users), the operating system (since there may not be versions for all operating systems) or available options, etc.
- The cost of obtaining a license is not negligible, especially since it must be constantly updated, which certain institutions cannot cover.
- If the owner company goes bankrupt or is taken over by another company, the software situation is uncertain, its development may stop, new versions may no longer appear, technical support may disappear, etc.

When a company evaluates the opportunity to invest in a Business Intelligence system, it must take into account, along with the financial aspects, several aspects related to the functionalities and limitations of the targeted applications, the experience of potential suppliers in previous projects, the long-term support offered by the company proprietary for upgrades or external studies and references, such as the study of the multinational technology consulting company Gartner, which is a prestigious company founded in 1979.

Magic Quadrants, designed and developed by Gartner, represent one of the best tools available for companies of any kind to evaluate software solutions suitable for their respective fields of activity. The Gartner Magic Quadrant for Business Intelligence and Analytics Platforms (Business Intelligence & Analytics) is a tool for evaluating providers of solutions in these areas, based on a graphical representation, which analyzes and ranks a large number of providers, over a certain period, using a set of criteria.

The report ranks proprietary and free software solutions providers according to the progress they show in their respective proposals and developed products and examines their strengths and weaknesses. This classification is carried out annually, following a study that indicates the progressive evolution of Business Intelligence tools, which are increasingly easier to use and include a greater number of functionalities. Gartner's analysis is useful for companies that want to implement new Business Intelligence and Analytics projects, through modern platforms, to take advantage of the dominant innovation in the market.



Figure 4.3: Gartner Magic Quadrant for Analytics and Business Intelligence Platforms [https://www.gartner.com/doc/reprints?id=1-2D773G95&ct=230411&st=sb]

The Gartner quadrant is represented in a scatter plot, in which the analyzed companies are divided along two axes:

► **The horizontal axis** represents the company's view of the technology market. To determine the positioning on this axis, eight criteria are used, which include:

- **market understanding:** the company's ability to listen and understand customer needs and use this information to provide appropriate solutions;
- **marketing strategy:** the effectiveness of the company's communication efforts, publicized both offline and online;
- **sales strategy:** the practices adopted by the company to develop a consumer base for the sale of products or services;
- **offering (product) strategy:** the approaches and techniques used by the company to enable the distribution and development of a product and the provision of services;
- **business model:** the effectiveness and logic of a supplier's business proposition to dominate a market share;
- **vertical/industry strategy:** the ability to direct resources, skills and offerings to meet the needs of specific market niches;
- **innovation:** the level of resources, investments and expertise invested in solutions that allow the creation of competitive advantages;
- **geographical strategic:** the ability of a supplier to meet the needs of various geographical regions important for the market, directly or through partners.

► **The vertical axis** indicates the company's ability to implement what it plans. For the positioning of each element in the graph, seven more criteria are used, among which:

- **products/services:** the quality, functionalities and level of differentiation (characteristics and skills) of the mix of products, goods and basic services offered by the supplier;
- **overall viability:** the relationship between the costs involved in operations and the overall financial sustainability presented by the company to assess the likelihood of continued investment in the product.
- **sales execution/pricing:** the company's ability to find effective strategies for negotiating and managing transactions, for pricing efficiency and sales channel efficiency;
- **market responsiveness and track record:** the company's ability to adapt to different opportunities identified in a dynamic market to achieve competitive success;
- **marketing execution:** the quality, creativity and effectiveness of the company in communicating its marketing message (advertising, promotions, discussions, etc.) to the consumer;
- **consumer experience:** the level of consumer satisfaction in relation to the service provided (customer assistance programs, technical support, etc.);

• **operations:** the company's ability (expressed in skills, experiences, programs, etc.) to achieve its goals and objectives.

The companies analyzed are divided into quadrants, according to the strength of each company within specific research aspects.

The four quadrants are made up of:

<ul style="list-style-type: none"> • Leaders 	<p>Companies that lead the market segment, having the highest score resulting from the combination of their ability to execute and vision (potential). With better market vision and more advanced technology, these companies demonstrate a solid understanding of product capabilities and are dedicated to their customers' success, executing and developing their projects with authority; the developed products allow complex analyzes to be performed by people with limited technical expertise and without the direct involvement of IT or technical experts.</p>
<ul style="list-style-type: none"> • Challengers 	<p>They rank below the market leaders in terms of effective marketing, sales channels, geographic presence, industry-specific content and innovation, but are well positioned to succeed in this market. They have a good ability to execute large projects and have a good share of customers in the analyzed market.</p>
<ul style="list-style-type: none"> • Visionaries 	<p>Companies focused on research and development, being thought leaders and innovators. They have a broad vision to provide a modern Business Intelligence platform, but they don't have much technological power and may have gaps when it comes to meeting broader functionality requirements.</p>
<ul style="list-style-type: none"> • Niche competitors 	<p>Companies that meet the specific requirements of the market, do well in a certain market segment, but do not have authority in certain projects. They do not have the scale to strengthen their market positions and have limited ability to outperform other suppliers in terms of innovation or performance.</p>

Based on the indices achieved against the two important approaches, completeness of vision and software execution capability, this current report, from 2023, highlights that the undisputed leader was Microsoft, closely followed by Tableau and then Qlik. Among the visionary companies are ThoughtSpot, SAP, Oracle and TIBCO Software, and the most important niche companies (specialized in a certain sector) were Zoho, Incorta and GoodData. As challenger companies, the report highlights Google, Domo, Amazon Web Services, MicroStrategy and Alibaba Cloud.

Based on the analysis of the EDUCBA sites [<https://www.educba.com/power-bi-vs-tableau-vs-qlik/>], selecthub [<https://www.selecthub.com/business-intelligence/tableau-vs-qlikview-vs-microsoft-power-bi/#4>] and the analysis of TechTarget's expert, Craig Stedman [Stedman, 2019], results in a comparative analysis of the three most important Business Intelligence tools, Microsoft Power BI, Qlik Sense and Tableau Desktop, based on the following features: visual capabilities, user interface, ease of learning, advanced analysis capabilities, price, availability in the cloud, cloud storage limit and memory requirements for installation:

	Microsoft Power BI	Qlik Sense	Tableau Desktop
Performance	<p>It lacks behind on data visualizations.</p> <p>The platform has a deficiency in presenting data visually.</p> <p>Has limited customization compared to Tableau.</p>	<p>Has good visualizations and takes all types of datasets.</p> <p>Needs a developer to work with reports and dashboards.</p>	<p>Is more user-friendly because non-technical users can work with this tool.</p> <p>The cubic technique employed by this tool makes it easily accessible to non-technical users, rendering it more user-friendly.</p>
User interface	<p>The user interface is fairly intuitive for users familiar with Excel, and allows integration with other Microsoft products.</p>	<p>The user interface is clean and intuitive.</p>	<p>Good user interface.</p> <p>Is well-organized, intuitive, easy to use and allows customizing.</p>
Ease of learning	<p>User-friendly: knowledge of Excel is enough.</p>	<p>They do not require any technical or programming skills to work with.</p>	<p>Easy to learn with Data science background.</p>
Advanced analysis capabilities	<p>Supports visualizations based on R programming language, including decision trees and forecasting.</p> <p>It uses EM algorithms and K-means for clustering analysis.</p> <p>It supports simple linear regression and predictive modeling.</p>	<p>Predictive analytics, bi-variate linear analysis, clustering and regressions are only possible through API connections with third-party software.</p>	<p>Fully integrated support of R, MATLAB and Python programming languages.</p> <p>It supports regression analysis using these languages.</p> <p>It provides built-in date/time functions for comparisons like year-over-year growth and moving averages.</p>
Cloud availability	<p>First software offered in the cloud through Microsoft's Azure platform.</p> <p>Desktop option available. Cloud accounts are required to share views.</p>	<p>Offers fully managed SaaS cloud product.</p> <p>Most clients choose to run the server version.</p>	<p>Can be deployed in the cloud managed by Tableau or on third-party platforms, including Amazon Web Services and Microsoft Azure</p>
Price	<ul style="list-style-type: none"> • Free trial version for 90 days. • Desktop version US \$9.99 per month per user. 	<ul style="list-style-type: none"> • Qlik Sense Desktop: Unlimited free version with some limited app features. • Enterprise: US\$1,500 per token (1 token buys unlimited use for one user or 10 temporary login passes). • Cloud: US\$20 per month per user, US\$25 per user for multiple logins. 	<p>Free trial version for 14 days</p> <ul style="list-style-type: none"> • Tableau Desktop Personal Edition and Server US\$ 35/month • Tableau Online: US\$42/month. • Tableau Desktop (Professional Edition) US \$70/month.
Cloud storage limit/ memory for installation	<p>10 GB cloud storage for data.</p> <p>May require additional costs to scale data capacity in the cloud.</p>	<p>500 GB of cloud storage per workgroup.</p> <p>Requires a minimum of 5 GB internal memory for installation.</p>	<p>100 GB of cloud data storage.</p> <p>2 GB minimum free disk space for download.</p>

Figure 4.4: Difference Between Power BI vs Tableau vs Qlik

4.3 Tools for implementing a Data Warehouse

The software market for these applications includes 2 market segments:

1. ETL Data Warehouse applications: These applications are used in the process of designing, cleaning, transforming, loading and managing data in the Data Warehouse. Among the most important ETL applications are: Informática PowerCenter, SAS ETL Studio, IBM Websphere DataStage (considered TOP tools on the current market, due to superior functionalities and average cost per license.), Oracle Warehouse Builder, Microsoft IntegrationServices, IBM Cognos Data Manager (considered mid-level tools) or Pentaho Data Integration (PDI), Jasper ETL, Palo ETL or Bee (open source tools).

2. Data Warehouse management applications: These applications are used to manage data in the Data Warehouse. Among the most important such applications are: Microsoft SQL Server (Microsoft Corporation), Oracle Express (Oracle Corporation), PostgreSQL (PostgreSQL Global Development Group) or Qlik Data Integration (QDI).

4.3.1 ETL solutions with Pentaho Data Integration

Pentaho includes a web server platform and various customizable and intuitive Business Intelligence tools to support reporting, analysis, graphing, data integration and data mining. Pentaho has two versions on the market; Pentaho Enterprise, which has a paid license, and Pentaho Community, which is an open source Business Intelligence tool with multiple features: it incorporates a large number of graphs, various ways of interpreting data and it also has different APIs to embed in other information systems.

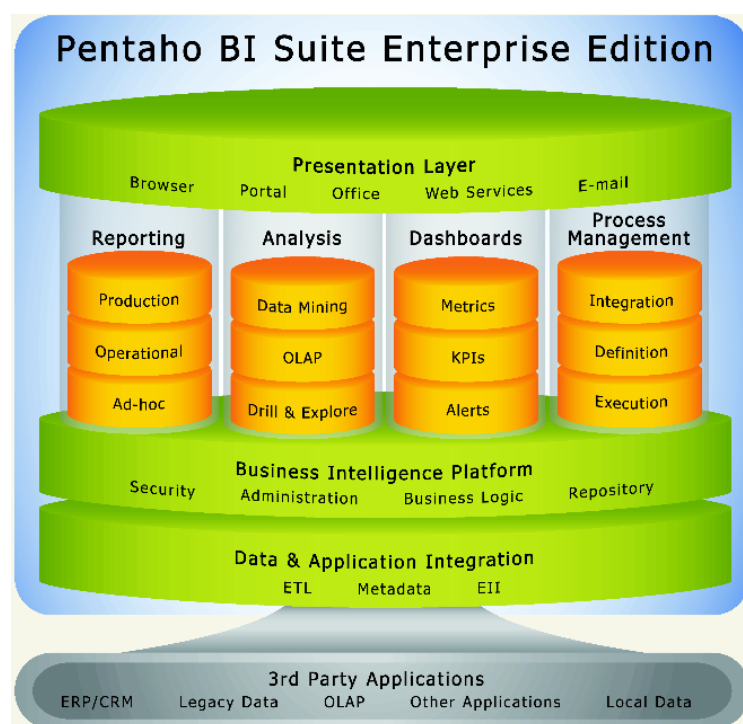


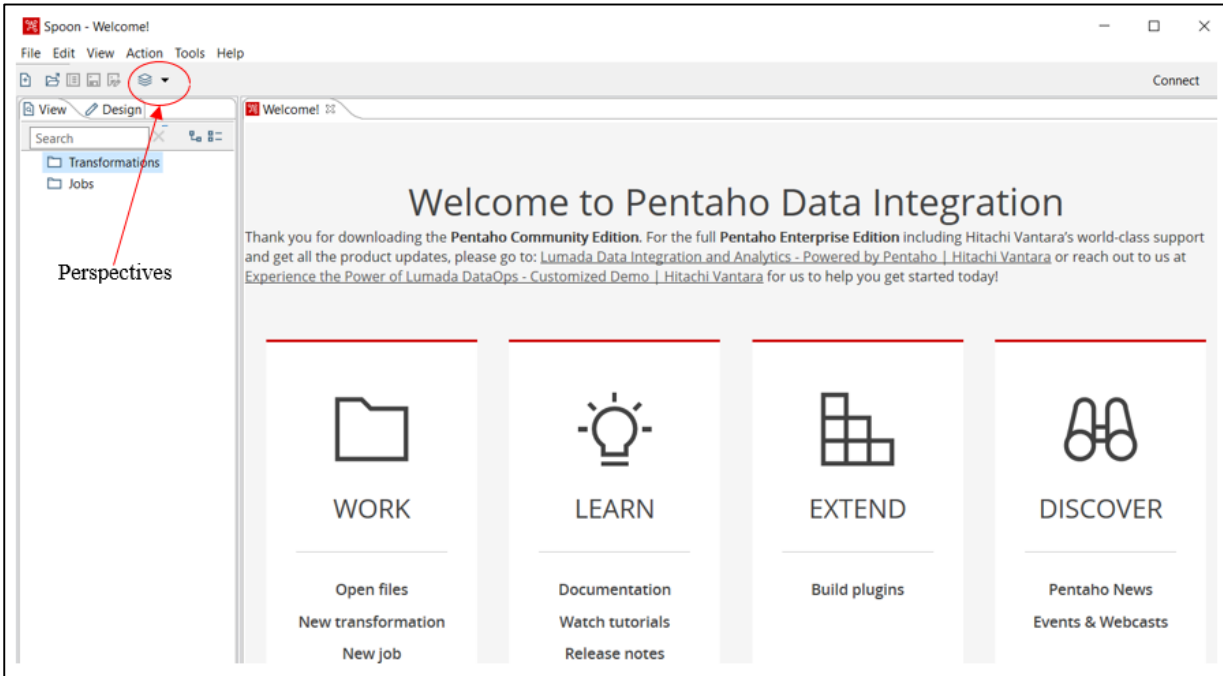
Figure 4.5: Pentaho suite [<https://www.wexer.ro/en/pentaho-bi-suite>]

As can be seen in Figure 4.5, the set of programs that make up the Pentaho suite can be visualized as a stack of components: reporting, analysis, dashboards and process management, which constitute the middle layer of the stack, while the Business Intelligence platform provides administrative resources basic and security. Data and application integration is required to obtain data from different sources, bringing them together in a shared Data Warehouse environment.

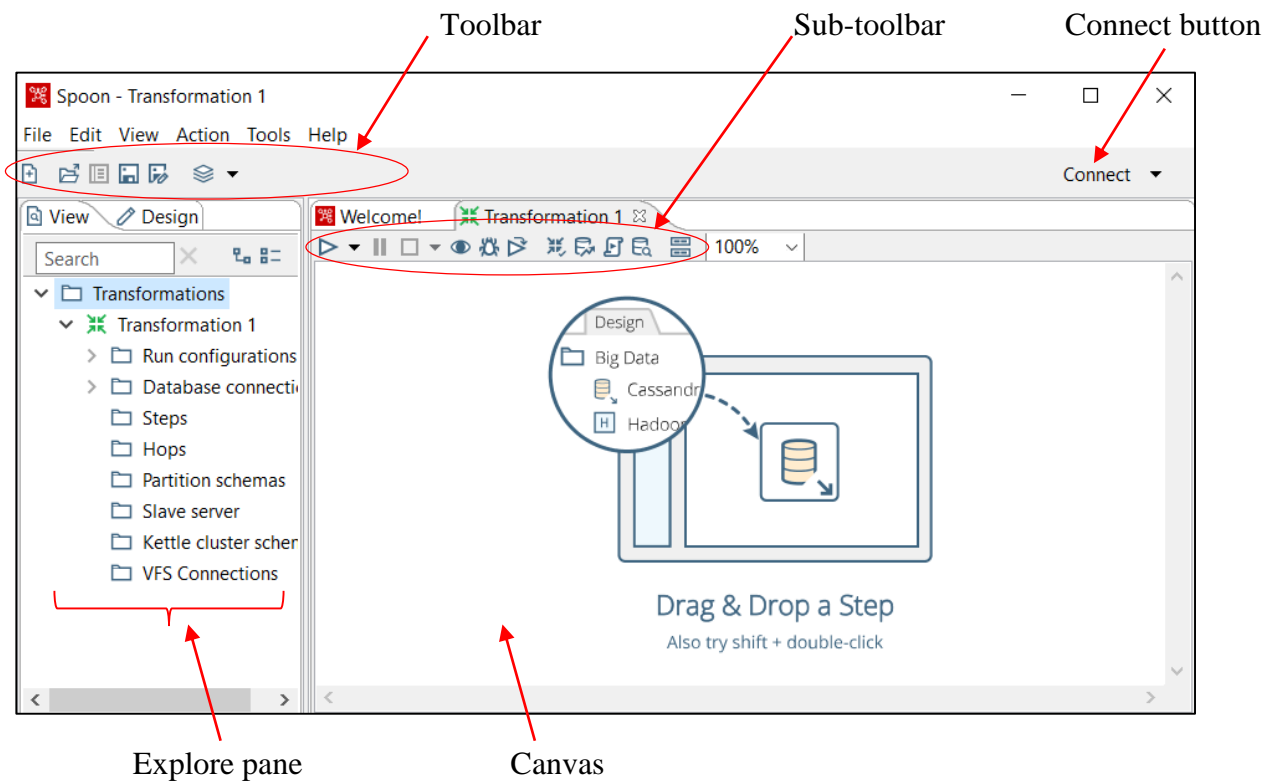
- ▶ **Pentaho BI Server** is a web application (accessed through an internet browser) that consists of a suite of programs that "provide the architecture and infrastructure needed to build business intelligence solutions" [Meadows et al., 2013]. Web server access connects the client user to the data cube and database information; also, multiple users are allowed to access information simultaneously without one user interfering with another's display.
- ▶ **Pentaho Reporting** is a tool that allows the creation of relational and analytical reports, in files with different extensions, including: PDF, Excel, HTML, Text, Rich-Text File, XML or CSV.
- ▶ **Pentaho Analysis** enables information exploration and provides intuitive and interactive analytical reporting, enabling non-technical business users to understand and gain the knowledge and understanding needed to make optimal decisions. It is integrated with other products from the Pentaho BI suite and allows advanced sorting and filtering of data, creating reports and choosing specific values and attributes to analyze, multiple graphic views.
- ▶ **Pentaho Dashboards** is a tool integrated with the Pentaho Analysis and Pentaho Reporting modules, which offers the user the opportunity to choose different ways of representing and displaying data. It provides dashboards, where analyzed data is grouped and can be represented synthetically, through a variety of pre-built templates.
- ▶ **Pentaho Data Mining** discover, from large volumes of data, patterns and relationships between information, through advanced grouping, segmentation algorithms, decision trees, logistic regression, Bayesian networks, neural networks, evolutionary algorithms.
- ▶ **Pentaho Data Integration (Kettle)** is an ETL tool - a Java-programmed, fully user-oriented data integration solution where ETL processes are encapsulated in metadata that is run through the ETL engine. It easily integrates into any IT infrastructure, has an intuitive graphical environment that allows data from multiple sources to be loaded into a Data Warehouse, and its features and capabilities are suitable for working efficiently with such ETL processes: the application allows users to extract, transform, clean and prepare various data from any source.

Through the Spoon graphical interface, Pentaho offers a graphical environment for rapid design, having two areas: the work area and the design/visualization area.



















After installing the Business Intelligence tool **Pentaho Data Integration (PDI)**, accessing the **Spoon.bat** file, the PDI interface opens through which you can create various data ingestion pipelines by drag-and-drop widgets:



Using the **Perspective** button in the toolbar one can change perspectives:



The table below describes the elements of the PDI window: [hitachivantara.com/Documentation/Pentaho/Data_Integration_and_Analytics/9.4/Products/Learn_about_the_PDI_client/]:

Feature	Description
Toolbar	<p>Use this toolbar to access commonly performed actions:</p> <ul style="list-style-type: none"> • New file button () to create a new job, transformation, database connection, or slave server. • Open file button () to open a transformation or job from a file. • Explore Repository button () to explore the repositories. • Save button () to save the current transformation or job to a file or repository. • Save As button () to save the transformation or job under a different file name or type. • Perspectives button () to switch between the different perspectives: <ul style="list-style-type: none"> ○ Data Integration Perspective: Create ETL transformations and jobs. ○ Schedule Perspective: Manage scheduled ETL activities on the Pentaho Server.
Connect button	<p>Use this button to access the menu to create and connect to repositories for central storage of your ETL jobs and transformations.</p>
Sub-toolbar	<p>Use this toolbar to perform transformation or job actions:</p> <ul style="list-style-type: none"> • Run button () to run a transformation or job: <ul style="list-style-type: none"> ○ Run: Runs the current transformation or job from an XML file or a repository. ○ Run Options: Sets the Run Options and then runs the current transformation or job from an XML file or a repository. • Pause button () to pause a running transformation or job. • Stop button () to stop a running transformation or job: <ul style="list-style-type: none"> ○ Stop: Stops the transformation or job immediately. ○ Stop input processing: Stops the input steps to the transformation or job, while allowing any records already retrieved or initiated to be processed and then stopped. • Preview button () to run the transformation in preview mode to examine the rows produced by the selected steps. • Debug button () to run the transformation in debug mode to troubleshoot execution errors. • Replay button () to replay the processing of a transformation. • Verify button () to verify the transformation. • Analyze button () to run an impact analysis on the database. • SQL button () to generate the SQL that is needed to run the loaded transformation. • Explore DB button () to launch the Database Explorer to perform actions such as preview data, run SQL queries, and generate DDL. • Results button () to show the Execution Results pane. • Lock button () to lock the transformation.
Explore pane	<p>Use this pane to access the Design and View tabs:</p> <ul style="list-style-type: none"> • The Design tab provides a list of steps or entries that are used to build transformations or jobs. • The View tab provides information about available database connections and the steps and hops used for the transformation or job.
Canvas	<p>Use this canvas for designing and building transformations and jobs for the ETL activities you want to perform.</p>

ETL solutions allows to extract data from the source environment, transform it according to technical and business needs and load this data into the destination environments. The source and destination media can be databases, text files, XML files, or other structured, semi-structured, or unstructured sources.

The process of moving/transporting data between source and destination environments may include various transformations on the data, such as conversions, filters, calculations, or statistics associated with data movement processes. The graphical interface of PDI allows the design, management and control of these transformations that enable an ETL process:

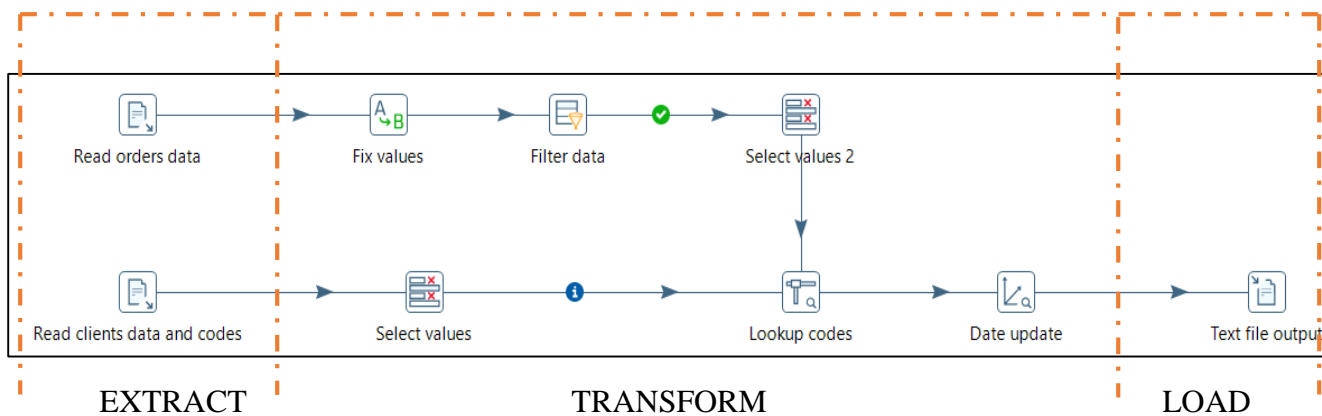


Figure 4.6: Example of a PDI ETL solution

ETL solutions using PDI are based on two types of objects within a workflow: transformations and jobs. The application integrates a PDI engine - a software component that is able to interpret and execute jobs and transformations. In addition to the PDI engine, the application offers a number of tools and utilities to create, manage and initiate transformations and jobs:

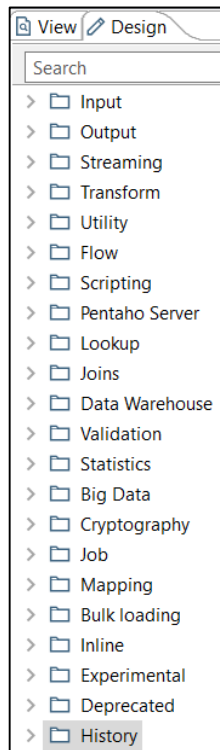
► **Transformations:** To implement an ETL process using PDI, transformations are required. A transformation is a flow of data implemented through a sequence of logical tasks called steps. Through these steps, specific operations on the data are performed, such as extracting, transforming and loading the desired data.

To create a new transformation: **File → New → Transformation:**

File	Edit	View	Action	Tools	Help
New					
Open...			CTRL-O		
Open Recent					
				Transformation	CTRL-N
				Job	CTRL-ALT-N
				Database Connection...	

After this command, a new Transformation tab will open to design the transformation.

The application contains a wide variety of steps to perform the transformations, which can be configured through numerous options and parameters to achieve the desired results. According to the function they perform, the steps are grouped into several categories, visualized in the tree shown in the upper left panel of the Spoon interface:

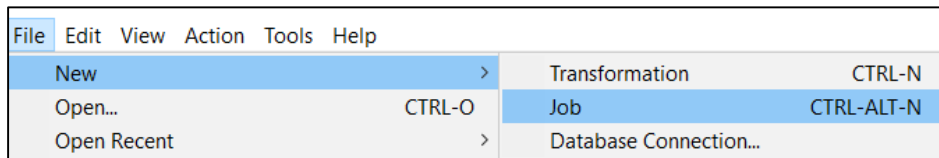


> Input	allows to read data from a variety of different file types: ODBC, Acces, CSV, Excel files, XML files, JSON structures, etc.
> Output	allows loading data into databases or other output formats: Excel files, CSV files, text file, SQL, XML, JSON, etc.
> Streaming	allows to configure the consumption of streaming data in PDI from the specified sources.
> Transform	allows to perform operations on data, such as filtering, sorting, splitting, selecting values, adding new fields, mapping, etc.
> Utility	allows to operate with rows, columns, tables and other various conditional and data processing tasks.
> Flow	allows to perform operations with the data flow, such as abort a transformation, blocking a step, detecting empty flows, performing different operations depending on a condition, etc.
> Scripting	pertain to formula and script execution: allows to build JavaScript expressions, SQL or regular expressions, add constants, input fields or specialized functions to create a script.
> Pentaho Server	allows to connect to the Pentaho server and perform operations (such as set the value of session variable or call API endpoints within a PDI transformation).
> Lookup	allows to add information to the data stream by searching databases, for example allows to look up values in a database table and these values are added as new fields in the stream.
> Joins	allows database and file merge operations: combinations of rows on the input streams, compare two streams of rows, etc.
> Data Warehouse	pertain to Data Warehouse functions: lookups of a more advanced nature.
> Validation	allows to validate credit cards, data, email addresses or other data based on a set of rules.
> Statistics	allows to perform statistical operations and analysis on a data stream.

> Big Data	allows to load and extract data from Avro, Cassandra, Hadoop, MongoDB: read/write to a Cassandra column family, serialization of data into JSON/Avro binary format from data stream followed by writing to file, etc.
> Cryptography	allows encrypting and decrypting files with PGP (Pretty Good Privacy) standard.
> Job	allows to perform operations of a job: read file names, setting file names, setting variables, etc.
> Mapping	allows the reuse of an existing transformation (a repetitive, reusable part of a transformation - a sequence of steps) as the subcomponent in other transformations.
> Bulk loading	pertain to bulk loading of data.
> Inline	inline data modification: inserting rows into the transformation using the Kettle API and Java, read/write a socket.
> Experimental	allows to upload a file/a stream file to a remote host via SFTP.
> Deprecated	contains steps that may be removed from a future version.
> History	contains the steps frequently used by the user.

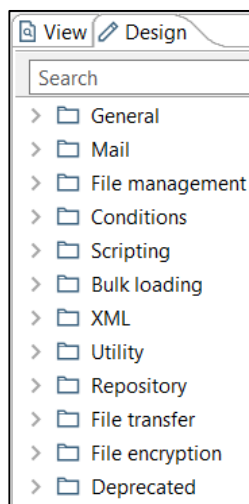
► **Jobs:** After designing and carrying out the transformations, a job is created, which defines and organizes the order in which the transformations will be executed, as well as defining the period in which they will be executed.

To create a new job: **File → New → Job:**



After this command, a new Job tab will be opened to design the job.

The application contains the following steps available for designing a job, grouped into the following categories, viewed in the tree shown in the upper left panel of the Spoon interface:



> General	offers a wide range of functionalities, from defining the starting point for the execution of the work, to running the transformation and obtaining the files on a web server.
> Mail	allows to send emails, recover accounts or check the validity of an email address.
> File management	allows file input/output operations, such as adding files/ folders to the result list of the job entry, creating, deleting, moving, comparing the contents of files or compressing, etc.
> Conditions	allows to perform checks necessary for ETL processes, such as the existence of a file, a folder, or table in a database connection, evaluate fields or variables, check if a folder is empty, etc.
> Scripting	allows to create and control scripts execution.
> Bulk loading	allows bulk loading of data to a MySQL table, to MSSQL, Access and other files.
> XML	allows XML validation and XSL execution.
> Utility	allows different functions to ensure the execution of transformations, ping a host or truncate tables, write message to log, etc.
> Repository	allows to perform operations with the repository of transformations and jobs: export repository or check the connection to a repository.
> File transfer	allows file transfer operations
> File encryption	allows use of PGP for sending and file reception and for verifying the signature.
> Deprecated	contains steps that may be removed from a future version.
> History	contains the steps frequently used by the user (appears only if the steps have been used previously)

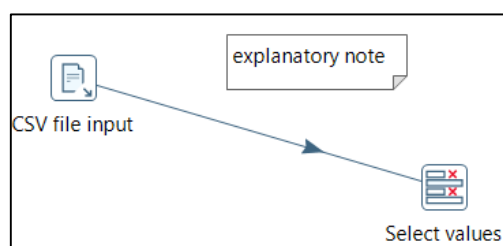
► **Hops:** The constituent elements of the transformations and jobs are interconnected by hops - graphic representations of the data that flows between the two elements.

To create a hop, there are several alternative possibilities:

Click on the source step → press the <SHIFT> key down → draw the line to the target step.

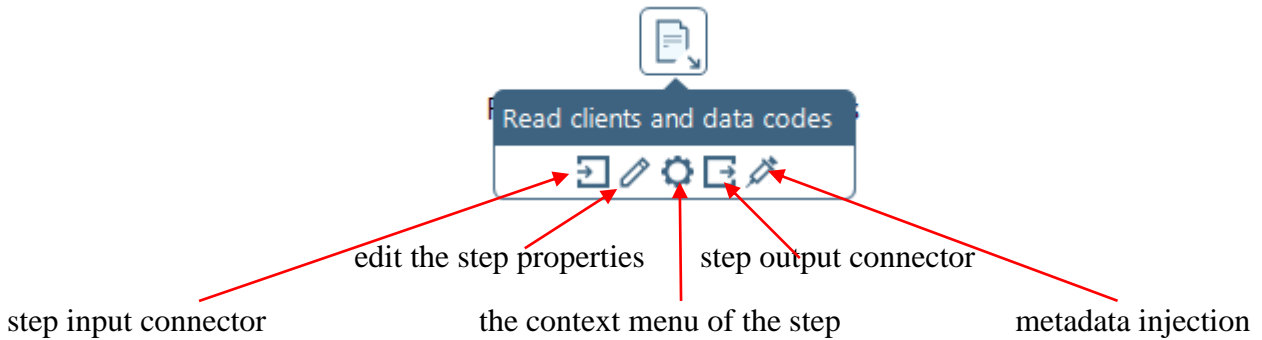
or:

While using the middle mouse button → drag on the graphical view between the steps.



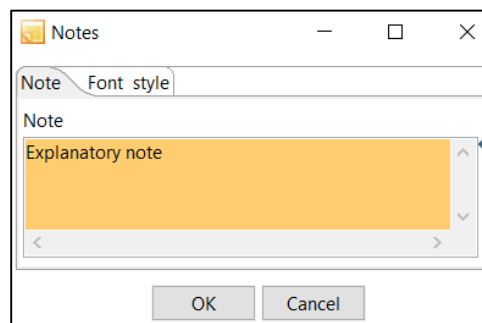
or:

Hover over a step until the hover menu appears → drag the hop painter icon from the source step to the target step.



► **Notes:** are small boxes in which one can enter explanatory text and can be placed anywhere in a transformation.

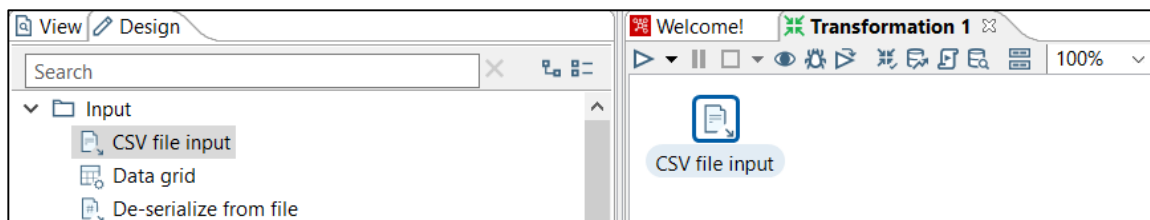
To add a note: **Right-click on the canvas** → select **New Note...** → in the **Notes window** that opens → **enter the text** and **configure the appearance of the note:**



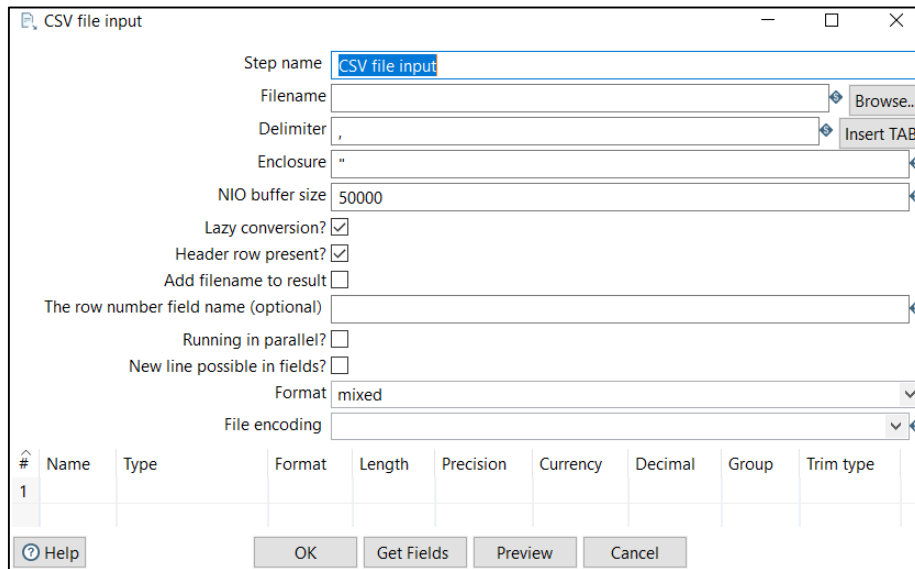
Examples of PDI solutions

► **Importing a .csv file to create a .txt file:**

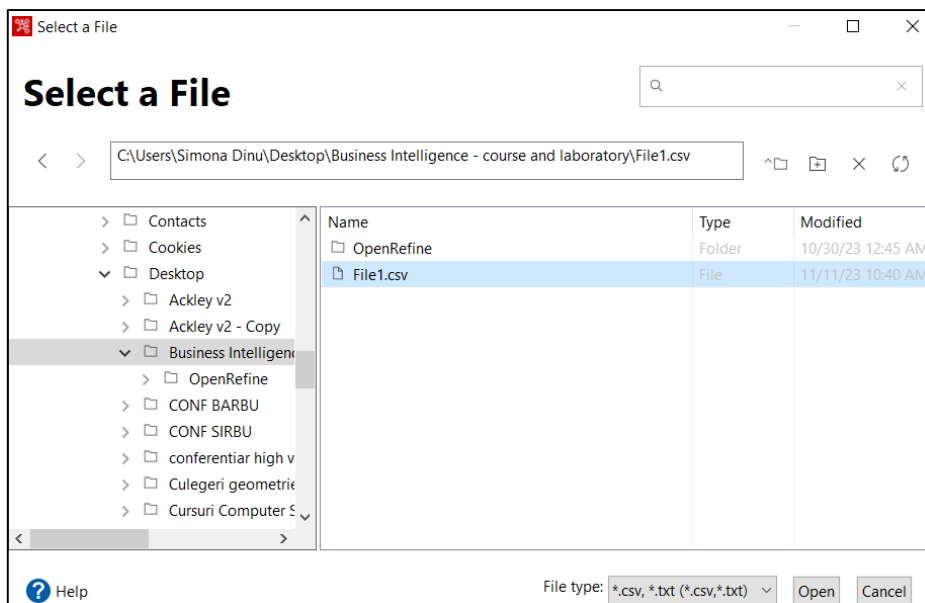
1. **Design** → **Input** → **CSV file input** → **drag-and-drop the component** in the canvas:



2. **Double click** on the **CSV file input** component:

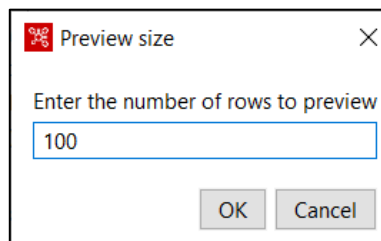


3. In the **Filename** field → Click on the **Browse** button to locate the file:



4. In the field **Delimiter** → **select the field delimiter.**

5. Click on the **Get Fields** button → Click on the **Preview** button to view the data:



6. In the **Examine preview** data window → **check the data that will be brought to the output file:**

Examine preview data

Rows of step: CSV file input (13 rows)

#	No	Department	Head of department/department name	Department name	Number of persons
1	1	department_1	Adam Smith/department_C3	department_C3	25
2	2	department_2	Carol Morgan/department_B2	department_B2	13
3	3	department_3	Paul Johnson/department_D1	department_D1	7
4	4	department_4	Martha Brown/department_D1	department_D1	4
5	5	department_5	John Murphy/department_C1	department_C1	13
6	6	department_6	Mary Cooper/department_B1	department_B1	6
7	7	department_7	Dona Raven/department_D2	department_D2	19
8	8	department_8	Will Bart/department_D3	department_D3	24
9	9	department_9	Paul Taylor/department_D2	department_D2	9
1..	10	department_10	Ana Barrel/department_B1	department_B1	18
1..	11	department_11	Beth Evans/department_C1	department_C1	7
1..	12	department_2	Carol Morgan/department_B2	department_B2	13
1..	13	department_13	Mary Coampbell/department_B3	department_B3	9

Close Show Log

7. Close → OK: the fields of the file appeared in the CSV file input window:

CSV file input

Step name: CSV file input

Filename: C:\Users\Simona Dinu\Desktop\Business Intelligence - course and laboratory\File1.csv

Delimiter: .

Enclosure: -

NIO buffer size: 50000

Lazy conversion?

Header row present?

Add filename to result

The row number field name (optional):

Running in parallel?

New line possible in fields?

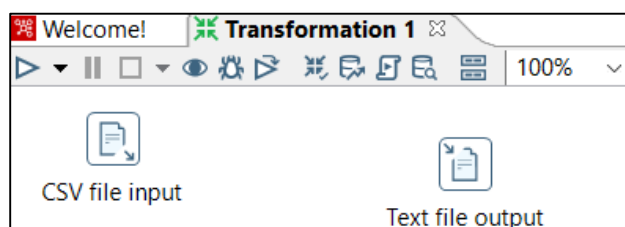
Format: mixed

File encoding:

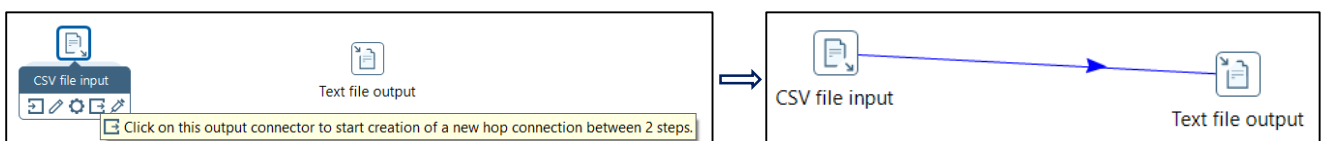
#	Name	Type	Format	Length	Precision	Currency	Decimal	Group	Trim type
1	No;Department;Head of department/department name;Department name;Number of persons	String		61		£	.	.	none

Help OK Get Fields Preview Cancel

8. Design → Output → Text file input → drag-and-drop the component in the canvas:

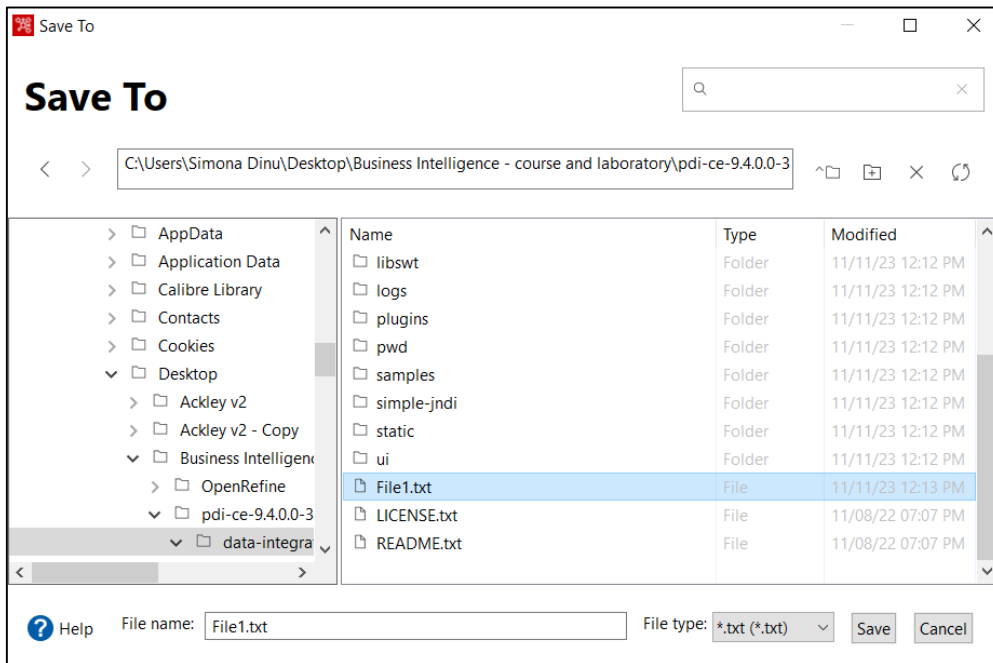


9. Create a new hop connection between the two components:

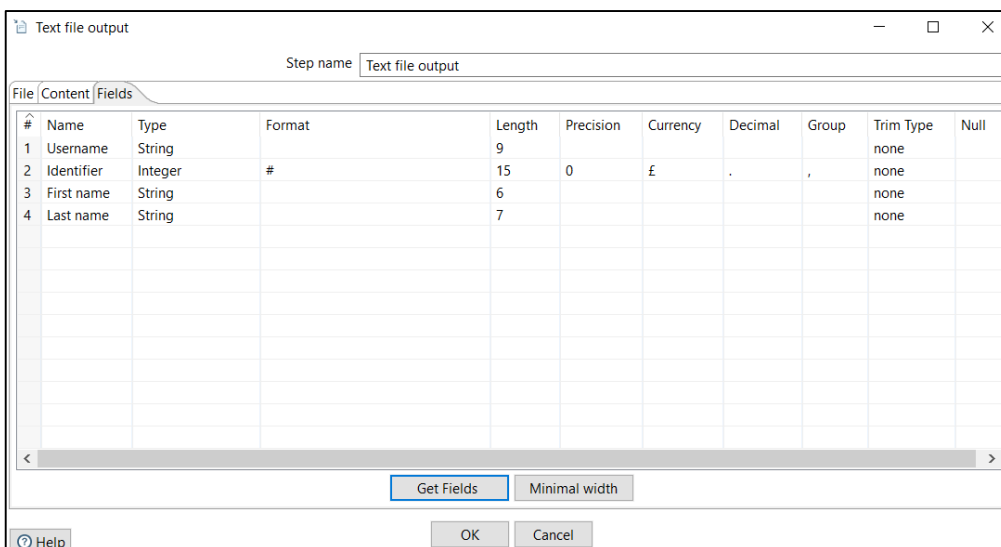


10. Double click on the Text file output component → Browse (to locate the text file where the data will be transferred)

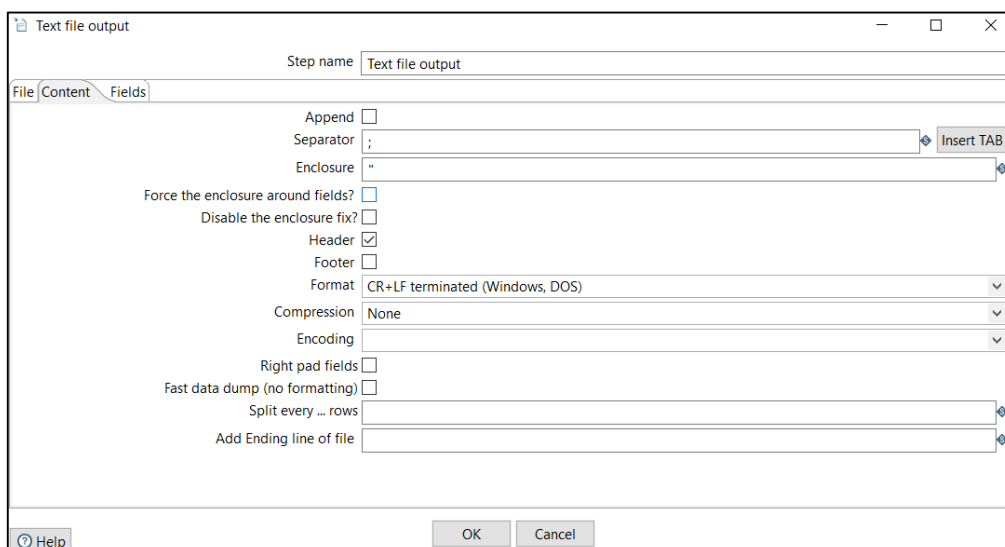
11. In the Save To window → enter the location and name of the output text file:



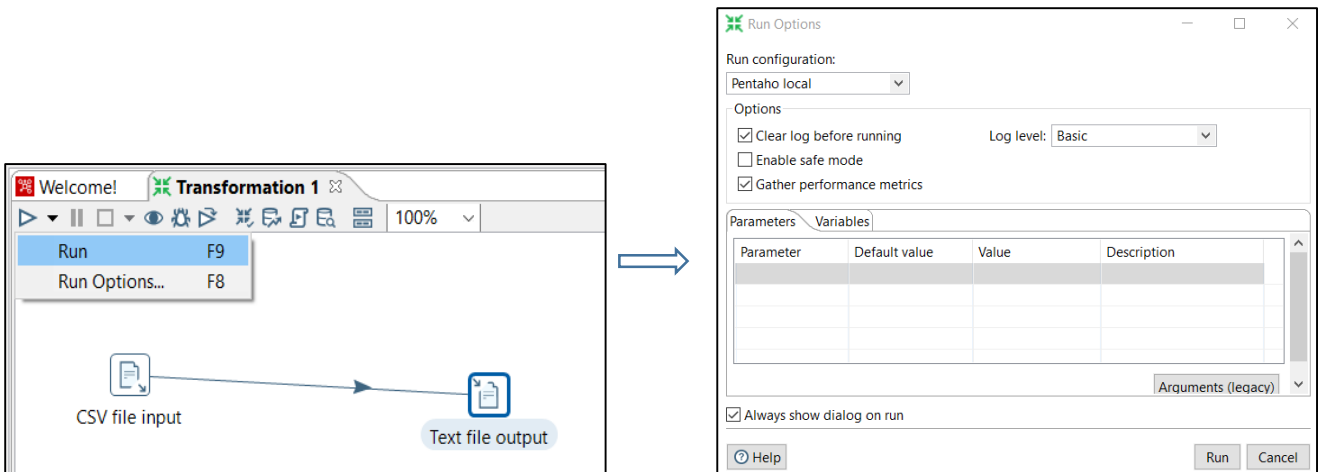
12. In the **Fields** tab → **Get fields**:



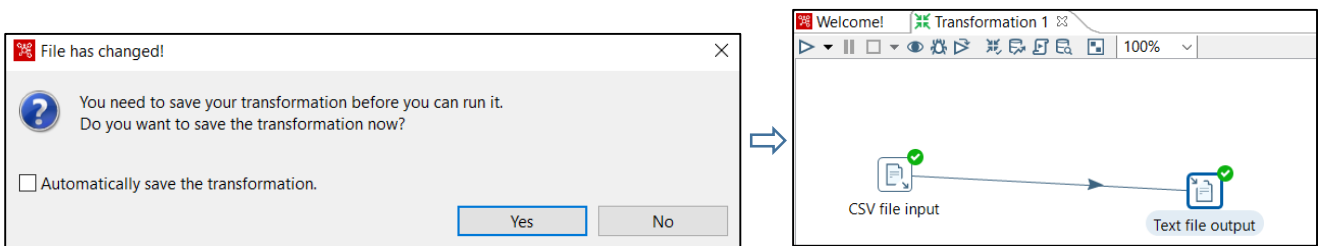
13. In the **Content** tab → in the **Separator** field → enter the field separator → **OK**:



14. Click on the **Run** button → run options can be set in the **Run Options** window → **Run**:



15. Save the transformation:



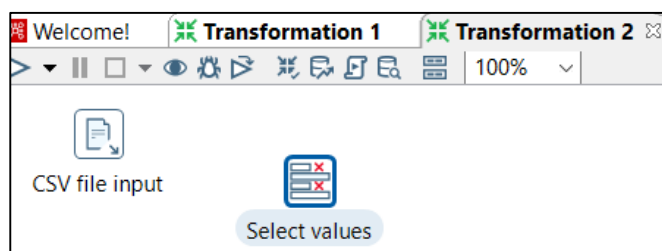
16. The text file is opened to verify the data:

*file2.csv - Notepad

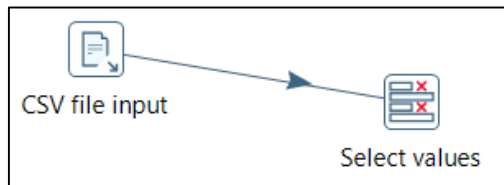
No	Department	Head of department/department name	Department name	Number of persons
1	department_1	Adam Smith/department_C3	department_C3	25
2	department_2	Carol Morgan/department_B2	department_B2	13
3	department_3	Paul Johnson/department_D1	department_D1	7
4	department_4	Martha Brown/department_D1	department_D1	4
5	department_5	John Murphy/department_C1	department_C1	13
6	department_6	Mary Cooper/department_B1	department_B1	6
7	department_7	Dona Raven/department_D2	department_D2	19
8	department_8	Will Bart/department_D3	department_D3	24
9	department_9	Paul Taylor/department_D2	department_D2	9
10	department_10	Ana Barrel/department_B1	department_B1	18
11	department_11	Beth Evans/department_C1	department_C1	7
12	department_2	Carol Morgan/department_B2	department_B2	13
13	department_13	Mary Coampbell/department_B3	department_B3	9

► **Changing data field formats:**

1. **Design** → **Transform** → **Select values** → **drag-and-drop** the component in the canvas:



2. **Create a new hop connection** between the two components:



3. Preview the fields of the CSV file input:

Examine preview data

Rows of step: CSV file input (13 rows)

#	No.	Department	Head of department/department name	Department name	Number of persons	Employment date
1	1	department_1	Adam Smith/department_C3	department_C3	25	12/04/2020
2	2	department_2	Carol Morgan/department_B2	department_B2	13	13/09/2014
3	3	department_3	Paul Johnson/department_D1	department_D1	7	14/04/2020
4	4	department_4	Martha Brown/department_D1	department_D1	4	15/04/2020
5	5	department_5	John Murphy/department_C1	department_C1	13	16/07/2021
6	6	department_6	Mary Cooper/department_B1	department_B1	6	17/04/2020
7	7	department_7	Dona Raven/department_D2	department_D2	19	18/04/2021
8	8	department_8	Will Bart/department_D3	department_D3	24	19/04/2020
9	9	department_9	Paul Taylor/department_D2	department_D2	9	20/05/2015
1..	10	department_10	Ana Barrel/department_B1	department_B1	18	21/04/2020
1..	11	department_11	Beth Evans/department_C1	department_C1	7	22/04/2020
1..	12	department_2	Carol Morgan/department_B2	department_B2	13	23/02/2015
1..	13	department_13	Mary Coampbell/department_B3	department_B3	9	04/04/2016

Close Show Log

4. Double click on the Select values component → Get fields to change:

Select values

Step name: Select values

Select & Alter Remove Meta-data

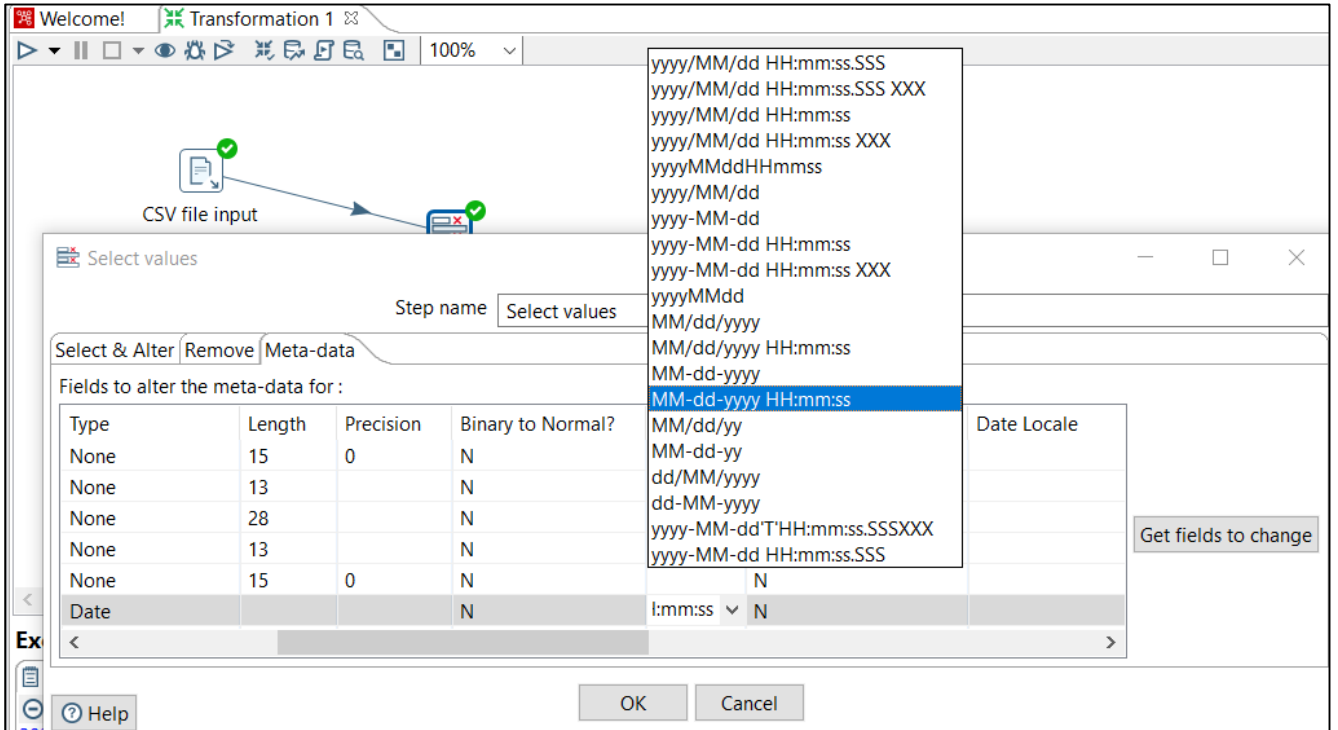
Fields to alter the meta-data for:

#	Fieldname	Rename to	Type
1	No.;Department;Head of department/department name;Department name;Number of persons		None

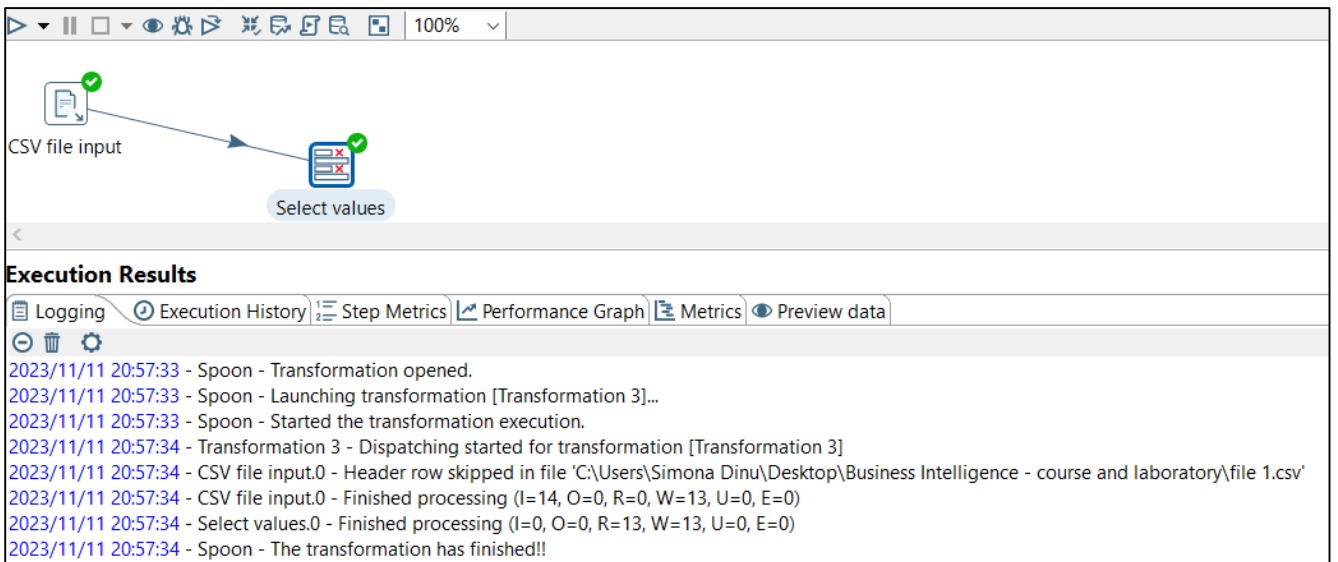
Get fields to change

Help OK Cancel

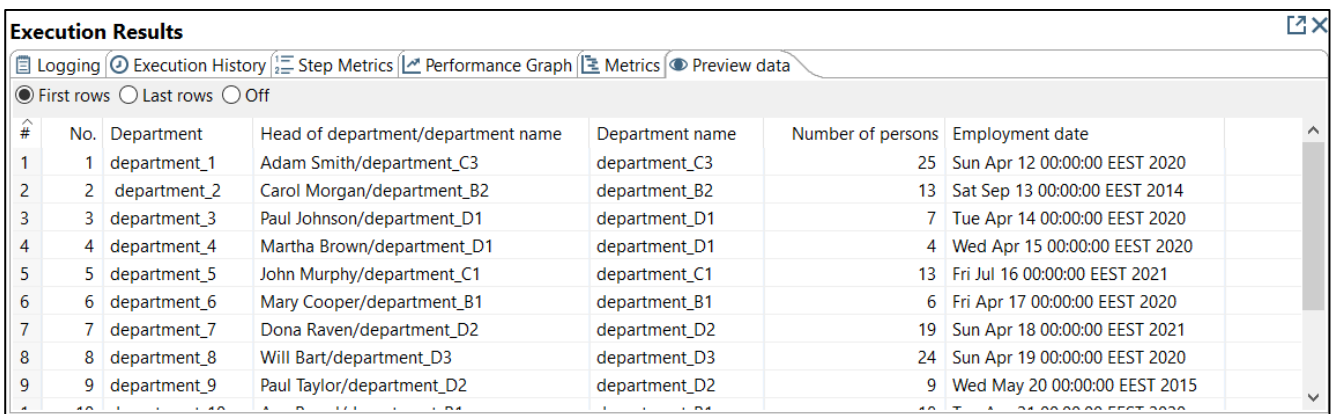
5. For example, the format of the Employment data field is changed:



6. Run:

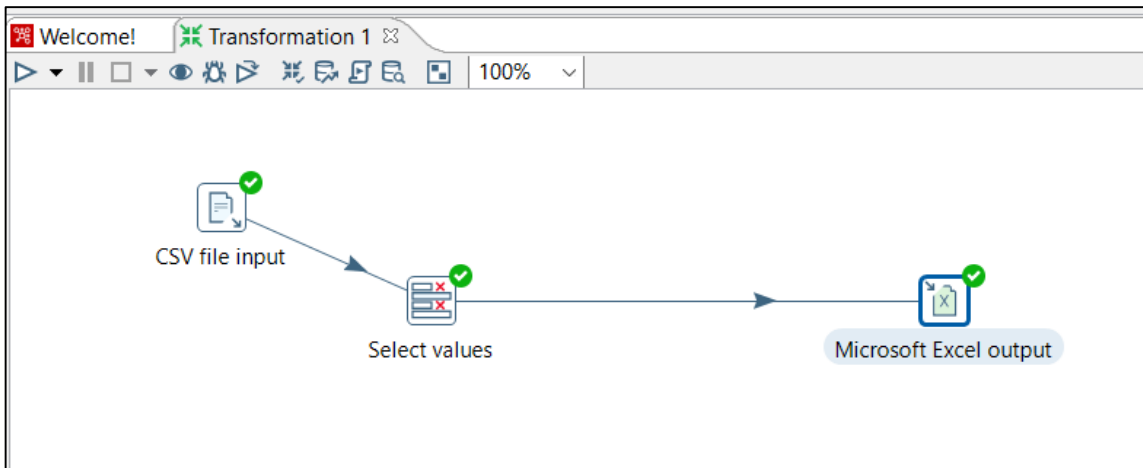


7. Click on the Preview data tab:

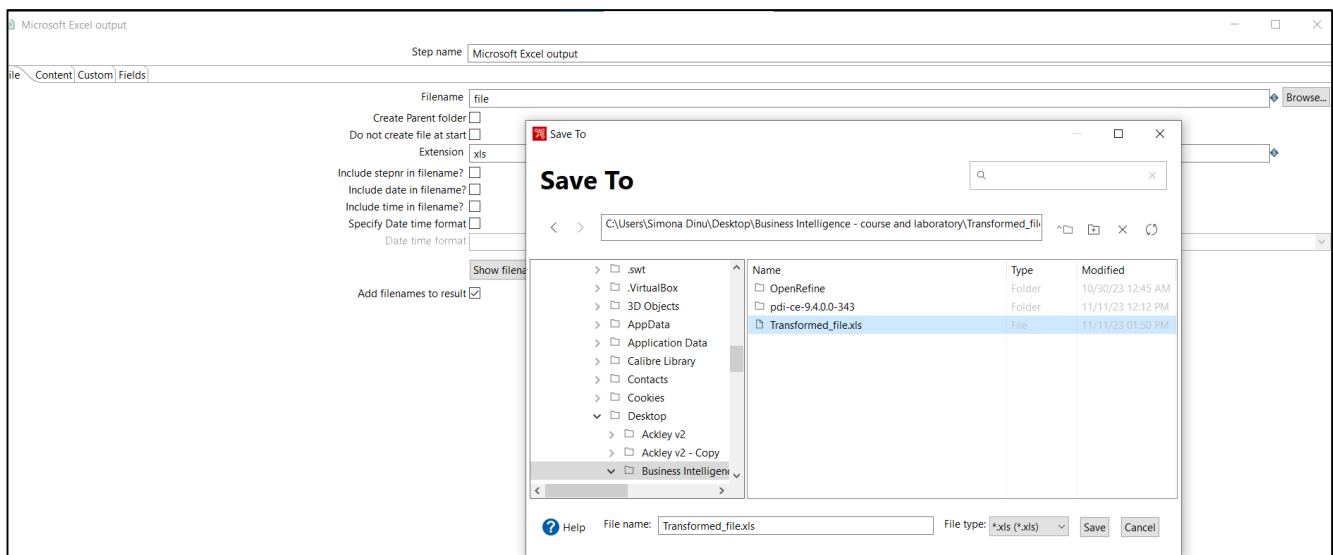


8. To transfer the new data to an Excel file:

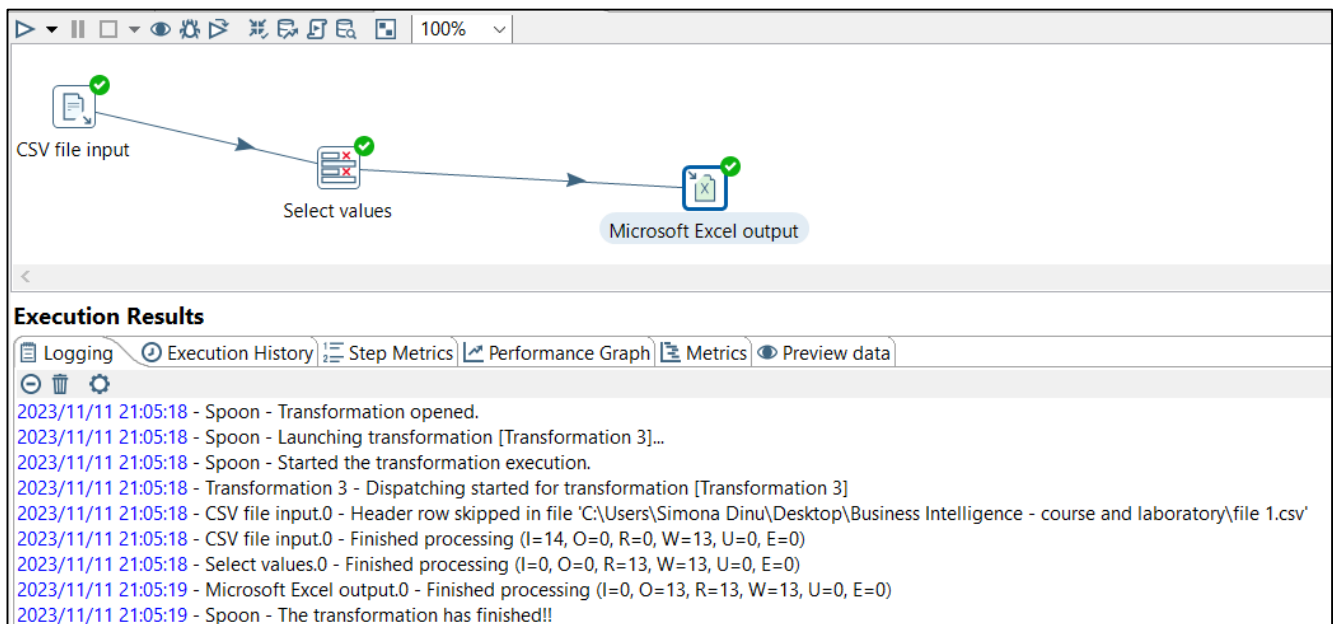
Output → Microsoft Excel output → Create a new hop connection between the two components:



9. Double click on the Microsoft Excel output component → Enter the location and name of the output Excel file:



10. Run:

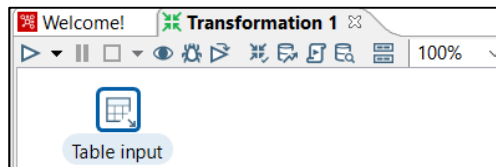


11. The Excel file is opened to verify the data export:

No.	Department	Head of department/department name	Department name	Number of persons	Employment date
1	department_1	Adam Smith/department_C3	department_C3	25	12/04/2020 00:00
2	department_2	Carol Morgan/department_B2	department_B2	13	13/09/2014 00:00
3	department_3	Paul Johnson/department_D1	department_D1	7	14/04/2020 00:00
4	department_4	Martha Brown/department_D1	department_D1	4	15/04/2020 00:00
5	department_5	John Murphy/department_C1	department_C1	13	16/07/2021 00:00
6	department_6	Mary Cooper/department_B1	department_B1	6	17/04/2020 00:00
7	department_7	Dona Raven/department_D2	department_D2	19	18/04/2021 00:00
8	department_8	Will Bart/department_D3	department_D3	24	19/04/2020 00:00
9	department_9	Paul Taylor/department_D2	department_D2	9	20/05/2015 00:00
10	department_10	Ana Barrel/department_B1	department_B1	18	21/04/2020 00:00
11	department_11	Beth Evans/department_C1	department_C1	7	22/04/2020 00:00
12	department_2	Carol Morgan/department_B2	department_B2	13	23/02/2015 00:00
13	department_13	Mary Coampbell/department_B3	department_B3	9	04/04/2016 00:00

► Getting data from a database:

1. Design → Input → Table input → drag-and-drop the component in the canvas:



2. Double click on the Table input component to create the connection to the database where data resides:

Table input

Step name: Table input

Connection: [Dropdown] Edit... New... Wizard...

SQL: `SELECT <values> FROM <table name> WHERE <conditions>` Get SQL select statement...

Line 1 Column 35

Store column info in step meta

Enable lazy conversion

Replace variables in script?

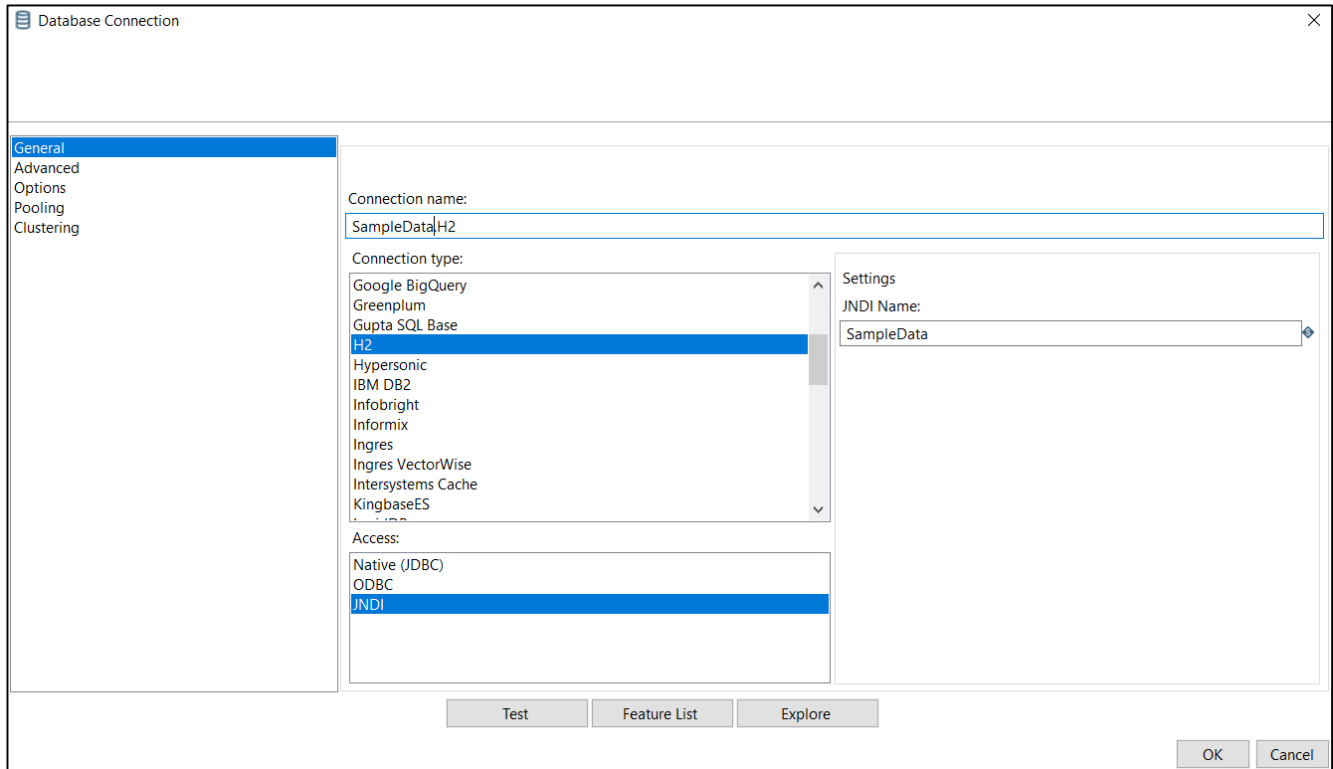
Insert data from step [Dropdown]

Execute for each row?

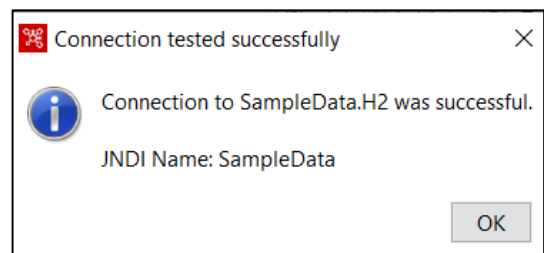
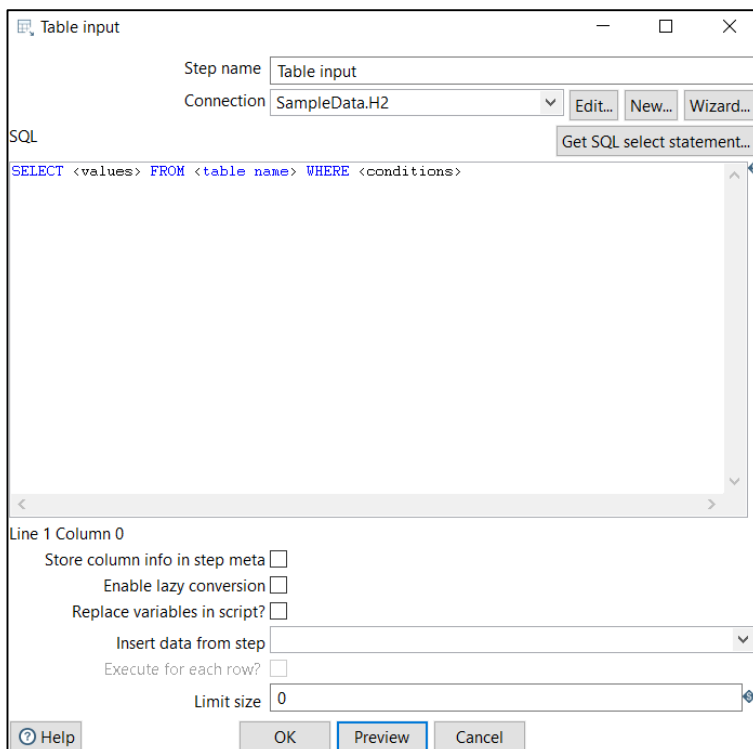
Limit size: 0

Help OK Preview Cancel

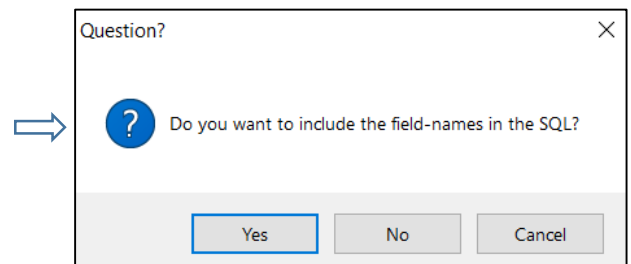
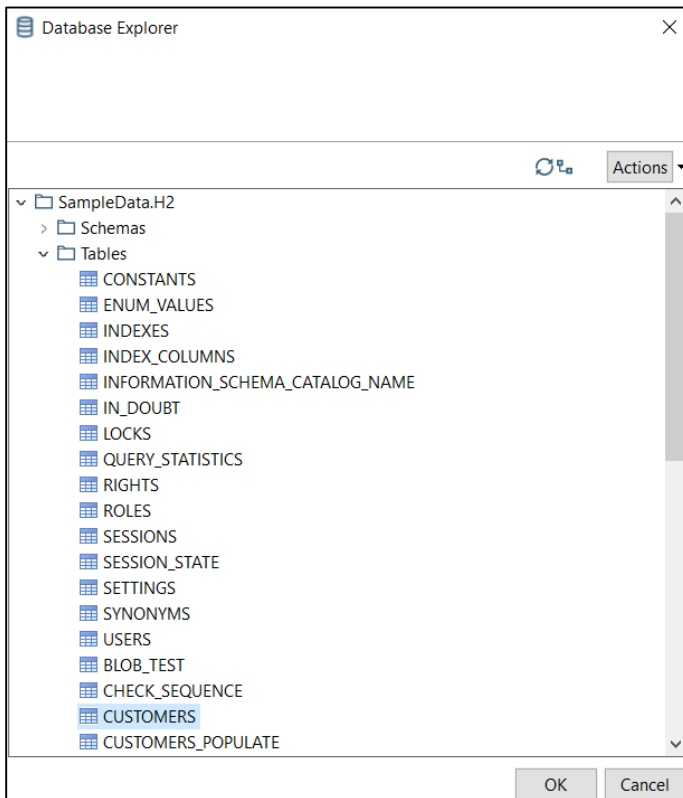
3. Click on the **New** button → in the **Database Connection** window, define the connection for the **SampleData.H2** database, which is the default database of Pentaho → **OK**:



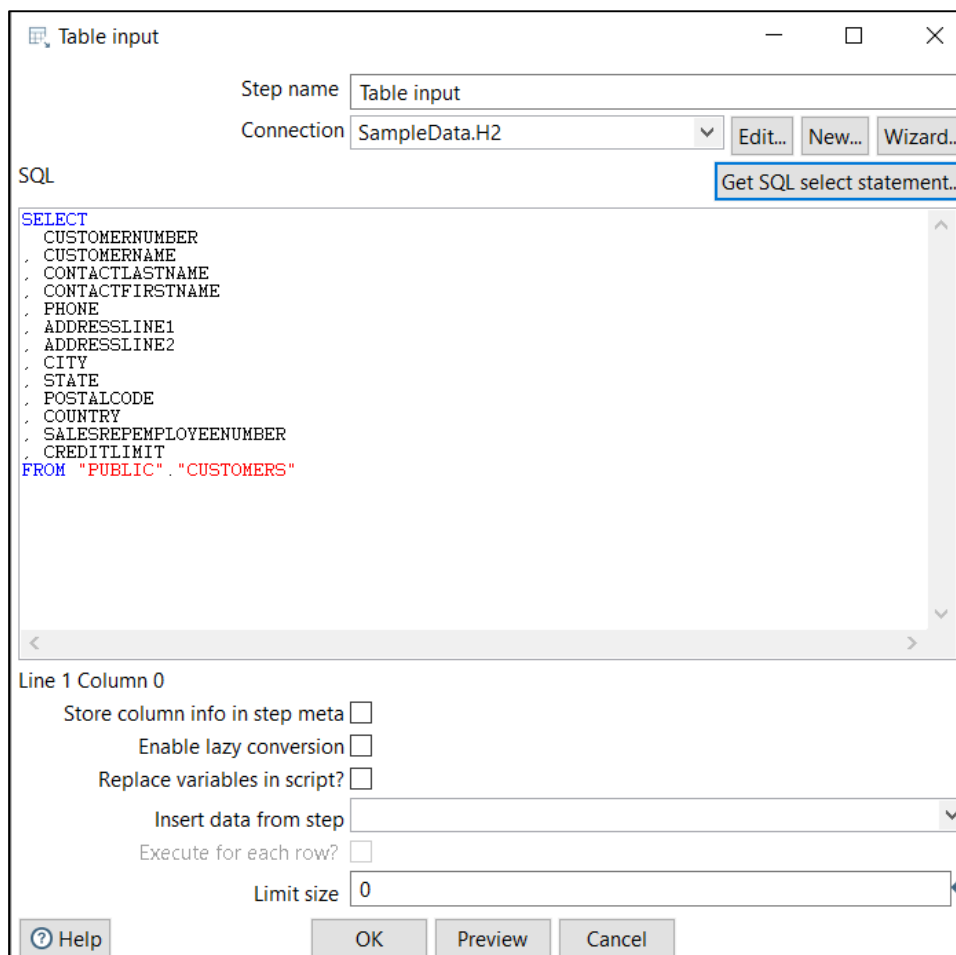
4. Click on the **Preview** button to test the connection:



5. Click on the **Get SQL select statement** button → the data from the **CUSTOMERS** table is fetched:



6. In the Table input window → click on Preview to check the imported data → OK:



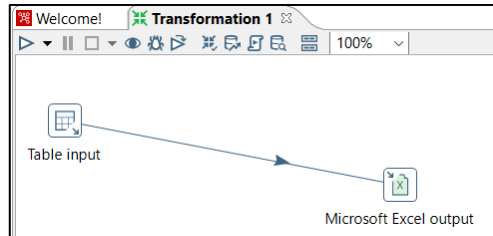
Examine preview data

Rows of step: Table input (122 rows)

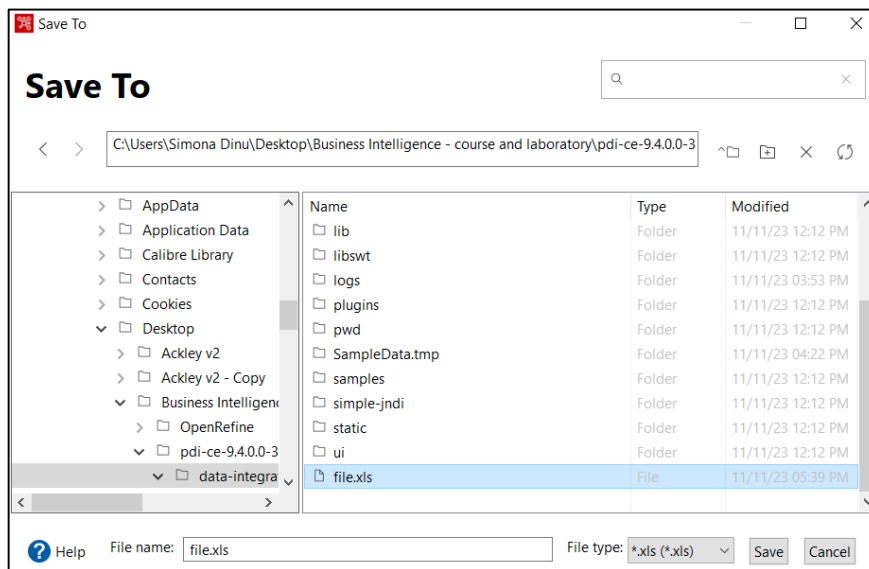
#	CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTNAME	CONTACTFIRSTNAME	PHONE	ADDRESSLINE1	ADDRESSLINE2	CITY	STATE
1	103	Atelier graphique	Schmitt	Carine	40.32.2555	54, rue Royale	<null>	Nantes	<null>
2	112	Signal Gift Stores	King	Sue	7025551838	8489 Strong St.	<null>	Las Vegas	NV
3	114	Australian Collectors, Co.	Ferguson	Peter	03 9520 4555	636 St Kilda Road	Level 3	Melbourne	Victoria
4	119	La Rochelle Gifts	Labruno	Janine	40.67.8555	67, rue des Cinquante Otages	<null>	Nantes	<null>
5	121	Baane Mini Imports	Bergulfsen	Jonas	07-98 9555	Erling Skakkes gate 78	<null>	Stavern	<null>
6	124	Mini Gifts Distributors Ltd.	Nelson	Valarie	4155551450	5677 Strong St.	<null>	San Rafael	CA
7	125	Havel & Zbyszek Co.	Piestrzeniewicz	Zbyszek	(26) 642-7555	ul. Filtrowa 68	<null>	Warszawa	<null>
8	128	Blauer See Auto, Co.	Keitel	Roland	+49 69 66 90 2555	Lyonerstr. 34	<null>	Frankfurt	<null>
9	129	Mini Wheels Co.	Murphy	Julie	6505555787	5557 North Pendale Street	<null>	San Francisco	CA
1.	131	Land of Toys Inc.	Yu	Kwai	2125557818	897 Long Airport Avenue	<null>	NYC	NY
1.	141	Euro+ Shopping Channel	Freyre	Diego	(91) 555 94 44	C/ Moralzarzal, 86	<null>	Madrid	<null>
1.	144	Volvo Model Replicas, Co	Berglund	Christina	0921-12 3555	Berguvsv\u000e4gen 8	<null>	Lule\u000e5	<null>
1.	145	Danish Wholesale Imports	Petersen	Jytte	31 12 3555	Vinb\u000e6t 34	<null>	Kobenhavn	<null>

Close Show Log

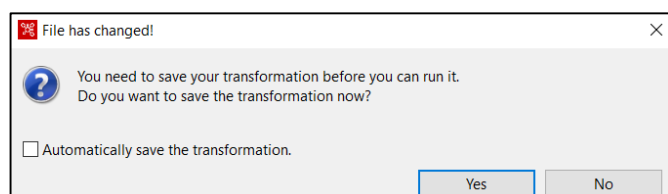
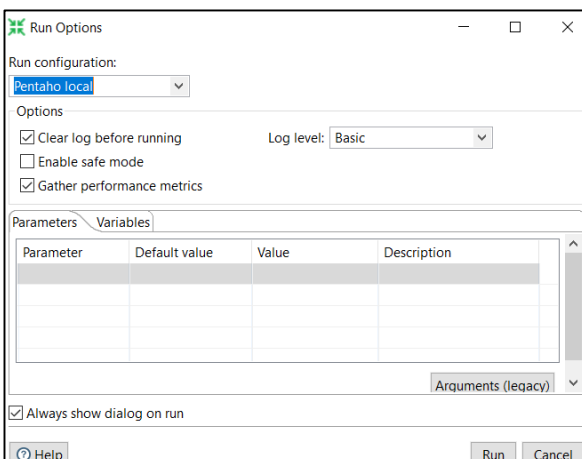
7. The data imported from the database can then be used in other processes, for example they can be exported to an Excel file:

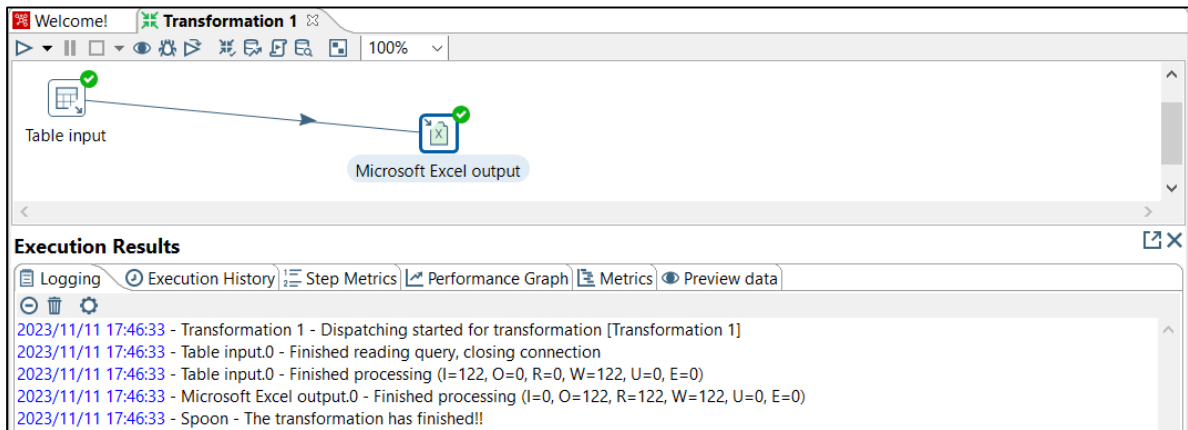


8. Double click on the Microsoft Excel output component → Enter the location and name of the output Excel file:



9. Run:





10. Open the Excel file to check the data export:

A	B	C	D	E	F	G	H	I	J	K	L
CUSTOMERNUMBER	CUSTOMERNAME	CONTACTLASTN	CONTACTF	PHONE	ADDRESSLINE1	ADDRESSLINE2	CITY	STATE	POSTAL.CODE	COUNTRY	SALESREPEMPLOYEE
103.00	Atelier graphique	Schmitt	Carine	40 32 2555	54, rue Royale		Nantes	NV	44000	France	1,370.00
112.00	Signal Gift Stores	King	Sue	7025551838	8489 Strong St		Las Vegas	NV	83030	USA	1,166.00
114.00	Australian Collectors, Co.	Ferguson	Peter	03 9520 4555	636 St Kilda Road	Level 3	Melbourne	Victoria	3004	Australia	1,611.00
119.00	La Rochelle Gifts	Labrunne	Janine	40 67 8555	67, rue des Cinquante Otages		Nantes		44000	France	1,370.00
121.00	Baane Mini Imports	Bergulfson	Jonas	07-98 9555	Erling Skakkes gate 78		Stavern		4110	Norway	1,504.00
124.00	Mini Gifts Distributors Ltd.	Nelson	Valarie	4155551450	5677 Strong St		San Rafael	CA	97562	USA	1,165.00
125.00	Havel & Zbyszek Co	Pieastrzeniewicz	Zbyszek	(26) 642-7555	ul. Filrowa 68		Warszawa		01-012	Poland	1,504.00
128.00	Blauer See Auto. Co.	Kottel	Roland	+49 69 66 90 2555	Lyonerstr. 34		Frankfurt		60528	Germany	1,504.00
129.00	Mini Wheels Co.	Murphy	Julie	650555787	5557 North Pendale Street		San Francisco	CA	94217	USA	1,165.00
131.00	Land of Toys Inc.	Yu	Kwai	2125557818	897 Long Airport Avenue		NYC	NY	10022	USA	1,323.00
141.00	Euro+ Shopping Channel	Freyre	Diego	(91) 555 94 44	C/ Moratalzaral, 86		Madrid		28034	Spain	1,370.00
144.00	Volvo Model Replicas, Co	Berglund	Christina	0921-12 3555	Berguvsvu00e4gen 8		Lulea	SE	S-958 22	Sweden	1,504.00
145.00	Danish Wholesale Imports	Petersen	Jytte	31 12 3555	Vimbu00e6let 34		Kobenhavn		1734	Denmark	1,401.00
146.00	Savelley & Henrot, Co.	Savelley	Mary	78 32 5555	2, rue du Commerce		Lyon		69004	France	1,337.00
148.00	Dragon Souvenirs, Ltd.	Nathidad	Enc	+65 221 7555	Bronx Stok - Bronx Apt. 3/6 Tesvikiye		Singapore		079903	Singapore	1,621.00
151.00	Muscle Machine Inc	Young	Jeff	2125557413	4092 Furth Circle	Suite 400	NYC	NY	10022	USA	1,286.00
157.00	Diecast Classics Inc.	Yu	Kyung	2155551555	7586 Pompton St.		Allenstown	PA	70267	USA	1,216.00
161.00	Technics Stores Inc.	Hirano	Juri	6505556809	9408 Furth Circle		Burlingame	CA	94217	USA	1,165.00
166.00	Handji Gifts & Co	Victorino	Wendy	+65 224 1555	Village Close - 106 Linden Road Sandown	2nd Floor	Singapore		069045	Singapore	1,612.00
167.00	Henkku Gifts	Oeztan	Veysel	+47 2267 3215	Drammen 121, PR 744 Sentrum		Bergen		N 5804	Norway	1,504.00
168.00	American Souvenirs Inc	Franco	Sue	2035557945	149 Spinnaker Dr.		New Haven	CT	97623	USA	1,286.00
169.00	Porto Imports Co	de Castro	Isabel	(1) 356-5555	Estrada da saeu00fاده n. 58	Suite 101	Lisboa		1756	Portugal	1,286.00
171.00	Daedalus Designs Imports	Rancu00e9	Martine	20 18 1555	184, chaussu00e9e de Tournai		Lille		59000	France	1,370.00
172.00	La Corne D'abondance, Cc	Bertrand	Marie	(1) 42 34 2555	265, boulevard Charonne		Paris		75012	France	1,337.00
173.00	Cambridge Collectables Cc	Tseng	Kyung	6175555555	4658 Baden Av.		Cambridge	MA	51247	USA	1,188.00
175.00	Gift Depot Inc.	King	Julie	2035552570	25593 South Bay Ln.		Bridgewater	CT	97562	USA	1,323.00
177.00	Osaka Souvenirs Co.	Kentary	Mory	+81 06 6342 5555	Dojima Avanza 4F, 1-6-20 Dojima, Kita-ku		Osaka		530-0003	Japan	1,621.00
181.00	Vitachrome Inc.	Frick	Michael	2125551500	2878 Kingston Rd.	Suite 101	NYC	NY	10022	USA	1,286.00
186.00	Toys of Finland, Co.	Karttunen	Matti	90-224 8555	Keskuskatu 45		Helsinki		21240	Finland	1,501.00
187.00	AV Stores, Co.	Ashworth	Victoria	(171) 555-1555	Fauntleroy Circus		Manchester		EC2 5NT	UK	1,501.00
189.00	Clover Collections, Co.	Cassidy	Dean	+353 1862 1555	25 Maiden Lane	Floor No. 4	Dublin		2	Ireland	1,504.00

4.3.2 Qlik Data Integration for Data Warehouse management

Qlik Data Integration (QDI) operates as an end-to-end solution platform, which through three products (Qlik Replicate, Qlik Compose and Qlik Catalog) efficiently captures large volumes of data from heterogeneous as well as homogeneous data sources and provides data ready for real-time analysis, giving enterprise data architects a new approach to automating data warehouse design, implementation, and management.

According to the specifications of the official website [qlik.com](https://www.qlik.com):

► **Qlik Replicate** is a software with advanced data ingestion and replication capabilities that moves data at high speed, providing access to a greater variety of source and target endpoints. Data are transferred from one system to another and are updated in real time, this being possible thanks to Change Data Capture (CDC) technology. Through this technology, administrators and data architects can easily configure, control and monitor loads for large data sets and create a comprehensive and up-to-date dataset for Business Intelligence analysis.

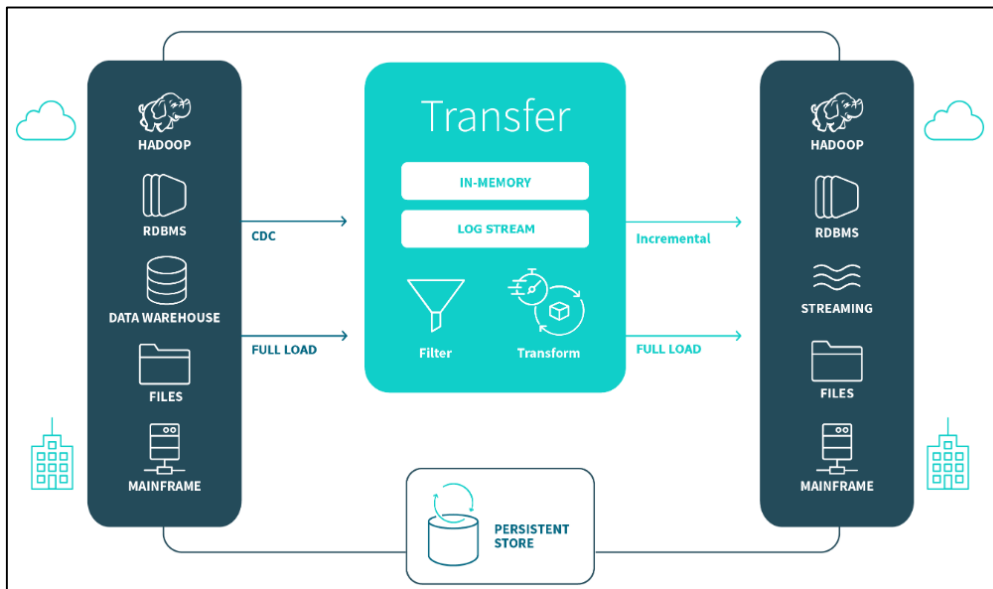


Figure 4.7: Basic architecture of Qlik Replicate

► **Qlik Compose** supports the entire life cycle of data warehouses and data stores through agile and modern automation of the manual and repetitive aspects of data warehouse design, implementation and updating, reducing the time, costs and risks of Data Warehouse projects. Using Qlik Compose, data architects and project managers can quickly update, create, load, and design data warehouses, automatically generate end-to-end workflows, and automatically generate insights for Big Data Analytics and Business Intelligence projects.

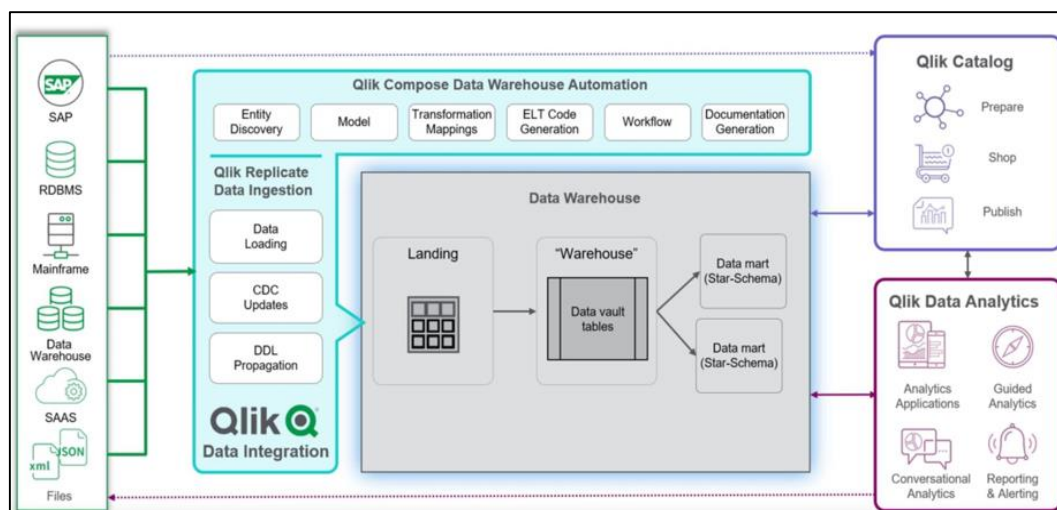


Figure 4.8: Qlik Data Warehouse automation architecture

► **Qlik Catalog** implements a modern, enterprise-scale, secure data catalog solution that provides access to all company data assets, available for various uses, various users, and numerous applications, in one simplified view.

This catalog provides users with a single destination to search, browse, preview, understand, compare and find appropriate datasets and ultimately derive valuable insights from all company data sources.

Chapter V. Microsoft Power BI for Business Intelligence

5.1 Microsoft Power BI tools

Power BI is a top Business Intelligence tool created by Microsoft, designed as a platform with various functionalities for business analysis. The application contains a powerful suite of analytical tools that provide the ability to access the data managed by a company, to merge data sources, to analyze them effectively and to convert them into graphs or reports, in order to present the information in various interactive, high quality visualizations.

According to the analysis carried out by the consulting firm Gartner, the application is considered a leader in this area of business solutions, because:

- guarantees the user ease of use, even with complex data sets;
- has an intuitive graphical interface through which the user can view data stored in the cloud or locally (data coming from a variety of data sources, from Excel spreadsheets to databases), can connect to this data and can access various applications to facilitate their use.
- integrates with all the Microsoft applications already installed within the company, so that the company can make the most of its data to obtain a better global perspective on operation and performance, but also to obtain a detailed picture of the most relevant and important data from organizational structures (as it can generate dedicated reports for each of the company's areas).
- carry out comparative studies between periods, which helps managers to analyze what happened in the past and what is happening now, but also to predict what could happen in the future, in order to better understand their customers and for to be able to generate concrete actions that make the business sustainable and more competitive; in addition, it provides functionality that allows the identification of patterns, trends and relationships hidden in data.
- provides quick response to queries and visualizations in which the data is presented in an easy to understand way, which facilitates and improves the communication of information to different interested parties, in order to obtain significant information for decision-making and solving detected problems.
- allows the elimination of manual and repetitive tasks, facilitates the reuse of calculations and offers the possibility to constantly update the reports, thus saving time and resources for the team that generates the reports, while minimizing human errors.
- through a collaborative work environment hosted in the cloud, data can be accessed and managed in real time by thousands of users simultaneously, thus facilitating collaborative work.
- last but not least, the application is very accessible, because it integrates the Power BI Desktop component which is completely free (it can be downloaded for free from the official Microsoft website) and offers a large number of benefits: it allows users to create data models , design reports and publish them through the Power BI Service.

Power BI is composed of three main tools: Power BI Desktop, Power BI Service and Power BI Mobile, which together form a complex but, at the same time, easy-to-use platform for data analysis and visualization:

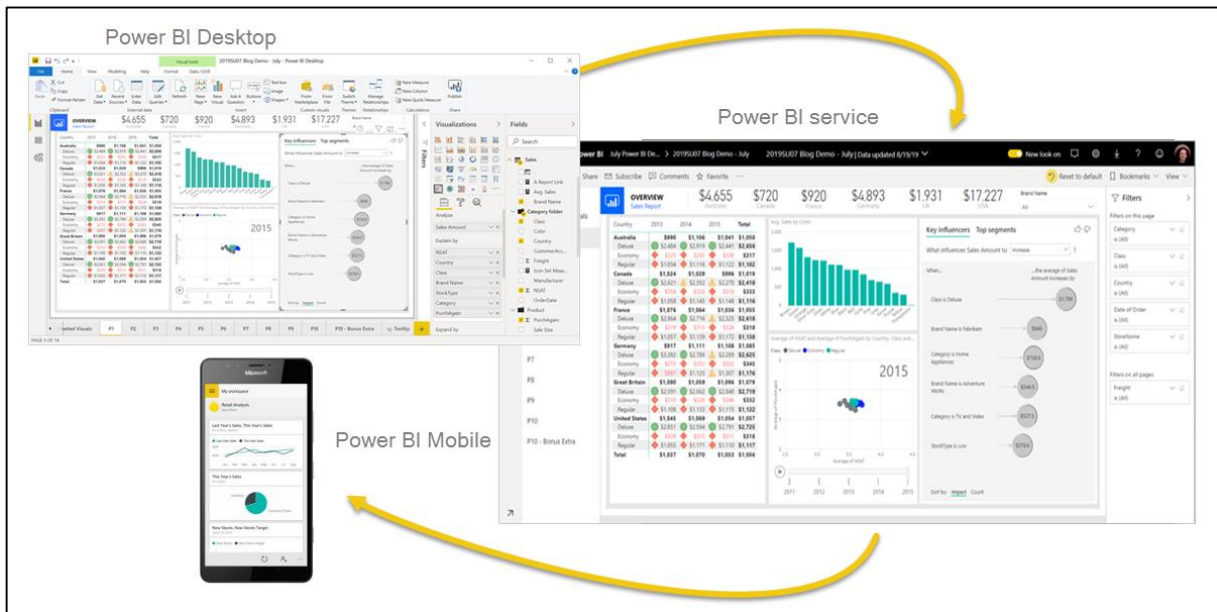


Figure 5.1: Key components of Power BI
[\[https://learn.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview\]](https://learn.microsoft.com/en-us/power-bi/fundamentals/power-bi-overview)

► **Power BI Desktop:** is a program that can be installed on the computer, which allows, through an intuitive and flexible interface, to import or connect to various data sources (text files, Excel, databases, web pages, cloud services, etc.) and create queries, reports, complex dashboards and custom visualizations using Power Query, Power Pivot and DAX (Data Analysis Expressions) language.

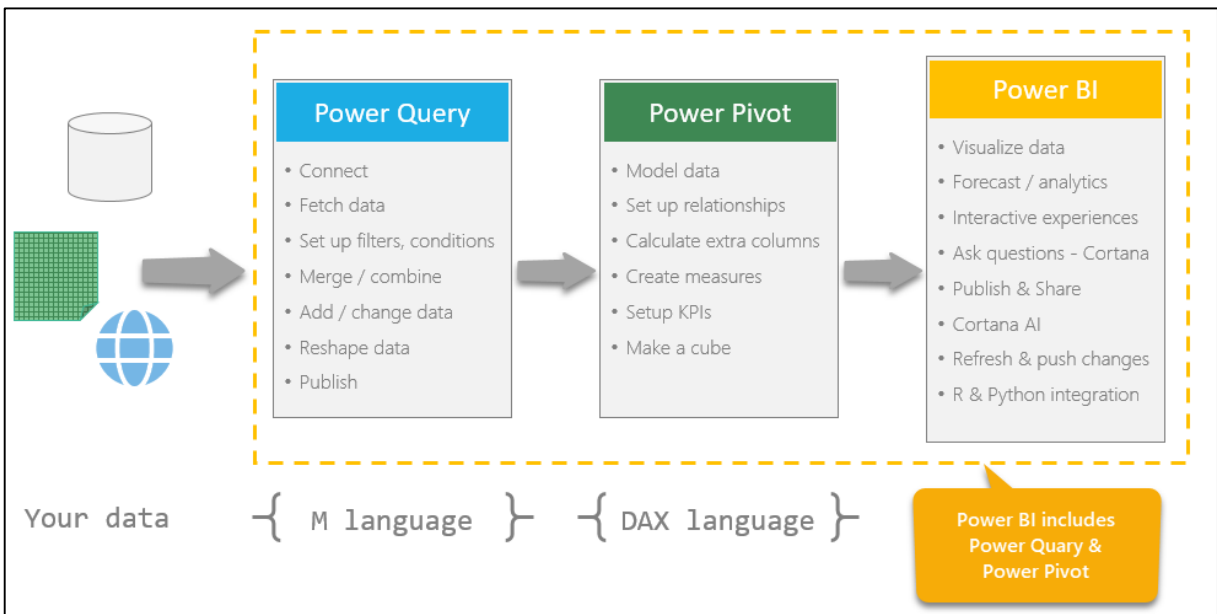


Figure 5.2: Power Query and Power Pivot – key components of Power BI
[\[https://community.fabric.microsoft.com/t5/Desktop\]](https://community.fabric.microsoft.com/t5/Desktop)

- **DAX** is a formula expression language: a set of over 200 functions, operators and constant values that can be used in expressions to perform specific data analysis tasks (advanced calculations) and can return one or more values.

The application includes two important tools for data processing, analysis and subsequent visualization, called Power Query and Power Pivot.

- **Power Query** allows the preparation and editing of data through transformations associated with ETL processes (data extraction, transformation and loading) to meet analysis standards. By connecting to the data source, the application extracts the database from that data source. Next comes the transformation of the data through various types of changes made to the database structure, and later the database is loaded into Power BI for future analyses.

The application thus automates the manual work of cleaning and pre-organizing information for future Business Intelligence analyses, offering:

- for regular users: an easy-to-use environment, without the need to use a programming language to create connections with several different data sources and to transform the data into the desired form;

- for advanced users: the possibility to use "M", Python and SQL programming languages for advanced data processing.

- **Power Pivot** is a data modeling tool that allows to efficiently create particular relational data model from multiple data sources within Power BI. In addition to ensuring the correct creation of the database, this tool is also responsible for the relationships established between the various tables that make up this data.

▶ **Power BI Service:** is an online option, hosted on Microsoft Azure cloud platform, through which users will be able to access their data from anywhere through the cloud. This web application facilitates information sharing and collaborative work on various reports and dashboards between multiple users.

▶ **Power BI Mobile:** is the version accessible from mobile devices with IOS, Android and Windows systems, through which users can access and view from anywhere and at any time the data, reports and dashboards created in the Desktop or Service version and can receive notifications and alerts based on data changes.

5.2 Key Features of Power BI Desktop

5.2.1 The Power BI Desktop user interface

Power BI Desktop has a simple and intuitive interface, with many elements similar to Microsoft Office programs:

▶ Most of the application's features are organized on the **Ribbon**, in thematic groups. Thus, at the top of the bar are command tabs that allow access to various command menus. The command tabs defined by default are File, Home, Insert, Modeling, View, Optimize and Help:

- **File:** when accessed, displays a drop-down menu that allows general operations in working with files, such as opening and saving Power BI Desktop files, importing data, exporting and publishing processed data, but also configuring options or managing settings for data sources.

Note: Power BI Desktop files have the .pbix file extension.

- **Home:** provides access to a wide list of operations, from the most common ones, such as copying and pasting, to obtaining data, inserting various visual elements, creating calculations and other actions to prepare and transform data.
- **Modeling:** provides access to many operations specific to the data modeling process, including the use of DAX expressions for creating calculations, sorting, formatting and classifying data columns, as well as operations related to the management of security roles, or the Q&A function.
- **Help:** includes useful links to get support in using Power BI, links to the Power BI Community site, various documentation, reports examples, guided learning and training videos for tasks and features in Power BI.
- **View:** offers operations that control the size of the page, formatting functions such as gridlines, aligning objects to the grid and locking objects, bookmarks, filters or the ability to activate additional panels.
- **Optimize:** offers access to tools for pausing and refreshing visuals, settings adapted to reporting needs with preset optimization settings or easy access to important tools for optimizing reporting performance.

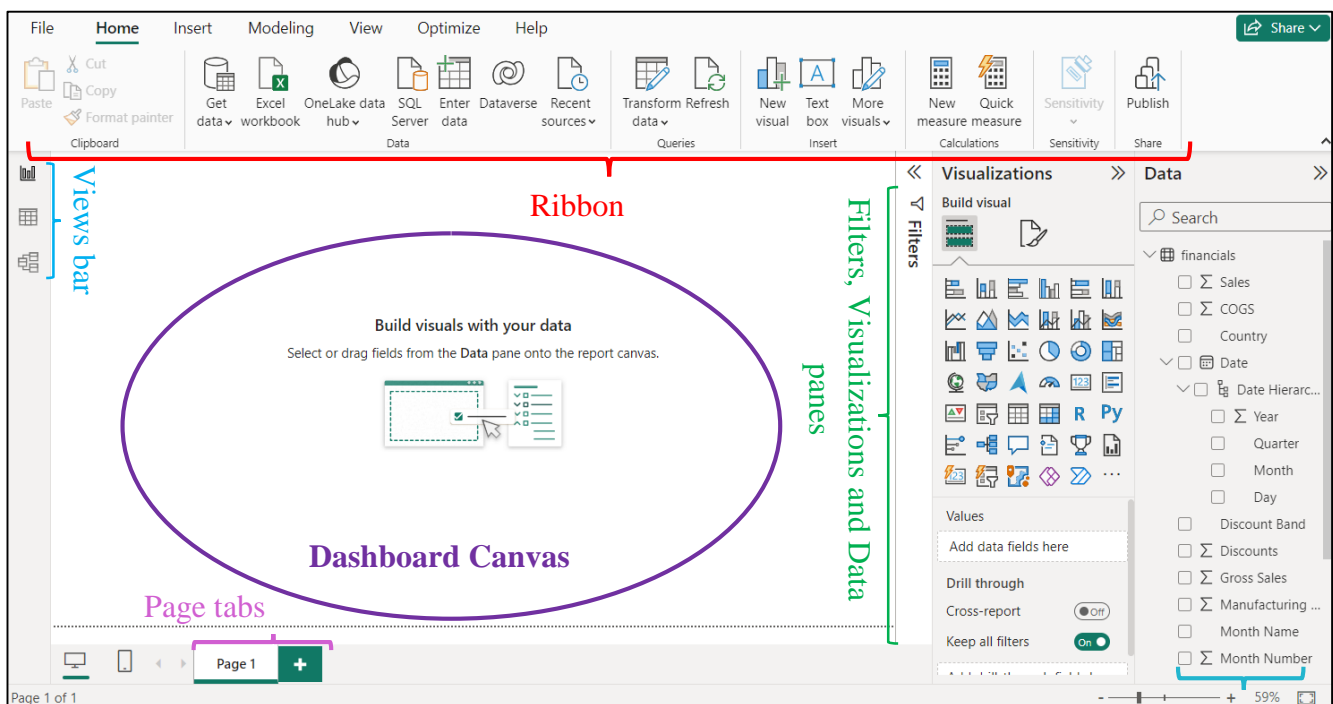


Figure: The Power BI Desktop home screen

Fields list

When a view is accessed from the view panel, two new tabs become active in the Ribbon:

- **Format:** contains formatting options for how visual elements interact with each other or are displayed/aligned relative to each other.
- **Data/Drill:** Provides access to drill actions or ways to explore data and see the raw data that makes up a visualization.

► In the **panes area** there are three panes that can be active, namely Filters, Visualizations and Data.

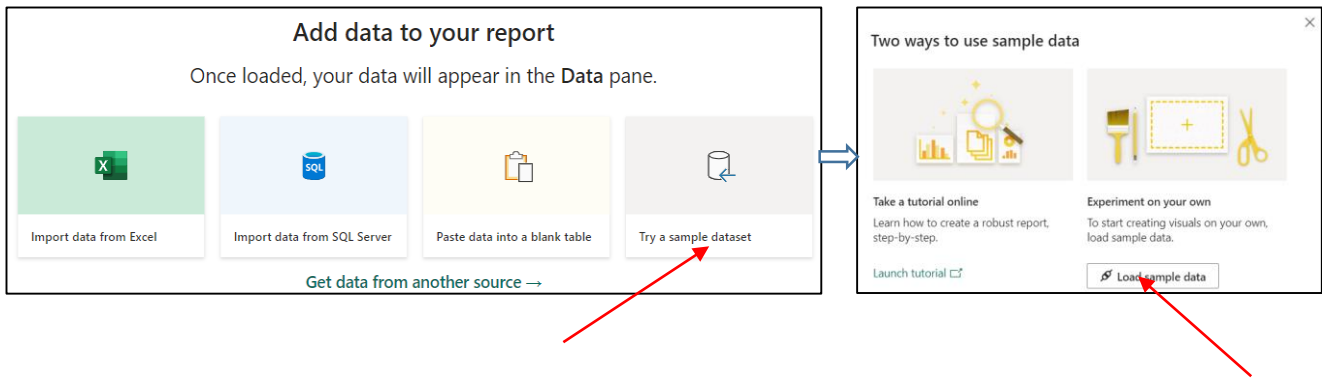
- The **Filters panel** is useful when you need to apply filters for reports; it displays and manages several types of filters active on reports, pages, and visualizations.
 - The **Data panel** is where the databases or tables that we bring into Power BI to create the interactive reports will be placed.
 - The **Visualizations panel** is where various visualization types reside, using various graphics and other resources, that can be added to a dashboard. In this panel, there is also the possibility to format the graphics, to configure the page size or to access other advanced functions.
- ▶ **Views bar** provides three different views of the data model. These views change what is displayed in the Canvas area, but contextually determine which tabs and operations are available in the Ribbon, as well as which Panels will be displayed.
- **Report view:** it is the main view, which allows the creation of views (reports with several pages, where each page can have several visual elements) to obtain information from the analyzed data.
 - **Table view:** the place where one can create new columns, bringing some specific calculations and also where you can view the content of the databases.
 - **Model view:** provides an overview of the tables contained in the data model and how these tables are related to each other.
- ▶ **Fields list:** shows all the fields in the source data that can be used to build various views.
 - ▶ **Page tabs:** is an area available in Report view; here one can generate new pages, rename pages and reorder pages in a report.
 - ▶ **Dashboard Canvas:** is the main area, where, in Report view, one can add visualizations and design dashboards.

In Table view, the data of a selected table in the data model is displayed here, and in Model view, all the tables in the data model and the relationships between them are displayed here.

5.2.2 Brief example of using the application by connecting to a data source and generating a report

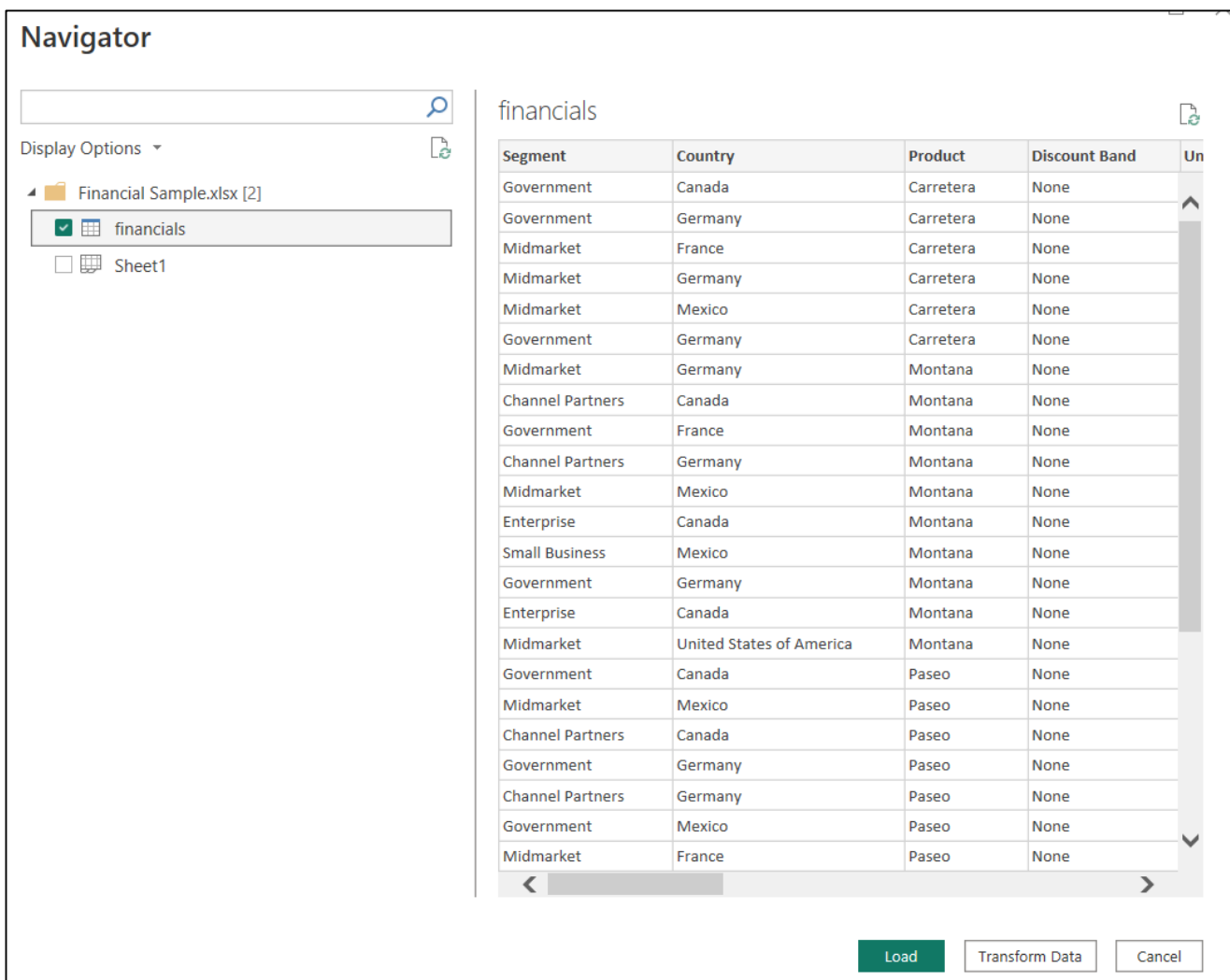
Power BI can connect to a wide range of data sources: Excel spreadsheets, databases (Access, SQL, Oracle, Azure), as well as web links that allow establishing a connection.

To generate visualizations and demonstrate the functionality of the Power BI Desktop application, a test database provided by Microsoft "financials.xlsx" can be used, which is available for download:



In the window that appears, the user has the option to load the data directly or go to the Power Query Editor using the "Transform data" option.

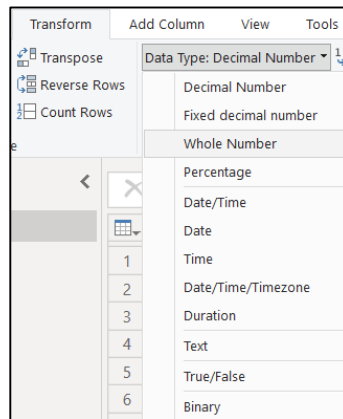
In this case, a separate window will open for the Power Query Editor, which allows the transformation of the data before loading it, in order to resolve possible errors or inconsistencies within the data.



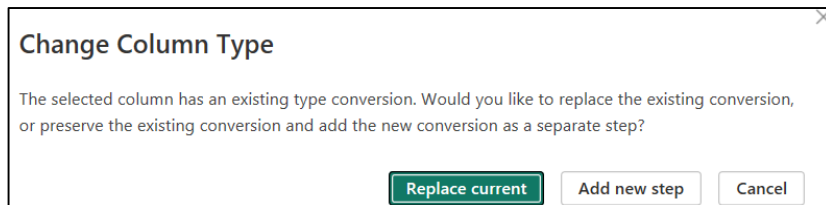
For this purpose, the following transformations can be applied, for example, to the following columns:

► For the "Sales" column, apply the data transformation in the whole numbers format:

1. Select Sales column → Transform tab → select Data Type → select Whole Number:

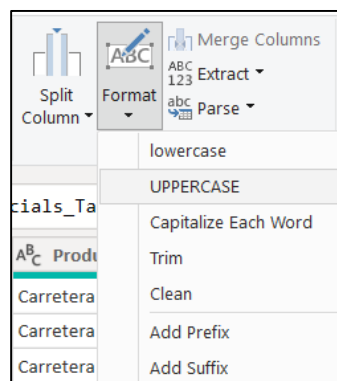


2. Choose **Replace current** to change the column type:



► For the "Country" column , apply the data transformation in the **UPPERCASE** format:

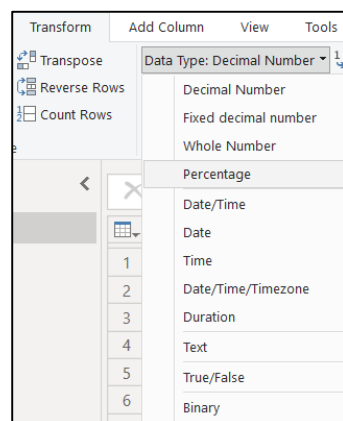
1. Select **Country** column → **Transform** tab → select **Format** → select **UPPERCASE**:



2. Choose **Replace current** to change the column type.

► For the "Discounts" column , apply the data transformation in the **Percentage** format:

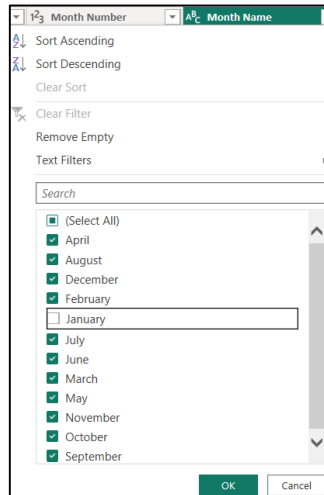
1. Select **Discounts** column → **Transform** tab → select **Format** → select **Percentage**:



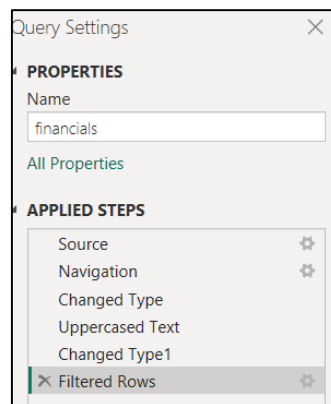
2. Choose **Replace current** to change the column type.

► Next, we assume that the company is interested in sales from all months of the year, less than January, so we want to filter this data from our report:

1. Select **Month name column** → **Drop-down menu** → **uncheck the box corresponding to the month of January** → **OK**:



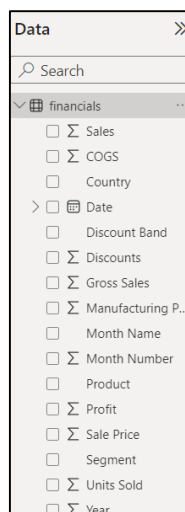
It can be seen that all these transformations have been added to the **APPLIED STEPS** section in area **Query Settings**:



To apply the pending changes, the Query Editor window must be closed:

1. Select **Home tab** → **Close & Apply**:

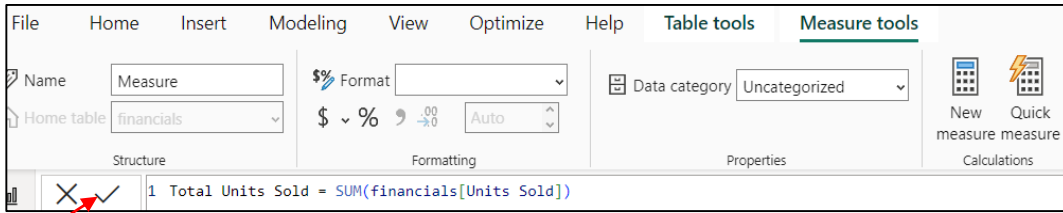
It can be seen that in the **Date pane**, the application recognized the numeric fields and marked them with the Σ symbol, and the date field marked it with a calendar symbol:



► In the next step, we assume that we have to write a basic expression in the DAX formula language to add all the values in the Units Sold column:

1. Select **Home tab** → **New measure**.

2. **Enter the formula:** Total Units Sold = **SUM(financials[Units Sold])**:



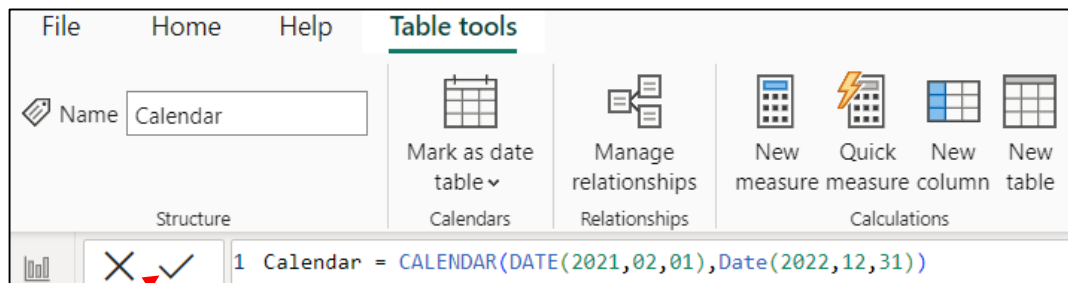
3. **Select the check box** to perform the action.

► In the next step, we want to generate a Calendar table containing all dates between February 1, 2021, and December 31, 2022.

1. In the **Views bar** → select **Table view**.

2. Select **Home tab** → **New table**.

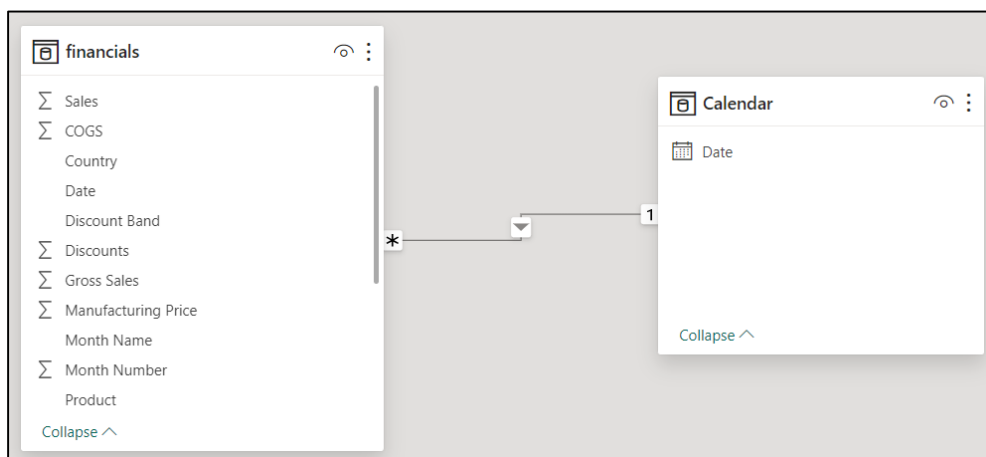
3. **Enter the formula:** Calendar = **CALENDAR(**DATE(2021,02,01),Date(2022,12,31))



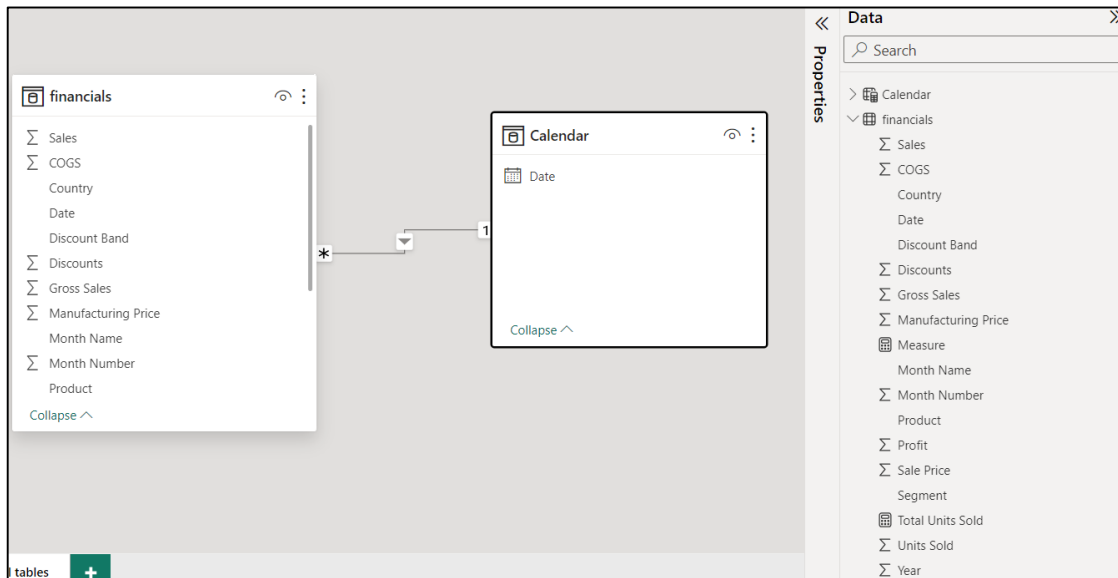
4. **Select the check box** to perform the action.

5. In the **Views bar** → select **Model view**.

6. **Drag the Date field from the Financial table over the Date field from the Calendar table** to create a relationship between them:

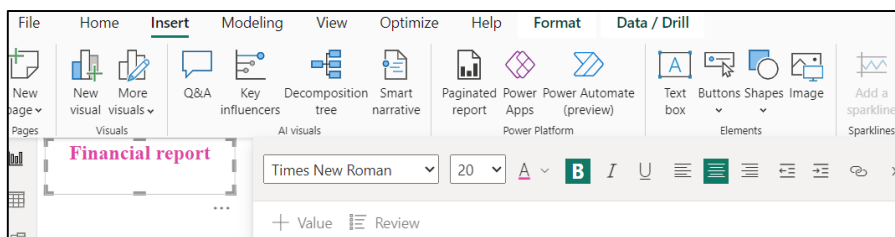


At this moment, **all the fields from the created data model have appeared in the Data pane** and one can proceed to generate a report. We will add visuals to the Power BI Desktop report, one visual at a time.



► Visual 1: Insert a title for the report

1. In the **Views bar** → select **Report view**.
2. Select **Insert tab** → **Text Box** → type "Financial report" → select the font type, text color, etc.



► Visual 2: Monthly profit

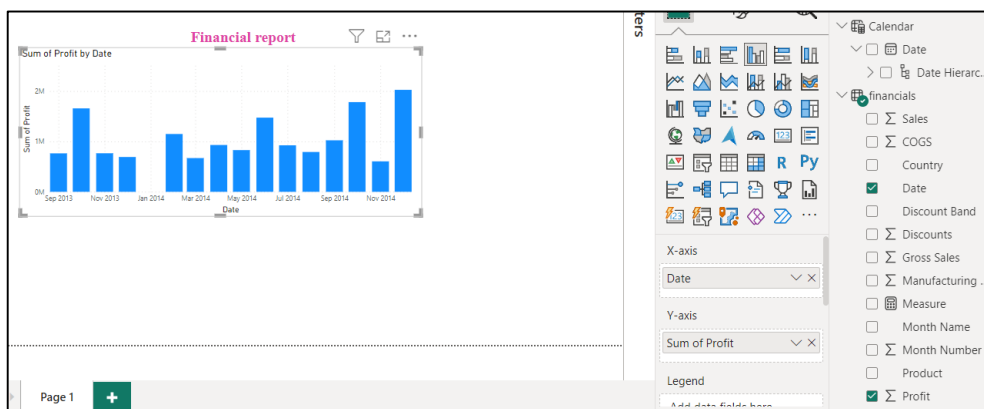
A bar graph will be created to see in which month and year the highest profit was recorded.

1. From the **Fields list** → bring the **Profit** field to an empty area of the **Dashboard Canvas**.

By default, the application displays a column chart with one column, Profit:

2. From the **Fields list** → the **Date** field is brought into the same image.

The application will update the column chart to show the profit over the two years:



Note: To change the graph type, choose a different graph type from the Visualizations pane, for example **Line chart**:

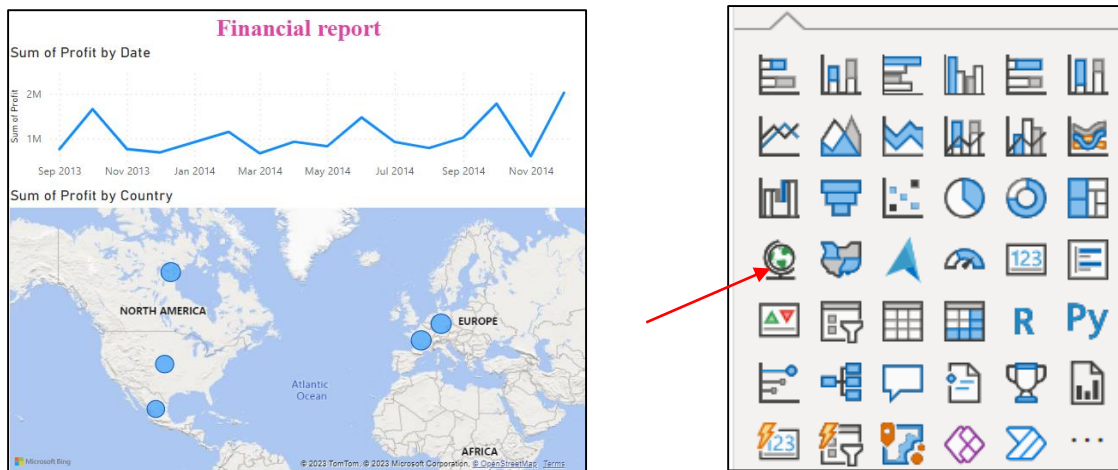


► **Visual 3: Country profit**

A map will be created to see which country recorded the highest profits.

1. From the **Visualizations pane** → choose the **Map graphic type**.
2. From the **Fields list** → the **Country** field is brought to an empty area of the **Dashboard Canvas** to create a map.
3. From the **Fields list** → the **Profit** field is added to the created map.

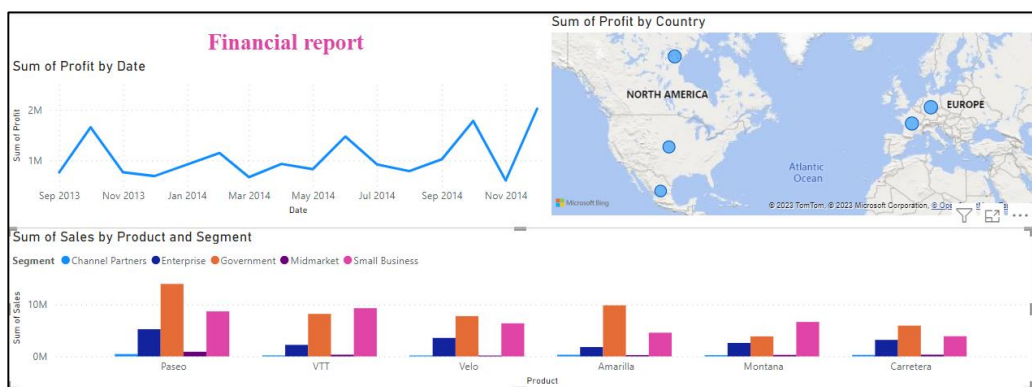
It can be seen that the application created a visual map with bubbles representing the relative profit of each location:



► **Visual 4: Product and Segment sales**

A bar chart will be created to see which companies and segments are worth investing in.

1. From the **Visualizations pane** → choose the **Column chart type**.
2. From the **Fields list** → the **Sales, Product, and Segment** fields are brought to an empty area:



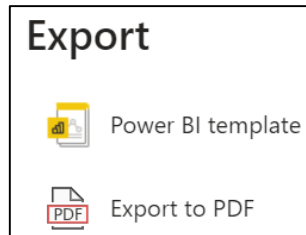
It can be seen that the application automatically created a chart with grouped columns.

► **Save the report:**

Select **File tab** → **Export** → **select the destination folder**.

► **Export the report:**

Select **File tab** → **Save:**



For sharing, the application allows exporting the report in two formats:

- as a Power BI template (.PBIT) - preferably if the report needs to be opened in the Power BI application on different devices. With the mention that the file with the extension .PBIT will store only the data schema of the report, not the real data, as in the case of saving the file in the .PBIX.
- as a PDF file - preferably if the report must be shared for viewing.

5.3 Using the Power BI Desktop Query Editor

To ensure processing performance and consistency of the data model created in the Power BI Desktop application, after loading the data into the model (extracting the database from the source where it is stored), the data transformation and preparation process follows. By transformation is meant any necessary change in the structure of the database to allow the use of data in other stages of processing with the Power BI Desktop application.

Among the most common transformation operations are:

- data preaggregation and filtering;
- removing unnecessary or empty rows and/or columns;
- elimination of redundant or erroneous data elements;
- modification of data types: although in the import stage, the Power BI application tries to automatically detect the data type, errors can creep in and the data types must always be checked for each data column;
- rationalization and standardization of data to make them easier to handle;
- combining data through transformations that have queries from several sources;

To perform these types of operations, the application has the Power Query component, which is a tool from the Microsoft product suite, dedicated to the data transformation and preparation process:

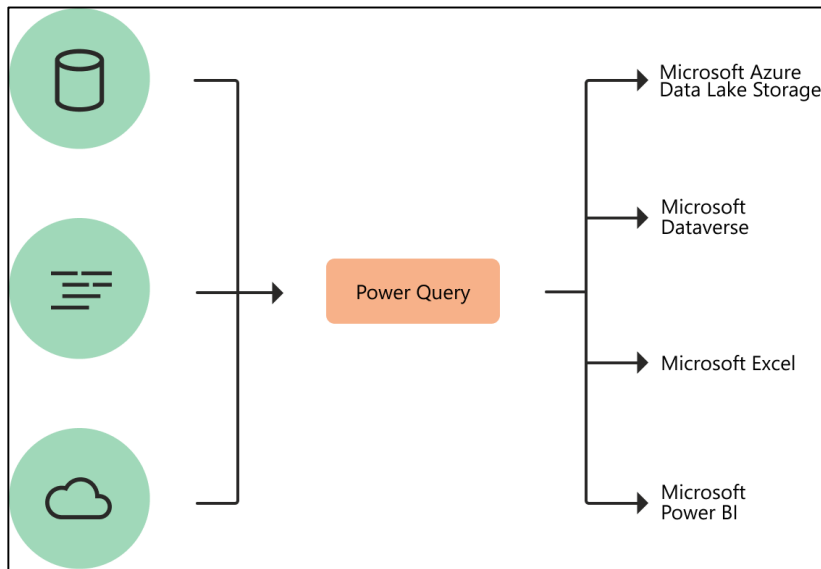
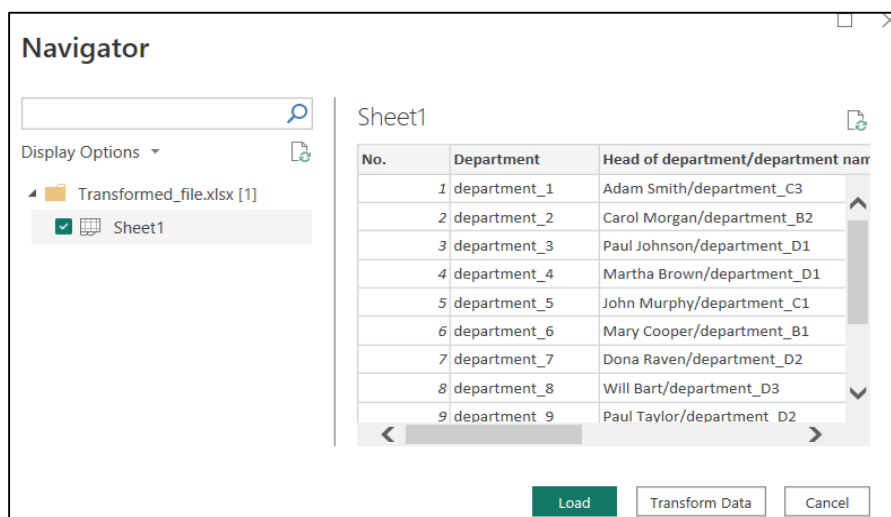


Figure 5.3: Power Query transformation

[<https://learn.microsoft.com/en-us/power-query/power-query-what-is-power-query>]

In the Power BI Desktop application, the data can be edited by applying the desired transformations:

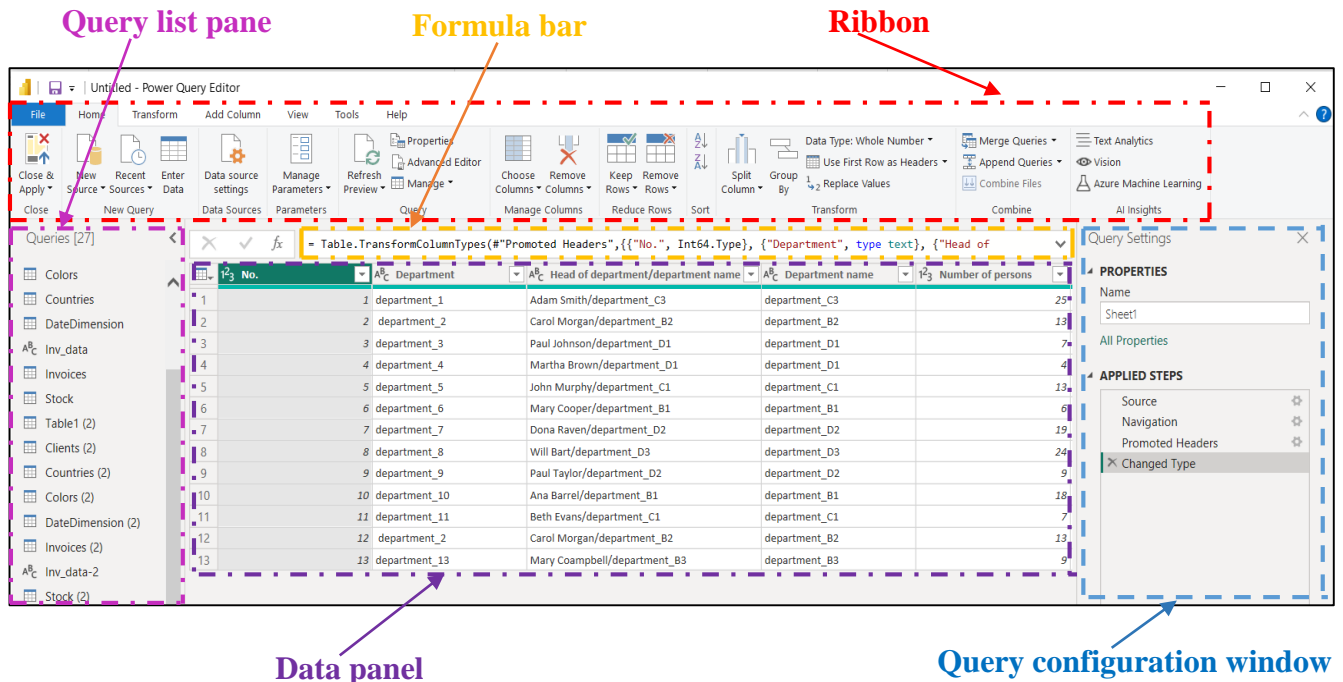
- either after loading the data, namely **in the Power BI Desktop Home ribbon** → **click the Transform Data button**;
- either in the data loading stage, if the **"Transform Data"** option is selected in the loading window:



In both cases, The Power BI Desktop Query Editor will open and display the source data as a table.

Through its graphical interface, Power Query provides users with an easy-to-use environment for obtaining data from various sources and applying the necessary transformations to bring it into the desired form. Advanced users can use "M", Python and SQL programming languages to develop script codes for advanced data processing.

The following figure shows the Power Query interface and its organization:



► **The Ribbon** with the command tabs that allow access to specific command menus. The command tabs defined by default are Home, Transform, Add Column, View and Tools. And when performing other specific types of data transformations, other command tabs will be available.

The collection of tools related to each tab in the Ribbon, which can be used to perform various data transformations or to load data into the working model are explained in [Aspin, 2020] and can be found in Appendix 2.

- **The data panel** that displays the data from the selected query.
- **The Query list pane** where the available queries for the required transformations are listed, containing all the queries that have been added to a Power BI Desktop file.
- **The Query configuration window** which displays the properties of the selected query and all changes made to the data model, step by step.
- **The formula bar** (located above the data panel) which shows the code written in the Power BI "M" language through which the respective transformation operation is carried out.

5.3.1 Power Query detailed views and transformations through contextual menus

When the data model contains a table with many fields, in order to avoid going through each column of the table to view the content of a record, the Power BI Desktop Query Editor offers as a solution to view the entire content of the selected record (by clicking on the row of the record) in a window visible under the data set in the table. Moreover, to make the content visible, the size of the window can be changed by dragging the dividing line:

	ReportingYear	ReportingMonth	Registration_Date	VehicleType	InvoiceNumb
10	2012	9	08/05/1993	Coupe	CFC6726D-152
11	2012	9	06/05/1975	Coupe	CFC6726D-152
12	2012	11	15/01/1985	Coupe	C4A55876-389
13	2012	11	29/03/1979	Coupe	4DFCF7EF-C85
14	2012	12	08/09/2004	Coupe	B47CA156-707
15	2012	12	04/06/1994	Coupe	7DBAF8FB-E34
16	2013	1	08/09/2006	Saloon	4799184A-499
17	2013	2	01/09/2000	Saloon	2AE72D9C-552
18	2013	3	10/05/1997	Saloon	5963EAA5-4F0
19	2013	4	01/09/2001	Saloon	C6ECD08D-53C
20	2013	5	08/06/2005	Saloon	EA924C34-F8F
21	2013	6	01/01/1999	Saloon	A44A6460-1BE
22	2013	7	05/05/2001	Saloon	00B971F3-33D
23					

InvoiceDate	04/12/2012 00:00:00
Make	Aston Martin
CountryName	United Kingdom
IsDealer	NULL
SalePrice	95000
CostPrice	155000
TotalDiscount	5000
DeliveryCharge	1500
SpareParts	600

Regarding performing data transformations, in addition to the dedicated options that are organized on the Ribbon in thematic groups, the application offers three types of useful contextual menus:

- The **Table contextual menu**: is available when clicking on the top left corner of the data table.
- The **Column contextual menu**: is available when right-clicking on the title of a column.
- The **Cell contextual menu**: is available when right-clicking inside a data cell.

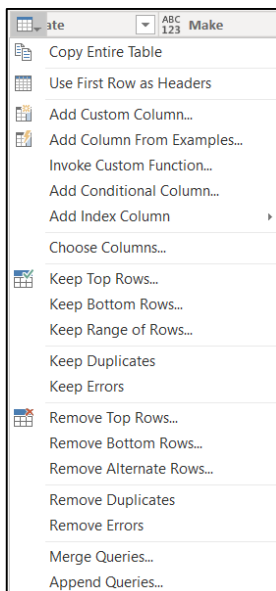
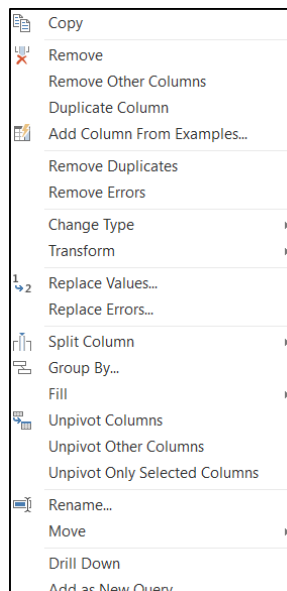
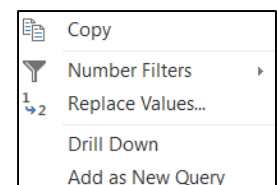


Table contextual menu



Column contextual menu



Cell contextual menu

5.3.2 Data cleaning and transformation operations in Power Query

The Power Query Editor allows to apply some basic techniques to model the initial data set. The data uploaded to Power BI Desktop in the following examples represents some Excel workbooks available for download or connection at github.com/Apress/pro-power-bi-desk/tree/master/Samples and chandoo.org/wp/power-query-tutorial.

■ **Removing unnecessary spaces and non-printing characters:**

To avoid processing errors, when using data containing leading, trailing spaces or non-printable characters, the following functions should be used:

- **Trim**, which removes any leading or trailing whitespace characters from text.
- **Clean**, which removes all non-printable characters from selected columns.

Select **Home** tab → **Right click on the column name** → **Transform** → select **Trim** or **Clean**.

■ **Adding a Prefix or a Suffix:**

To add a prefix or suffix to the data in a column:

Select **Transform** tab → **Click on the column name** → **Format** (section **Text Column**) → select **Add Prefix** or **Add Suffix**.

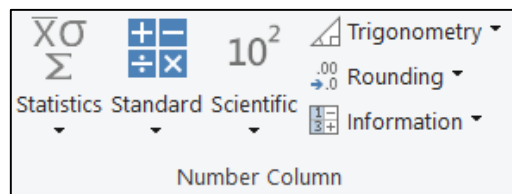
■ **Text formatting by converting to lowercase, uppercase or converting the first letter of each word to a capital:**

Select **Transform** tab → **Click on the column name** → **Format** (section **Text Column**) → select **lowercase**, **UPPERCASE** or **Capitalize Each Word**.

■ **Transformations applied to numerical data:**

The content of the number columns can be modified by accessing one of the corresponding transformations:

Select **Transform** tab → **Click on the name of the numeric column** → **select a numeric transformation** (section **Number Column**):



- The **Rounding** drop-down box has the following three options:

- **Round:** round each number in the column up to a specified number of decimal places.
- **Round Up:** round each number in the column up to the nearest whole number.
- **Round Down:** round each number in the column down to the nearest whole number.

- The **Trigonometry** drop-down box has the following options: **Sine**, **Cosine**, **Tangent**, **ArcSine**, **ArcCosine**, **ArcTangent** and returns the value obtained by applying the respective trigonometric function.

- The **Information** drop-down box has the following three options:

- **Is Even:** returns TRUE if the values in the column are even, FALSE otherwise.

- **Is Odd:** returns TRUE if the values in the column are odd, FALSE otherwise.
- **Sign:** returns 1 if the values in the column are positive, -1 if they are negative, 0 otherwise.

- The **Scientific** drop-down box has the following options:

- **Absolute Value:** returns the absolute value of the numbers in the column (positive values).
- **Power:**
 - **Cube:** returns the cube of the numbers in the column.
 - **Square:** returns the square of the numbers in the column.
 - **Power:** raises the numbers in the column to a specific power.
- **Square Root:** returns the square root of the numbers in the column.
- **Exponent:** returns the exponent of the numbers in the column.
- **Logarithm:**
 - **Natural:** returns the natural logarithm of the numbers in the column.
 - **Base-10:** returns the base-10 logarithm of the numbers in the column.
- **Factorial:** returns the factorial of the numbers in the column.

- The **Statistics** drop-down box has the following options: **Sum, Minimum, Maximum, Median, Average, Standard Deviation, Count Values** and **Count Distinct Values** and returns the value obtained by applying the respective statistical calculation for all the values in the column.

■ **Calculations performed with numerical data:**

The application can automatically apply various simple arithmetic operations to the numbers in a column, which can be found in the **Standard** drop-down box.

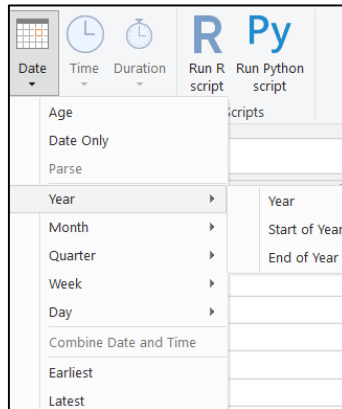
The **Standard** drop-down box has the following options:

- **Add:** Adds a certain user-defined value to the numbers in a column.
- **Multiply:** Multiplies the numbers in a column by a certain user-defined value.
- **Subtract:** Subtracts a certain user-defined value from the numbers in a column.
- **Divide:** Divides the numbers in a column to a certain value defined by the user.
- **Integer-Divide:** Same as the Divide operation, but returns only the whole part of the result.
- **Modulo:** Same as the Divide operation, but only returns the remainder (modulo division).
- **Percentage:** Applies the respective user-defined percentage to the column.
- **Percent Of:** Expresses the value in the column as a percentage of the user-defined value.

■ **Transformations applied to the time data columns:**

The content of the time data columns can be modified by accessing one of the corresponding transformations:

Select **Transform** tab → **Click on the name of the numeric column** → **select a Date/Time/Duration transformation** (section **Date&Time Column**):



- The **Date** drop-down box has the following options:

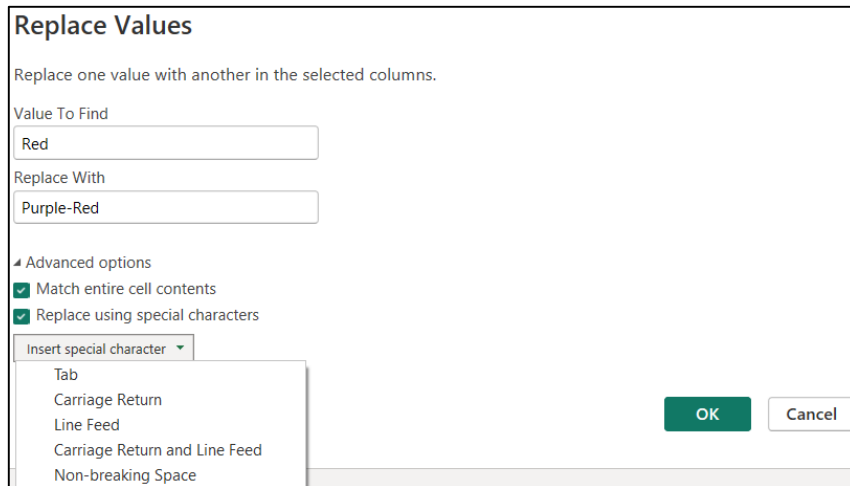
- **Age:** calculates the difference between the existing date in the cell and the current local time, in days and hours.
- **Date:** for a value defined as a Date/Time value, it will return only the Date part of that value.
- **Year:**
 - **Year:** extracts only the year part from the data in the column.
 - **Start of Year:** returns the first day of the year for each date in the column
 - **End of Year:** returns the last day of the year for each date in the column
- **Month:**
 - **Month:** returns the number of the month corresponding to that date.
 - **Start of Month:** returns the first day of the month corresponding to that date.
 - **End of Month:** returns the last day of the month corresponding to that date.
 - **Days in Month:** returns the number of days in the month corresponding to that date.
 - **Name of Month:** Returns the name of the month for the given date.
- **Quarter:**
 - **Quarter of Year:** returns the calendar quarter of the year corresponding to that date.
 - **Start of Quarter:** returns the first date in the quarter of the year corresponding to that date.
 - **End of Quarter:** returns the last date in the quarter of the year corresponding to that date.
- **Week:**
 - **Week of Year:** returns the week number of the year corresponding to that date.
 - **Week of Month:** returns the week number of the month corresponding to that date.
 - **Start of Week:** returns the first date of the week (Sunday) corresponding to that date.
 - **End of Week:** returns the last date in the week corresponding to that date.
- **Day:**
 - **Day:** returns the day number of the date.
 - **Day of Week:** returns the weekday as a number, starting from Monday = 1, Tuesday = 2, etc.

- **Day of Year:** returns the day number of the year for the date.
 - **Start of Day:** returns the start of the day corresponding to the Date/time values in that column.
 - **End of Day:** returns the end of the day corresponding to the Date/time values in that column.
 - **Earliest:** returns the earliest Date value in the selected column
 - **Latest:** returns the latest Date value in the selected column
- The **Time** drop-down box contains transformations that can only be applied to columns of the date/time or time data types and has the following options:
- **Time Only:** returns the Time part of any Date/Time value in the column.
 - **Hour:**
 - **Hour:** returns the hour from a date/time or date value.
 - **Start of Hour:** returns the start of the hour from a time value.
 - **End of Hour:** Returns the end of the hour from a time value.
 - **Minute:** returns the minute number of the time.
 - **Second:** returns the second number of the time.
 - **Earliest:** returns the earliest time value in the selected column.
 - **Latest:** returns the latest time value in the selected column.
- The **Duration** drop-down box has the following options:
- **Days:** returns the days component of a duration value.
 - **Hours:** returns the hours component of a duration value.
 - **Minutes:** returns the minutes component of a duration value.
 - **Seconds:** returns the seconds component of a duration value.
 - **Total Days:** returns the total number of days in the duration.
 - **Total Hours:** returns the total number of hours in the duration.
 - **Total Minutes:** returns the total number of minutes in the duration.
 - **Total Seconds:** returns the total number of seconds in the duration.
 - **Multiply:** multiplies the duration value by a value defined by the user.
 - **Divide:** divides the duration value by a value defined by the user.
 - **Statistics:**
 - **Sum:** returns the sum of all the duration values in the column.
 - **Minimum:** returns the minimum of all the duration values in the column.
 - **Maximum:** returns the maximum of all the duration values in the column.
 - **Median:** returns the median value for all the duration values in the column.
 - **Average:** returns the average for all the duration values in the column.

■ **Replacing values:**

In columns containing text data, certain text elements can be replaced with other values, as in the case of an editing operation in Word:

1. Select **Home** tab → **Click on the column header** → **Replace values** (section **Transform**):



2. The value to be replaced and the replacement value are entered respectively in the two fields.

3. When expanding **Advanced options**, do one or more of the following:

- select **Match entire cell contents** to replace the whole cell value;
- select **Insert special characters** to replace the search value with a nonprinting character: **Tab**, **Carriage Return**, **Line Feed**, **Carriage Return and Line Feed**, and **Non-breaking Space**.

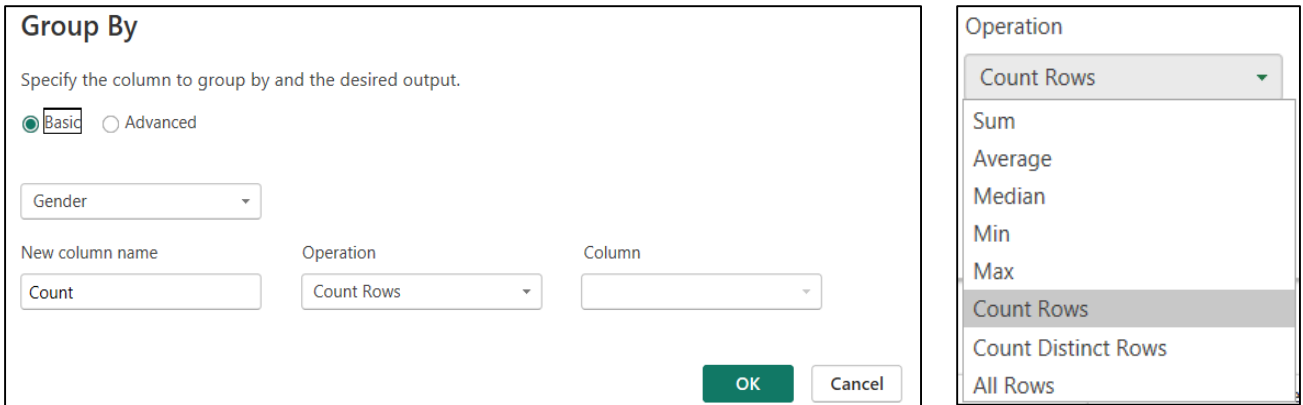
■ **Group by:** allows the grouping of attributes found on several rows into a single element, grouping the rows according to the values in one or more columns.

For example, in the data model we want to group employees according to their gender, male/female:

1. Select **Transform** tab → select the **Gender** column → **Group by**:

	ABC 123 Name	ABC 123 Gender	ABC 123 Department	ABC 123 Salary	ABC 123 Location	ABC 123 Start Date
1	Ab Lehrian	Male	NULL	82240.77	3 Redmond Way Bellevue, WA USA	04/03/2020
2	Abbie Tann	Female	Business Development	116518.12	3 Redmond Way Bellevue, WA USA	24/05/2019
3	Abe Gayter	Male	Training	null	3 Redmond Way Bellevue, WA USA	03/05/2019
4	Abigael Basire	Male	Engineering	61624.77	8 Parliament Lane - Wellington, NZ	19/08/2019
5	Abigael Basire	Male	Engineering	61624.77	3 Redmond Way Bellevue, WA USA	21/05/2020
6	Abramo Labbez	Female	Research and Development	76998.38	3 Redmond Way Bellevue, WA USA	27/07/2019
7	Abran Danielsky	Female	Engineering	32716.22	1 Infinite Loop, Los Angels, CA, USA	17/05/2020
8	Addi Studdeard	Female	Product Management	72502.61	3 Redmond Way	11/07/2020

2. In the **Group By** window that opens:



- In the **New column name** box, a name for the newly created column must be entered;
 - In the **Operation** drop-down list, the aggregation function used for grouping the values must be chosen.
- One of the following functions can be selected as an aggregation function:

Sum	returns the sum of values (total) in the column selected
Average	returns the average of values in the column selected
Median	returns the median of values in the column selected
Min	returns the minimum of the values in the column selected
Max	returns the maximum of the values in the column selected
Count Rows	returns a count of the number of records
Count Distinct Rows	returns a count of the number of unique records
All Rows	returns a value of type Table of records for each grouped element

3. The settings made will return a table with two columns: a column called **Gender** that contains all distinct values in the Gender column in the dataset, and a column called **Count** that contains the number of rows in the original table that are associated with each value in the Gender field:

	ABC 123 Gender	123 Count
1	Male	515
2	Female	487
3	null	44

In the following database, a complex grouping can be made, for example to find out from each brand, depending on the type of model, how many cars are in the database and what was the maximum selling price for a car of the type respectively

1. Select **Transform** tab → **Group by**.
2. The following settings are made in the **Group By** window that opens:

Group By

Specify the columns to group by and one or more outputs.

Basic Advanced

Make

Model

New column name Operation Column

Count Count Rows

Max sales price Max SalePrice

3. The settings performed will return a table with two additional columns: a column called **Count** that contains for each car brand, how many cars are in the database depending on the type of model and a column called **Max sales price** that shows the maximum price with such a car was sold:

	1.1 Make	1.2 Model	1.3 Count	1.2 Max sales price
1	Aston Martin	DB4	8	112750
2	Aston Martin	Zagato	2	181250
3	Aston Martin	DB9	54	181250
4	Aston Martin	Vantage	12	97750
5	Aston Martin	DB7	16	127750
6	Aston Martin	DBS	4	122750
7	Aston Martin	Rapide	4	132750
8	Aston Martin	Vanquish	10	181250
9	Bentley	Arnage	2	46750
10	Bentley	Azure	8	112750
11	Bentley	Turbo R	10	112750
12	Bentley	Continental	51	112750
13	Jaguar	XK	95	122750
14	Jaguar	XJ12	8	112750
15	Jaguar	XJ6	26	112750
16	MGB	GT	36	42250
17	Rolls Royce	Phantom	2	181250
18	Rolls Royce	Silver Seraph	4	181250
19	Rolls Royce	Silver Ghost	12	132750
20	Rolls Royce	Silver Shadow	4	181250
21	Rolls Royce	Wraith	2	181250
22	Rolls Royce	Camargue	39	181250
23	TVR	Tuscan	10	44000
24	TVR	Cerbera	4	32500
25	Triumph	TR7	4	28000
26	Triumph	TR5	8	28000
27	Triumph	TR4	22	28000

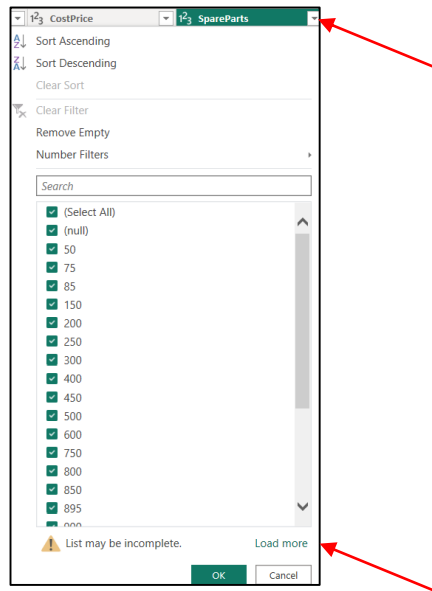
■ **Filter data to select values which meet specific criteria:**

Once the source data is loaded, there are situations where not all data is needed, and certain records must be excluded. For example, only certain values must be selected from a list of elements of a column, or a range of data must be specified that must be kept or must be excluded. Consequently, a filtering of the rows will have to be carried out.

In Power Query there are several possibilities to perform filters:

► **Filtering rows using the Auto-Filter option:**

1. **Choose the column** to be filtered → **click on the down arrow icon** located next to the column name:



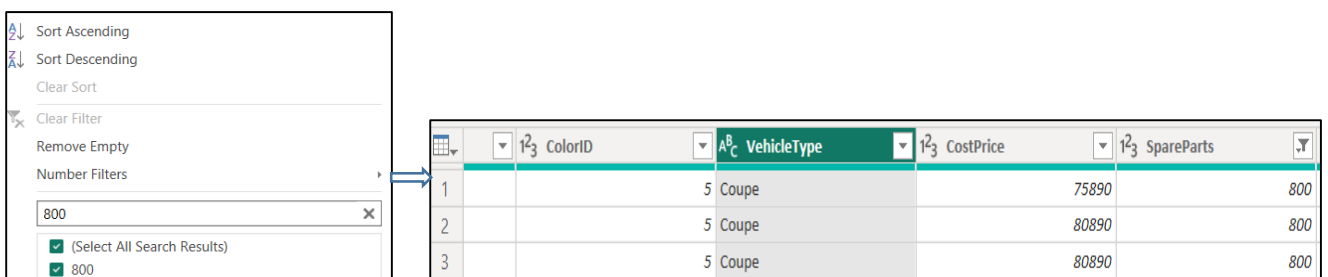
In the **Auto-Filter** menu that opens, the first 1,000 distinct values from that column are displayed; in case the column contains more than 1,000 distinct values, the **Load more** button appears which, if selected, allows loading another 1,000 distinct values.

2. **Check all the elements that must be kept and uncheck all the elements that must be removed.** Unchecking an element meant that any row containing that element for the selected column would be excluded from the data model.

3. If you uncheck the first option, (**Select All**), which is checked by default, then all the elements in the filtered column will be deselected.

4. If the **Remove empty** option is selected, then all records that do not contain a value in the filtered column will be removed.

5. In the **Search box**, one can search for certain values. Only records matching the search value will be displayed:

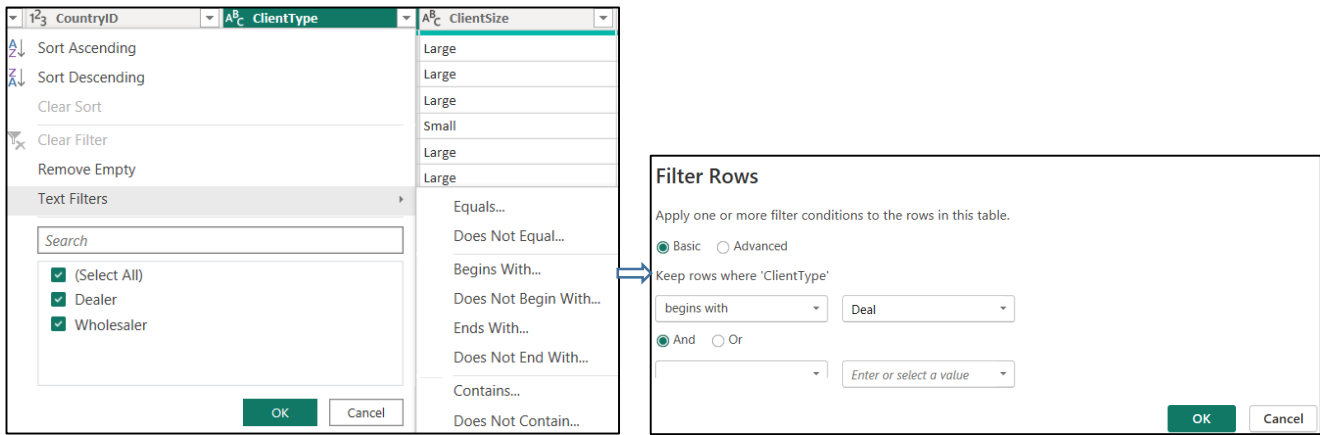


► **Filtering text content:**

For a column containing text, the following types of filters can be applied, which can be found in the **Text Filters** menu below.

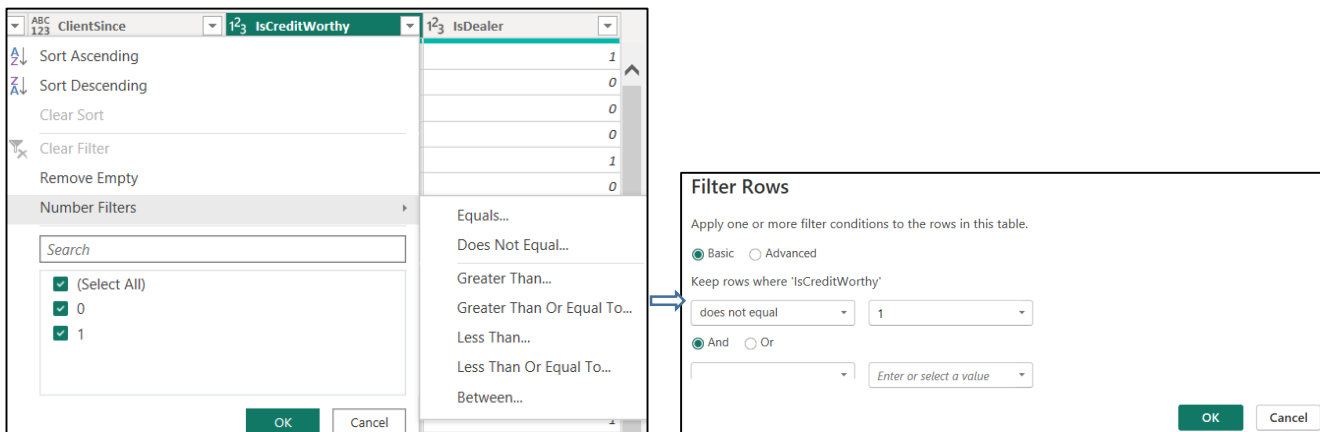
Rows in a table can be filtered by text that is equal or not equal to a text value, by text that starts with/does not start with a text value or ends/does not end with a text value, or by text that contains/does not contain a text value.

Note: The filtered text is not case sensitive.



► Filtering number content:

For a column that contains numerical data, the following types of filters can be applied, which can be found in the **Number Filters** menu:



The specific options that can be applied for filtering in this case are: if a value in the selected column is equal to/is not equal to, is greater than/greater than or equal to, is less than/is less than or equal to a certain value.

One can also filter by value ranges.

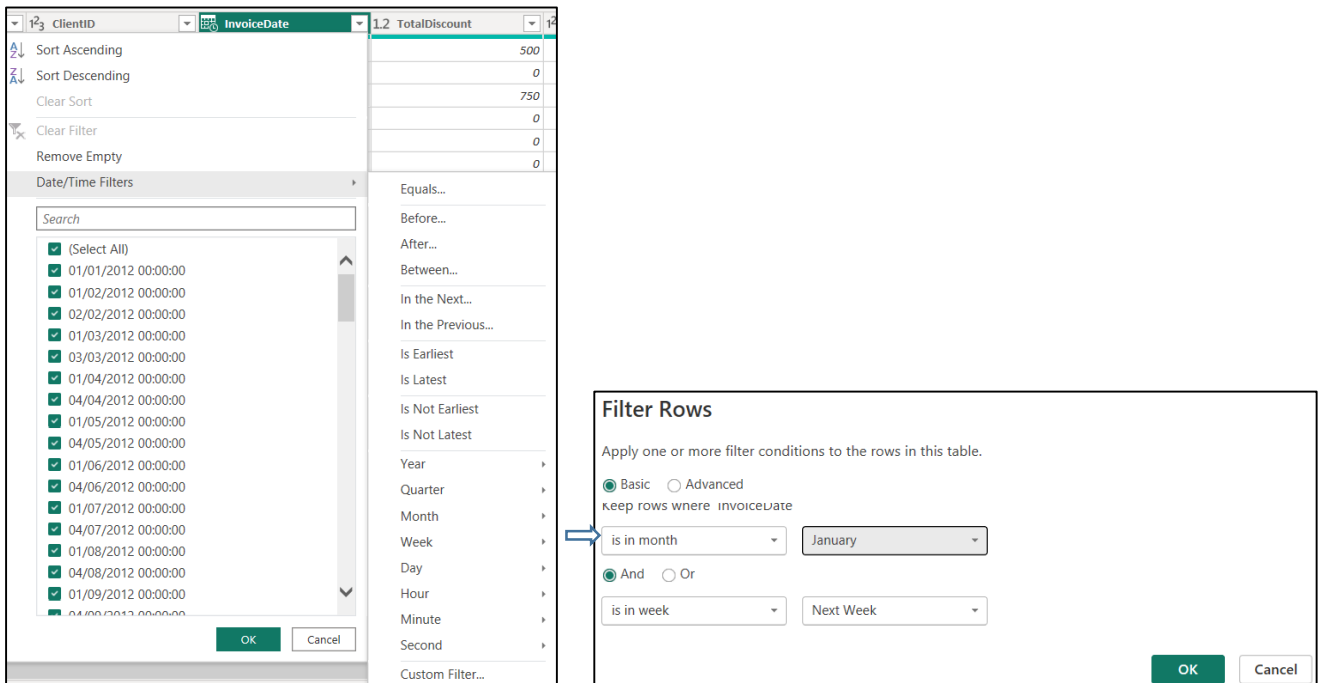
► Filtering date and time content:

For a column with content of the date and time type, several types of filters can be applied, which can be found in the **Date/Time Filters** menu.

Thus, rows can be filtered based on whether the date in that row is equal to a given date, is a date before or after a given date, or based on whether the date is in one of several predefined date ranges, such as the current week or the current month, the previous week, or the previous month, etc.

These date ranges are based on a standard calendar that cannot be customized.

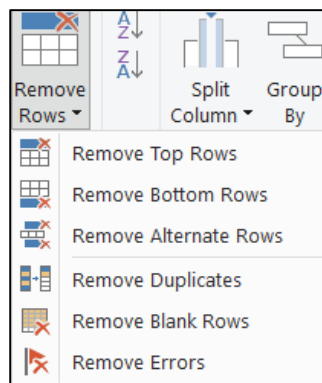
If the **Custom Filter...** option is selected, the **Filter Rows** dialog box will open, where more complex date filters can be created:



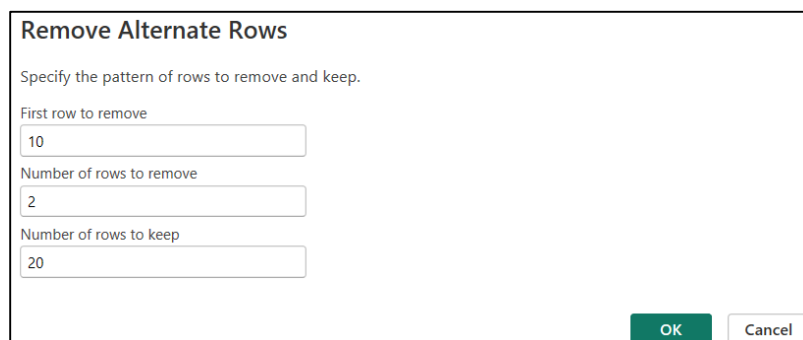
► **Filtering rows by range:**

To filter the rows according to their position in the table, there are several options available by accessing the menu:

1. Select **Home** tab → **Remove Rows** (section **Reduce Rows**):



- **Remove Top Rows/ Remove Bottom Rows:** allows removing "n" rows from the top or bottom.
- **Remove Alternate Rows:** allows to remove an alternating number of rows. A subset of data is thus generated, an operation that can be useful in the implementation of sampling:



- **Remove Duplicates:** allows removing duplicates if the data table contains duplicates, where some fields contains two or more identical records:

ClientID	ClientName	Address1	Address2	Town	County
1	Aldo Motors	4, Scale Street		Uttoxeter	Staffs
2	Honest John	99a Baker Street	NULL	London	NULL
3	Bright Orange	17, Arcadia Way	NULL	Birmingham	NULL
4	Cut'n'Shut	Grange Avenue	NULL	Manchester	NULL
5	Wheels'R'Us	Buckingham Drive	NULL	London	NULL
6	Les Arnaqueurs	33, Rue Des Bleus	NULL	Paris	NULL
7	Crippen & Co	1012 Princess Street	NULL	Glasgow	NULL
8	Rocky Riding	5205 108th Ave	NULL	New York	New York
9	Voitures Diplomatiques S.A.	NULL	NULL	Geneva	NULL
10	Karz	NULL	NULL	Stuttgart	NULL
11	Bright Orange	17, Arcadia Way	NULL	Birmingham	NULL
12	Costa Del Speed	NULL	NULL	Madrid	NULL

Note: When checking for duplicates, Power Query is case sensitive.

- **Remove Errors:**

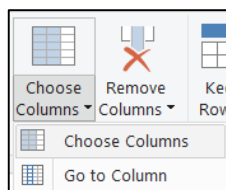
Error records can appear in columns for multiple reasons: when the columns are converted to other data types or due to some syntax errors, etc.

i) **To remove errors from several columns simultaneously:**

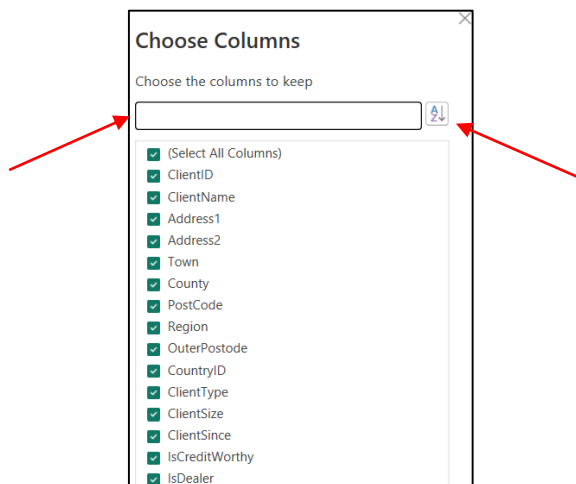
Ctrl + Click on the headings of the columns containing the errors → **click on the down arrow icon next to the column name** → **Remove Errors**.

Notes: In the case of a table with a lot of columns, in order to more easily select a lot of columns to which a transformation should be applied, proceed as follows:

1. Select **Home** tab → **Choose Columns** (section **Manage Columns**) → **Choose Columns:**



2. From the **Choose Columns** window that opens → **Click (Select All Columns)** to deselect all the columns in the dataset → **Select the desired columns:**



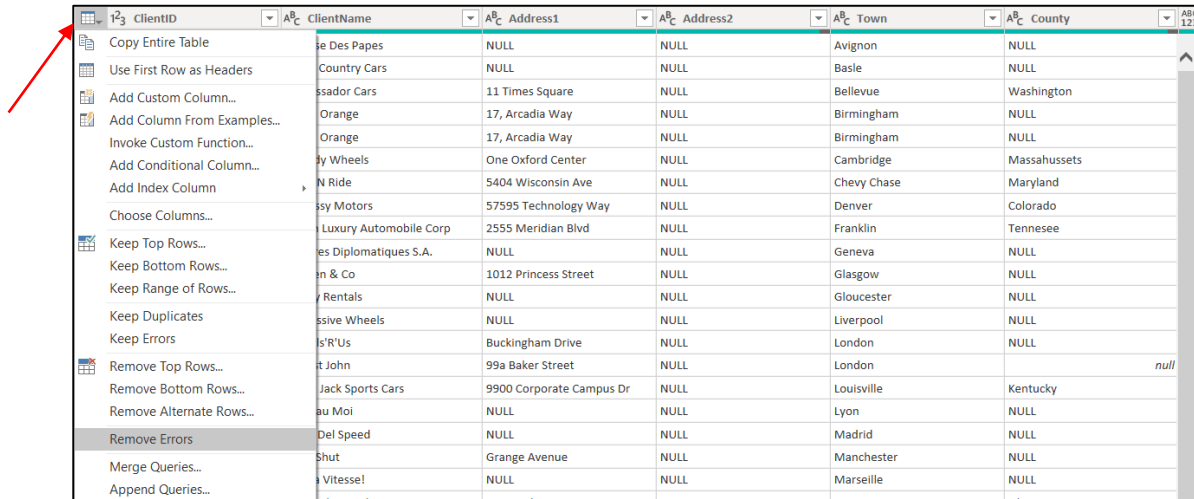
Note: The **Choose Columns** window contains some additional functions:

- by clicking on the icon at the top right of the window, one can sort the list of columns in alphabetical order.

- the list of columns that is displayed can be filtered if a few characters are entered in the search field **Choose the columns to keep** and then pressing the **Enter** key.

ii) To remove errors from the entire table:

Click on the table icon → Remove Errors:

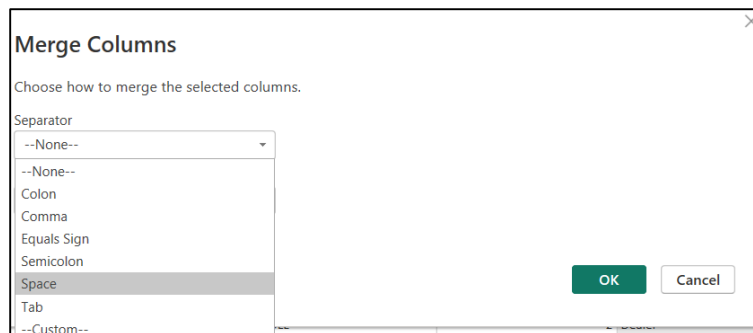


■ Merging text data columns:

The application allows combining data from several columns into a single column, performing the following steps:

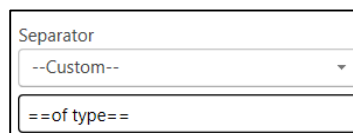
1. **Ctrl-click the headers of the columns** that must be merged (any columns can be selected).

2. Select **Transform** tab → **Merge Columns** (section **Text Column**):



3. In the **Merge Columns** window that appears → in the **Separator** field choose **one of the available separator elements** → choose a name for the column that will be created → **OK**.

Note: If the **Custom** option is chosen in the **Separator** field, then you can define your own separator, composed of several characters:



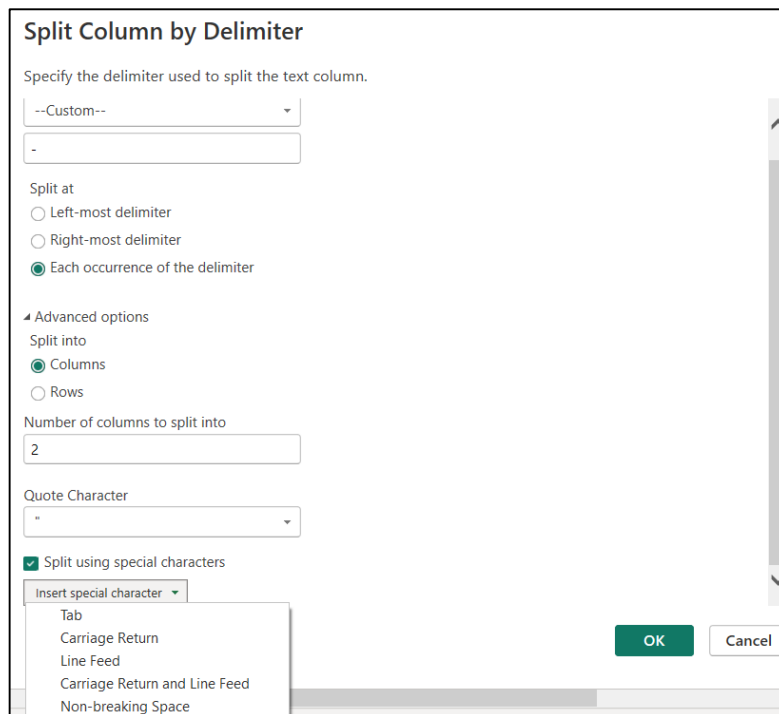
4. The selected columns will be replaced with a column that contains the data from all these columns.

Note: When merging data, the order in which the columns are selected is very important: the first column selected must be the one whose data must appear on the left side of the merged column, and so on.

■ Splitting columns by a delimiter:

In the event that a column contains rows in which there is a list of elements separated by a delimiter character, and these elements must appear in separate columns, the application allows this, following the following steps:

1. **Click the header of the column** that must be splitted.
2. Select **Transform** tab → **Split Column** → **By Delimiter** (section **Text Column**):



3. In the **Split Column by delimiter** window that appears → select the delimiter in the **Select or enter delimiter** field → check the option **Each occurrence of the delimiter**.

In the box **Select or enter delimiter** other types of delimiters can be chosen: Colon (:), Comma (,), Equals Sign (=), Semi-Colon (;), Space (), Tab or Custom (user defined delimiter).

In the **Split At** section, there are the following possible options:


- **Left-most delimiter:** the column is divided only once, namely at the first appearance of the delimiter.
- **Right-most delimiter:** the column is divided only once, namely at the last appearance of the delimiter.
- **Each Occurrence of the Delimiter:** the column is divided into several columns, corresponding to the number of occurrences of the delimiter.

When click the **Advanced options** element another list of options opens:

- **Split into columns:** a new column is created for each new element resulting from the division operation, without changing the number of rows in the data table:

As a result, it can be seen that **the initial column has been replaced by two other columns:**

A ^B C Type and color
Camargue - Red
DBS - Blue
Silver Ghost - Green
Silver Ghost - Blue
Camargue - Canary Yellow
Camargue - British Racing Green
DBS - Dark Purple
DB7 - Red
DB9 - Blue
DB9 - Silver




A ^B C Type and color.1	A ^B C Type and color.2
Camargue	Red
DBS	Blue
Silver Ghost	Green
Silver Ghost	Blue
Camargue	Canary Yellow
Camargue	British Racing Green
DBS	Dark Purple
DB7	Red
DB9	Blue
DB9	Silver

- **Split into rows:** the data string will be split into multiple rows so that each row contains the data that was previously separated by the delimiter:

As a result, it can be seen that **two new rows have appeared:**

A ^B C VehicleType	A ^B C InvoiceNumber
Saloon	8B3D7F83-F42C-4523-A737-CDCBF7705B77, 8B3D7F83-F42C-4523-A737-CDCBF8906X33
Coupe	139BEEEF-FF32-4BE9-9EF1-819AC888B85C
Saloon	D35D72CD-5FF3-4701-A6D1-265A4F4E7CD5, D35D72CD-5FF3-4701-A6D1-265A4F4I9DR4
Saloon	2ABAA300-E2A5-4E37-BFCA-7B80ED88A2BD
Saloon	A1C2D846-EC39-46FA-A399-0C194AAD4DC8



A ^B C VehicleType	A ^B C InvoiceNumber
Saloon	8B3D7F83-F42C-4523-A737-CDCBF7705B77
Saloon	8B3D7F83-F42C-4523-A737-CDCBF8906X33
Coupe	139BEEEF-FF32-4BE9-9EF1-819AC888B85C
Saloon	D35D72CD-5FF3-4701-A6D1-265A4F4E7CD5
Saloon	D35D72CD-5FF3-4701-A6D1-265A4F4I9DR4
Saloon	2ABAA300-E2A5-4E37-BFCA-7B80ED88A2BD
Saloon	A1C2D846-EC39-46FA-A399-0C194AAD4DC8

■ Splitting columns by number of characters:

It is also possible for a text type column to be divided into several columns, extracting a number of characters and leaving the rest in the column, following the steps:

1. Click the header of the column that must be splitted.
2. Select **Transform** tab → **Split Column** → **By Number of Characters** (section **Text Column**):

Split Column by Number of Characters

Specify the number of characters used to split the text column.

Number of characters

Split

Once, as far left as possible
 Once, as far right as possible
 Repeatedly

▸ Advanced options

3. In the **Split Column by Number of Characters** window that appears → in the field **Number of Characters** introduce a number that represents the **number of characters before dividing the column**.

4. In the **Split** section, there are the following possible options:

- **Once, As Far Left As Possible:** Splits the column into two columns, the first containing the number of characters counting from the left, and the second containing the rest of the characters from the right.

- **Once, As Far Right As Possible:** Splits the column into two columns, the second containing the number of characters counting from the right, and the first containing the rest of the characters from the left.

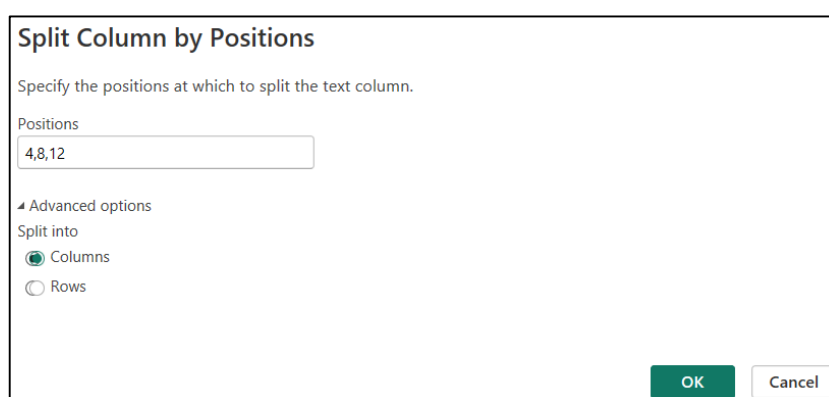
- **Repeatedly:** Splits the column into several columns, each having a number of characters equal to the defined number of characters.

■ **Splitting columns by positions:**

Another possibility offered by the application is to divide a text type column by defining fixed numerical positions of the characters. For example, if the sequence 4,8,12 is entered, the column will be divided into three columns, each having 4 characters. The steps to be followed in this case are:

1. **Click the header of the column** that must be splitted.

2. Select **Transform** tab → **Split Column** → **By Positions** (section **Text Column**):



3. In the **Split Column by Positions** window that appears → in the **Positions** field, enter the position numbers to split the text column.

When click the **Advanced options** element, another list of options opens:

- **Columns:** the default option, explained above.

- **Rows:** based on the specified positions, a new row will be added each time, instead of a new column.

For example, if the sequence 4,8,12 is entered, the column will be divided into 3 rows of 4 characters each.

■ **Splitting columns by letter case combinations:**

- **Lowercase to uppercase:** for each occurrence of two consecutive letters, where the first is a lowercase letter and the second is an uppercase letter, the column is divided so that the second column starts with the uppercase letter.

- **Uppercase to lowercase:** for each occurrence of two consecutive letters, where the first is an uppercase letter and the second is a lowercase letter, the column is divided so that the second column starts with the lowercase letter.

■ **Splitting columns by digit and non-digit combinations:**

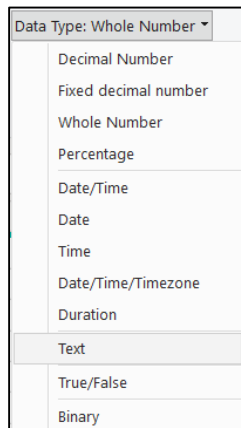
- **Digit to non-digit:** for each occurrence of two consecutive characters, where the first is a digit and the second is not a digit, the column is split so that the second column begins with the character that is not a digit.

- **Non-digit to digit:** for each occurrence of two consecutive characters, where the first is not a digit and the second is a digit, the column is split so that the second column starts with the character that is the digit.

■ **Changing the data type of a column:**

To set another type of data for a certain column, there are two possibilities:

1. **Click the column header** whose type needs to be changed.
2. Select **Home** tab → **Data Type** (section **Transform**) → select the appropriate type:



or:

1. **Right-Click the column header** whose type needs to be changed.
2. Select **Change Type** (from the right-click menu) → **select the appropriate type**.

The following table shows the data formats commonly used in Power Query analysis:

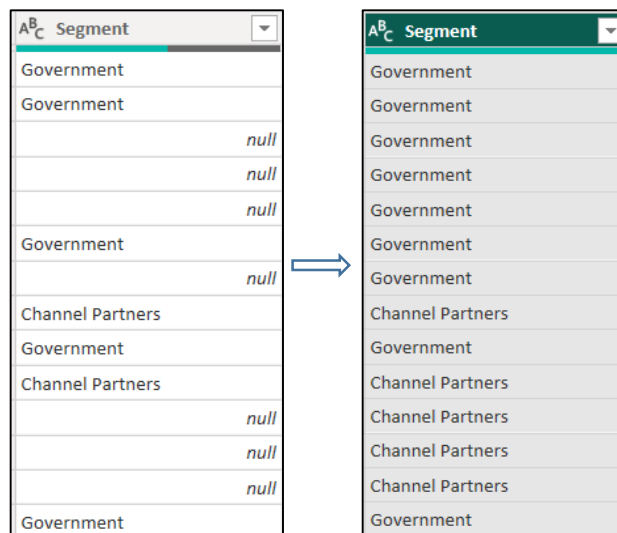
Binary	It can be used to represent data in binary format (byte sequences).
Date	It can be used to represent calendar date values, indicating only the date, not the time.
Time	It can be used to represent values that represent only the time, not the date.
Date/Time	It can be used to represent values that represent a combination of date and time.
Date/Time/Timezone	It can be used to represent values that represent a date, a time, and a time zone; the time zone is calculated as the difference between the local time and the Universal Coordinated Time (UTC).
Duration	It can be used to represent values from subtraction operations on Date/Time fields (difference between two dates).
True/False	It can be used to represent true or false values (based on boolean logic).
Decimal Number	It can be used to represent both integer and fractional values with a maximum of 15 significant decimal digits (64-bit - eight-byte floating-point number).

Fixed Decimal Number	Can be used to represent numeric values with a specified number of decimal places.
Whole Number	Can be used to represent eight-byte (64-bit) integer numeric values, with a maximum format of 19 digits, representing values in the range $[-2^{63}, 2^{63}-1]$.
Percentage	Can be used to represent percentage values. When converting whole numbers to percentage format, the values are multiplied by 100, two decimal places are added and then the percentage symbol.
Text	It can be used to represent Unicode character data strings, with the observation that Power BI Desktop Query Editor distinguishes between uppercase and lowercase text.
Any	It can be used to represent any type of value, in situations where a column lacks an explicit data type definition.

■ **Fill Down/Fill Up:**

When data is improperly imported from various files, there is a possibility that the data set contains a cell with a certain value, and the cells below are empty until the next non-null value. Power Query offers the ability to replace all null values below that cell until another non-null value appears:

1. **Click the header of the column** that must be filled:
2. Select **Transform** tab → **Fill Down** (section **Any Column**):



Note: To replace all null values above a non-null value, the **Fill Up** option will be used:

1. **Click the header of the column** that must be filled:
2. Select **Transform** tab → **Fill Up** (section **Any Column**).

■ **Converting rows into columns and vice versa:**

In Power Query it is possible to change the orientation of a range of cells, by converting rows into columns and vice versa, i.e. transposing the data:

1. Select the columns to be transposed:

	ABC Segment	ABC Country	ABC Product
1	Government Agency	CANADA	Carretera
2	Government Agency	GERMANY	Carretera
3	Midmarket Leaders	FRANCE	Carretera
4	Midmarket Leaders	GERMANY	Carretera
5	Midmarket Leaders	MEXICO	Carretera
6	Government Agency	GERMANY	Carretera

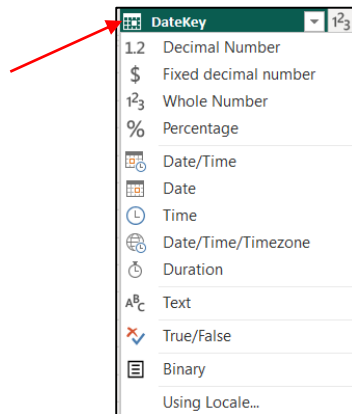
2. Select **Transform** tab → **Transpose** (section **Table**):

	ABC Column1	ABC Column2	ABC Column3	ABC Column4	ABC Column5	ABC Column6
1	Government Agency	Government Agency	Midmarket Leaders	Midmarket Leaders	Midmarket Leaders	Government Agency
2	CANADA	GERMANY	FRANCE	GERMANY	MEXICO	GERMANY
3	Carretera	Carretera	Carretera	Carretera	Carretera	Carretera

■ **Configuring Regional Settings:**

A source of error in the Power Query application is the different format of date fields: the original date format is from a different region, and when importing the data into the application, the format is not recognized. For example, most regions use the DD/MM/YYYY format, while the U.S. use the MM/DD/YYYY format. To solve this type of error and to make it adapted to the local formatting conventions, there are two simple ways:

i) 1. **On the left of the column title** → **Click the data type icon** → select **Using Locale...**



2. In the **Change Type with Locale** window that opens:

Change Type with Locale

Change the data type and select the locale of origin.

Data Type

Locale

i Sample input values:
 29.03.2016
 marți, 29 martie 2016
 29 martie
 martie 2016

- choose **Date** as the **Data Type**.

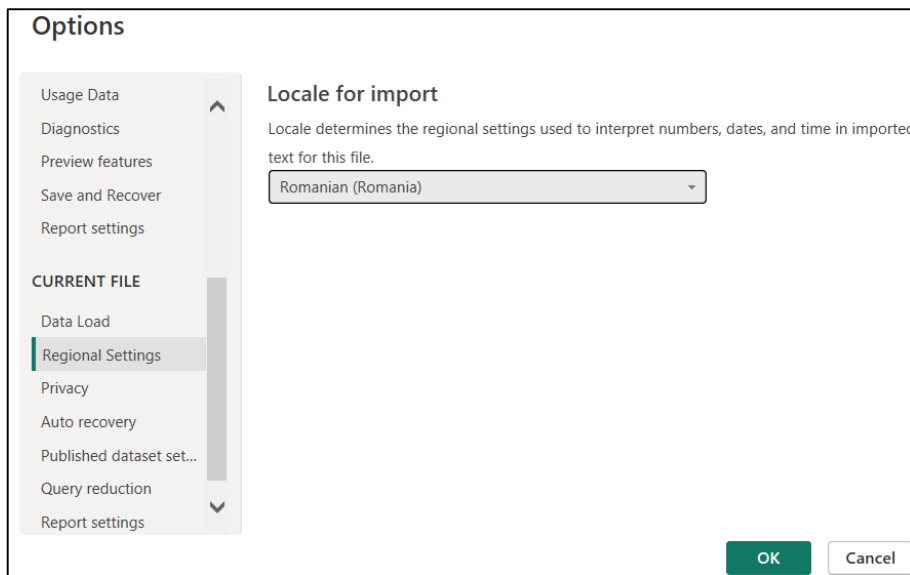
- in **Locale** field choose the **required locale** from the list of worldwide locales.

3. Click **OK** → **The data type is converted to the selected locale.**

ii) If the query has several data type columns or there are several queries, it would be more appropriate to change the regional settings for the entire file:

1. Select **File** tab → select **Options and settings** → **Options**.

2. In the **Options** window that opens → In the **CURRENT FILE** section (left side) select **Regional Settings**:



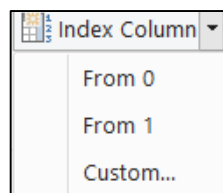
3. In the **Locale** field → **choose the respective selection** (where the data comes from) → **OK**.

5.3.3 Advanced data transformations

■ Add an index column:

An index column is a new column with specific positional values that is typically created to work with other transformation models. To insert such a column, follow these steps:

1. Select **Add Column** tab → **Column** (section **General**):



2. The starting index will default to 0, incrementing by 1 per row.

3. To change this aspect, choose the **Custom** option.

4. In the **Add Index Column** window that appears:

- In the **Starting Index** box → the **index start value** is specified.

- In the **Increment** box → the **increment value** is specified.

Add Index Column

Add an index column with a specified starting index and increment.

Starting Index

Increment

OK Cancel

5. Click **OK** → A column containing the values set in this window will be added.

■ **Add a modulo column based on an index column:**

If a repetition of the values contained in the index column is needed, for example the values from 1 to n must be repeated, then follow the steps:

1. Select the newly added index column.
2. Select **Add Column** tab → **Standard** (section **From Number**) → **Modulo**:

Modulo

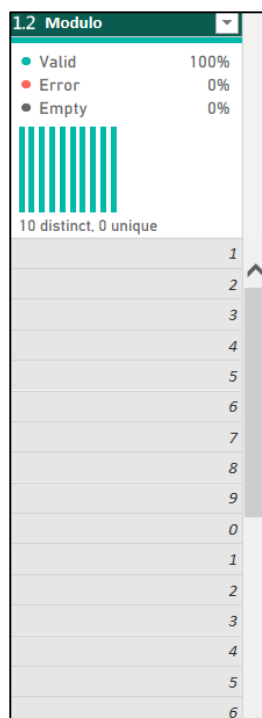
Enter a number from which to find the remainder for each value in the column.

Value

OK Cancel

3. In the **Modulo** window that appears → In the **Value** box the **repetition value** is specified.

5. Click **OK** → A column titled **Modulo** will be added, which contains the values set in this window:



■ **Add a conditional column:**

In the case of advanced data analyses, there are situations where, in order to provide more interactivity to reports and dashboards, it is necessary to insert a conditional column: a column in which the respective values are generated based on certain criteria, using the IF-THEN -ELSE principle to define the field result.

In the following example, we want to add a column that contains a comment about the type of discount for the products sold, namely for an existing discount, defined in the **Discount Band** field; the offer will be included in the **favorable offer**, and if there is no discount, the offer will be an **unfavorable offer**.

To achieve this, the following steps are performed:

1. Select **Add Column** tab → **Conditional Column** (section **General**):

Column Name	Operator	Value	Output
Discount Band	equals	None	No

2. In the **Add Conditional Column** window that appears:

- In the **New column name** box → **define a name for the new conditional column** (in this example: **Favorable offer**).
- In the **Column name** list box → **select the column name** (in this example: **Discount Band**).
- In the **Operator** list box → **select an operator** (in this example: **equals**).
- In the **Value** box → **enter a suitable value** (in this example: **None**).
- In the **Output** box → **enter the value** that must be displayed in the new column **when the IF condition is true** (in this example: **No**).
- In the **Else** box → **enter a valid expression when the condition is false** (in this example: **Yes**).

3. Click **OK** → **A column containing the values Yes or No will be added**, depending on the existing discount.

Note: In the **Operator** list box there are **other types of operators** that can be selected, depending on the data type of the column selected in the **Column name** list box:

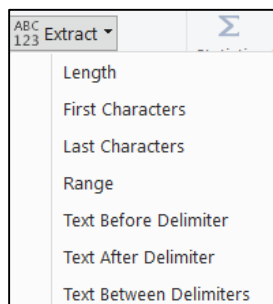
Selected field type in Column Name		
Text	Number	Data
Equals	Equals	Equal
Does not equal	Does not equal	Does not equal
Begins with	Greater than	Is before
Does not begin with	Greater than or equal to	Is before or equal to
Ends with	Less than	Is after
Does not end with	Less than or equal to	Is after or equal to
Contains		
Does not contain		

■ **Data extraction operations from the content of a column:**

To extract a specific part of the content of a column, the Power Query application offers two possibilities:

► **If the existing column must remain unchanged and an additional column must be added with the extracted content, the Extract function from the Add Column menu will be used:**

1. **Click the column header** from which the respective content must be extracted (in this example: **Segment**).
2. Select **Add Column** tab → **Extract** (section **From Text**) → **Choose one of the content extraction options** (in this example: **Text Before Delimiter**):



3. In the **Text Before Delimiter** window that appears → In the **Delimiter** field, enter the delimiter that marks the end of the content to be extracted (in this example: **Space**) → **OK**:

Text Before Delimiter

Enter the delimiter that marks the end of what you would like to extract.

Delimiter

▶ Advanced options

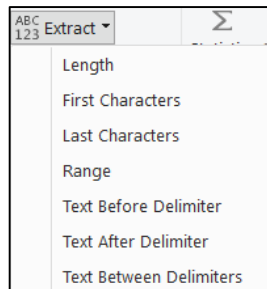
OK
Cancel

4. **The Segment column remains unchanged** and the automatically titled **Text before Delimiter column** appears in addition with the extracted content:

ABC Segment	ABC Text Before Delimiter
Government Agency	Government
Government Agency	Government
Midmarket Leaders	Midmarket
Midmarket Leaders	Midmarket
Midmarket Leaders	Midmarket
Government Agency	Government
Midmarket Leaders	Midmarket
Channel Partners	Channel
Government Agency	Government
Channel Partners	Channel

► **If the existing column needs to be modified, use the Extract function from the Transform menu:**

1. **Click the column header** from which the respective content must be extracted (in this example: **Segment**).
2. Select **Transform** tab → **Extract** (section **Text Column**) → **Choose one of the content extraction options** (in this example: **Text Before Delimiter**):



3. In the **Text Before Delimiter** window that appears → In the **Delimiter** field, **enter the delimiter that marks the end of the content to be extracted** (in this example: **Space**) → **OK**:

Text Before Delimiter

Enter the delimiter that marks the end of what you would like to extract.

Delimiter

> Advanced options

OK
Cancel

4. **The Segment column will be modified**, because in this column the existing content before the space delimiter was extracted:

ABC Segment	ABC Segment
Government Agency	Government
Government Agency	Government
Midmarket Leaders	Midmarket
Midmarket Leaders	Midmarket
Midmarket Leaders	Midmarket
Government Agency	Government
Midmarket Leaders	Midmarket
Channel Partners	Channel
Government Agency	Government
Channel Partners	Channel

Notes:

1) The **Extract** menu also contains other options:

Length	returns the length of the text in the selected field
First Characters	returns a specified number of characters to be extracted (from the left)
Last Characters	returns a specified number of characters to be extracted (from the right)
Range	allows to define a starting position in the text and how many characters to extract from this starting position
Text Before Delimiter	returns the text occurring before a specified delimiter
Text After Delimiter	returns the text occurring after a specified delimiter
Text Between Delimiters	returns the text occurring between two specified delimiters

2) In the **Text Before Delimiter** window that appears → When click the **Advanced options** element, another list of options opens:



- **Scan for the delimiter (From the start of the input or From the end of the input):** allows locating the searched delimiter, scrolling the column content forward, from the beginning of the column or backward, from the end of the column.

- **Numbers of delimiters to skip:** allows defining the number of occurrences of the delimiter that will be ignored before performing the extraction.

■ **Creating a custom column based on M formula language:**

If you want to create a custom column, apart from the functions provided by the Power Query application in the menu bar, one can use the **Power Query M formula language**.

For example, in the data set below, which contains the columns **Units Sold** and **Manufacturing Price**, a new column should be added to express:

Total sales = Units Sold * Manufacturing Price.

Units Sold	Manufacturing Price
888	891
2470	891
1513	891
921	1368
2518	1368
1899	1368
1545	1368

The steps to be followed in this case are:

1. Select **Add Column** tab → **Custom Column** (section **General**):

2. In the **Custom Column** window that appears:

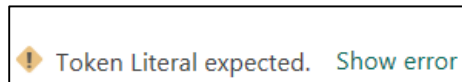
- In the **New column name** box → specify the **name of the custom column**.
- In the **Available columns** list → select the **names of the columns that will enter the calculation formula**.
- In the **Custom column formula** → the **Power Query M formula** is edited.

To write the Power Query M formula:

1. **Select the column** name from the **Available columns** list → **Click on the Insert button** → immediately, the **column name will be surrounded by square brackets**, which shows that it is a column reference.
2. Continue editing the formula, **specifying the names of the columns that will enter the calculation formula and the corresponding operators** → **OK**.
3. As a result, a **new column will appear on the right side of the table containing the values calculated according to the defined formula**:

1 ² ₃ Units Sold	1 ² ₃ Manufacturing Price	ABC 123 Total sales
1618	891	1441638
1321	891	1177011
2178	891	1940598
888	891	791208
2470	891	2200770
1513	891	1348083
921	1368	1259928

Note: If there is a syntax error when creating the custom column, then a yellow warning icon will appear along with an error message:



■ **Append data from multiple data sources:**

Data specialists regularly face situations in which several data sources must be combined into a single file, based on which various analyzes will be performed later. There are various situations:

- simpler scenarios, in which although many files must be joined (in the order of hundreds), the data sets have the same structure, so they contain the same columns, which have the same order; in this case, a new set of data will be obtained simply by annexing all the data sources
- more complicated scenarios, in which the data sets do not have the same structure; in this case, a new set of data will be obtained and it will be observed that if an attached table does not have certain column headers from other tables, the obtained table will contain null values in those columns.

To exemplify the first type of scenario, consider two text files, Countries_01.txt and Countries_02.txt, which will be combined into a single output file:

1. **Open Power BI Desktop interface.**
2. Select **Home** tab → **Get data** → **Text/CSV** → the **first file is located**, Countries_01.txt → **Open** → **Load**.

The same procedure is followed for the second file, Countries_02.txt:

Countries_01.txt:

1 ² ₃ CountryID	A ^B _C Name
1	1 Ireland
2	2 Norway
3	3 Denmark
4	4 Poland
5	5 France

Countries_02.txt:

1 ² ₃ CountryID	A ^B _C Name
1	6 Spain
2	7 Romania
3	8 Bulgaria
4	9 Greece
5	10 Island

3. Select **Home** tab → **Transform data** → **Transform data**.
4. In the Power Query Editor, to append these queries, first **select the Countries_01 query**.
5. Select **Home** tab → **Append Queries** (section **Combine**) → **Append Queries**:

Append

Concatenate rows from two tables into a single table.

Two tables
 Three or more tables

Table to append

Countries_02

OK Cancel

6. In the **Append** window that opens → In the **Table to append** list, choose the query **Countries_02** → **OK**.

7. As a result, a new query is obtained, where the data from the two tables is appended:

	1 ² ₃ CountryID	A ^B _C Name
1	1	Ireland
2	2	Norway
3	3	Denmark
4	4	Poland
5	5	France
6	6	Spain
7	7	Romania
8	8	Bulgaria
9	9	Greece
10	10	Island

In the case of the second type of scenario, the application will perform the addition operation based on the names of the column headers from both tables, and the output table will have all the columns from the two tables appended. If a table does not contain columns that can be found in the other table, then null values will appear in that column:

1 ² ₃ CountryID	A ^B _C CountryName	A ^B _C CountryISOCode	1 ² ₃ StockID	A ^B _C Make	A ^B _C Model
1	United Kingdom	GBR		null	null
2	France	FRA		null	null
3	USA	USA		null	null
4	Germany	DEU		null	null
5	Spain	ESP		null	null
6	Switzerland	CHE		null	null
null	null	null		1	Rolls Royce
null	null	null		2	Aston Martin
null	null	null		3	Rolls Royce
null	null	null		4	Rolls Royce
null	null	null		5	Rolls Royce
					Camargue
					DBS
					Silver Ghost
					Silver Ghost
					Camargue

■ **Merging tables:**

When working with large data sets, to avoid manually copying and pasting data from one source to another, one can use the Merge queries option. More specifically, when two or more data sets contain one or more common fields, the Merge queries option can be used to combine the respective data tables in a single unified set. Basically, through this type of data aggregation, a new table is obtained, which contains all the columns from all the tables involved, or new columns are added to an existing table.

To exemplify this, consider two Excel files, Table1_for_merging.xlsx and Table2_for_merging.xlsx, which will be merged into a single output file:

1. Open Power BI Desktop interface.

2. Select **Home** tab → **Get data** → **Excel workbook** → the first file is located, Table1_for_merging.xlsx → **Open** → **Load**.

The same procedure is followed for the second file, Table2_for_merging.xlsx:

Table1_for_merging.xlsx :

	Client ID	Address	Reporting Year	Sales
1	78	Address_1	2013	759
2	45	Address_2	2014	682
3	21	Address_3	2015	531
4	74	Address_4	2016	225
5	85	Address_5	2017	870
6	65	Address_6	2018	49
7	32	Address_7	2019	485
8	31	Address_8	2020	268
9	21	Address_9	2021	157
10	51	Address_10	2022	198
11	54	Address_11	2023	143

Table2_for_merging.xlsx :

	Client ID	Postal code	Reporting Year	Profit
1	78	LK6541	2013	45937
2	45	MN6530	2014	58004
3	21	FG4320	2015	65927
4	74	LK6549	2016	8926
5	85	RT7895	2017	29956
6	65	LT6765	2018	3746
7	32	LR6545	2019	11076
8	31	LK6548	2020	45900
9	21	LR6519	2021	17473
10	51	PK6550	2022	14631
11	54	ER6351	2023	63940

3. Select **Home** tab → **Transform data** → **Transform data**.

4. In the Power Query Editor, to merge these queries, first select the **Table1_for_merging** query.

5. Select **Home** tab → **Merge Queries** (section **Combine**) → **Merge Queries**:

Merge

Select a table and matching columns to create a merged table.

Table1_for_merging

Client ID	Address	Reporting Year	Sales
78	Address_1	2013	759
45	Address_2	2014	682
21	Address_3	2015	531
74	Address_4	2016	225
85	Address_5	2017	870

Table2_for_merging

Client ID	Postal code	Reporting Year	Profit
78	LK6541	2013	45937
45	MN6530	2014	58004
21	FG4320	2015	65927
74	LK6549	2016	8926
85	RT7895	2017	29956

Join Kind

Left Outer (all from first, matching from second)

Use fuzzy matching to perform the merge

Fuzzy matching options
 Similarity threshold (optional)

Ignore case

The selection matches 11 of 11 rows from the first table.


OK
Cancel

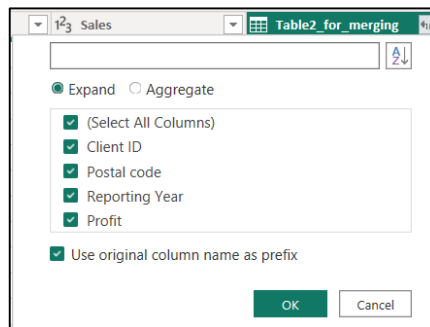
6. In the **Merge** window that opens:

- At the top of the dialog box, where the first data source is displayed, **Table1_for_merging** → **Click on the common column, Client ID.**
- In the box-list below → select the query **Table2_for_merging** → **Click on the common column, Client ID.**
- In the **Join Kind** box-list → **select Left outer (all from first, matching from second)** option:

	1 ² Client ID	A ^B C Address	1 ² Reporting Year	1 ² Sales	Table2_for_merging
1		78 Address_1	2013	759	Table
2		45 Address_2	2014	682	Table
3		21 Address_3	2015	531	Table
4		74 Address_4	2016	225	Table
5		85 Address_5	2017	870	Table
6		65 Address_6	2018	49	Table
7		32 Address_7	2019	485	Table
8		31 Address_8	2020	268	Table
9		21 Address_9	2021	157	Table
10		51 Address_10	2022	198	Table
11		54 Address_11	2023	143	Table

7. **Click OK** → As a result, a new column, **Table2_for_merging**, has been added to the right of the existing data table, representing the merged table.

8. **Click on the double arrow button**  in the right corner of the new column header → a window opens with the available fields from the second table that should be included in the merged dataset>



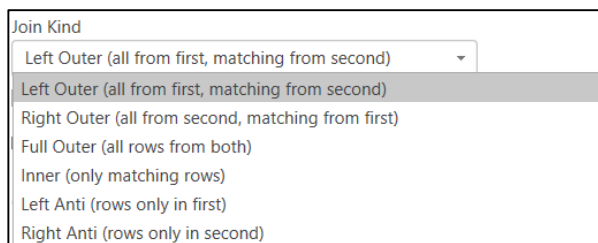
9. **Select the columns to be added** → **OK:**

	1 ² Client ID	A ^B C Address	1 ² Reporting Year	1 ² Sales	A ^B C Table2_for_merging.Postal code	1 ² Table2_for_merging.Profit
1		78 Address_1	2013	759	LK6541	45937
2		45 Address_2	2014	682	MN6530	58004
3		21 Address_3	2015	531	FG4320	65927
4		21 Address_3	2015	531	LR6519	17473
5		21 Address_9	2021	157	FG4320	65927
6		21 Address_9	2021	157	LR6519	17473
7		74 Address_4	2016	225	LK6549	8926
8		85 Address_5	2017	870	RT7895	29956
9		65 Address_6	2018	49	LT6765	3746
10		32 Address_7	2019	485	LR6545	11076
11		31 Address_8	2020	268	LK6548	45900
12		51 Address_10	2022	198	PK6550	14631
13		54 Address_11	2023	143	ER6351	63940

Notes:

1) In the window with the available fields from the second table that should be included in the merged data set, checking the **Use original column name as prefix** option will prefix the table name to each inserted column; if this is not desired, uncheck this option.

2) The **Join Kind** box-list contains 6 different types of joins:



- **Left Outer (all from first, matching from second):** is the default option; is used when the LEFT query is the important one and then all records from this query will be displayed in the result set, plus their matching rows on the right (from the second table).

- **Right Outer (all from second, matching from first):** is used when the RIGHT query is the important one and then all the records from this query will be displayed in the result set, to which their matching rows from the left (from the first table) are added.

- **Full Outer (all rows from both):** is used when the new data set must contain all records: all rows from the first table, all rows from the second table, and all matching rows.

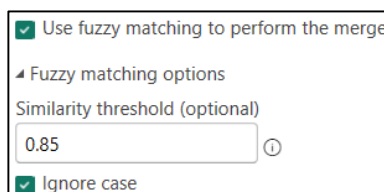
- **Inner (only matching rows):** is used when the new data set must contain only matching rows.

- **Left Anti (rows only in first):** is used when the new data set must contain only the rows that are in the first table and do not match what is in the other table.

- **Right Anti (rows only in second):** is used when the new data set must contain only the rows that are in the second table and do not match what is in the first table.

3) Use **fuzzy matching to perform the merge** is a way to join two tables together, but not by looking for exact matches in the data, but by using a certain similarity threshold, which can take values from 0 (match every row) to 1 (exact matches).

To use this facility:



- **Check the option Use fuzzy matching to perform the merge.**

- **Choose a similarity threshold (a value from 0 to 1):** when the similarity of the two text values is greater than the threshold, a successful match will be considered.

- One can choose one of the options of Fuzzy Merge:

- Ignore case** the similarity algorithm ignores uppercase/lowercase characters in columns.
- Ignore spaces** the similarity algorithm ignores whitespace characters in columns.
- Maximum number of matches** sets the maximum number of matching rows that can be associated with a single value.
- Transformation table** provides the option to use your own mapping table so that some values can be mapped automatically

■ **Cross joins (cartesian product):**

A cross join is a type of join that returns all possible combinations of the rows of one table together with the rows of a second table (the Cartesian product of the rows in the two tables). In an equivalent expression, it combines each row of the first table with each row of the second table, thus creating a large list of all possible combinations of records.

To exemplify this, consider two Excel files, car_models.xlsx and color.xlsx, which will be combined by a join operation:

1. **Open Power BI Desktop interface.**

2. Select **Home** tab → **Get data** → **Excel workbook** → the **first file is located**, car_models.xlsx → **Open** → **Load**.

The same procedure is followed for the second file, color.xlsx:

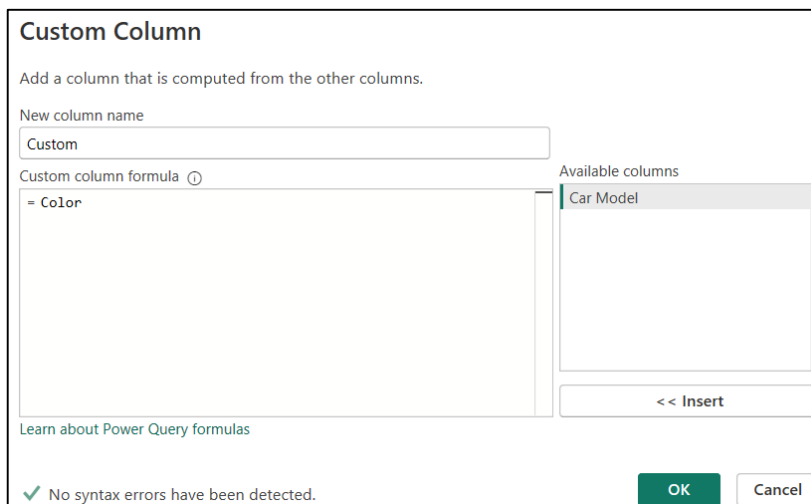
	A ^B _C Column1
1	Car Model
2	Model_X
3	Model_Y
4	Model_Z
5	Model_T
6	Model_Q
7	Model_R

	A ^B _C Color
1	Red
2	Blue
3	Green
4	Silver
5	Canary Yellow
6	Night Blue
7	Black
8	British Racing Green
9	Dark Purple
10	Pink

3. Select **Home** tab → **Transform data** → **Transform data**.

4. In the Power Query Editor, to join these queries, first **select the Car Model query**.

5. Select **Add column** tab → **Custom column** (section **General**):




6. In the **Custom Column** window that opens:

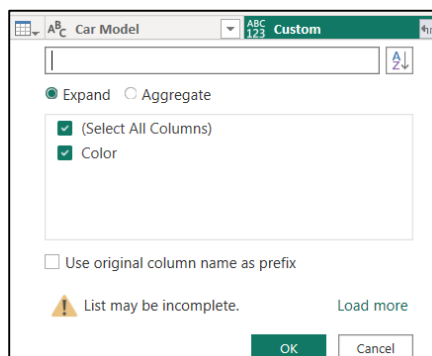
- In the **New column name** box → **enter a specific name**.

- In the **Custom column formula** box → **enter Color**.

7. **Click OK** → As a result, a new column, **Custom**, has been added to the right of the Car Model data table, representing the merged table.

	ABC Car Model	ABC 123 Custom
1	Model_X	Table
2	Model_Y	Table
3	Model_Z	Table

8. Click on the double arrow button  in the right corner of the new column header → a window opens with the available fields from the second table that should be included in the merged dataset.



9. Select the columns to be added → OK:

	ABC Car Model	ABC 123 Color
1	Model_X	Red
2	Model_X	Blue
3	Model_X	Green
4	Model_X	Silver
5	Model_X	Canary Yellow
6	Model_X	Night Blue
7	Model_X	Black
8	Model_X	British Racing Green
9	Model_X	Dark Purple
10	Model_X	Pink
11	Model_Y	Red
12	Model_Y	Blue
13	Model_Y	Green
14	Model_Y	Silver
15	Model_Y	Canary Yellow
16	Model_Y	Night Blue
17	Model_Y	Black
18	Model_Y	British Racing Green
19	Model_Y	Dark Purple
20	Model_Y	Pink
21	Model_Z	Red
22	Model_Z	Blue
23	Model_Z	Green
24	Model_Z	Silver
25	Model_Z	Canary Yellow
26	Model_Z	Night Blue
27	Model_Z	Black
28	Model_Z	British Racing Green

As the first table contains 3 variables and the second table has 10 variables, the result has $3 \times 10 = 30$ unique combinations.

■ Pivot columns:

When managing large data sets, before the data can be used in Power BI for analysis, it is necessary to transform it through aggregation and grouping or summarizing operations. Thus, the Power Query application groups each unique value and allows the creation of a table that contains an aggregated value for each unique value in a column.

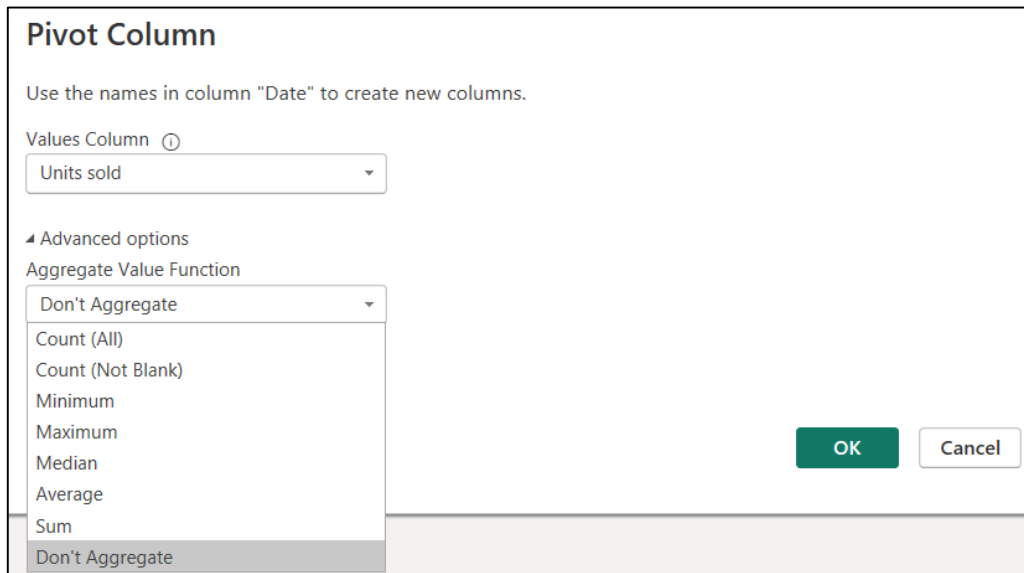
Also, columns can be pivoted without aggregation, when there are columns that cannot be aggregated or when aggregation is not necessary to obtain a better visualization of the data.

For example, in the data table below, every four rows of data represents a record and for better visualization, Power Query can group each unique value by doing an aggregate calculation for each value and then pivot column in a new table:

	A ^B C Store	Date	1 ² 3 Units sold
1	Store 1	05/07/2023	45
2	Store 1	06/07/2023	23
3	Store 1	07/07/2023	47
4	Store 1	08/07/2023	56
5	Store 2	05/07/2023	89
6	Store 2	06/07/2023	74
7	Store 2	07/07/2023	25
8	Store 2	08/07/2023	64
9	Store 3	05/07/2023	49
10	Store 3	06/07/2023	62
11	Store 3	07/07/2023	54
12	Store 3	08/07/2023	77

To pivot a column:

1. **Select the column** that you want to apply the Pivot Column feature (in this example: **Date**).
2. Select **Transform** tab → **Pivot column** (section **Any column**):



3. In the **Pivot Column** window that opens:

- In the **Values Column** list → **choose the column** that will be transformed (in this example: **Units sold**).
- In the **Aggregate Value Function** list → **choose between available aggregations** (in this example: **Don't Aggregate**).

4. Click **OK** → **The result below will be obtained:**

	A ^B C Store	1 ² 3 05/07/2023	1 ² 3 06/07/2023	1 ² 3 07/07/2023	1 ² 3 08/07/2023
1	Store 1	45	23	47	56
2	Store 2	89	74	25	64
3	Store 3	49	62	54	77

In the following example, the table contains values (units sold) according to Store and Date of sale. The table must be transformed by pivoting the date column:

Store	Date	Units sold
Store 1	05/07/2023	45
Store 2	05/07/2023	23
Store 1	05/07/2023	47
Store 4	05/07/2023	56
Store 2	05/07/2023	89
Store 2	06/07/2023	74
Store 4	06/07/2023	25
Store 1	06/07/2023	64
Store 3	06/07/2023	49
Store 2	06/07/2023	62
Store 3	07/07/2023	54
Store 4	07/07/2023	77
Store 3	07/07/2023	78
Store 2	07/07/2023	44
Store 1	07/07/2023	58
Store 4	08/07/2023	56
Store 2	08/07/2023	63
Store 3	08/07/2023	28
Store 4	09/07/2023	47
Store 1	08/07/2023	29
Store 3	09/07/2023	45
Store 4	08/07/2023	78
Store 4	09/07/2023	41

The settings made can be found below:

Pivot Column

Use the names in column "Store" to create new columns.

Values Column ⓘ

Advanced options

Aggregate Value Function

[Learn more about Pivot Column](#)

Which generates the following result:

Date	Store 1	Store 2	Store 4	Store 3
05/07/2023	92	112	56	null
06/07/2023	64	136	25	49
07/07/2023	58	44	77	132
08/07/2023	29	63	134	28
09/07/2023	null	null	88	45

Note: The Aggregate Value Function list contains the following options: **Don't aggregate, Count (all), Count (not blank), Minimum, Maximum, Median, Sum** or **Average**.

■ **Unpivot columns:**

When managing large data sets that need to be analyzed, there are situations where it is observed that the data is delivered in a nested or stacked format (instead of certain information being found on one column, it is found on several columns) . Since the visualization and analysis of this data would be difficult to achieve in this format, first the data set should be transformed by deactivating the stacking, in a tabular format of attribute-value pairs, where the columns are transformed into rows. So it would be much more convenient to have a single column for sales values and a row for each date.

For example, in the table with the structure below, the columns Stores and Date rows, generate this matrix of values, because there is a column for each date of sales. It is observed that the table includes several numerical columns with different contexts for the same attribute, which alters the ability to analyze this data. Tables with this type of format are common for spreadsheets and reports.

Date	Store 1	Store 2	Store 3	Store 4	Store 5	Store 6	Store 7	Store 8	Store 9	Store 10
05/01/2023	4	73	55	78	83	23	13	99	92	58
09/01/2023	50		42	80	15	56	49	8	36	80
12/01/2023	20	10	28	5	42	95	67		57	
19/01/2023	5	31	19	12		99	79	61	36	33
21/01/2023	29		49	3	77	78	46	58	51	6
25/01/2023	12	71	29	50	57	89		23	68	
30/01/2023	37	21	55		26	96	26	89	83	99
02/02/2023	57	92	76	70	96	97	60		71	60
06/02/2023		96	62		87	90	75	14	43	75
15/02/2023	9	90	66	8		29	91	5	65	66
20/02/2023	63	29		80	43	69		91	89	88
26/02/2023	29	96	57	1	55	21	33	24	91	
27/02/2023	78		52	30		37		83	76	89
28/02/2023	99	2	89	54	7	96	19	2	50	74
02/03/2023	28	44		68		33	84	74	22	79
09/03/2023		53	59	18	7	38	81	83	49	75
10/03/2023	87	64	12	58	95	22		37	39	79
14/03/2023	2		20	95	50	75	56	97	97	81
15/03/2023	28	98			35	74	64	77	4	54
22/03/2023	49	85	74	4	41	67	2		77	
26/03/2023		99	17	11	57	13	9	66	30	65
28/03/2023	18	83	78	88	39	63	86	54	85	57
03/04/2023	67	87	12	25		75	39	99	64	35
07/04/2023	21	16		65	2	92	19	34		8
14/04/2023	22	62	8	18	58	23	28	38	59	36
15/04/2023		42	18		8	23		41	94	29
19/04/2023	65	21	40	76	76	61	5	19	66	79
27/04/2023	55	74	47	79	5	1	52	59	19	24

As such, this table can be transformed into a table with non-pivot columns, in which the date will be used as the filtering attribute.

In general, when the unpivot operation is applied, there are two types of columns:

- columns that will not be unpivoted, because they represent the attributes that must be kept in the data structure;
- columns that will be unpivoted, because they represent different contexts for the same attribute.

Through the unpivot operation, each cell in the unpivoted columns will be represented in a row containing the unpivoted columns, along with the two new columns, Attribute and Value for the name and value of the cell's original column.

Thus, **to unpivot data with Power Query Editor**, the following steps must be performed:

1. **Load the dataset** into the Power Query Editor.
2. **Select the set of columns that must be unpivoted** (in this example **columns Store1 to Store 10**).

3. Select **Transform** tab → **Unpivot columns** (section **Any column**) → **Unpivot column:**

Date	Attribute	Value
05/01/2023	Store 1	4
05/01/2023	Store 2	73
05/01/2023	Store 3	55
05/01/2023	Store 4	78
05/01/2023	Store 5	83
05/01/2023	Store 6	23
05/01/2023	Store 7	13
05/01/2023	Store 8	99
05/01/2023	Store 9	92
05/01/2023	Store 10	58
09/01/2023	Store 1	50
09/01/2023	Store 3	42
09/01/2023	Store 4	80
09/01/2023	Store 5	15
09/01/2023	Store 6	56
09/01/2023	Store 7	49
09/01/2023	Store 8	8
09/01/2023	Store 9	36
09/01/2023	Store 10	80
12/01/2023	Store 1	20
12/01/2023	Store 2	10
12/01/2023	Store 3	28
12/01/2023	Store 4	5
12/01/2023	Store 5	42
12/01/2023	Store 6	95
12/01/2023	Store 7	67
12/01/2023	Store 9	57

4. Save the resulting table as a new query.

Notes:

1) The **Unpivot Columns transform** (from the **Transform** tab) contains three options:

- **Unpivot Columns:** previous scenario.
- **Unpivot Other Columns:** allows the selection of columns to which the unpivot operation should not be applied. It is useful in queries with an unknown number of columns. In this case, the unpivot operation will be applied to all columns in the table, except for the selected ones.
- **Unpivot Only Selected Columns:** It is useful in queries with an unknown number of columns, where you want to apply the unpivot operation only to the selected columns.

2) The unpivot operation can be applied to a **dataset imported from the web in the Power Query Editor**.

In the following example, a table of historical data of the census carried out in the states of India, existing at the following address, will be loaded into the Power Query Editor:

[https://en.wikipedia.org/wiki/List_of_states_in_India_by_past_population:](https://en.wikipedia.org/wiki/List_of_states_in_India_by_past_population)

The following steps must be performed:

1. Select **Home** tab → **New Source** → **Web**.

2. In the **From Web** window that opens → in the **Url field** → enter the above address → **OK**:

3. Select the census table from the displayed tables, **By past population (1947 to 2022)** → **OK**:

Rank	State or union territory	Population (1951 Census)[11]	Popu
1	Uttar Pradesh	60,274,800	71
2	Maharashtra	32,002,500	35
3	Bihar	29,085,900	34
4	West Bengal	26,300,670	34
5	Madhya Pradesh	18,615,700	23
6	Tamil Nadu	30,119,680	33
7	Rajasthan	15,971,130	20
8	Karnataka	19,402,500	23
9	Gujarat	16,263,700	20
10	Andhra Pradesh	31,115,000	33
11	Odisha	14,646,100	17
12	Telangana	N/A	N
13	Kerala	13,549,000	14
14	Jharkhand	9,697,300	11
15	Assam	8,029,100	11

4. **Unnecessary information is removed:** the first column, Rank and the last line that makes up the total.

A table containing data about the census was obtained: the first column contains the states, and the following columns the populations reviewed in the respective years.

By applying the unpivot operation, we will transform this table into a table that will contain three columns: state, year and population.

5. Click on the column **State or union territory** → Select **Transform** tab → **Unpivot columns** (section **Any column**) → **Unpivot other columns:**

	A ^B _C State or union territory	A ^B _C Attribute	1 ² ₃ Value
1	Uttar Pradesh	Population (1951 Census)[11]	60274800
2	Uttar Pradesh	Population (1961 Census)[11]	70144160
3	Uttar Pradesh	Population (1971 Census)[11]	83849775
4	Uttar Pradesh	Population (1981 Census)[11]	105113300
5	Uttar Pradesh	Population (1991 Census)[11]	132062800
6	Uttar Pradesh	Population (2001 Census)[11]	166053600
7	Uttar Pradesh	Population (2011 Census)[11]	199581477
8	Maharashtra	Population (1951 Census)[11]	32002500
9	Maharashtra	Population (1961 Census)[11]	39554900
10	Maharashtra	Population (1971 Census)[11]	50412240
11	Maharashtra	Population (1981 Census)[11]	62782820
12	Maharashtra	Population (1991 Census)[11]	78937190
13	Maharashtra	Population (2001 Census)[11]	96752500
14	Maharashtra	Population (2011 Census)[11]	112372972
15	Bihar	Population (1951 Census)[11]	29085900
16	Bihar	Population (1961 Census)[11]	34841490
17	Bihar	Population (1971 Census)[11]	42126800
18	Bihar	Population (1981 Census)[11]	52303000

Note: The Power Query Editor application keeps track of the steps applied for each query, in the form of text, which can be viewed and possibly modified. To access this record:

Select **View** tab → **Advanced Editor**:

```

let
    Source = Web.BrowserContents("https://en.wikipedia.org/wiki/List_of_states_in_India_by_past_populati
    #"Extracted Table From Html" = Html.Table(Source, {"Column1", "TABLE.wikitable.sortable > * > TR >
    #"Promoted Headers" = Table.PromoteHeaders("#Extracted Table From Html", [PromoteAllScalars=true]),
    #"Changed Type" = Table.TransformColumnTypes("#Promoted Headers",{"Rank", type text}, {"State or un
    #"Removed Columns" = Table.RemoveColumns("#Changed Type",{"Rank"}),
    #"Removed Bottom Rows" = Table.RemoveLastN("#Removed Columns",1),
    #"Unpivoted Other Columns" = Table.UnpivotOtherColumns("#Removed Bottom Rows", {"State or union terr
    #"Changed Type1" = Table.TransformColumnTypes("#Unpivoted Other Columns",{"Value", Int64.Type})
in
    #"Changed Type1"
  
```

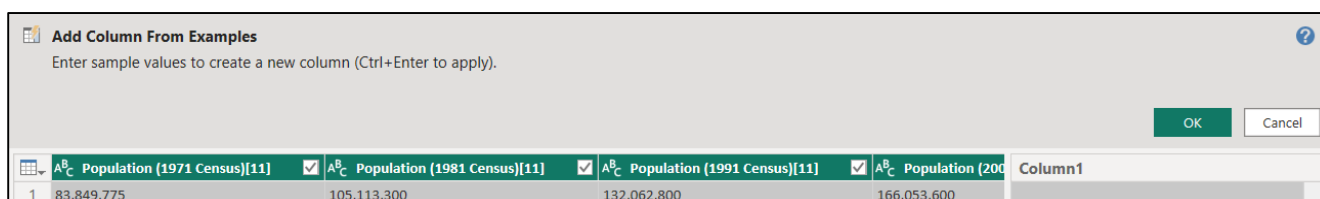
✓ No syntax errors have been detected.

■ **Add supplemental fields of data:**

Power Query allows you to enhance a data set by adding additional data fields. Using the **Column From Examples** feature, new columns of data can be added to the Power Query Editor. Basically, this function provides a sample of how the data needs to look and can then automatically determine what transformations are needed to achieve this. The application waits for the user to provide some sample values for the new column, then it will perform the necessary calculation to generate the values in the new column and then fill it automatically.

To use this function in a concrete example, the following steps must be performed:

1. **Load the dataset** into the Power Query Editor.
2. Select **Add Column** tab → **Column From Examples** (section **General**) → select **From All Columns**:



It can be seen that **a new, empty column was created** (Column 1) at the right of the existing data columns and **a new formula bar appeared above the data**.

In this example, the goal is to enter codes associated with the states listed in the first column of the data set:

3. **Double click in the first cell in the column Column 1** → type the code UP → **OK**.

Double click in the second cell in the column Column 1 → type the code MA → **OK**.

One can immediately see how the **entered values were automatically translated into a M query, which applied to each row in the new data column**:

Add Column From Examples

Enter sample values to create a new column (Ctrl+Enter to apply).

Transform: `let splitStateorunionterritory = Splitter.SplitTextByDelimiter(" ", QuoteStyle.None)/([State or union territory]) in Text.Combine(Text.Combine(List.Transform(splitStateorunionter...`

OK Cancel

	A ^B C State or union territory	A ^B C Population (1951 Census)[11]	A ^B C Population (1961 Census)[11]	Custom
1	Uttar Pradesh	60,274,800	70,144,160	UP
2	Maharashtra	32,002,500	39,554,900	MA
3	Bihar	29,085,900	34,841,490	B
4	West Bengal	26,300,670	34,926,000	WB
5	Madhya Pradesh	18,615,700	23,218,950	MPA
6	Tamil Nadu	30,119,680	33,687,100	TN
7	Rajasthan	15,971,130	20,156,540	RT
8	Karnataka	19,402,500	23,587,910	KT
9	Gujarat	16,263,700	20,633,305	GA
10	Andhra Pradesh	31,115,000	35,983,480	APA
11	Odisha	14,646,100	17,549,500	OA
12	Telangana	N/A	N/A	TG
13	Kerala	13,549,000	16,904,560	KA
14	Jharkhand	9,697,300	11,606,504	JH

4. Click **OK** → the transformation is completed and the new column is added to the data set.

5. Double-click the column header to rename the column:

A ^B C State or union territory	A ^B C Population (2001 Census)[11]	1 ² 3 Population (2011 Ce...	A ^B C State codes
Uttar Pradesh	166,053,600	199581477	UP
Maharashtra	96,752,500	112372972	MA
Bihar	82,879,910	103804630	B
West Bengal	80,221,300	91347736	WB
Madhya Pradesh	60,385,090	72597565	MPA
Tamil Nadu	62,111,390	72138958	TN
Rajasthan	56,473,300	68621012	RT
Karnataka	52,734,986	61130704	KT
Gujarat	50,597,200	60383628	GA
Andhra Pradesh	75,728,400	49386799	APA
Odisha	36,707,900	41947358	OA
Telangana	N/A	35193978	TG
Kerala	31,839,000	33387677	KA
Jharkhand	26,946,070	32988134	JH

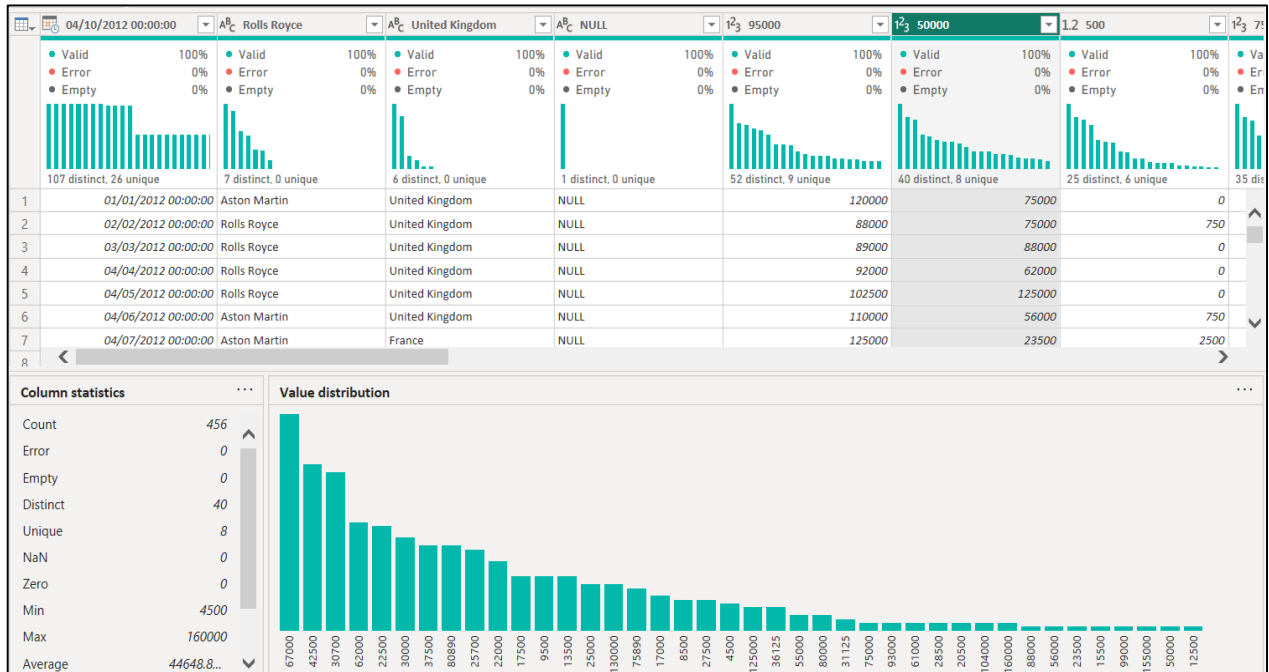
Note: The **Column from Examples** function menu has two options for choosing data samples: **From All Columns** and **From Selection**. If the second option is used, then only the data from the selected columns will be visible when double-clicking inside the new column to see data samples.

5.3.4 Power Query profiling tools for data analysis

To evaluate the data quality of the data model and to discover additional characteristics of data, the application has the following powerful dedicated analysis tools: **Column quality**, **Column distribution** and **Column profile**, which become available by activating the corresponding settings in the **View** tab:

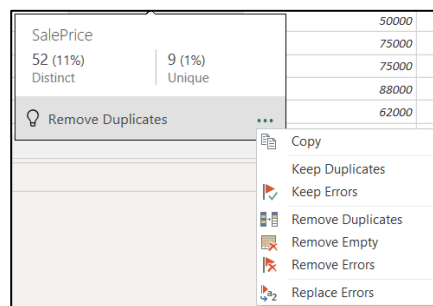
File	Home	Transform	Add Column	View	Tools	Help
 Query Settings	<input checked="" type="checkbox"/> Formula Bar	<input type="checkbox"/> Monospaced	<input checked="" type="checkbox"/> Column distribution	 Go to Column	<input type="checkbox"/> Always allow	 Advanced Editor
	<input checked="" type="checkbox"/> Show whitespace	<input checked="" type="checkbox"/> Column profile	<input checked="" type="checkbox"/> Column quality			
Layout	Data Preview			Columns	Parameters	Advanced Dependencies

After activating these settings, above the data columns one can view certain diagrams and statistics dedicated to the evaluation of data quality:



► **Column quality:** is expressed in percentage values of valid data (shown in green), of errors (shown in red) and of empty fields (shown in dark gray).

If the mouse is positioned on a column, a new box appears with available data about the numerical distribution of the quality of the values in that column; clicking on the ellipsis button (...) opens another menu with quick action options for operations on values:

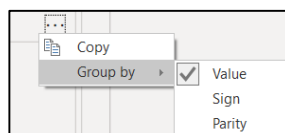


From this menu, the desired option for correcting the anomalies will be selected.

► **Column distribution:** shows both numerically and graphically the unique and distinct values that a column contains (values that are calculated from the first 1,000 rows returned by the query).

► **Column profile:** offers a more detailed view of the data in the selected column through two additional sections, **Column statistics** and **Value distribution**.

By selecting the button with the ellipsis (...) in the upper right corner of these areas, an additional menu is displayed with shortcuts for copying the data displayed in the section or grouping the values from the diagram according to a set of options:



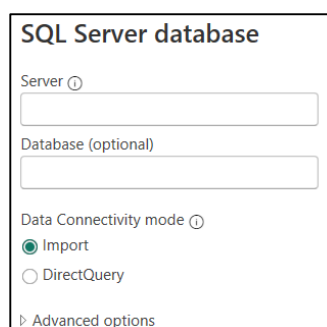
Chapter VI. Designing a data model

6.1 Overview of Power BI for data modeling

After the data loading and transformation stage in the Power BI application, the resulting data set, which is structured in several tables, must be converted into a coherent data model, so that the obtained information is visualized correctly and efficiently, in order to be able to be later analyzed to create useful insights.

Power BI Desktop is an integrated work environment that allows to build a data model going through the steps of connecting to data sources, transforming data, and up to creating reports and dashboards. Building a data model involves building visual representations of the connections between dataset tables, detailing the individual attributes contained in these data structures, and organizing the data into a structured format for analysis.

First of all, to build the data model, run Power BI Desktop and connect to the data source. Here comes an important observation, namely the way to connect to certain data sources: the Import mode or the DirectQuery mode:



SQL Server database

Server

Database (optional)

Data Connectivity mode Import DirectQuery

Advanced options

- **Import** is a data connection method that involves loading data from various sources into the Power BI analytical engine. This engine loads the data locally, in an internal data model in the Power BI file, and later, the modification of the data is done at the level of the local copy, which was imported.

The advantage is represented by the ability to work efficiently with large data sets, but also the high speed of data analysis through various queries and visualizations.

- **DirectQuery** is an alternative data connection method that doesn't load data into Power BI Desktop, so it doesn't store data locally. The method connects Power BI Desktop live to the data source, so data is queried directly from the source database in real time.

The advantage is the speed of updating the information, because when it changes in the data source, the data is also updated in Power BI Desktop. Also, the fact that the local computer's memory is not used, allows the management of very large data sets.

Once the data source is located, all related tables are loaded into a new model, simultaneously, using the Navigator dialog box:

Check the selection boxes corresponding to all the data model tables → click the Load button:

Navigator

CarSalesData.xlsx (8)

- Table1
- Clients
- Colors
- Countries
- DateDimension
- InvoiceLines
- Invoices
- Stock

Suggested Tables (1)

- Table 1 (Stock)

Stock
Preview downloaded on 20 November 2023

StockID	Make	Model	ColorID	VehicleType	Co
1	Rolls Royce	Camargue		1 Saloon	
2	Aston Martin	DBS		2 Coupe	
3	Rolls Royce	Silver Ghost		3 Saloon	
4	Rolls Royce	Silver Ghost		2 Saloon	
5	Rolls Royce	Camargue		5 Saloon	
6	Rolls Royce	Camargue		8 Saloon	
7	Aston Martin	DBS		9 Coupe	
8	Aston Martin	DB7		1 Coupe	
9	Aston Martin	DB9		2 Coupe	
10	Aston Martin	DB9		4 Coupe	
11	Aston Martin	DB4		6 Coupe	
12	Aston Martin	Vantage		5 Coupe	
13	Aston Martin	Vanquish		6 Coupe	
14	Aston Martin	Rapide		7 Coupe	
15	Aston Martin	Zagato		8 Coupe	
16	Rolls Royce	Silver Ghost		5 Saloon	
17	Rolls Royce	Wraith		4 Saloon	
18	Rolls Royce	Silver Ghost		3 Saloon	
19	Rolls Royce	Camargue		2 Saloon	
20	Rolls Royce	Silver Shadow		1 Saloon	
21	Rolls Royce	Silver Seraph		1 Saloon	
22	Rolls Royce	Silver Ghost		7 Saloon	

Buttons: Load, Transform Data, Cancel

Alternatively, to build the data model, the data can be loaded using the Power BI Query Editor:

Click the **Transform data** button to open Query Editor:

Processing Queries
Determining automatic transformations...

- Table1 (2) Evaluating...
- Clients (3) Evaluating...
- Colors (3) Evaluating...
- Countries (3) Evaluating...
- DateDimension (3) ✓

Buttons: Skip

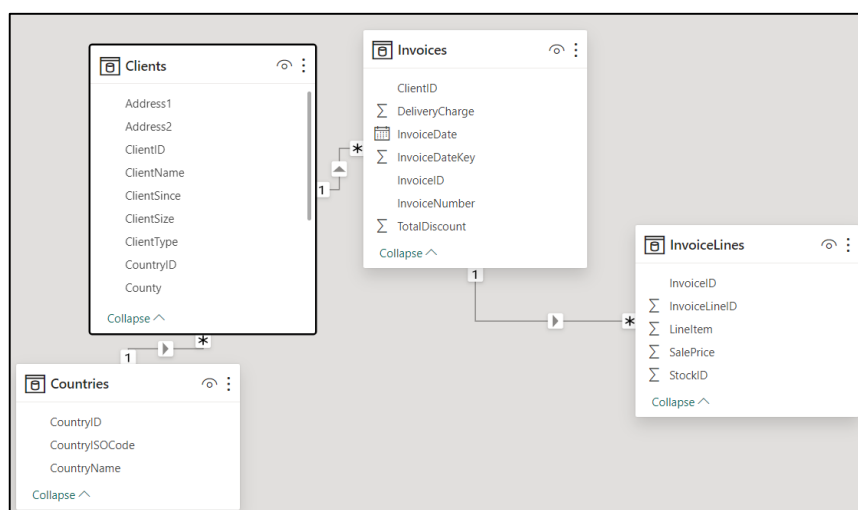
Close & Apply, New Source, Recent Sources, Enter Data, Close, New Query

Queries [8]

- Table1
- Clients
- Colors
- Countries
- DateDimension
- InvoiceLines
- Invoices
- Stock

After **closing the Query Editor**, by selecting the **Close & Apply** option (Close section), **Power BI Desktop** will load the data tables into the model and automatically build relationships among them:

Select **Home** tab → **Model view** (left pane):



It can be seen that at this stage, the application has not correctly and completely established all the relationships between the tables, therefore the data model must be perfected.

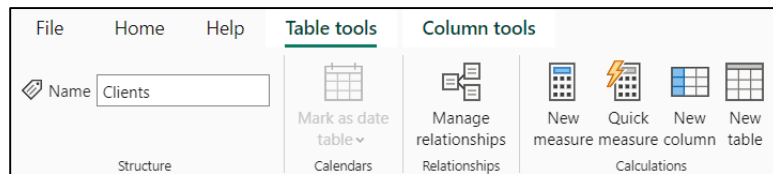
But, returning to the **Table view** (changing the view in the left panel):

ClientID	ClientName	Address1	Address2	Town	County	PostCode	Region	OuterPostcode	CountryID	ClientType
1	Aldo Motors	4, Scale Street		Uttoxeter	Staffs	ST17 99RZ	East Midlands	ST		1 Wholesaler
2	Honest John	99a Baker Street	NULL	London		NSW1 1A	Greater London Authority	EC		1 Dealer
3	Bright Orange	17, Arcadia Way	NULL	Birmingham	NULL	B1 50AZ	West Midlands	B		1 Dealer
4	Cut'n Shut	Grange Avenue	NULL	Manchester	NULL	M1 5AZ	North West	M		1 Dealer
5	Wheels'R'Us	Buckingham Drive	NULL	London	NULL	SE1 4YY	Greater London Authority	NE		1 Dealer
6	Les Arnaqueurs	33, Rue Des Bleus	NULL	Paris	NULL	75010	NULL	NULL		2 Dealer
7	Crippen & Co	1012 Princess Street	NULL	Glasgow	NULL	G1 8GH	NULL	NULL		1 Dealer
8	Rocky Riding	5205 108th Ave	NULL	New York	New York	NULL	NULL	NY		3 Dealer

There is an important difference between this view (Table View of Power BI Desktop) and the Power BI Desktop Query Editor window, namely that in the Table View all the data of the model (data from all related tables) are available, while the Power BI Desktop window Query Editor displays only part of the data.

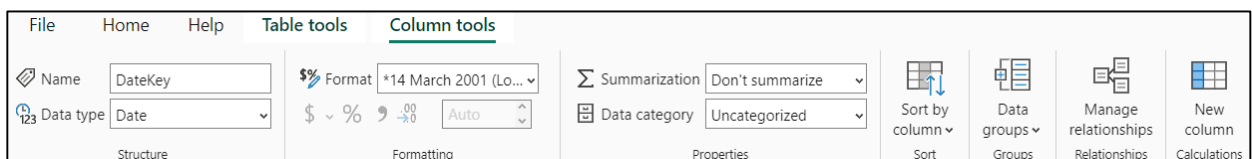
Also, the Ribbon bar menus are different and contain the following tools:

► The **Table tools** tab:



- **Name (Table name):** allows to modify the table name.
- **Mark as date table:** allows to transform the selected table as a date table.
- **Manage relationships:** allows to add, edit or remove joins (called relationships) between tables.
- **New measure:** allows to write a DAX expression to add a new value or calculation to a table.
- **Quick measure:** allows to create a measure using examples from a list of common calculations.
- **New column:** allows to add a new calculated column in the selected table using a DAX expression.
- **New table:** allows to write a DAX expression to create a new table.

► The **Column tools** tab (available when selecting a column):



- **Name (column name):** allows to modify the column name.
- **Data Type:** allows to modify the data type for a column.

- **Format:** allows to set how the values in that column are displayed.
- **Summarization:** allows to set the default way to summarize values for that column in visualizations.
- **Data category:** allows to define that a certain column is of a specific category; it is used for columns containing various types of data used for maps or links and allows the selection of the following types: Uncategorized, Address, Place, City, County, State or Province, Postal code, Country, Continent, Latitude, Longitude, Web URL, Image URL, Barcode.
- **Sort by column:** allows to sort the data in one column by the contents of another column.
- **Data groups:** allows to create a new group to combine multiple values into one.

6.2 Specific operations that use the Table Tools and Column Tools

■ Apply a Summarization:

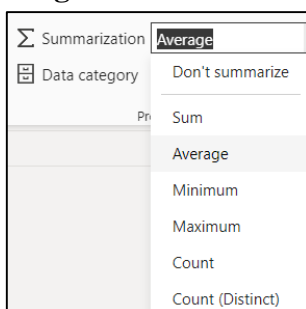
When viewing a diagram, or when creating a dashboard, it is necessary to modify the way in which a numerical field is aggregated, to display the sum of the numerical elements in the column or the total, average, etc. The application allows to set the default way to aggregate values for that field, when used in a visual.

To choose one of the aggregation methods, perform the following steps:

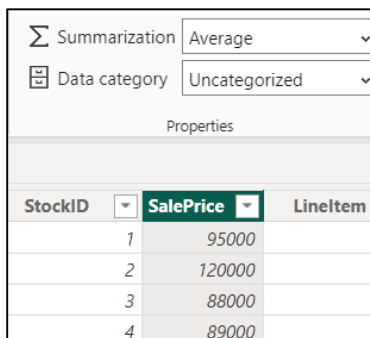
1. The **Table view** → in the **Fields** list → select **InvoiceLines** table → in Power BI Desktop window the **InvoiceLines** table is displayed:

InvoiceLineID	InvoiceID	StockID	SalePrice	Lineltem
3	1	1	95000	1
4	2	2	120000	1
5	3	3	88000	1
6	4	4	89000	1
7	5	5	92000	1
8	6	6	102500	1
9	7	7	110000	1
10	8	8	125000	1
11	9	9	130000	1
12	10	10	75000	1
13	10	11	68500	2
14	11	12	95000	1

2. Select **SalePrice** column → The **Column tools** tab becomes active → click the pop-up to the right of the **Summarization** option → select **Average**:



3. In the ribbon, the **Summarization** option will show for the SalePrice field the default value that was set, Average:



Notes:

1) The **Summarization** menu has the following **available options**:

- Do Not Summarize each value in this column is treated separately and is not summarized
- Sum adds up the values in this column
- Average returns an average value of the data in this column
- Minimum returns the smallest value of the data in this column
- Maximum returns the largest value of the data in this column
- Count returns the number of values in this column
- Count (Distinct) returns the number of distinct values in this column

2) The **Count** and **Count (Distinct)** options **can be applied to any type of data field**, but **the math aggregation options can only be applied to numeric data fields**.

■ Set a sort based on another column:

In reports or dashboards, the application allows to change the way of sorting the values of a certain column, from the default sort mode (for example, the Power BI application will sort the text fields by default, alphabetically) to a sort type that allow the recorded data to follow a personalized sorting (for example, chronologically, the months of the year appear in order of January, February, March, etc.).

Practically, in this case, the months of the year will be sorted according to a column that will contain the numbers 1, 2, ..., so that the name of the respective month is sorted according to the number contained in the other column.

To achieve this sorting mode, perform the following steps:

1. The **Table view** → in the **Fields** list → select **MonthAndYear** table → in Power BI Desktop window the **MonthAndYear** table is displayed:

MonthFull	Mon	Quar	Quarte	Quar	Year	Quar	Month	Mon	MonthName	Mon	Quar	Quar	Year
April	Apr	2	Quarter 2	Q2	20142	Q2 2014	01 April 20	01 April	April	Apr	Quarter	Qtr 2 20	201404
April	Apr	2	Quarter 2	Q2	20122	Q2 2012	01 April 20	01 April	April	Apr	Quarter	Qtr 2 20	201204
April	Apr	2	Quarter 2	Q2	20122	Q2 2012	01 April 20	01 April	April	Apr	Quarter	Qtr 2 20	201204
April	Apr	2	Quarter 2	Q2	20122	Q2 2012	01 April 20	01 April	April	Apr	Quarter	Qtr 2 20	201204
August	Aug	3	Quarter 3	Q3	20143	Q3 2014	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201408
August	Aug	3	Quarter 3	Q3	20143	Q3 2014	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201408
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208
August	Aug	3	Quarter 3	Q3	20123	Q3 2012	01 August	01 August	August	Aug	Quarter	Qtr 3 20	201208

Data

Search

- > MonthAndYear
- MonthFull
- MonthName
- MonthNameAbbr
- Σ MonthNum
- QuarterAbbr
- QuarterAbbrAndYear
- QuarterAndYear

2. In this example, a column of months is sorted alphabetically. To sort a column of month names chronologically, we need a column containing a number for each month. Thus, we switch to the Power BI Desktop Query Editor:

Select **Home** tab → **Transform data** → an additional column is added, **MonthNumber** (which will be used as the support column for sorting the data):

	r	MonthAndYear	A ^B _C MonthName	A ^B _C MonthNameAbbr	A ^B _C QuarterAndYear	A ^B _C QuarterAndYearAbbr2	I ² ₃ YearAndMonthNum	I ² ₃ MonthNumber
1	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	1
2	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	2
3	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	3
4	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	4
5	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	5
6	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	6
7	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	7
8	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	8
9	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	9
10	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	10
11	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	11
12	1/2012	01/01/2012	January	Jan	Quarter 1 2012	Qtr 1 2012	201201	12

3. Select **Home** tab → **Close&Apply**:

4. Select **MonthFull** column → In the **Table view** → The **Column tools** tab becomes active → click the pop-up to the right of the **Sort by column** option → from the available columns select the supporting column, **MonthNumber**:

Sort by column

- MonthFull
- DateKey
- MonthAbbr
- MonthAbbrAndYear
- MonthAndYear
- MonthName
- MonthNameAbbr
- MonthNum
- MonthNumber

MonthNumber ↑

1
2
3
4
5
6
7
8
9
10
11
12

5. In the respective visualization, a chronological sorting will be automatically performed, in the order of the months of a year.

Note: In general, custom sorting is applied to data fields that contain information about months of the year or days of the week.

■ **Organizing data in groups:**

In order to be used in reports or dashboards, the Power BI application allows the organization of data fields (both numerical fields and text fields) by grouping or classifying similar rows of data in a table.

To achieve this way of organizing data, perform the following steps:

1. The **Table view** → in the **Fields** list → select **Clients** table → in Power BI Desktop window the **Clients** table is displayed:

Town	County	PostCode	Region	OuterPostcode
Zurich	NULL	NULL	NULL	NULL
Uttoxeter	Staffs	ST17 99RZ	East Midlands	ST
Telford	NULL	TF6 9RR	West Midlands	NULL
Stuttgart	NULL	NULL	NULL	NULL
Shrewsbury	NULL	SY10 9AX	West Midlands	TF
San Francisco	California	NULL	NULL	CA
Portland	Oregon	NULL	NULL	OR
Pittsburgh	Pennsylvania	NULL	NULL	PA
Paris	NULL	75010	NULL	NULL
Newcastle upon Tyne	NULL	NE3 3SS	North East	NE
New York	New York	NULL	NULL	NY
Mason	Ohio	NULL	NULL	OH
Marseille	NULL	13002	NULL	NULL

2. Select **Town** column → The **Column tools** tab becomes active → click **Data groups** option → select **New data groups**:

3. In the **Groups** window that opens:

- in the **Ungrouped values** section → **select the items to be grouped.**

- in the **Groups and members** section → **double click on the group name to rename the group.**

4. Click **OK** → **A column containing the group associated with that row will be added in the table:**

Town (groups)
Shrewsbury
San Francisco
Portland
Pittsburgh
Paris
Newcastle upon Tyne
New York
Mason
Marseille
Manchester
Madrid
Lyon
Louisville
London
London
Liverpool
Gloucester
Avignon & Geneva & Glasgow - first priority group
Avignon & Geneva & Glasgow - first priority group
Bellevue & Birmingham & Franklin - second priority group

■ **Creating Data Categories:**

In order to control how the data is aggregated and displayed in reports or dashboards, the Power BI application allows data columns to be assigned geographic categories (such as City, Country, Continent, etc.) or hyperlinks to various websites or documents. Otherwise, without an assigned data category, the application would have to guess what that data represents.

To categorize data for this purpose, perform the following steps:

1. The **Table view** → in the **Fields list** → select **Clients** table → in Power BI Desktop window the **Clients** table is displayed:

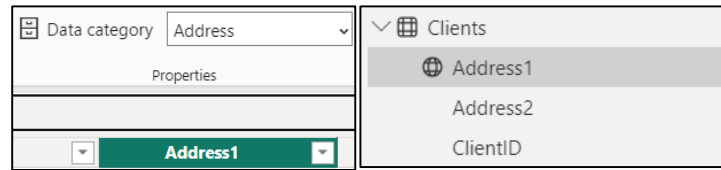
Address1	Address2	Town	County	PostCode	Region
NULL	NULL	Zurich	NULL	NULL	NULL
4, Scale Street		Uttoxeter	Staffs	ST17 99RZ	East Midlands
NULL	NULL	Telford	NULL	TF6 9RR	West Midlands
NULL	NULL	Stuttgart	NULL	NULL	NULL
NULL	NULL	Shrewsbury	NULL	SY10 9AX	West Midlands
13550 Market Street	NULL	San Francisco	California	NULL	NULL
1414 NW Northrup Street	NULL	Portland	Oregon	NULL	NULL
30 Isabella St	NULL	Pittsburgh	Pennsylvania	NULL	NULL
33, Rue Des Bleus	NULL	Paris	NULL	75010	NULL
NULL	NULL	Newcastle upon Tyne	NULL	NE3 3SS	North East
5205 108th Ave	NULL	New York	New York	NULL	NULL

2. Select **Address1** column → The **Column tools** tab becomes active → click the pop-up to the right of the **Data category** option → from the available options select **Address**:

Data category	Uncategorized
	Uncategorized
	Address
Address1	Place
NULL	City
4, Scale Street	County
NULL	State or Province
NULL	Postal code
13550 Market Street	Country
1414 NW Northrup Street	Continent
30 Isabella St	Latitude
33, Rue Des Bleus	Longitude
NULL	Web URL
5205 108th Ave	Image URL
4605 Duke Drive	Barcode
NULL	
Grange Avenue	
NULL	
NULL	

3. Now the **Address** category for this data field will be displayed in the **Data category**, and the application will be able to correctly use the content of this field in reports or dashboards.

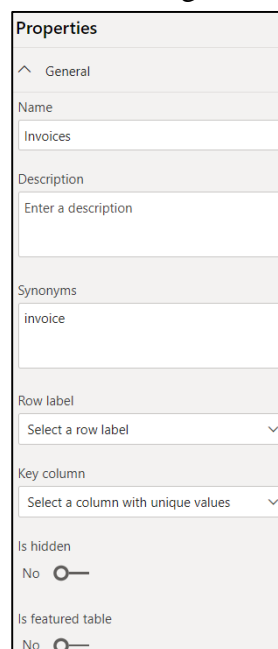
In addition, after classifying the data as Address, in the **Fields** list one can see an icon with a small globe displayed in front of the column name:



■ **Setting the properties of columns and tables:**

Tables and columns have various properties, which can be viewed, updated or configured by accessing the **Properties pane** in **Model view**. After selecting an object, its properties become visible in the Properties pane:

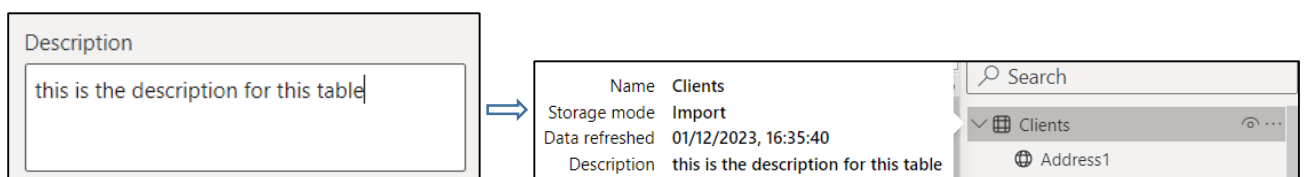
► **For tables**, the following features can be configured:



Name: allows to configure the table name.

Description: allows to edit a description, an explanation for the table.

This description will then be visible in all views: in Data view, as well as in the Model view and in the Report view, as can be observed in the following screenshot:



Synonyms: allows to add a variety of terms as additional names for tables, so that the application identifies that table, even if a different name is used.

Row label: allows the selection of that column that best identifies each row in a table.

Key column: allows the selection of that column that has unique values for every row.

Is hidden: allows to hide a table so that it is kept in the data model, but is no longer visible from the Fields pane.

Is featured table: this property makes the table's data accessible via Excel, more precisely, it allows the column that was selected as Row label to be used in Excel to identify the row quickly.

Storage mode: shows the storage mode of the table.

► **For columns,** the following features can be configured:

Properties	
^ General	
Name	DateKey
Description	Enter a description
Synonyms	date key, DateKey, date, key
Display folder	Enter the display folder
Is hidden	No <input type="radio"/>

Formatting	
^ Advanced	
Data type	Date
Date time format	*14 March 2001 (Long Date)
Sort by column	DateKey (Default)
Data category	Uncategorized
Summarize by	None
Is nullable	Yes <input checked="" type="radio"/>

Name: allows to configure the column name.

Description: allows to edit a description, an explanation for the column.

Synonyms: as in the case of tables, allows to add a variety of terms as additional names for columns, so that the application identifies that column, even if a different name is used.

Display folder: allows grouping columns from a table into display folders.

Is hidden: allows hiding a column, so that it is kept in the data model, but not visible from the Fields pane.

Data type: allows to select one of the available data types, specifying that these data types are different from those available in Power Query.

Format: in this box list, different formatting properties will be displayed, depending on the type of data: numeric, text, calendar date type, etc.

Sort by column: allows to sort the data in one column by the contents of another column.

Data category: property that can be useful for reports and dashboards, allowing to define that a certain column is of a specific category.

Summarize by: property that determines the aggregation mode of the column when it is used in reports and dashboards.

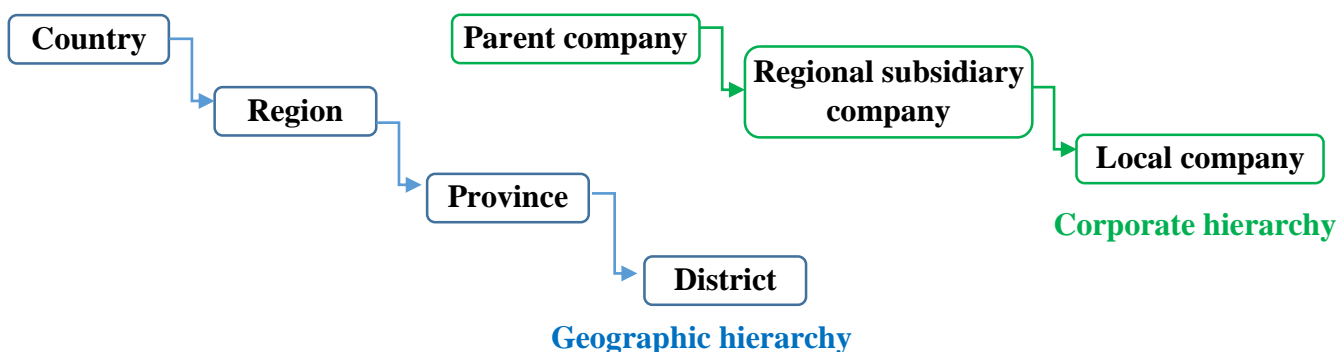
Is nullable: allows disallowing null values for a column.

6.3 Adding hierarchies and measures in the data model

► A hierarchy is an abstract data structure that shows how data is organized into levels of detail, thus simplifying complex data sets while providing a more detailed view of information. Hierarchy is useful for drilling down and exploring data in more advanced views, such as detailed reports, charts, or interactive maps.

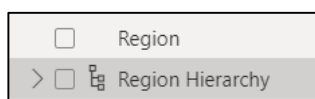
In Power BI, a hierarchy is created by classifying a set of fields hierarchically on multiple levels (at least two levels), so that one level is the parent of another level in the hierarchy.

A classic hierarchy pattern in a Power BI data model is a calendar hierarchy, which typically includes years, quarters, months, and days. But there are other types of hierarchies obtained by grouping other types of associated data fields together, such as product hierarchies or geographic hierarchies, etc:



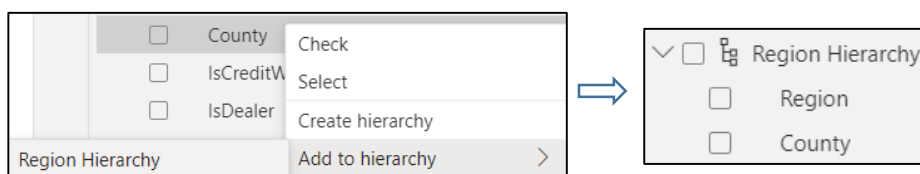
To create a new hierarchy from the data in the Clients table, so that we set Region as level 1, then County as level 2 and Town as level 3, perform the following steps:

1. The **Report view** → in the **Fields** list → select **Clients** table.
2. **Right-click on the field to be set as level 1 of the hierarchy (Region field) → Create hierarchy:**



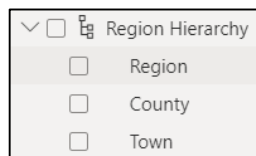
It can be seen that a new hierarchy has appeared in the data model, with the name of the chosen field, titled Region Hierarchy, marked by a hierarchy icon next to it and having the option to expand to the hierarchy fields.

3. **To add another level to the hierarchy → right click on the field that will be the subcategory (County field) → Add to hierarchy:**



It can be seen that the County field was added to the hierarchy as the second level of the hierarchy.

4. Similarly, **the Town field is added as the third level in the hierarchy:**



5. To rename the hierarchy: **Right click on the name of the hierarchy → Rename.**

Note: The fields within the constructed category are not duplicates of the original columns, but only references to those columns. Therefore, to avoid confusion, the original columns can be hidden:

Right click on that field → Hide.

► In data modeling with Power BI, a measure is a calculated field that performs a specific task to help analyze data. They are useful when working with aggregate values, so when complex calculations must be performed, which would not be possible using simple fields or filters of the data model.

In Power BI, an explicit measure is defined using the DAX () language, which includes many types of functions and operators that can be used in writing complex calculations and aggregations with subsets of data of the analyzed model. For example, one can write a DAX measure that realizes a sum of several values, which will later be divided by a number. Once created, the measure can be used in any visualization that relies on that data model to display meaningful information.

To see how Power BI Desktop creates a measure, perform the following steps:

1. The **Table view** → in the **Fields** list → select **Table1** table → in Power BI Desktop window the **Table1** table is displayed:

OrderDate	City	Seller	Item	Units	Unit Cost
06 January 2021	Arad	Ion	Box	15	4.33
07 January 2021	Constanta	Ana	Map	40	15.66
08 January 2021	Constanta	Maria	Notebook	39	9.99
09 January 2021	Constanta	Gabriel	Pen	17	15.66
10 January 2021	Arad	Ion	Pencil	56	7
11 January 2021	Bucuresti	Ion	Sharpener	60	9.99
12 January 2021	Constanta	Mihai	Pencil	75	4.33
13 January 2021	Constanta	Maria	Pencil	90	9.99
14 January 2021	Arad	Sergiu	Pencil	32	4.33
15 January 2021	Bucuresti	Ion	Sharpener	60	8.99
16 January 2021	Constanta	Mircea	Pencil	90	9.99
17 January 2021	Bucuresti	Ion	Sharpener	29	4.33
18 January 2021	Bucuresti	Corina	Sharpener	81	15.66
19 January 2021	Constanta	Ion	Pencil	35	9.99

2. Select **Table tools** tab → **New measure (Calculations field)**:



3. In the **formula bar** that appeared → **edit the calculation formula:**

In this example, we want to calculate from the **Table1**, the total number of units sold by the seller **Ion**:

SALESforIon = `CALCULATE(SUM(Table1[Units]),FILTER(Table1,Table1[Seller]="Ion"))`

The formula consists of the following elements:

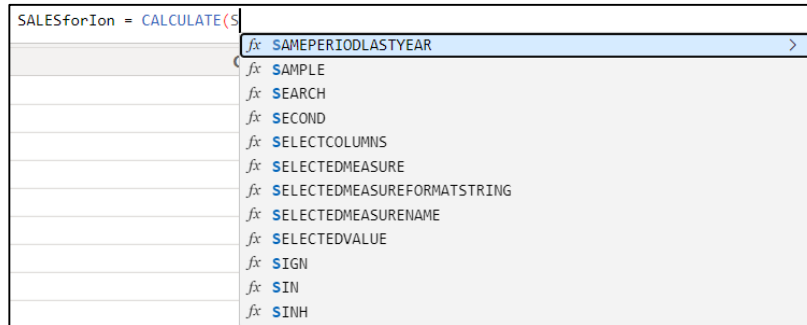
SALESforIon is the name assigned to the measure.

SUM(Table1[Units]) represents the sum of all the values found in the **Units** column of **Table1**.

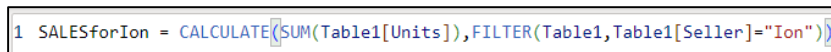
FILTER(Table1,Table1[Seller]="Ion") will filter the rows in **Table1** where the value in the **Seller** column is **Ion**.

CALCULATE () brings together the two expressions within the function: the main calculation, **SUM(Table1[Units])** which is filtered to retain only those rows for which **FILTER(Table1,Table1[Seller]="Ion")** has the value TRUE.

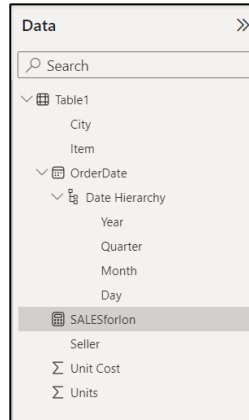
When editing the formula in the formula bar, a **list of suggested DAX formulas** appears in the **pop-up window**:



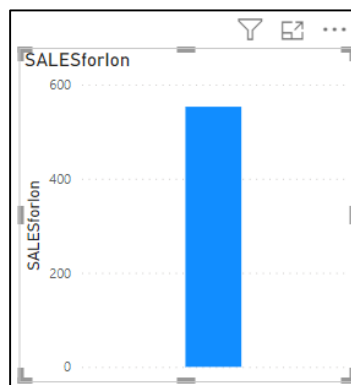
In the formula bar, the entered formula will look like this:



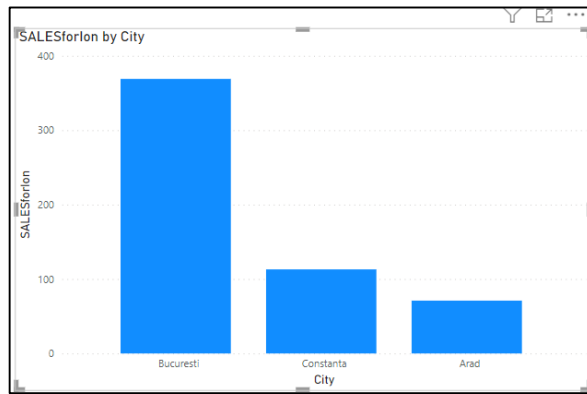
4. It can be seen that in the **Fields pane**, a new field containing the constructed measure has appeared:



5. Then **drag the new SalesforIon measure into the chart**:

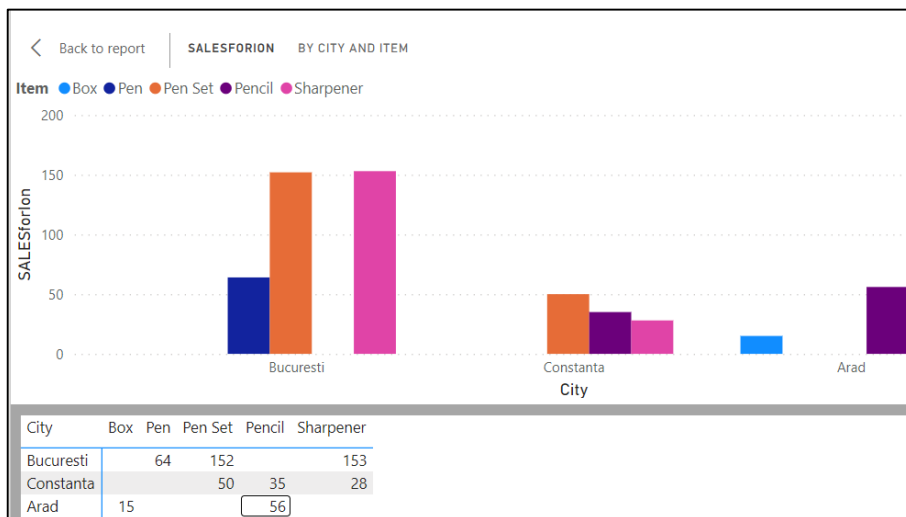


6. In the displayed graph, drag the **City** field from the **Table1**:



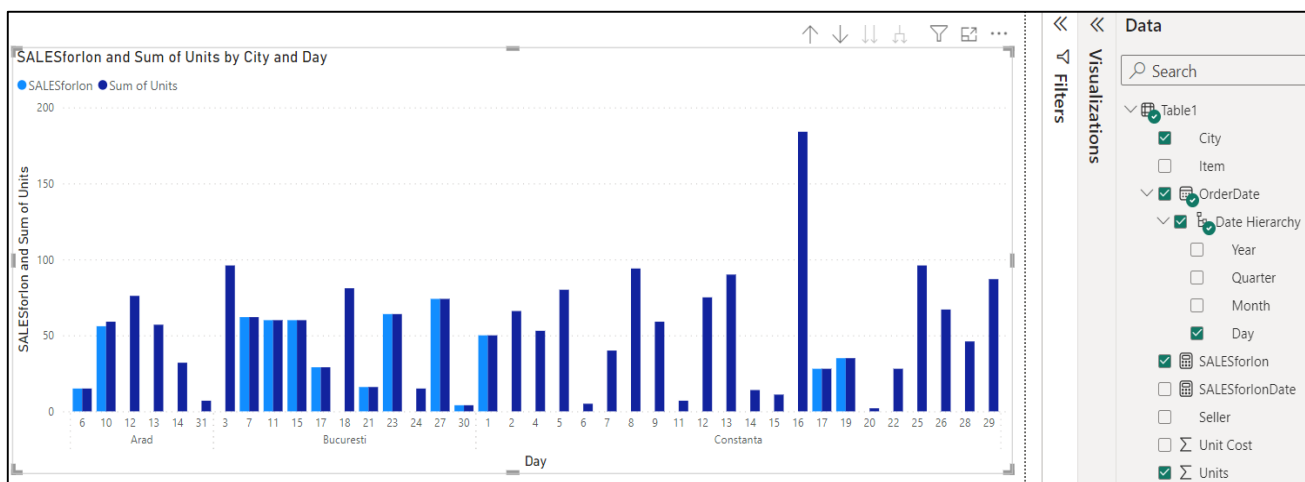
It can be seen that the graph changes, to display the sales recorded for the seller Ion, for each city. This is because the measure is a dynamic tool that responds to user interactions. Thus, based on the context of the data displayed in the views, the values calculated from the SalesforIon measure change in response to the interactions with the generated report.

7. Next, in the displayed graph, drag the **Item** field from the **Table1**:



It can be seen that the graph changes, to display the sales recorded for the seller Ion, for each city and for each type of item.

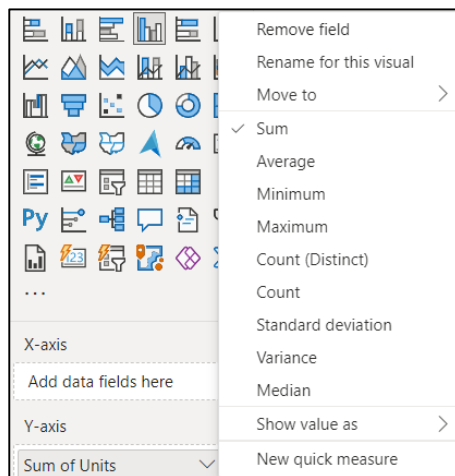
8: Next, we want to visualize in each city the total sales on the days on which there were sales and how many of these items were sold by the seller Ion. Correspondingly, in the graphic display area, we will drag the selected fields into the **Fields** pane:



Measures created by using DAX functions, like the one above, are called explicit measures. Apart from this type of measures, in the Power BI application there are default measures, created automatically, when a column is used in a visual. By dragging a numeric field such as Units or Unit cost into the field of a visual object, the application will automatically detect these default measures based on the data type of that column.

In the Fields pane you will see a Σ icon next to Units or Unit cost, which means that these fields contain numerical data that will be aggregated into an amount.

To verify that the aggregation type is a sum: Click on the arrow next to Units or Unit cost and see which of the aggregation methods is selected. In this case, this is indeed a sum:

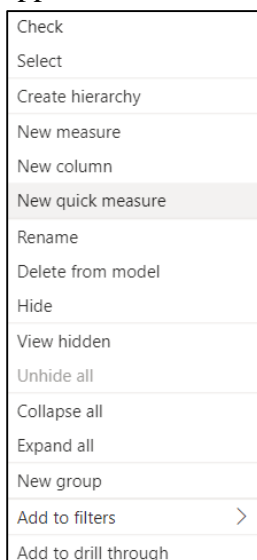


Also from this list one can select another type of aggregation (average, maximum, minimum, standard deviation, average, etc.) for the respective measure.

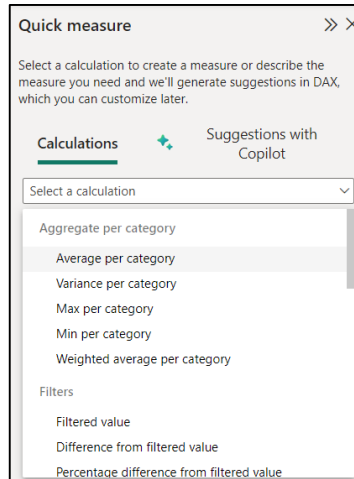
Another important functionality of the Power BI application is quick measures. A quick measure allows to easily create and apply calculations, without having to explicitly write DAX functions. Such a measure contains a set of DAX commands that run in the background, then the result will appear in the report.

To see how Power BI Desktop creates a quick measure, perform the following steps:

1. The **Table view** or **Report view** → in the **Fields** list → **select the ellipsis ...** next to any item → select **New quick measure** from the menu that appears:



2. In the **Quick measure window** that appears → in the **Select a calculation** list box → select an available quick measure to perform a specific calculation:



The application has the following quick measure calculation types:

Aggregate per category

- Average per category
- Variance per category
- Max per category
- Min per category
- Weighted average per category

Filters

- Filtered value
- Difference from filtered value
- Percentage difference from filtered value
- Sales from new customers

Time intelligence

- Year-to-date total
- Quarter-to-date total
- Month-to-date total
- Year-over-year change
- Quarter-over-quarter change
- Month-over-month change
- Rolling average

Totals

- Running total
- Total for category (filters applied)
- Total for category (filters not applied)

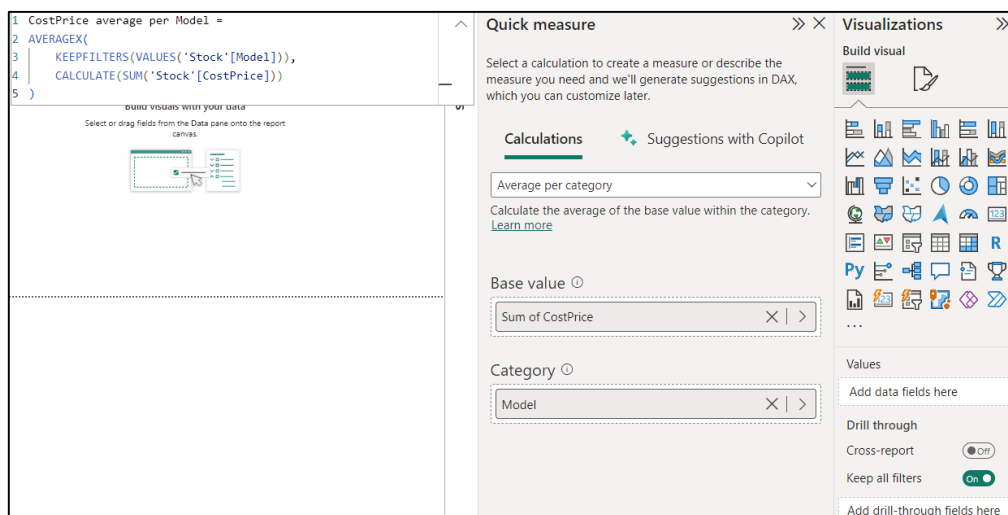
Mathematical operations

- Addition
- Subtraction
- Multiplication
- Division
- Percentage difference
- Correlation coefficient

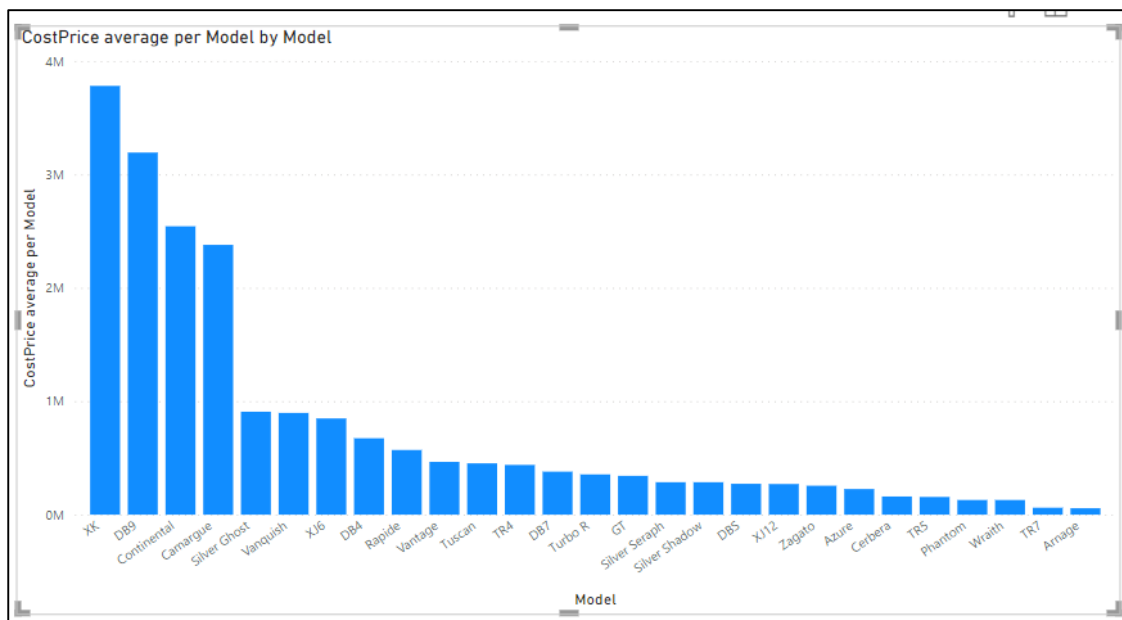
Text

- Star rating
- Concatenated list of values

For example, in the data model below, if we want to calculate the average selling price for each car model, the following parameters must be set in the Quick measure window:



In the graph display area, the following visualization is obtained:

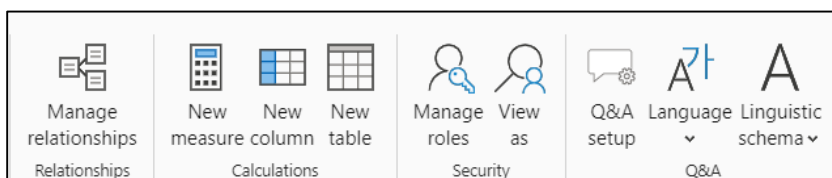


6.4 Building relationships in the data model

After loading the data in the Power BI Desktop application and cleaning it using the Power Query component, the data model becomes visible by accessing the Model view. This provides a visual representation of the tables and columns of the model in one place and allows to handle the relationships between the tables of the model.

It also offers the possibility to further improve the data model through calculations and specific operations, accessed from the Ribbon bar.

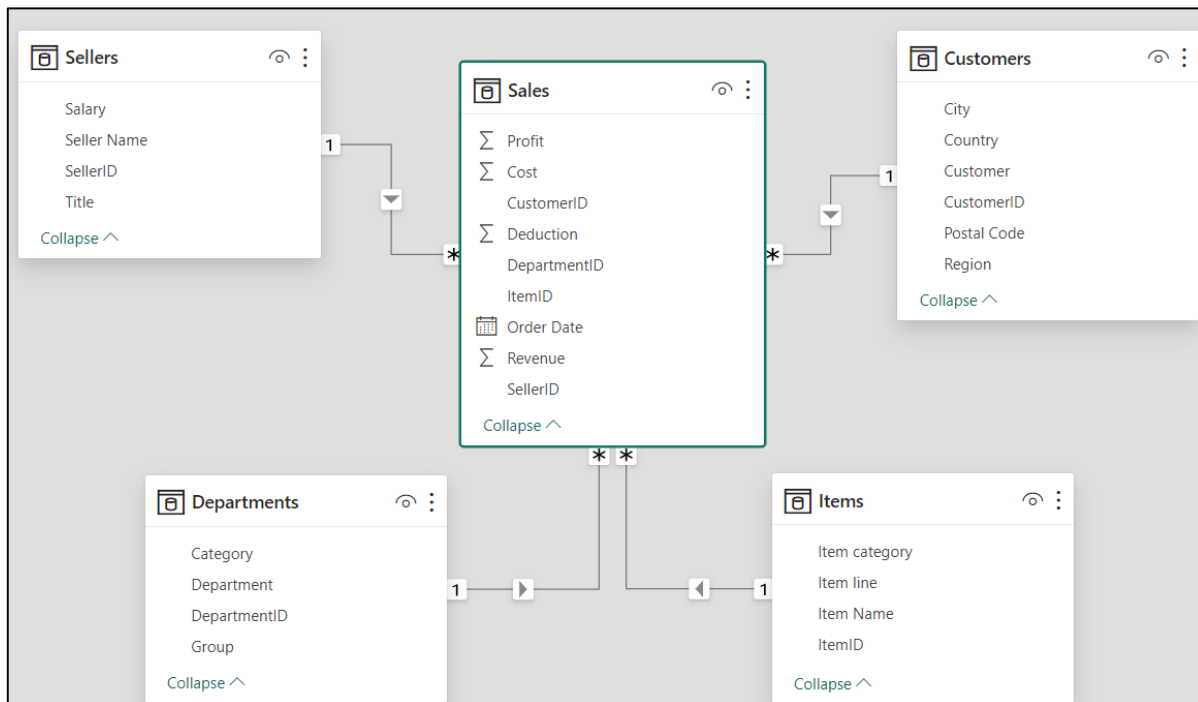
In **Model View**, the Ribbon bar contains the following specific tools:



- **Manage relationships:** allows to add, edit or remove joins (called relationships) between tables.
- **New measure:** allows to write a DAX expression to add a new value or calculation to a table.
- **New column:** allows to add a new calculated column in the selected table using a DAX expression.
- **New table:** allows to write a DAX expression to create a new table.
- **Manage roles:** allows to create, change or remove data access for particular users.
- **View as roles:** allows to see the data visible for a given role and permission.
- **Q&A setup:** teaches Q&A to better understand people’s questions and manage new terms.
- **Language:** allows to change the language used by Q&A.

- **Linguistic schema:** allows to import or to export information used by Q&A.

In the Model view, Power BI Desktop offers an overview of the data set, being able to see the names of the tables and columns, but not being able to view the records from these structures (the records being available through the Table view):



Also, it can be seen that links were automatically created between the tables of the data model. Relationships make it simple to apply filters between tables and perform cross-table calculations. But, although the application tries to correctly detect these relationships, it must always be checked that they were created correctly.

Before continuing with additional explanations regarding relationship management, some clarifications about the Power BI modeling technique are required. The application uses dimensional data modeling as a data modeling technique, which is based on several basic concepts:

- **Facts:** are the measurable data elements that users analyze and synthesize to better understand various aspects of the business. For example, for a sales business process, facts might include quarterly sales revenue, units sold, profit margins, etc.

- **Dimensions:** are descriptive data elements, business concepts, typically in the form of a noun as client, customer, seller, department, location, etc.

- A dimension contains several members (**attributes**) that describe the dimension. For example, the location dimension might include attributes such as city, district, country, region. Attributes are used to search, filter, or classify facts.

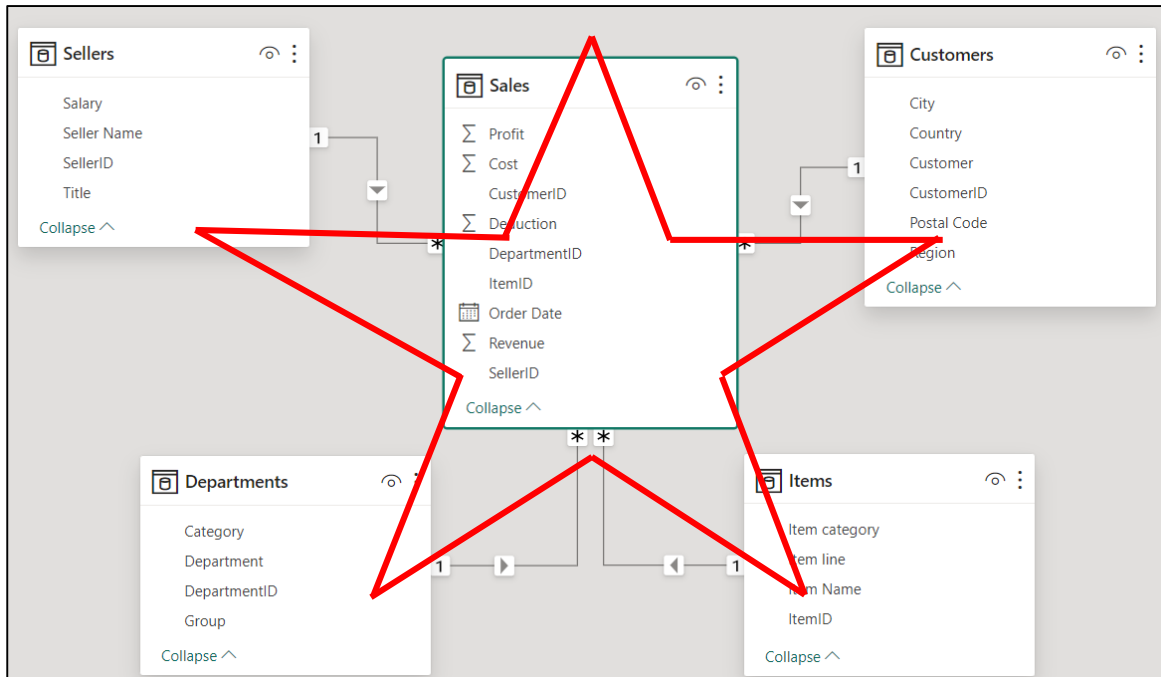
- **Fact table (sometimes known as data table):** is a primary table in dimension modelling, that contains facts (measurements) and one or more keys (which are usually numeric values) that link the dimension tables. The fact columns must be either a numeric or aggregated field.

- **Dimension table (sometimes known as lookup table):** contains several fields - dimensions that describe the fact table and a primary key that relates back to the fact table.

- One of the most important concepts in modeling a data set refers to how tables are organized in different types of data models (**schemas**). A scheme shows the logical relationships between the data, the way in which the tables are related to each other.

In Power BI analysis, the most common schemes are Flat, Star or Snowflake scheme:

► **Star schema:** there is one fact table in the middle, surrounded by some associated dimension tables:



The most common data modeling approach in Power BI modeling is to use a star schema. In the star schema, each of the dimension tables includes a primary key - a column in which each record (or row) in that column is unique, and the corresponding columns in the fact table are foreign keys. So, every foreign key of the fact table must have its counterpart in a dimension table.

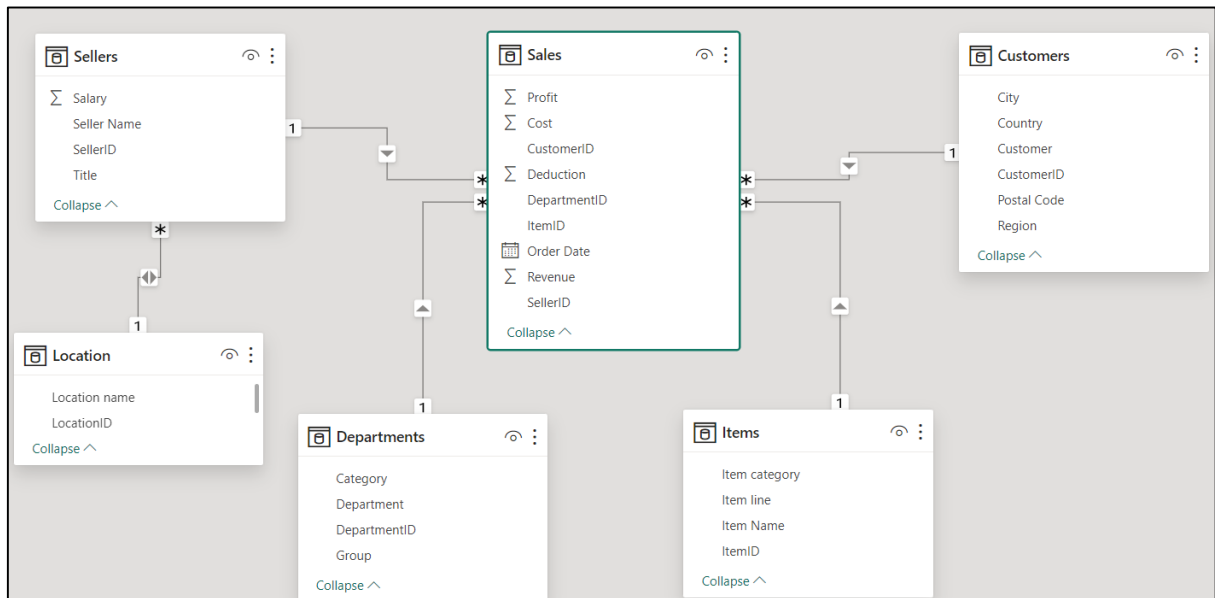
A relationship between the two tables can connect one or more rows. The type of join is determined by the cardinality of the relationship, which is a measure of the relationship between the rows in one table and the rows in another table:

- **one-to-many or many-to-one relationship (1:*)** describes a relationship in which multiple instances of a value in one column of one table are linked to a single corresponding unique instance in another column of the other table.

In the Star Schema, the fact table relates to every dimension table in a “many to one” relationship.

- **one-to one relationship (1:1):** happens when there are unique values in both tables per column.
- **many-to many relationship (*:*):** describes a relationship in which two tables share many values.

► **Snowflake schema:** is an extension of the star schema that allows relationships between dimension tables. So, the dimension tables are not only connected to the fact table but also to other dimension tables. The snowflake schema is obtained from the star schema by normalizing the dimension tables:



Normalization involves decomposing the tables into smaller (simpler) and better structured tables so that the modification of a field in a table propagates throughout the database through the relationships established between the tables. Normalization of data structures helps to reduce data redundancy and increases data integrity.

► **Flat schema** it consists of a single table that contains several columns and rows (all attributes are fully denormalized into a single table) or it can contain several tables that are isolated, have no links between them.

Coming back to creating relationships between database tables, there are several ways to achieve this:

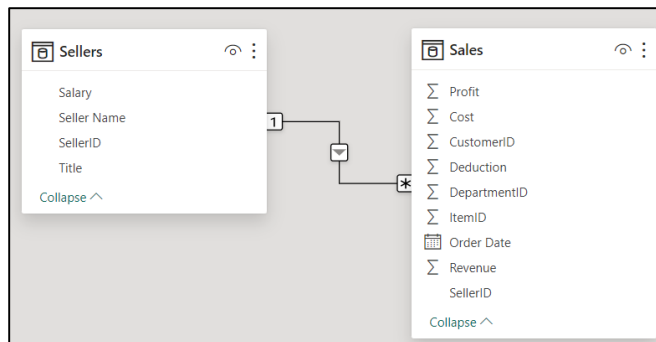
► **By drag and drop:**

1. Select **Home** tab → **Model view** (left pane):

The screenshot shows the 'Data' properties pane with the following structure:

- Item category
- Item line
- Item Name
- ItemID
- Location
 - Location name
 - LocationID
 - SellerID
- Sales
 - Profit
 - Cost
 - CustomerID
 - Deduction
 - DepartmentID
 - ItemID
 - Order Date
 - Revenue
 - SellerID
- Sellers
 - Salary
 - Seller Name
 - SellerID

2. Drag the SellerID field from the Sales table over the SellerID field in the Sellers table:



3. It can be seen that a line was created between the two tables, which indicates that the two tables are now joined.

Proceed in a similar way to create relationships between the other component tables of the model.

► **Manually:**

1. Select **Home** tab → **Model view** (left pane) → **Manage Relationships**.

2. In the **Manage Relationships window** which appears → select **New**.

3. In the **Create Relationships window** which appears:

- in the two list boxes → select the two tables that will be related.

- in **Cardinality** list box → select the relationship type:

Create relationship

Select tables and columns that are related.

Sales

Order Date	ItemID	CustomerID	SellerID	DepartmentID	Revenue	Cost	Deduction	Profit
25 June 2021	35	112	100	9	754	89	10	611
11 July 2021	35	112	100	9	754	89	0	611
15 August 2021	35	112	100	9	754	89	0	611

Sellers

SellerID	Seller Name	Title	Salary
1	Ioana Toma	Central Europe Sales Manager	78456
2	Maria Popa	Sales representative	4253
3	Dan Popescu	Director of sales	5689

Cardinality: Many to one (*:1) Cross filter direction: Single

Make this relationship active Apply security filter in both directions

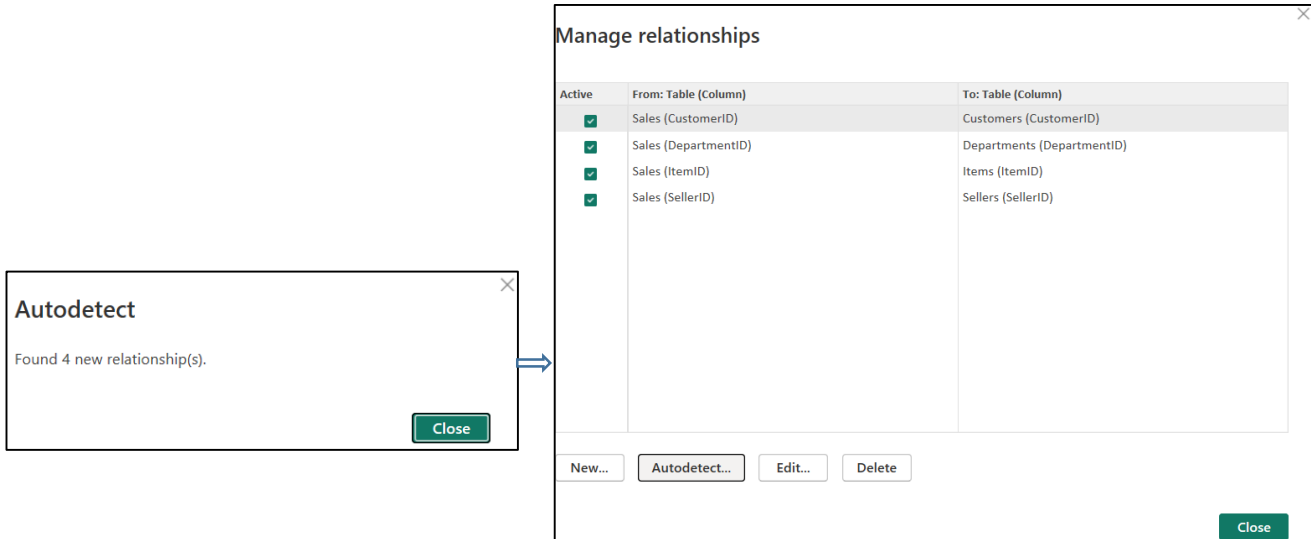
Assume referential integrity

4. Click **OK** → the two tables are now joined.

► **Automatically:**

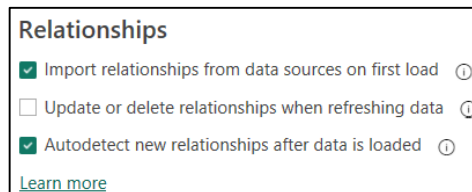
1. Select **Home** tab → **Model view** (left pane) → **Manage Relationships**.

2. In the **Manage Relationships window** which appears → select **Autodetect** → the following result is shown:



Note: The Power BI Desktop app identifies and defines relationships between tables by default. If we want to deactivate this setting, the following steps are performed:

File → **Options and settings** → **Options** → select **Data Load** (section **CURRENT FILE**) → **disable the default option "Autodetect new relationships after data is loaded"**:

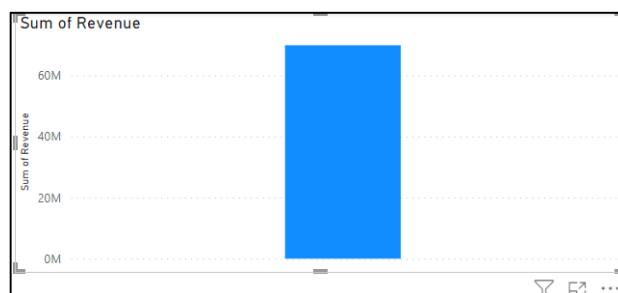


6.5 Create interactive visuals

■ Perform an analysis over time:

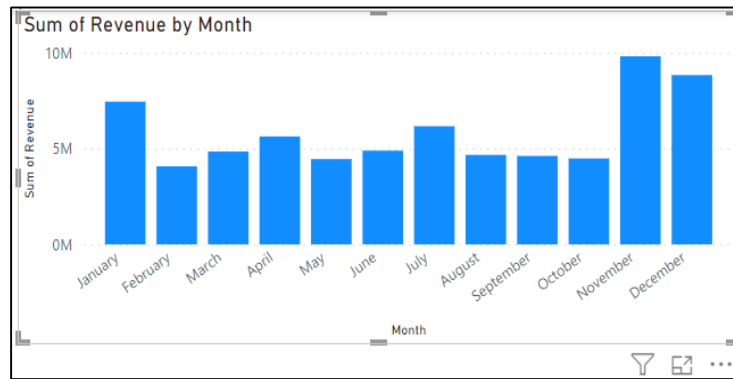
The following example will extract specific data from the model tables to perform an analysis over time:

1. The **Report view** → in the **Fields list** → **expand the Sales table** → **click the check box** to the left of the **Revenue** field:



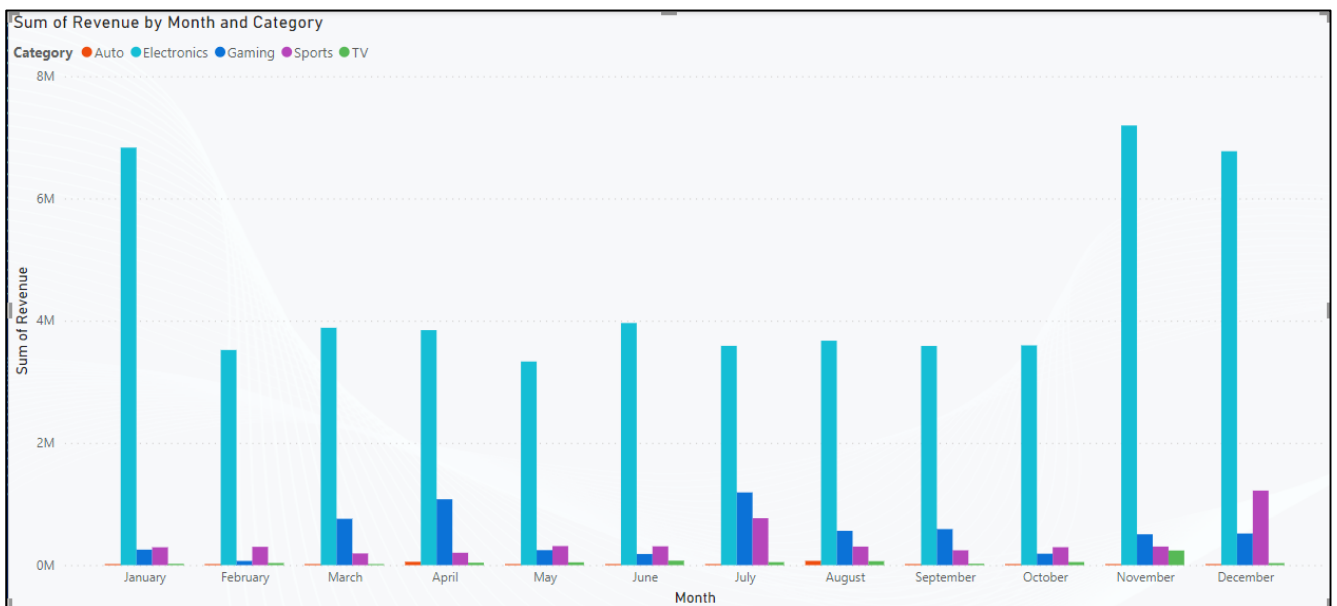
A column displaying the sum of revenue appeared.

2. **Click the check box** to the left of the **Month** field:



The sum of revenue by month was displayed.

3. Expand the **Departments** table → **Click the check box** to the left of the **Category** field:



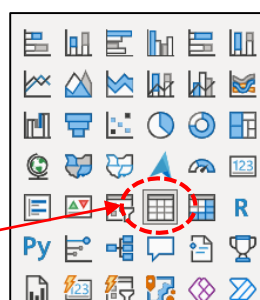
This will add the cumulative revenues per month and category of product.

■ **Create and format a table visual:**

Power BI enables the visualization of data in reports in the form of dynamic and interactive tables, which allow end users to make quantitative comparisons with large data sets, quickly and efficiently. In these tables, users can sort, filter, and perform complex calculations to obtain refined analysis and more targeted information that provides various insights from specific subsets of data.

To create a table visual, perform the following steps:

1. The **Report view** → in the **Visualizations** pane → select **Table visualization**:



2. **Expand the Sales table** → **click the check box** to the left of the **Revenue** field → **Click the check box** to the left of the **Month** field.

Expand the **Departments** table → **Click the check box** to the left of the **Category** field:

Month	Sum of Revenue	Category
October	763.99	Auto
May	920.45	Auto
September	3,107.24	Auto
January	3,925.49	Auto
December	4,593.84	Auto
February	4,754.58	Auto
June	5,675.73	Auto
November	8,098.31	Auto
July	8,344.66	Auto
March	12,124.03	TV
January	21,396.84	TV
September	22,539.33	TV
December	32,562.85	TV
February	34,016.72	TV
Total	44,707,304.78	

There are many ways to format a table to customize its visual elements:

► **Change table appearance by conditional formatting:**

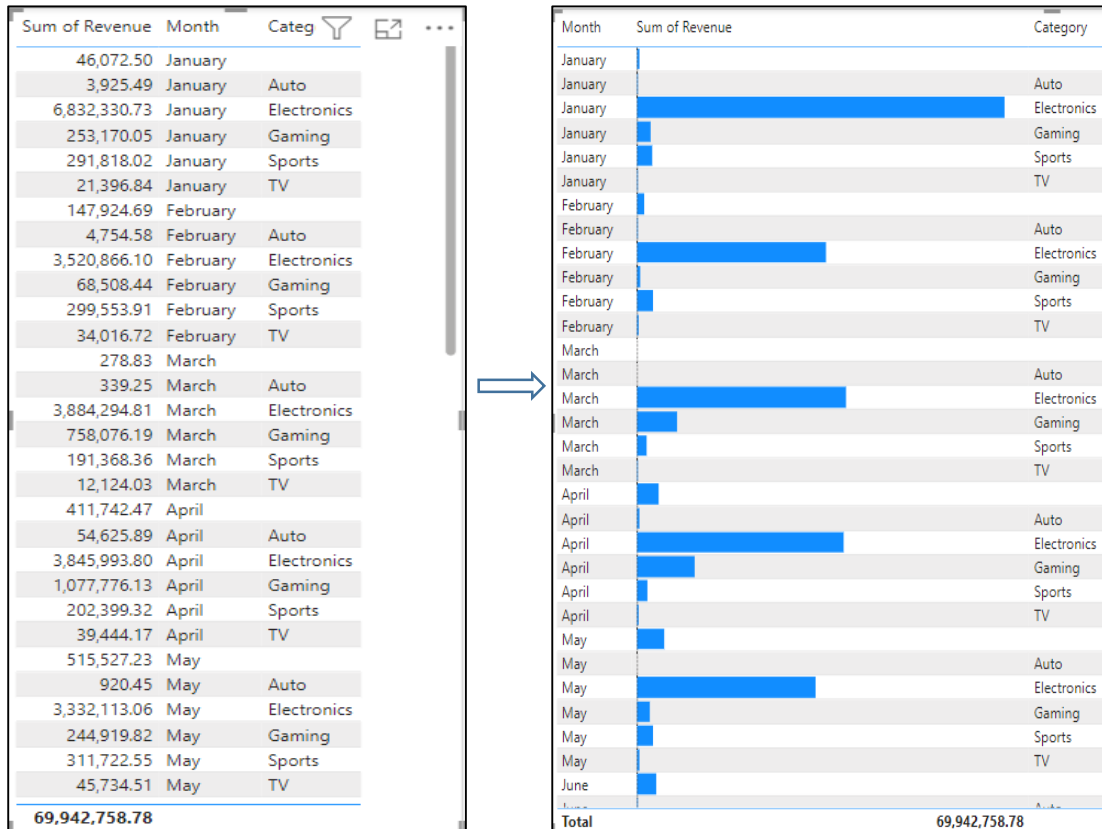
Conditional formatting can be applied to the constructed tables so that the data contained is relevant for the analysis performed. The Power BI Desktop application offers different types of formatting that can be applied to increase the degree of understanding the results provided by the data in the table:

► **Replace numbers with data bars:**

An interesting type of visualization is obtained by replacing numerical values with data bars, the length of each bar being proportional to the numerical value of the data represented.

1. **Visualizations** pane → **Expand the menu** for the **Sum of revenue** field → select **Conditional formatting** option → **Data bars**:

2. In the **Data bars - Sum of Revenue** window which appears → select the **Show bar only** option → **select the colors** for the **Positive bar** and **Negative bar** options:



3. Click **OK** → it can be seen that in the new presentation of the data, **the numerical values are replaced by Data bars**, for the selected column, **Sum of Revenue**.

► **Replace numbers with icons:**

1. **Visualizations** pane → **Expand the menu** for the **Sum of revenue** field → select **Conditional formatting** option → **Icons**.

2. In the **Icons - Sum of Revenue** window which appears → configure how to use the icons to represent the data in the selected column, **Sum of Revenue**:

Columns

- Sum of Revenue
- Order Date
- Month
- Category

Icons - Sum of Revenue

Format style: Rules | Apply to: Values only

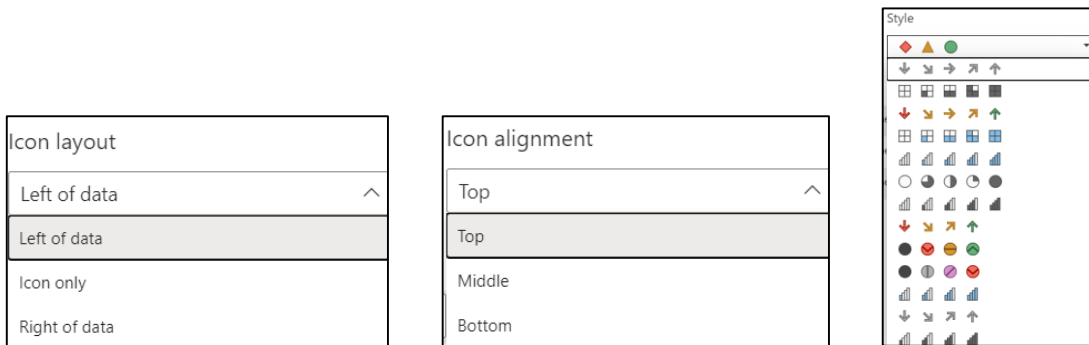
What field should we base this on? Sum of Revenue | Summarization: Sum

Icon layout: Left of data | Icon alignment: Top | Style: [Red Diamond] [Yellow Triangle] [Green Circle]

Rules

If value	>=	0	Percent	and	<	33	Percent	then	[Red Diamond]
If value	>=	33	Percent	and	<	67	Percent	then	[Yellow Triangle]
If value	>=	67	Percent	and	<=	100	Percent	then	[Green Circle]

- In the **Icon layout** box list → choose where the icon should appear (to the left/right of the date) and whether to replace the data in the table
- in the **Icon alignment** box list → choose the icon alignment.
- In the **Style** box list → choose the set of icons:



- At the bottom of the page, the application method of the icons is shown, depending on the numerical value: the first 33% (lowest) of the range of values is represented with the first icon, the next 33% (middle values) with the second icon and the last set of 33% of values with the third icon:

The image shows two tables side-by-side, with an arrow pointing from the left table to the right table. The left table is a standard data table with columns 'Sum of Revenue', 'Month', and 'Category'. The right table is the same data table but with conditional formatting applied to the 'Sum of Revenue' column. The values are grouped into three ranges, each represented by a different icon: a red diamond for the lowest values, a green circle for the middle values, and a yellow triangle for the highest values.

Sum of Revenue	Month	Category
46,072.50	January	
3,925.49	January	Auto
6,832,330.73	January	Electronics
253,170.05	January	Gaming
291,818.02	January	Sports
21,396.84	January	TV
147,924.69	February	
4,754.58	February	Auto
3,520,866.10	February	Electronics
68,508.44	February	Gaming
299,553.91	February	Sports
34,016.72	February	TV
278.83	March	
339.25	March	Auto
3,884,294.81	March	Electronics
758,076.19	March	Gaming
191,368.36	March	Sports
12,124.03	March	TV
411,742.47	April	
54,625.89	April	Auto
3,845,993.80	April	Electronics

► **Apply background color scales:**

The background color of a field containing numeric data can be changed depending on the value of each row. The gradient will change from minimum to maximum value for that field.

1. **Visualizations** pane → **Expand the menu** for the **Sum of revenue** field → select **Conditional formatting** option → **Background color**.
2. In the **Background color - Sum of Revenue** window which appears → configure how to use the icons to represent the data in the selected column, **Sum of Revenue**:

- in the **Format style** list box → select **Gradient**.
- **set the color** for the **lowest value**, for the **middle value** and for the **highest value**:

Background color - Sum of Revenue

Format style: Gradient

Apply to: Values only

What field should we base this on?: Sum of Revenue

Summarization: Sum

How should we format empty values?: As zero

Minimum: Lowest value (Green)

Center: Middle value (Yellow)

Maximum: Highest value (Red)

Add a middle color

Month	Sum of Revenue	Category
May	2,150,431.06	Electronics
June	2,271,444.48	Electronics
July	2,800,560.66	Electronics
August	2,491,345.31	Electronics
September	2,405,039.25	Electronics
October	2,414,265.97	Electronics
November	3,316,875.25	Electronics
December	3,044,529.01	Electronics
January	253,170.05	Gaming
February	68,508.44	Gaming
March	758,076.19	Gaming
April	733,934.13	Gaming
May	244,919.82	Gaming
June	182,470.27	Gaming
July	1,189,533.38	Gaming
August	561,089.25	Gaming
Total	44,707,304.78	

As a result of the conditional formatting settings, one can observe that the color of the field data changes, with the lowest value being green, while the highest value is red:

► **Apply font color:**

Similar to applying conditional formatting to the cell background, one can apply conditional formatting to the text of the values:

1. **Visualizations** pane → **Expand the menu** for the **Sum of revenue** field → select **Conditional formatting** option → **Font color**.

2. In the **Font color - Sum of Revenue** window which appears → configure how to use the icons to represent the data in the selected column, **Sum of Revenue**:

- in the **Format style** list box → select **Gradient**.

- **set the color** for the **lowest value**, for the **middle value** and for the **highest value**:

Font color - Sum of Revenue

Format style: Gradient

Apply to: Values only

What field should we base this on?: Sum of Revenue

Summarization: Sum

How should we format empty values?: As zero

Minimum: Lowest value (Blue)

Center: Middle value (Orange)

Maximum: Highest value (Red)

Add a middle color

Month	Sum of Revenue	Category
October	355,533.88	
October	763.99	Auto
October	2,414,265.97	Electronics
October	187,020.23	Gaming
October	293,072.45	Sports
October	50,325.95	TV
November	1,017,513.73	
November	8,098.31	Auto
November	3,316,875.25	Electronics
November	504,722.98	Gaming
November	303,877.50	Sports
November	239,291.06	TV
December	287,650.60	
December	4,593.84	Auto
December	3,044,529.01	Electronics
December	517,121.40	Gaming
Total	44,707,304.78	

► **Apply rules-based formatting:**

Similar to applying gradient-based conditional formatting, rule-based formatting can be applied to value text:

In the **Font color - Sum of Revenue** window which appears → configure how to use the icons to represent the data in the selected column, **Sum of Revenue**:

- in the **Format style** list box → select **Rules**.

- in section **Rules** → configure one or more value ranges, and set a color for each one. On each row must be set an **If value condition**, an **and value condition**, and a **color**:

The configuration window 'Font color - Sum of Revenue' shows the following rules:

Condition	Value	Operator	Value	Color	
If value >=	0	Number	and <=	1000	Green
If value >	1000	Number	and <=	5000	Blue
If value >=	5000	Number	and <	10000	Yellow
If value >=	10000	Number	and <	30000	Purple
If value >=	30000	Number	and <	30000	Red

The data table shows the following data:

Month	Sum of Revenue	Category
February	4,754.58	Auto
March	339.25	Auto
April	54,625.89	Auto
May	920.45	Auto
June	5,675.73	Auto
July	8,344.66	Auto
August	69,695.58	Auto
September	3,107.24	Auto
October	763.99	Auto
November	8,098.31	Auto
December	4,593.84	Auto
January	2,975,408.73	Electronics
February	2,339,184.10	Electronics
March	2,353,148.81	Electronics
April	2,664,311.80	Electronics
May	2,150,431.06	Electronics
Total	44,707,304.78	

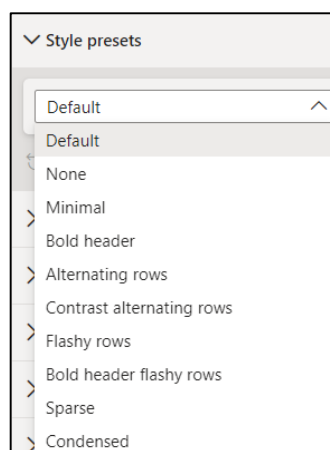
Note: this rule-based formatting can be applied similarly for the background color of the field data.

► Apply a table style:

Power BI Desktop offers the possibility to access a set of predefined styles for configuring the appearance of a table, with possibilities for formatting and adjusting the appearance of a table.

To apply a predefined formatting style to a table, the following steps are performed:

1. The **Report view** → in the **Visualizations** pane → select **Table visualization**.
2. **Create a visual with a table** → **Click inside the table** to format it.
2. **Visualizations** pane → click the **Format** icon → select a table style from the **Style presets** options (**Visual section**):



Default: Add a gray background to the rows in the table, alternatively.

None: Removes the formatting applied to the table.

Minimal: Add a little spacing between the rows of the table.

Bold header: Adds a dark background to the title row.

Alternating rows: Adds a gray background to alternating rows and a dark background to title and totals rows.

Contrast alternating rows: Alternates the background color of the rows between two shades of gray.

Flashy rows: Alternates the background color of the rows between two shades of colors.

Bold header flashy rows: Alternates the row background color between two shades of color and adds a gray background to the alternating rows and a dark background to the header and totals rows.

Sparse: Remove the row separator and add a dark background to the title and total rows.

Condensed: Adds vertical and horizontal lines between table rows and columns and adds a dark background to the title and totals rows.

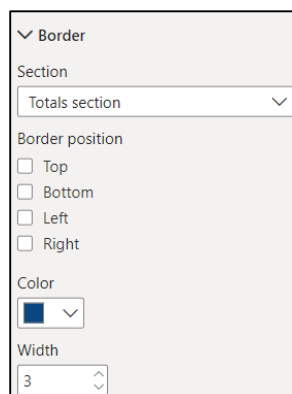
The figure below shows a change in the appearance of the table, after applying some of these formats:

Category	Auto		Electronics		Gaming		Sports		TV			
Month	Sum of Revenue	Sum of Cost	Sum of Revenue	Sum of Cost	Sum of Revenue	Sum of Cost	Sum of Revenue	Sum of Cost	Sum of Revenue	Sum of Cost		
December	287,650.60	466.49	4,593.84	17.25	6,773,103.01	4,274.81	517,121.40	502.61	1,219,717.14	1,454.85	32,562.85	73.84
November	1,568,053.73	846.52	8,098.31	15.76	7,194,980.25	4,045.71	504,722.98	659.80	303,877.50	1,037.93	239,291.06	59.91
October	355,533.88	400.66	763.99	6.26	3,585,947.07	3,598.86	187,020.23	421.49	293,072.45	1,192.85	50,325.95	102.71
September	174,909.29	119.55	3,107.24	15.34	3,586,721.25	3,416.15	589,217.07	419.46	242,157.86	1,026.50	22,539.33	62.71
August	648.30	5.41	69,695.58	84.77	3,673,027.31	3,516.39	561,089.25	544.09	303,173.24	1,287.36	64,629.38	70.01
July	563,584.91	546.04	8,344.66	20.37	3,588,348.66	3,414.19	1,189,533.38	801.08	767,960.00	990.42	48,727.77	72.81
June	364,201.82	318.35	5,675.73	6.80	3,062,179.48	3,469.44	182,470.27	536.27	307,574.84	1,034.10	73,066.08	62.71
May	515,527.23	350.19	920.45	23.80	3,332,113.06	3,062.63	244,919.82	314.66	311,722.55	873.62	45,734.51	77.41
April	411,742.47	141.37	54,625.89	15.00	3,645,093.00	3,326.13	1,077,776.13	670.85	202,399.32	649.89	39,444.17	66.81
March	278.83	155.73	339.25	1.50	3,884,294.81	3,412.34	758,076.19	569.39	191,368.36	1,044.82	12,124.03	219.51
February	147,924.69	176.39	4,754.58	9.30	3,520,886.10	3,014.47	68,508.44	138.70	299,553.91	907.90	34,016.72	358.71
January	46,072.50	6.27	3,925.49	6.39	6,832,330.73	3,926.59	253,170.05	334.67	291,818.02	1,281.39	21,396.84	59.71
Total	4,436,128.25	3,532.97	164,845.01	222.54	53,789,906.43	42,477.71	6,133,625.21	5,913.07	4,734,395.19	12,781.63	683,858.69	1,287.21

► **Apply a table border:**

The appearance of the table can be improved by adding an outer border:

1. The **Report view** → in the **Visualizations** pane → select **Table visualization**.
2. **Create a visual with a table** → **Click inside the table** to format it.
3. **Visualizations** pane → click the **Format** icon → select **Grid** option (**Visual section**) → select **Border** option:



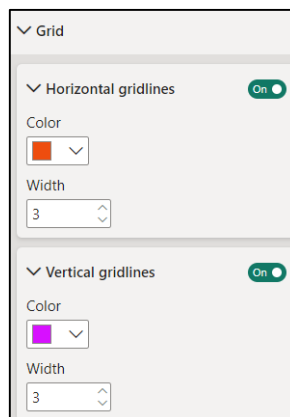
4. Choose the **section** that will be framed, the **border position** and the **thickness of the border line**.

The figure below shows a change in the appearance of the table, after applying a table border:

Month	Sum of Revenue	Category
January	46,072.50	
January	3,925.49	Auto
January	6,832,330.73	Electronics
January	253,170.05	Gaming
January	291,818.02	Sports
January	21,396.84	TV
February	147,924.69	
February	4,754.58	Auto
February	3,520,866.10	Electronics
February	68,508.44	Gaming
February	299,553.91	Sports
February	34,016.72	TV
March	278.83	
March	339.25	Auto
March	3,884,294.81	Electronics
March	758,076.19	Gaming
March	191,368.36	Sports
Total	69,942,758.78	

► **Apply gridlines to the table:**

1. **Visualizations** pane → click the **Format** icon → select **Grid** option (**Visual** section):

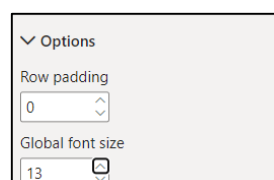


2. Choose the color and thickness for vertical and horizontal gridlines:

Month	Sum of Revenue	Category
December	287,650.60	
December	4,593.84	Auto
December	6,773,103.01	Electronics
December	517,121.40	Gaming
December	1,219,717.14	Sports
December	32,562.85	TV
November	1,568,053.73	
November	8,098.31	Auto
November	7,194,980.25	Electronics
November	504,722.98	Gaming
November	303,877.50	Sports
November	239,291.06	TV
October	355,533.88	
October	763.99	Auto
October	3,595,947.97	Electronics
October	187,020.23	Gaming
Total	69,942,758.78	

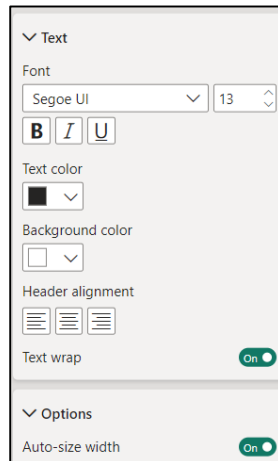
► **Modify row padding and global font size:**

Visualizations pane → click the **Format** icon → select **Grid** option (**Visual** section) → **Options**:



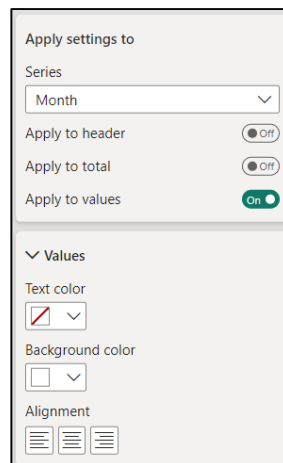
► **Modify the formatting of column headers:**

1. **Visualizations** pane → click the **Format** icon → select **Column headers** option (**Visual** section):



► **Individual formatting of table fields:**

Select the fields that need to be formatted differently and which specific sections. Choose the color of the text and the background and the alignment of the values in the respective field:



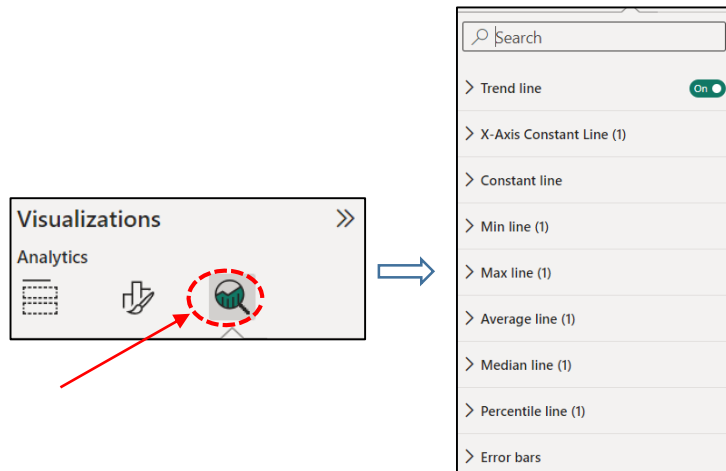
The figure below shows a change in the appearance of the table, after applying these formats:

Month	Sum of Revenue	Category
January	46,072.50	
January	3,925.49	Auto
January	6,832,330.73	Electronics
January	253,170.05	Gaming
January	291,818.02	Sports
January	21,396.84	TV
February	147,924.69	
February	4,754.58	Auto
February	3,520,866.10	Electronics
February	68,508.44	Gaming
February	299,553.91	Sports
February	34,016.72	TV
March	278.83	
March	339.25	Auto
March	3,884,294.81	Electronics
March	758,076.19	Gaming
March	191,368.36	Sports
March	12,124.03	TV
Total	69,942,758.78	

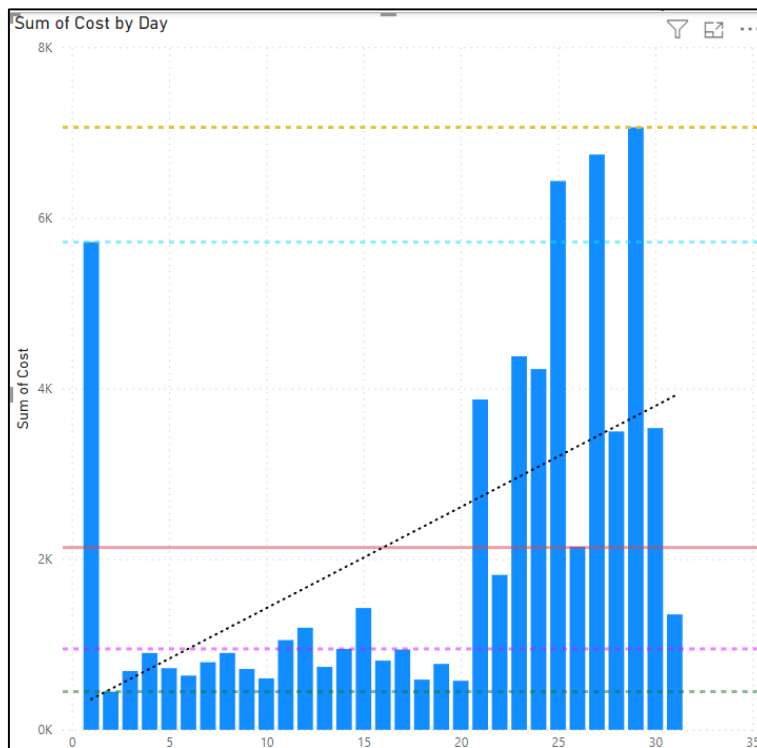
► **Use the Analytics pane:**

Analytics pane allows the addition of several dynamic reference lines, which can help to optimally visualize data and trends.

1. The **Report view** → in the **Visualizations** pane → select **Analytics icon**:



The panel allows the creation of the following types of dynamic reference lines: X-Axis constant line, Y-Axis constant line, Min line, Max line, Average line, Median line, Percentile line, Symmetry shading:



► **Use report themes:**

To generate a unified and customized design for a report, the Power BI application offers various themes through which design changes can be applied to the entire report. When a particular theme is selected, all visual elements in the report use the color set and formatting options from the selected theme as default values.

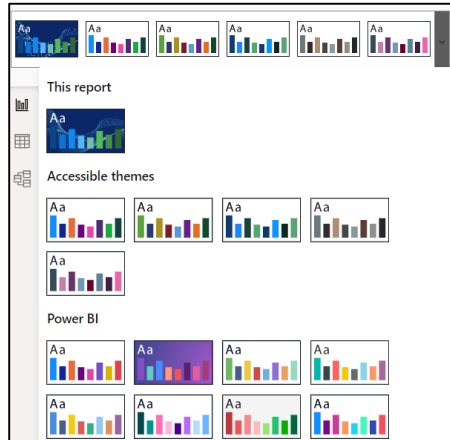
There are two types of themes that can be used:

- themes already installed in Power BI, with predefined color schemes and preset formatting options; they can be selected directly from the Power BI Desktop menu.

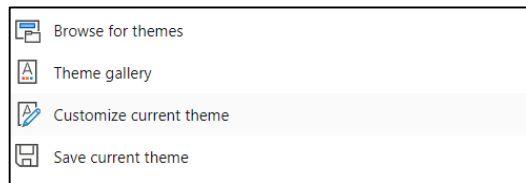
- themes that can be created by either customizing a specific theme, importing a custom theme from the Theme Gallery, or importing your own custom theme using a JSON file.

To apply a theme to a report, the following steps are performed:

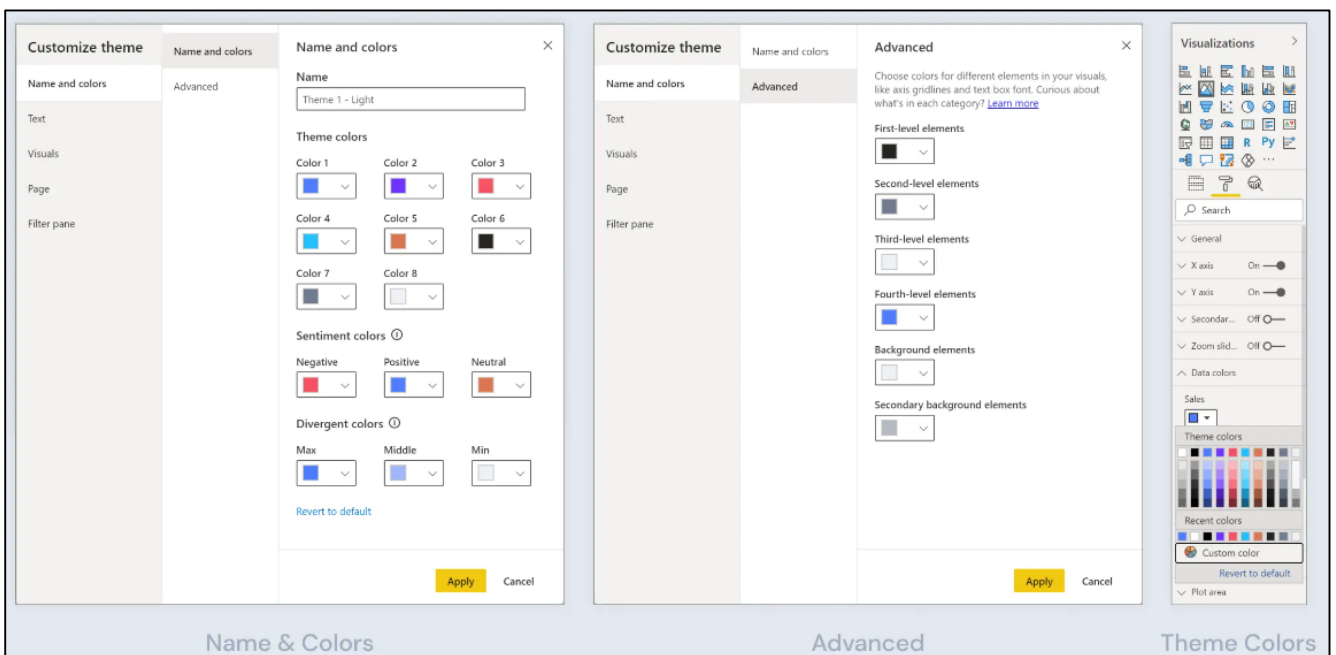
1. The **Report view** → **View** tab → **Themes** section → **select a theme**:



2. To customize a specific theme: **View** tab → **Themes** section → click the drop-down menu → select **Customize Current Theme** option:



3. The window that opens allows to edit colors and customize other formatting elements:



Name: the name of the current theme.

3. Another possible formatting is to expand the row headers. Right-clicking on a row brings up a window with options that allow expanding the selected row header, the entire level, or everything down to the last level of the hierarchy.

4. Likewise in the case of table visual, a matrix style can be applied. Power BI Desktop offers the possibility to access a set of predefined styles for configuring the appearance of a matrix, with options for formatting and adjusting the appearance:

Visualizations pane → click the **Format** icon → select a table style from the **Style presets** options (**Visual** section).

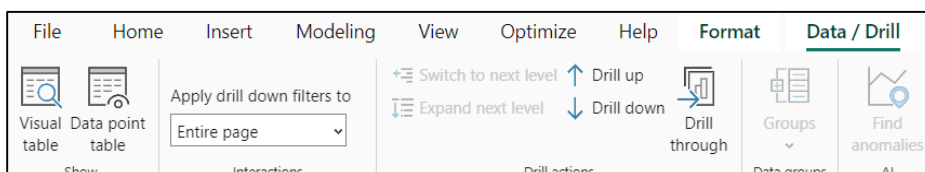
Also, all the formatting options available for table visual can be applied.

5. Next, additional data fields can be added in this matrix visual.

For example, the **Sum of profit** field is added, directly, by dragging it into the table:

Category	Sports		Outdoor		Total	
Department	Department1		Department1		Department1	
Group	Indoor	Outdoor	Indoor	Outdoor	Indoor	Outdoor
Item category	Sum of Revenue	Sum of Profit	Sum of Revenue	Sum of Profit	Sum of Revenue	Sum of Profit
Category2	16.67	129.26			16.67	129.26
Line 80	16.67	129.26			16.67	129.26
Product 1375	16.67	129.26			16.67	129.26
2016	7.18	121.18			7.18	121.18
2017	9.49	8.08			9.49	8.08
Category3	6,473.94	2,409.16			6,473.94	2,409.16
Line 135	3,283.40	2,355.29			3,283.40	2,355.29
Product 1665	3,283.40	2,355.29			3,283.40	2,355.29
2016	13.40	1,166.00			13.40	1,166.00
Qtr 4	13.40	1,166.00			13.40	1,166.00
November	13.40	1,166.00			13.40	1,166.00
9	13.40	1,166.00			13.40	1,166.00
2017						
2018	3,270.00	1,189.29			3,270.00	1,189.29
Line 3	3,166.00	31.55			3,166.00	31.55
Line 36	13.51	13.40			13.51	13.40
Product 2557	13.51	13.40			13.51	13.40
2018	13.51	13.40			13.51	13.40
Qtr 2	13.51	13.40			13.51	13.40
May	13.51	13.40			13.51	13.40
Line 59	6.85	5.60			6.85	5.60
Product 1576	6.85	5.60			6.85	5.60
2016	6.85	5.60			6.85	5.60
Qtr 3	6.85	5.60			6.85	5.60
Line 9	4.18	3.32			4.18	3.32
Product 2240	4.18	3.32			4.18	3.32
Total	79	275,159.80	90,303.96	23.04	140.06	275,182.84

6. Navigating between hierarchical data inside a matrix visual it can be done with the help of the options available in **Data/Drill** tab:



Visual table: shows the data as a table.

Data point table: allows to see a table of the data used to calculate a single data point:

Item category	Auto	Electronics	Gaming	Sports	TV	Total	
30				28.99		28.99	
Category2				16.67		16.67	
Line 80				16.67		16.67	
Product 1375				16.67		16.67	
2016				7.18		7.18	
Qtr 2				7.18		7.18	
April				3.97		3.97	
20				3.97		3.97	
May				3.21		3.21	
18				3.21		3.21	
2017				9.49		9.49	
Qtr 2				9.49		9.49	
June				9.49		9.49	
Category3	13.38			6,473.94		6,487.32	
Line 135	13.38			3,283.40		3,296.78	
Line 3				3,166.00		3,166.00	
Product 2137				3,166.00		3,166.00	
2018				3,166.00		3,166.00	
Qtr 3				3,166.00		3,166.00	
Total	39,942,228.74	68,506.12	879,309.24	3,401,739.32	384,897.27	30,624.09	44,707,304.78

← Back to report

Item category	Item line	Item Name	Year	Quarter	Month	Day	Revenue	Category	Department	Group
Category2	Line 80	Product 1375	2017	Qtr 2	June	15	9.49	Sports	Department1	Indoor

Drill down: shows the selected data together with any sublevel of that data.

Drill up: returns to the previous level that was drilled down.

Switch to next level: switch the horizontal axis to show the next level in the hierarchy.

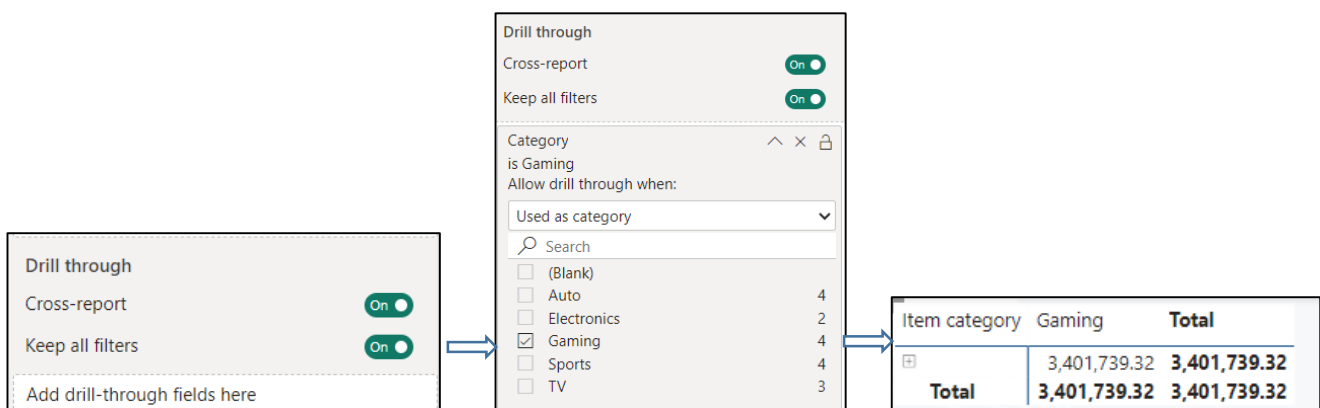
Expand next level: expand the horizontal axis to show the current level and the next level in the hierarchy at the same time.

Groups: produces a group to highlight the selected values in a visual.

Drill through: select a data point to see a page with more information about it.

This option is active through the existing configuration at the bottom of the **Visualizations** pane in the **Drill through** section.

In the example below, the **Category** field was added to the **Add drill-through fields here** field and by ticking the **Gaming** option, a separate page appeared with the detailed information requested:



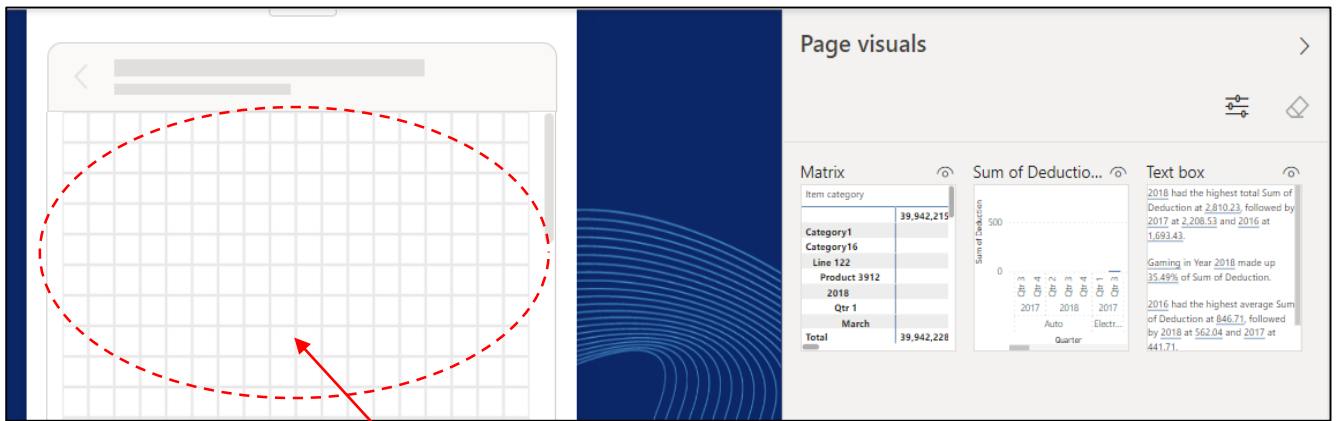
6.7 Viewing reports on mobile devices

A report page can be viewed on a mobile device dynamically, so that if a change is made in the visual, it will be automatically made on the mobile device as well.

This functionality is available through the integrated Mobile layout option. Using this option, users can define how the component elements of a visual are configured on a smartphone to match the appearance of the phone.

To view a Power BI report page on a mobile device, the following steps must be performed:

1. The **Report view** → **View tab** → **Mobile layout (Mobile section)**:

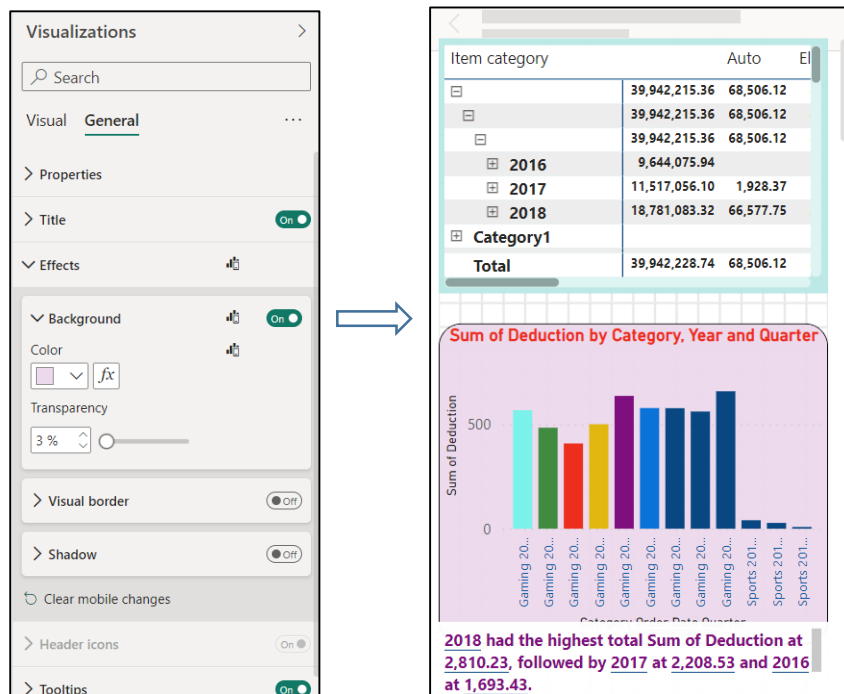


interactive phone emulator canvas

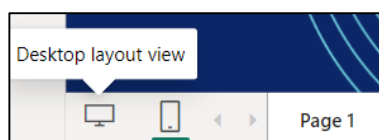
The component elements of the created report are available in the section **Page visuals**.

2. **Drag and drop the visual elements** inside the interactive phone emulator canvas.

3. **Visualizations pane** contains various formatting options that allows to modify the visual elements so that they are more suitable for the mobile layout: grid orientation and image style settings, font size adjustment, caption placement, title adjustment, background formatting, etc.



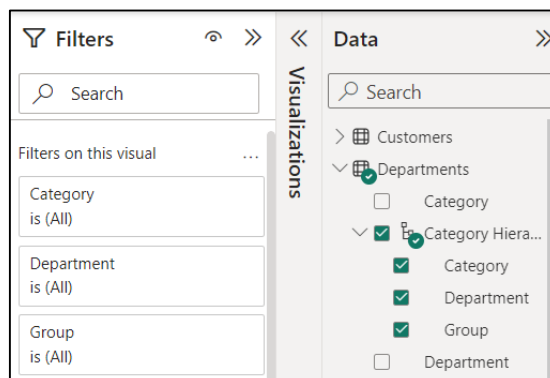
4. Click the **Desktop Layout** button (left, bottom of page) to switch back to the desktop layout



6.8 Filtering data in visuals

Visual filters are useful to restrict the data displayed in a report, so that only what is relevant for the respective analysis remains and as many visual images can be loaded into the Dashboard Canvas.

Power BI Desktop filters can be accessed from the **Filters pane** on the visuals of the report:



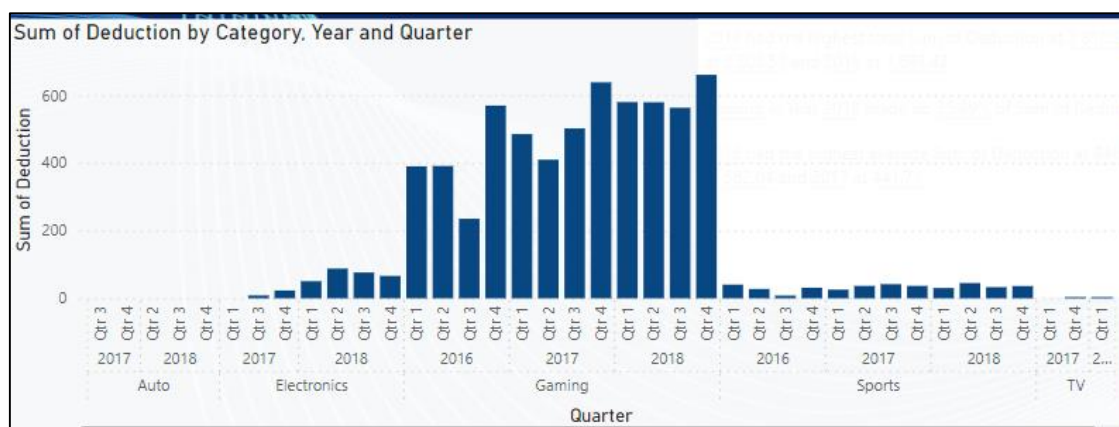
Filters can be applied to various data formats (numeric, text, date format or logical values), with specific ways of setting the ranges of values that can be included or excluded. In this sense, there are three levels of filtering:

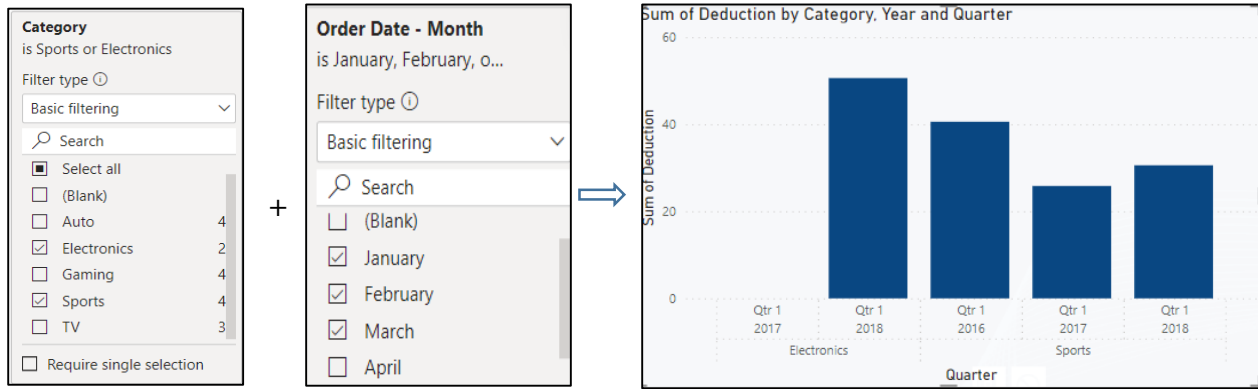
- *Visual-level filter*: can be applied only on a specific visual to focus attention on certain elements of the respective diagram or graph.
- *Page-level filters*: applies to all the visuals on a specific page, allowing to limit the data set appearing in the active page.
- *Report-level filters*: applies to all pages in the report, allowing to apply filters on all the visuals in the report.

■ Visual-Level Filters:

Filters pane has a section for **Visual-level filter**, where, by default, the fields used in the respective visual are displayed. For each of these fields, the filter settings can be changed.

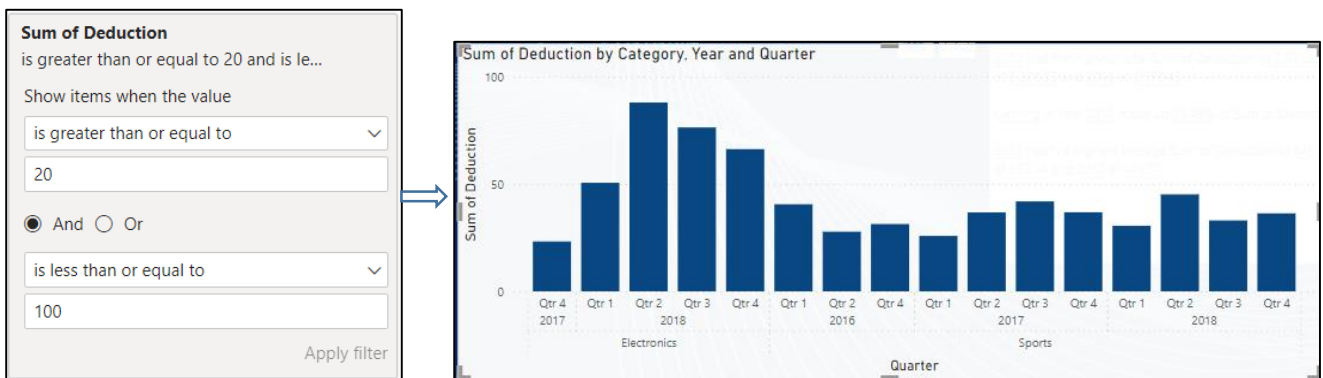
For example, for **Sum of Deduction by Category, Year and Quarter** visual, choosing as filter elements: **Category (Electronics and Sports)** and **Order Date – Month (first Quarter)** only information about these aspects will be found in the view:



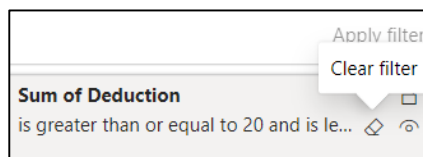


In addition to filtering the visual elements on the data fields used to create the visual elements, a new data field from the data model can be added in this section, by dragging it inside the **Add data fields here** field.

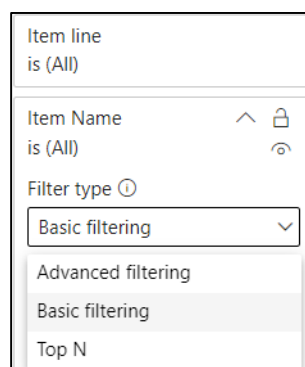
For filtering numerical data, the application allows filtering based on value ranges. Thus, one can select the lower and/or upper limits of the range of numbers to be displayed in the visual:



To clear a filter, click **Clear filter** icon to the right of the selected filter:

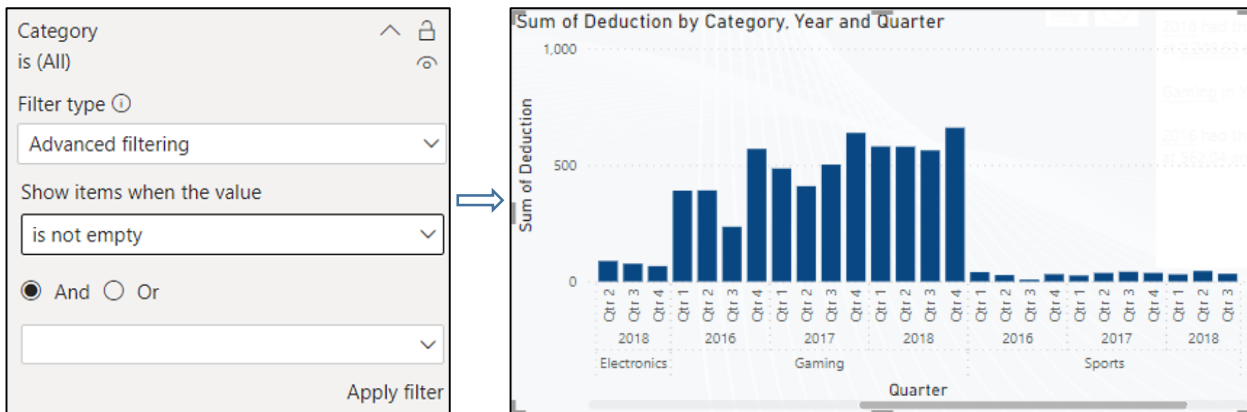


The filters contained in the application have two modes of use, **Basic filtering** and **Advanced filtering**. The difference is in the filtering mode, i.e. if in Basic filtering the application provides a list of values that can be scrolled and searched, in Advanced filtering rules are used to determine a certain range of values that will be obtained when running the report:

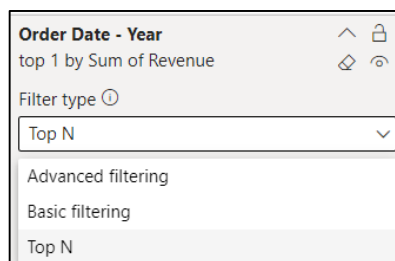


Advanced filters allow the use of more complicated filters. Depending on the type of data in the respective field, we look for values that contain or do not contain, start with or do not start with a certain value, is blank, is not blank, is empty, is not empty, is after, is before, etc.

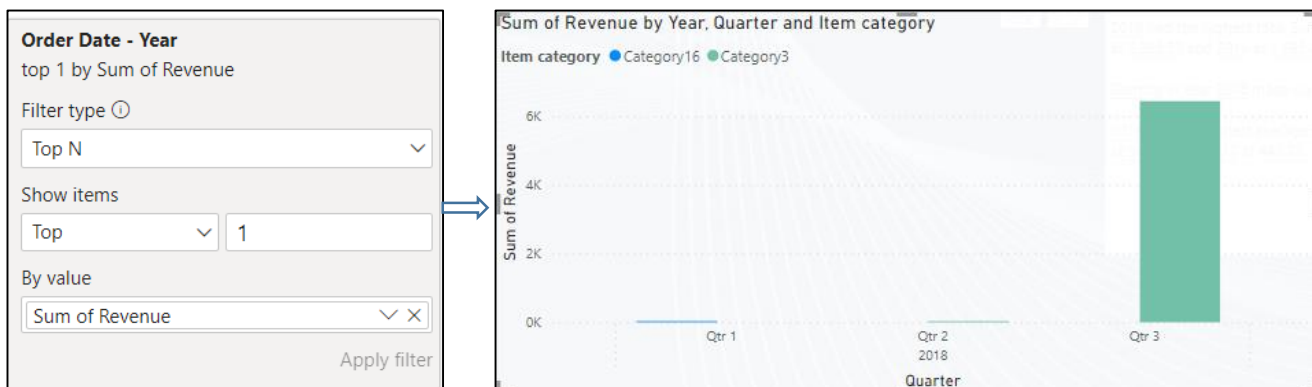
For example, for the **Sum of Deduction by Category, Year and Quarter** visual, choosing as filter elements: **Show items when the value is not empty** only information about these elements will be found in the view:



A special type of filtering offers the possibility of selecting the first N best elements from the respective selection or the weakest ones. The option selected in this case for filtering is **Top N**:



For example, for the **Sum of Revenue by Year, Quarter and Item category** visual, when choosing as filtering elements: **Show items Top 1**, the visualization will only contain information about the first Item category from each Quarter and each Year, the best performing in relation to the analyzed metric, **Sum of Revenue**:



■ Page-Level Filters:

Page level filters are useful when the report contains pages that focus on certain subsets of data. These filters are configured similarly to visual-level filters, they are used in parallel with these filters (so they do not replace them) and they only affect the current page, in which they were defined.

To apply such a filter, place the filter fields in the **Filters on this page** area:

The 'Filters on this page' panel shows the following filter configuration:

- Item line: is Line 131, Line 133, Line 134, or Line ...
- Filter type: Basic filtering
- Search: [Search box]
- Line 129: 3
- Line 13: 23
- Line 131: 41
- Line 133: 14
- Line 134: 9
- Line 135: 38
- Line 136: 1
- Require single selection

The resulting data table is as follows:

Item category	Sports	Total	
Category3	11.09	2,355.29	2,366.38
Line 135	11.09	2,355.29	2,366.38
Product 1665	11.09	2,355.29	2,366.38
2016		1,166.00	1,166.00
Qtr 4		1,166.00	1,166.00
November		1,166.00	1,166.00
9		1,166.00	1,166.00
2017	6.25		6.25
2018	4.84	1,189.29	1,194.13
Qtr 3	4.84	1,189.29	1,194.13
August	4.84	1,167.00	1,171.84
1	4.84		4.84
28		1,167.00	1,167.00
September		22.29	22.29
12		22.29	22.29
Total	11.09	2,355.29	2,366.38

■ **Page-Level Filters:**

To apply such a filter, place the filter fields in the **Filters on all pages** area.

6.9 Using slicers

To apply particular filter selections to an individual page of a report, a slicer can be added to the dashboard canvas, which is a multi-selection visual filter where one or more items can be chosen to filter the data in a report. Unlike filters, slicers remain visible while analyzing the report.

For example, for a profit analysis report, a department category slicer can be made. From that slicer, one can select the department category for which we want to see the profit values. The report views will automatically change, showing the information for that quarter. Moreover, several different slicers can be configured for the same report, so as to slice and dice the data using multiple criteria.

To see the result of applying a slicer, in the following example the following steps were applied:

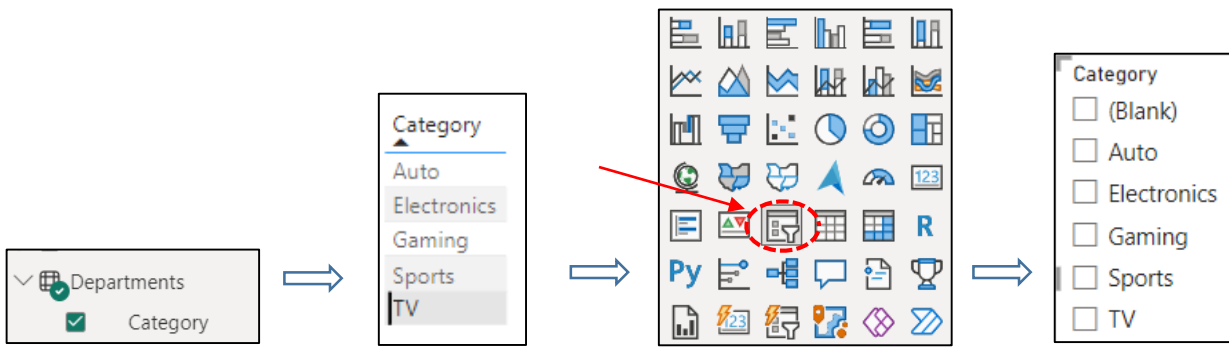
1. The **Report view** → **Expand the Sales table** → **Create a table using the fields: Σ Profit and Order_Date (Quarter):**

The field list shows the following configuration:

- Sales
- Σ Cost
- CustomerID
- Σ Deduction
- DepartmentID
- ItemID
- Order_Date
- Date Hierarc...
- Year
- Quarter
- Month
- Day
- Σ Profit

2. **Expand the Departments table** → drag the field **Category** to an empty side of the dashboard canvas.

3. **Visualizations pane** → select the **Slicer** icon. The table of categories will become a slicer:

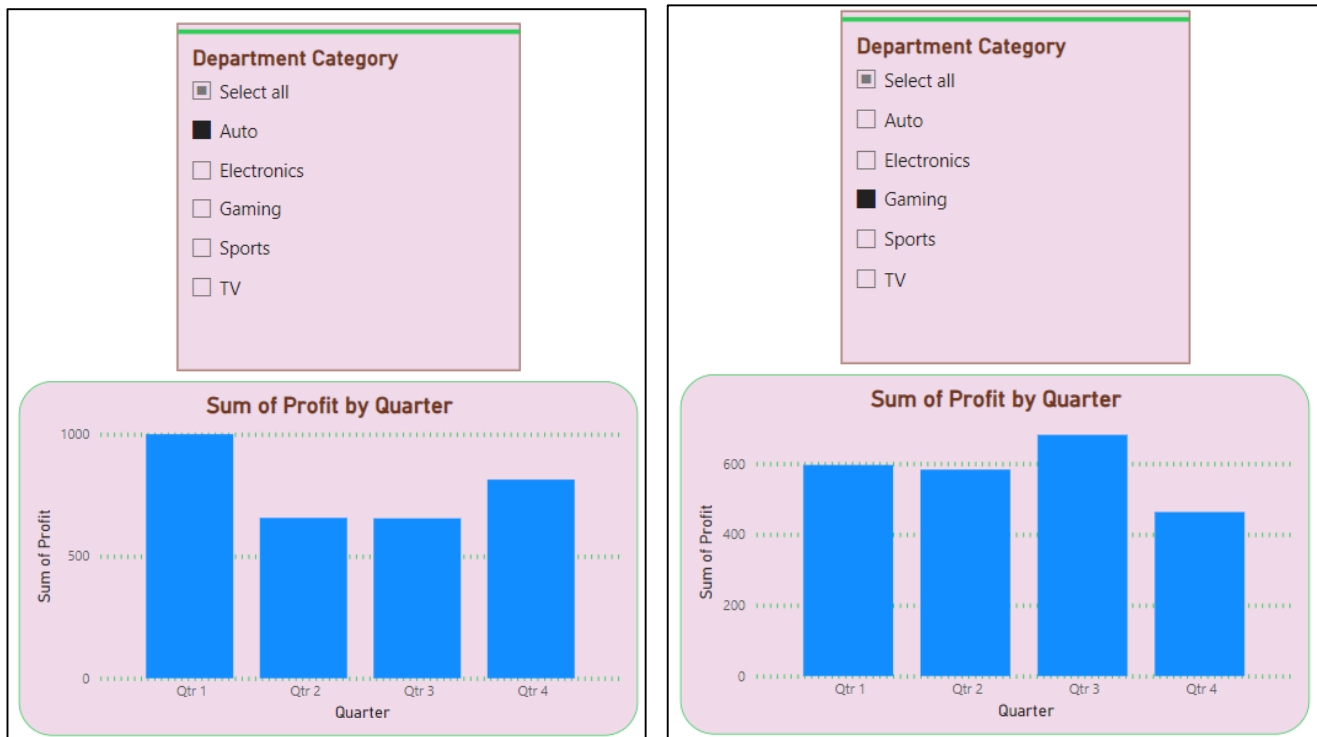


4. Similarly, by repeating these steps, several slicers can be created for each page of the report, using another field as data for the new slicer.

5. **To apply a slicer** to filter data on a page → **click on a single item in the slicer or Ctrl-click on multiple items** → **the objects on the report page will be filtered according to the selection made.**

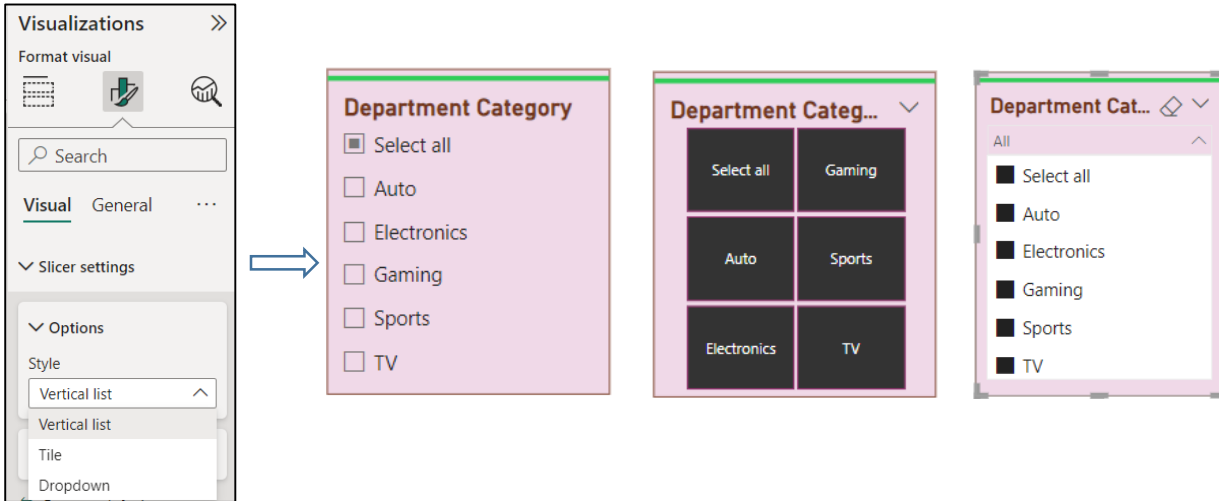
6. Each active element in the slicer now has a small rectangle on the left, indicating that this element is selected.

7. To apply formatting elements to the slice → **Visualizations pane** → **select the slicer** → select the **Format tab** → **configure all the formatting options:**



To change the shape of a slicer:

1. **Visualizations pane** → **select the slicer** → select the **Format tab** → **Slicer settings** section → in the **Style drop-down box** → select one of the options: **Vertical list, Tile or Dropdown:**



The application allows the creation of slicers for different types of data. For example, for a slicer based on calendar date ranges, the slicer will take the form of a slider bar with two buttons at each end. Sliding these buttons forward or backward will adjust the filter interval:



6.10 Using maps

In the variety of visual elements and presentation techniques contained in the Power BI Desktop platform, one can find the map visualization, which is a useful tool for the representation and analysis of spatial data. Displaying data in the form of a map offers the possibility to view valuable information, which increases the clarity of the presentations and reports made. This type of visualization is simple to create and behaves just like any other visual image.

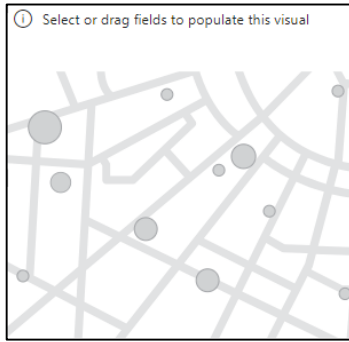
The application contains four types of maps that can be used: Basic Map, Filled Map, Shape Map and Azure Map, each with specific characteristics that allow adapting the map to the reporting requirements.

► The **Basic Map** it is also called Bubble Map because it shows the data points with circles - bubbles of different sizes to represent different values.

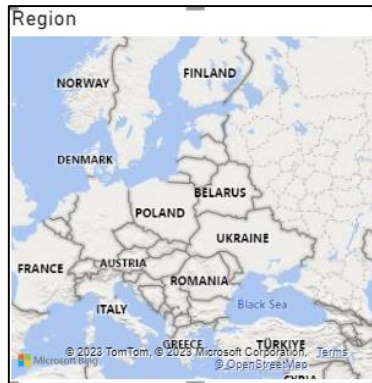
The bubble map in the example below uses the Region field as the location input and the Department Category column as the legend input. The size of the bubble represents a third variable, the Profit measure, which allows the observation of patterns and trends in the analyzed data.

To create this map, the following steps must be performed:

1. The **Report view** → Click **Map** in the visuals gallery → **a blank map will appear:**



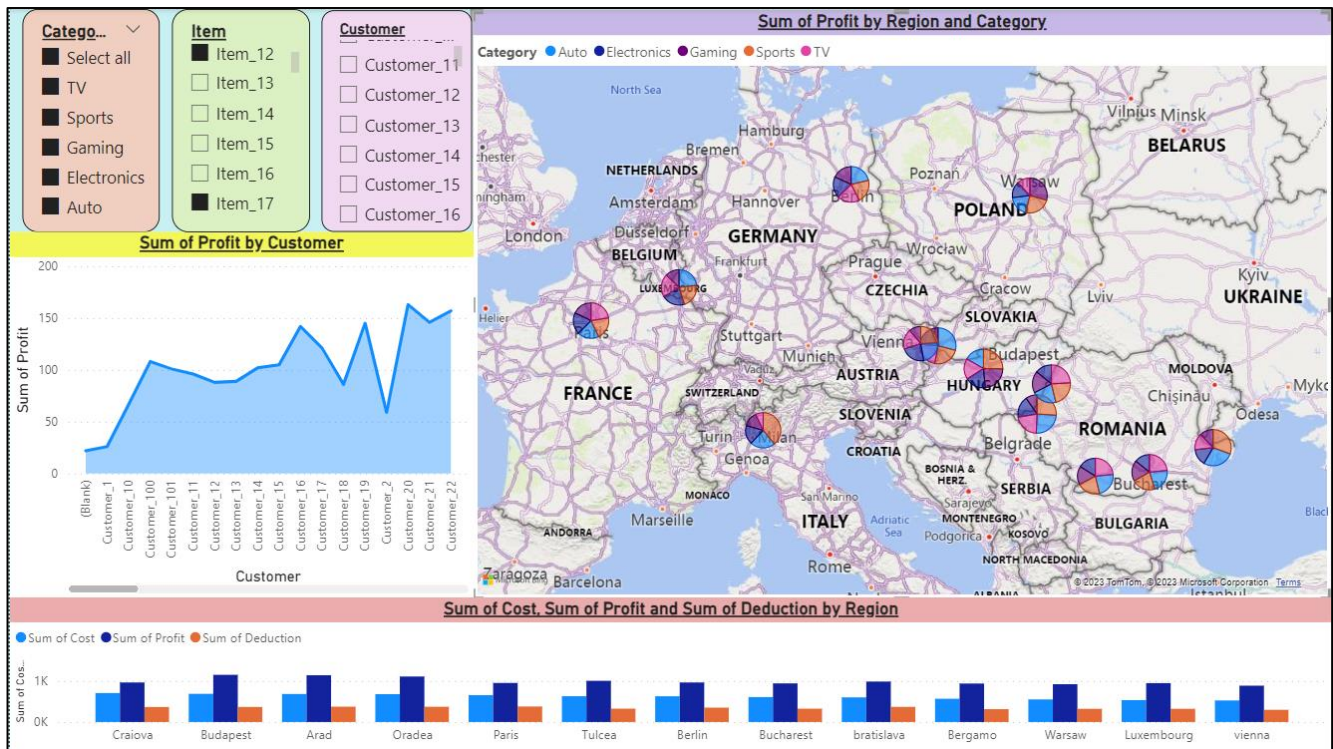
2. Expand the **Region** table → drag the **Region** field to the map visual → a map with all countries appear:



3. Expand the **Departments** table → drag the **Category** field to the map visual.

4. Expand the **Sales** table → drag the **Profit** field to the map visual.

As one can see on the map, the visual will change the size of each pie of data to reflect the sum of profit per category:



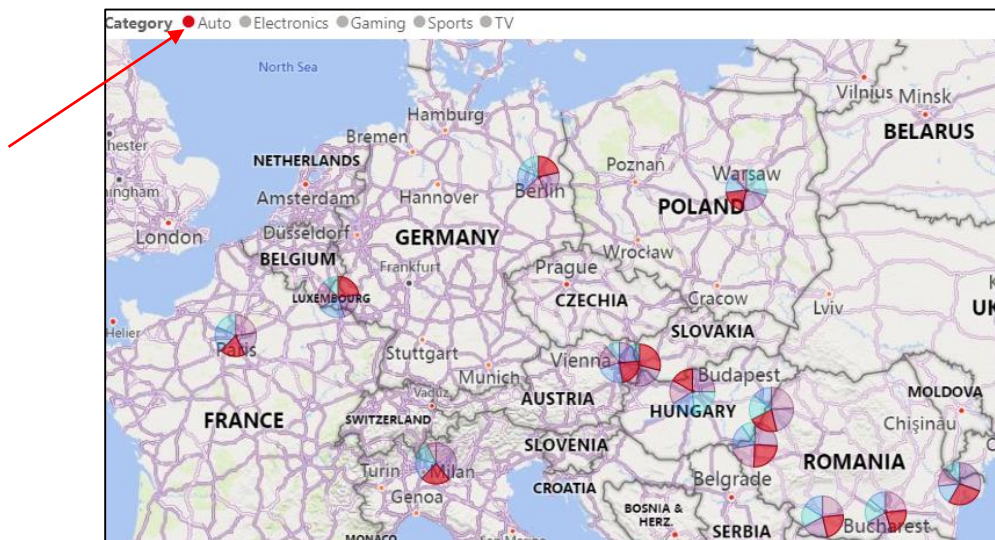
5. Furthermore, various filters and slices can be applied for the report.

Notes:

- 1) The location input set must contain data columns that can be interpreted as geographic information: countries, counties, cities, latitudes & longitudes.
- 2) When hovering the mouse cursor over the data representation, an explanatory box appears that provides details about the data:

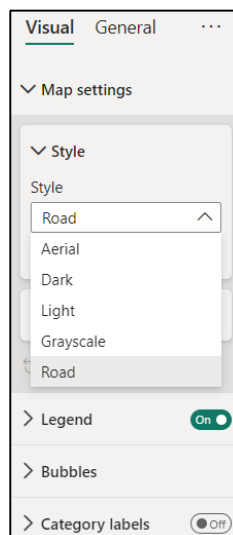


- 3) Based on the elements of the Legend box, segments of data can be highlighted within the representation. For example, by selecting the first element in the legend, the **sum of profit by category: Auto** will be highlighted:



- 4) The map can be refined by changing its appearance and style:

1. **Visualizations pane** → select the map → **Format tab** → **Visual** section → **Map settings** → in the **Style drop-down box** → select one of the options: **Road, Aerial, Dark, Light, Grayscale** or **Road:**



5) Further, one can apply settings for the data colors, for the title, for the legend, for the font color, for the background color or aspect ratio, etc.

▶ The **Filled Map** visual is somewhat similar to a thermal map, in that it uses color scaling and saturation to display the geographic data: the lighter an area is represented, the lower the representative value.

▶ The **Shape Map** visual also uses shading and saturation to represent the geographic data, but showing polygon shapes on a blank background with embedded geographic areas.

▶ The **Azure Map** visual provides a rich set of data visualizations for spatial data, requiring latitude and longitude as two of the attributes in the dataset to render the data on a map.

Chapter VII. Types of analysis in the current business scenario – the connection between Business Intelligence, Big Data and Data Analytics

7.1 The new economic context and the impact of digital transformation on the business environment

The current business scenario, anchored in the Internet Economy (or Web Economy), is mainly characterized by digitization and new information technologies, which have simplified and improved the efficiency of processes in all companies and imposed a new paradigm of terms. Concepts and technologies such as Business Intelligence, Big Data, Data Analytics, Artificial Intelligence, Machine Learning, Cloud Computing, augmented reality, computer simulation, Internet of Things (IoT), 4G and 5G technologies, mobile communications, etc. connect the physical world and the digital world and are often used in modern business solutions, in companies facing digital business transformation. These concepts are relevant to the storage and management of large amounts of data and have changed the business management.

High global competitiveness and the need to compress production times, without compromising on quality, make the digitization of economic processes a necessity and an important concern for companies. To be competitive, to create a stronger and more digitized value chain, companies must adopt technological and innovative solutions in their business models, which favors the obtaining of data and the use of information in a faster, more direct and cheaper way, in order to facilitate decision-making and thus obtain a competitive advantage.

As these technologies are on the rise, there are many consulting companies or companies specializing in information analysis that prepare detailed and well-documented reports on how the combination of these technologies has brought about a notable increase in the global flow of available, distributed information through different supports or technological devices (computers, mobile phones, tablets, Smart TV, etc.).

According to the report "Data Never Sleeps 10.0" by the consulting firm DOMO (2023), the current phase of development of the digital economy places us at a defining moment, where the generation and analysis of data have acquired a unique importance in history. The digital participation has gained hundreds or perhaps thousands of percentage points over the past ten years through peer-to-peer payments, online shopping, social networking, streaming entertainment, and other activities. One thing has remained consistent in society despite pandemics, economic ups and downs, and worldwide unrest: we are using new digital technologies more and more for interacting, communicating, and conducting business as well as for personal purposes.

The same report estimates that in April 2022, approximately 5 billion people, or 63% of the world's population, had access to the Internet. 4.65 billion of them, or more than 93%, were social media users. Statista estimates that 97 zettabytes of data were generated, recorded, copied and consumed globally in 2022, and by 2025, this amount is expected to grow to 181 zettabytes. All this data requires some kind of automated analysis.

This situation is also found in business, as large volumes of data are generated at each stage of the value chain.

7.2 Big Data: architectures, technologies and solutions

Big Data technology was born with the goal of obtaining and collecting large volumes of data very quickly, almost in real time, and processing this data to gain new insights. Big Data encompasses technology, hardware, software, services, telecommunications and, most importantly, a new way of thinking about managing, analyzing, interpreting and drawing conclusions from data.

The technology combines structured, semi-structured and unstructured data coming from disparate sources and different formats: sensors of various types (GPS/ RFID/ telemetry/ etc.) and devices connected to the Internet, GPS satellites, meteorological satellites, mobile applications, social platforms, photos, videos, online transactions, financial reports, experimental data, electronic medical records, environmental data, government statistics, biometric data, etc.

Big Data often contains much more unstructured data than standard datasets, requiring real-time analytics. The seven attributes that characterize a Big Data dataset – the seven Vs serve as the basis for distinguishing between a normal dataset and a huge dataset:

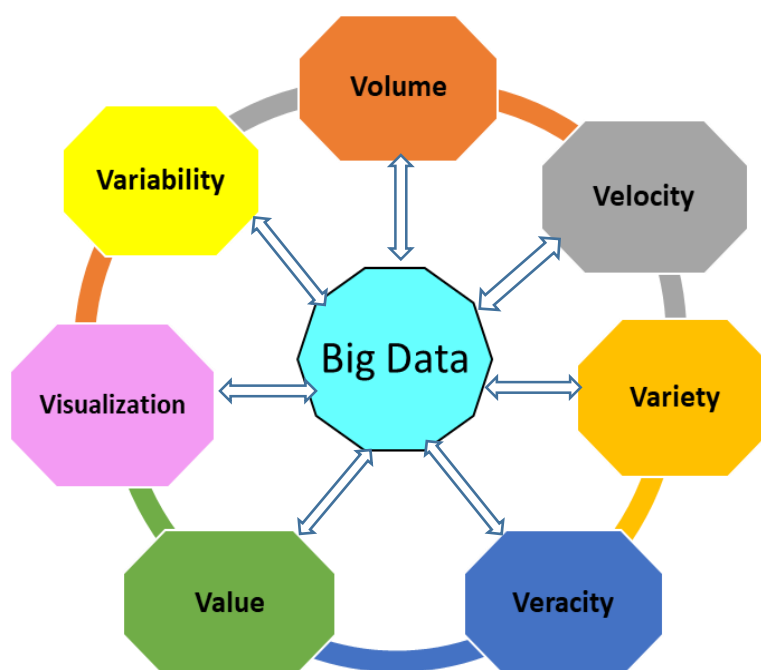


Figure 7.2: Seven V's of Big Data

► **Volume**, which is the most important characteristic associated with the concept of Big data, refers to the size of the data, the amount of data that is generated, collected and processed with every moment in our highly digitized world. In general, connecting to the 2.0 world generates more and more data, but the largest volumes are generated by unstructured data, and the previous figure illustrates everything that happens every minute of a day.

Specific to large data volumes is the fact that the size used to determine whether a data set is considered Big Data is not measured in megabytes or gigabytes, but in petabytes (PB), exabytes (EB) or zettabytes (ZB).

► **Velocity** refers to the rate at which data is generated or updated and the speed at which it must be analyzed and applied.

In most applications involving large volumes of data, the data is constantly generated (with real-time streaming) and real-time processing is required. Velocity gives the Big Data process the ability to process and analyze in real time to create real value for beneficiaries.

► **Variety** refers to the diversity of data: data coming from different sources, inside or outside the company, which differ in the type of information, having structured and unstructured characteristics, with different degrees of complexity. Therefore, the difficulty lies in determining what information will be taken from the variety of sources and formatted after analysis and filtering.

► **Veracity**: refers to the accuracy, quality and fidelity of the data – important aspects for the next stages of data analysis.

► **Value**: another feature that marks the importance of Big Data within Information Technologies refers to the fact that the resources generated from the analysis of large volumes of data allow users to better visualize and understand their business in order to reduce risks, detect new opportunities and improve decisions of business.

► **Visualization**: allows data analysts, with the help of reporting tools, to expose and communicate the multitude of information relevant to the company in an easily accessible and easy-to-understand form: in the form of balanced scorecards, dashboards, graphs or interactive tables. Analyzing this information through images is easier for understanding the current state of the business, reducing the time needed to analyze the key performance indicators and facilitating the prediction of trends.

► **Variability**: refers to the inconsistency, unpredictability and variation of different data flows; they can have predictable cyclical behavior or can be completely random, which makes them particularly difficult to manage; moreover, data formats can change, but also the context in which the results of the analysis must be interpreted can change. Therefore, the technologies that make up a Big Data architecture must be flexible, to adapt to these changes.

The challenge of managing the data flows described by these seven characteristics requires a new data management model, a new generation of systems, technologies and architectures capable of extracting information from large and varied volumes of data.

Due to the complexity of Big Data systems, standards and technical studies have recently been launched to provide a solid foundation for addressing the opportunities and problems presented by big data. The framework provided by the ISO/IEC 20547 series of standards (consisting of five parts) provides a Big Data Reference Architecture (BDRA) that organizations can use to effectively and consistently describe their Big Data architecture and implementations, capable to extract information from the large and varied volumes of available data, to process and analyze them in a timely manner and at low costs.

According to Wael William Diab, head of one of the two ISO/IEC committees that created the ISO/IEC 20547 series of standards, these standards address the rising need for more uniformity and

clarity in ideas and procedures when handling this kind of data: “At the heart of the fourth industrial revolution is the ability to derive insights in an increasingly data-centric world. Big data computational systems enable this digital transformation across a wide variety of industry verticals. These standards are in response to an increasing demand for greater clarity and consistency of concepts and processes in handling big data....The BDRA addresses requirements, architecture, security and privacy, use cases and considerations that architects, application providers and decision makers will want to consider in deploying a big data system. This will serve to increase trust and understanding amongst stakeholders and across the whole industry, ensuring big data technologies are used safely and effectively.” [iso.org/news/ref2578.html].

In practical implementations, Big Data architecture varies according to infrastructure and business/industry requirements. In general, the four main logical levels of a Big Data architecture are: data acquisition, transformation and storage, processing and analysis, querying and presentation of results.

A generic architecture is shown in the following figure:

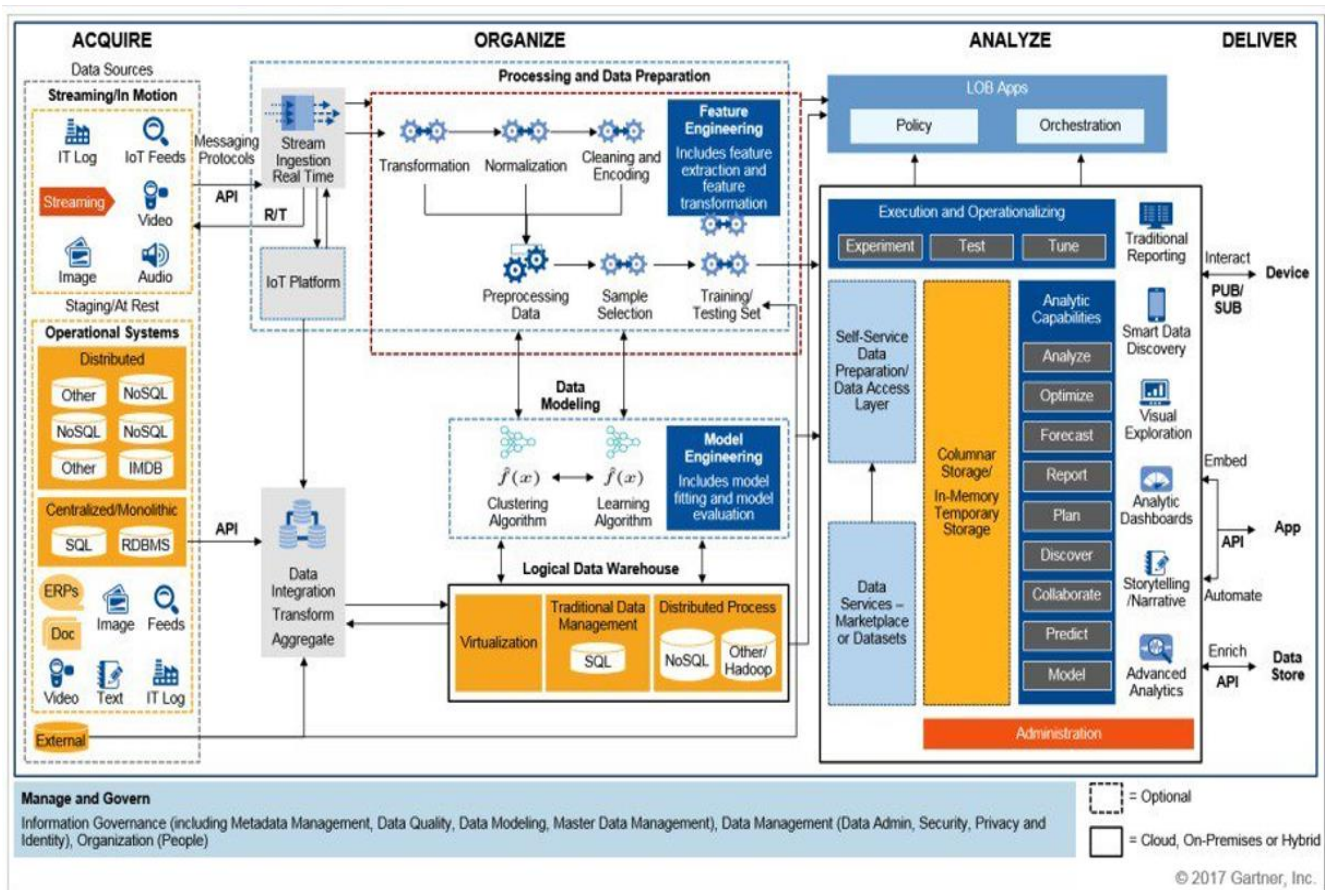


Figure 7.3: Gartner Research: “2017 Planning Guide for Data and Analytics” [https://www.gartner.com/en/documents/3471553]

In this architecture, used in the Business Intelligence solutions that exist today, each of the logical processing layers has adapted and provided new technologies, opening up new opportunities at the same time.

► According to the flow of information in a Big Data architecture, the process begins with the **collection of data**, directly and indirectly, with tools specially developed for such a function. Data is collected regardless of source, volume, speed and type: it can be real-time data sources such as IoT devices (sound sensors, meters that record temperature, light, height, pressure, biometric readers - retina scanners, fingerprint scanners, etc.), application data stores such as relational databases, web server log files, data generated by people sending emails, WhatsApp messages or Facebook posts, etc., weather data, geolocation data for population and traffic information, etc.

► **Data storage** is realized through two basic elements: distributed file systems and databases.

► *Distributed file systems* can contain significant amounts of large files in various formats and are preferred for storing unstructured data. The architectural principle is based on the idea that instead of using a single very powerful server to store and process information, groups of servers can be used to analyze data in parallel. Thus, the solution consists of a network/cluster of interconnected computers/nodes, which allow the storage and processing of large volumes of data. By distributing data across multiple machines, distributed file stores can significantly improve performance and fault tolerance.

Choosing the right storage solution is essential to efficiently manage and access collected data, and many of the best tools used in Big Data are open source. One such system is Apache Hadoop - considered to be the standard framework for the storage, distributed processing and analysis of large volumes of data.

Hadoop contains a distributed file system on each cluster node: HDFS (Hadoop Distributed File System). Clusters can grow from a few nodes to thousands or tens of thousands of nodes, making the solution highly scalable. New machines are added to the cluster as the amount of information and processing needs increase. The big advantage is that computers with more storage and processing power do not have to be bought to improve the performance of the platform. This goal of scalability is pursued by Big Data systems: the ability to vary the size (either increasing or decreasing) according to needs, which does not affect the overall performance of the entire system.

To perform information processing and analysis in this distributed file system, the MapReduce framework is used. The MapReduce programming paradigm allows calculations to be divided and parallelized between an indefinite number of low-cost computers from the cluster (Map process), and then combining the partial results into a single final result. (Process reduction).

The Hadoop platform supports various operating systems, is used by companies such as Facebook and Yahoo, and is also frequently used on cloud platforms such as Amazon EC2/S3 or Google Cloud.

► *Relational databases*, which emerged as a new paradigm in the 1970s, follow the relational model that allows data (stored in tables) to be interconnected by relationships that allow them to be correlated from different tables. Languages based on SQL (Structured Query Language) – the specific language for operating with these databases, such as MySQL, PostgreSQL and Microsoft SQL Server, are suitable for managing structured data, with well-defined schemas and less suitable for datasets massive or unstructured data.

► *Non-relational databases* or NoSQL ("Not Only SQL"), unlike relational databases, store data without structured mechanisms, which makes it possible to use different types of data in a flexible way. Thus, these databases are useful in working with large amounts of data, one of the necessary requirements for data analysis of Big data technology.

NoSQL systems are designed for storing data from applications that support millions of users every day and for parallel processing of data on multiple interconnected servers, enabling low levels of latency in node processors. It provides scalability in areas such as Big Data analysis, Business Intelligence and social networks, flexibility, high performance and supports thousands of concurrent users and millions of daily queries, as well as predictive and exploratory analytics.

Like relational databases, NoSQL databases rely on the ability to respect two of the properties of Brewer's CAP theorem, where C is consistency, A is availability, and P is partition tolerance. The CAP theorem (formulated by the scientist Eric Brewer in 2000) states that in a distributed computing system it is possible to ensure only 2 out of 3 fundamental characteristics of these systems. These are: consistency, availability and partition tolerance:

- *consistency*: it means that the data in the database remains consistent after an operation is executed. For example, after an update operation, all clients see the same data.

- *availability*: it means that all information in the distributed data storage system will always be available, so the system has no downtime.

- *partition tolerance*: it means that the different parts of the distributed system (the nodes) will continue to function normally, even if the communication between them is interrupted.

In other words, any Big Data solution that is implemented on a distributed computing system cannot respect all three dimensions at the same time and will be forced to give up one of these 3 features. More precisely:

- a distributed environment that ensures consistency and availability will not have partition tolerance: CA-type systems - systems where if communication between its nodes fails, the whole cannot function.

- a system that has partition availability and tolerance will be poorly consistent: AP-type systems - systems where when a failure occurs at one of the nodes, information will be available, but may not be consistent.

- a consistent and partition-tolerant system may not always be characterized by availability: CP-type systems - systems in which if an incident occurs, part of the information will not be available, but the system continues to work, and the information available will be consistent.

The figure below synthetically presents a classification of data storage systems, depending on the provision of characteristics, according to the CAP theorem. In the context of the CAP theorem, most relational databases focus on providing capabilities C and A of Brewer's theorem, that is, consistency and availability, while sacrificing partition tolerance.

NoSQL solutions prioritize different aspects of the CAP theorem based on the specific type of NoSQL database.

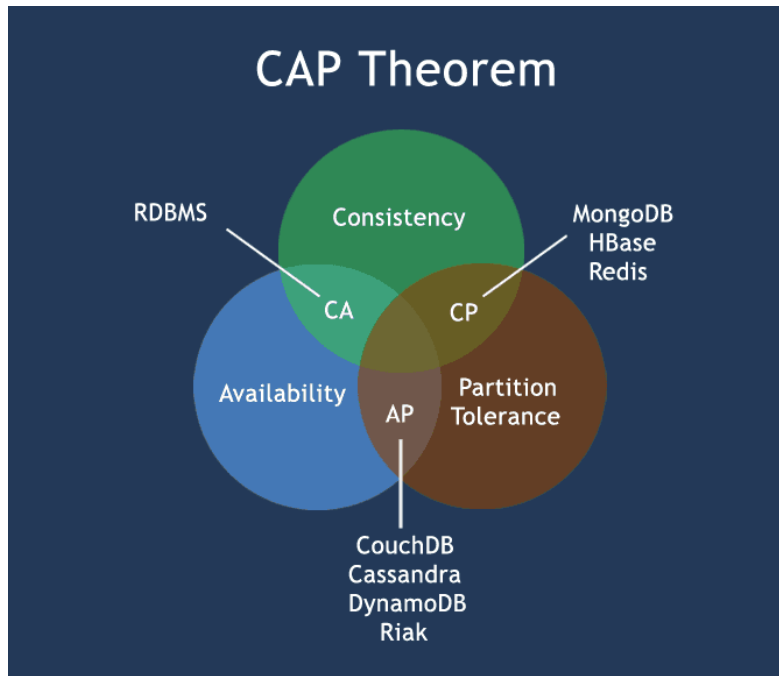


Figure 7.4: CAP Theorem (Brewer's Theorem)
[\[https://docs.deistercloud.com/content/Technology.50/NoSQL\]](https://docs.deistercloud.com/content/Technology.50/NoSQL)

One of the most famous NoSQL databases is MongoDB. The related technology involves working with several connected servers, storing in each of the nodes certain information that must be communicated with the rest of the nodes that make up the system. It is a document-oriented database (it stores data in documents, not in records), which means that each record can have a different data schema. Documents are stored in BSON format, which is a binary representation of JSON.

Apache Cassandra is a free and open-source software for managing NoSQL databases, built to manage large volumes of information, distributed across multiple servers. It has its own query language, Cassandra Structure Language (CQL). It is used for example by Netflix to store all its streaming data and user viewing history. Another example is Spotify, which uses Cassandra to manage its music suggestion system for its users.

► **Data processing** is the next step in a Big Data system. Processing involves integrating, cleaning and transforming collected data to obtain consistent and reliable datasets for further analysis that will extract meaningful insights. Information processing tools have evolved considerably, especially for unstructured data, the distributed alternative being of great interest. The distributed architecture ensures the sharing of the set of resources (computing, storage, memory resources) between several computers in a network, which can be close or distant from a geographical point of view, but connected to each other, to perform a series of operations.

It is a flexible architecture, because computers (called nodes) can be connected or removed from the network without having a significant impact on its operation.

Several tools have been developed to accomplish this process, offering different ways to perform distributed processing and analysis of large data sets:

► *batch processing*: the data sets collected over time are periodically transferred for processing in an analysis system. Although batch processing data is very efficient in processing large volumes of data, the duration of the process can be very variable, from minutes to hours, therefore, in this type of processing, the time required should not be a priority aspect.

In batch processing, which is based on the Map-Reduce model, batch jobs are configured to run without manual intervention and involve three steps: collecting data in a batch, processing the batch, and storing the output data

► *streaming processing*: involves continuous processing of input data as soon as it is submitted for processing. The processing therefore takes place immediately after the interception of the input data and involves performing independent calculations on small data sets in a short period of time or almost in real time. Compared to batch processing, which has the advantage of a large data processing capacity, but which requires a relatively long data collection time, followed by the actual processing, in the streaming model the results are produced almost instantly and can be viewed in real time. So, this type of processing focuses on the speed of reading and analyzing data, which is a necessity in many practical applications in Industry 4.0.

► *data processing based on the Lambda architecture*: it is a hybrid processing solution, which combines the ability to process large data sets, under the conditions of a real-time response. The design of this architecture features an approach where real-time and batch processing work in independent frameworks, combined with a third global component that unifies the results of both processes:

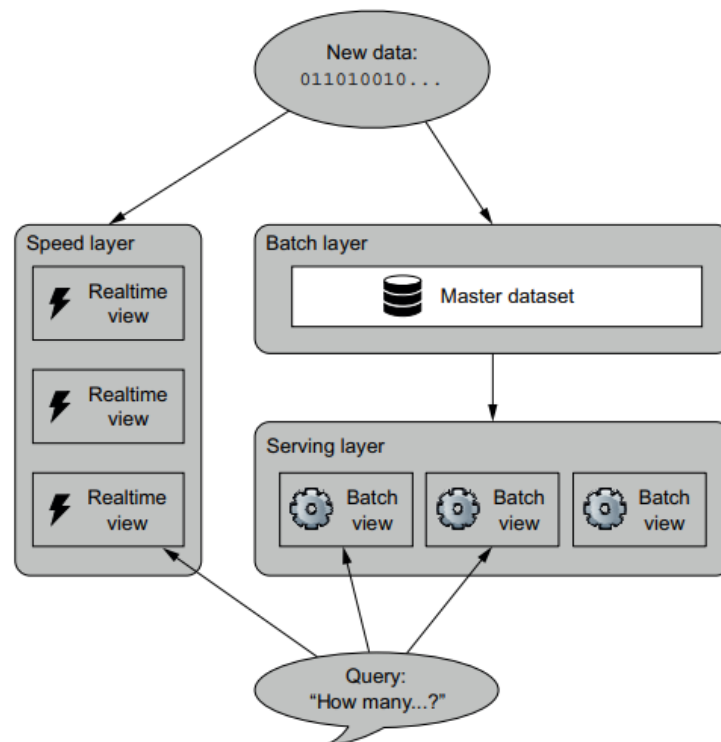


Figure 7.5: The elements of the Lambda architecture [Marz & Warren, 2015]

As shown in the figure above, this architecture is composed of three levels:

- *The batch layer* (or Cold Layer) and *the streaming layer* (Speed Layer or Hot Path), which performs the data processing using the two methods described previously. The main idea of how this architecture works is that all information entering the system is replicated, being submitted for both batch and streaming processing. In this way, any query made to the system can be solved by combining the results obtained from the views of both layers.

Unlike the speed layer, where the data is transient, being kept only for a period of time, the batch processing layer saves the data in a historical manner, or in other words, all processed data is saved persistently.

- *The serving layer*: indexes the views of both previous layers so that they can be consulted in an ad hoc, low-latency manner, which basically allows fast and advanced access to processing results.

► **Data analysis and visualization** represent the last stage in the flow of information in a Big Data architecture. Due to the ability of organizations to store, analyze and extract value from this information, Big Data has grown rapidly. This expansion has also been facilitated by the development and introduction of several technologies and methodologies that facilitate the management, analysis of data and visualization of the obtained results.

Oracle company [Oracle, 2018] gathers some of the Big Data solutions that organizations have used and groups them under the following use cases:

- *Improving the customer interaction experience*: Big Data enables the integration of data from customer interaction channels, such as data from social networks, web visits, call logs and other sources to understand and improve the customer interaction experience, reducing customer dissatisfaction and maximizing the value delivered (for example by delivering personalized offers).

- *Operational efficiency*: Big Data allows the analysis and evaluation of data collected along the supply chain: production data, customer feedback, product returns, but also other factors necessary to analyze current market demand, to reduce inactivity and anticipate future requests.

- *Launch of new products and services*: many companies use Big Data analysis in the analysis of test markets, interest groups, social networks to plan, produce and launch new products. By ranking the key attributes of past and current products or services and analyzing the relationship between these attributes and the commercial success of the offerings, companies build predictive models for new products and services.

- *Predictive maintenance*: the analysis of large data sets resulting from the recordings of various devices and sensors can predict mechanical failures or blockages in the operation of parts and equipment, making maintenance profitable and maximizing their service life.

- *Fraud*: Big Data helps identify patterns in large volumes of historical, transactional and customer behavioral data that may signal fraudulent activity.

- *Driving innovation*: By studying the interdependencies between people, institutions, entities and processes, Big Data can drive innovation. The information obtained can improve planning or financial decisions, can help to implement dynamic prices, etc.

Today, due to the complexity of large data sets, several types of tools are used in their analysis process: technologies, techniques and algorithms adopted from statistics, computer science, applied

mathematics and economics, which have been adapted to operate together to obtain the maximum value from data to discover trends, consumer preferences, behavioral patterns and correlations hidden in data.

The current trend is to use advanced combined analysis, which combines both structured and unstructured data and to develop new analysis solutions, new tools and new technologies, as the volume of data involved in the analysis increases.

In evaluating current or potential solutions for analyzing large data sets, using Gartner's four-stage analytics maturity model provides valuable insights. Gartner distinguishes 4 incremental levels of value and difficulty of data analysis within companies - descriptive, diagnostic, predictive and prescriptive analysis [Elliot, 2013]:

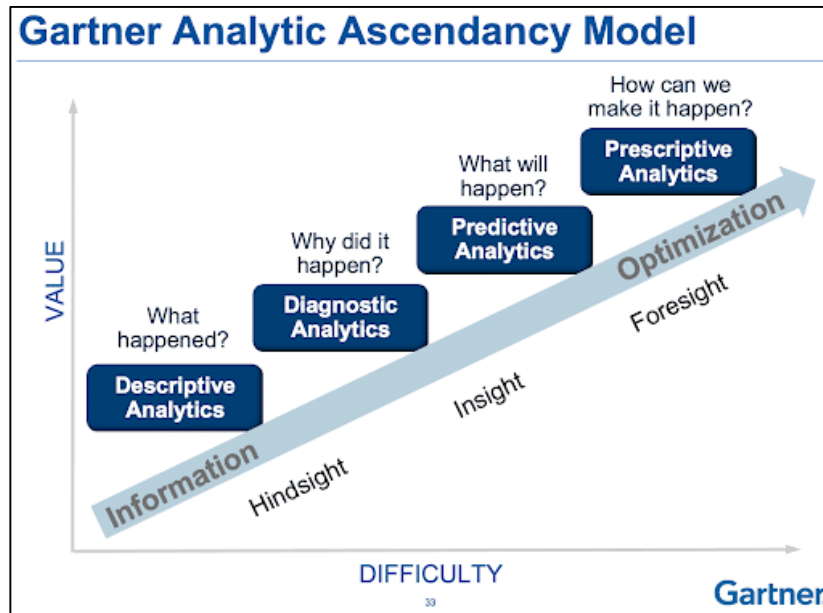


Figure 7.6: Gartner Analytics Maturity Model [Elliot, 2013]

► *Descriptive analysis level:* it is the starting point in data analysis, which focuses on understanding the past. Its function is to describe and summarize existing data to diagnose and discover what happened in the past and why: what trends and patterns emerge in a given process from the study of historical data.

Analysis methods include descriptive statistics (totals, averages, percentage changes, etc.), reports, tables, dashboards and domain-specific reports, ad hoc reports, graphs and numerical summaries that provide a retrospective view of the data.

This type of analysis can be used in many businesses, covering a wide range of purposes, from tracking inventory, production, operations or customers (and their demographics), to highlighting sales performance over a certain period of time or number of incidents reported.

► *Diagnostic analysis level:* it focuses on deepening the exploration and analysis of the data to discover or determine why a certain event happened and to determine, based on the examination of the relationship between the key variables, what factors contributed to the result obtained, so that there is the possibility of taking measures to mitigate the identified inefficiencies.

Analysis methods include correlation analysis and clustering techniques, operational or ad hoc reports, text mining, web mining, graph mining, network analysis, etc.

This type of analysis can be used to answer questions such as: why did sales decrease in the current period compared to the same period of the previous year, why did incidents increase, why did costs increase, etc.

► *Predictive analysis level:* focuses on using past data to make predictions about what events might happen in the future, to predict trends or likely outcomes. By developing a prediction model, it tries to answer the question: "what is likely to happen?"

Analysis methods include more advanced mathematical methods, statistical analysis, predictive modeling, machine learning, and data mining algorithms to capture relationships between various data sets.

This type of analysis can be used to forecast buying patterns, to anticipate how customers will respond to a marketing campaign, trends in sales activities or customer behavior, stock market behavior, or to predict customer rate: the probability that customers to make future credit payments on time or the probability of a customer's credit default.

► *Prescriptive Analysis Level:* It is the highest level of the analytical maturity model. Starting from the predicted results generated by a predictive model, it focuses on what needs to be done, what specific actions need to be taken, trying to analyze the effect of future decisions to provide advice on possible outcomes before decisions are made.

Achieving effective prescriptive analytics requires more advanced Artificial Intelligence models, complex algorithms based on Neural Networks, heuristic learning, machine learning and optimization to suggest the best possible decisions based on available data and predefined goals.

This type of analysis can be used to answer questions such as what should be done to increase sales by a certain amount in the next period or what should be done to retain customers, to optimize customer experience, to optimize production, etc.

In an article posted on the website of the Mckinsey company [Levene et al., 2018] these analyzes are grouped into three main stages and it is graphically illustrated how the different levels of data analysis provides a greater competitive advantage, the greater the effort made by the analytical team.

According to this representation:

- Data management, i.e. raw data, clean data and reporting (standard reports and ad-hoc queries) requires little effort and hardly provides a competitive advantage.
- Descriptive analysis i.e. data filtering, alerts, clustering techniques, trend forecasting and statistical analysis are labor intensive projects that provide greater competitive advantage.
- Advanced analytics i.e. predictive analytics, optimization & simulation modeling and prescriptive analytics are very difficult and labor intensive projects that provide exceptional competitive advantage.

All these types of data analysis require that the obtained results are presented to the responsible people in the right format, easy to understand, accessible, and when they need it, so that they can make the right decisions at the right time. The presentation of these data is much more intuitive, much easier

to understand and process by the human brain when it is made in a graphic form. Thus, data visualizations by creating graphic representations facilitate the identification of patterns and trends, can reveal opportunities and identify risks, and allow a quick sharing of information.

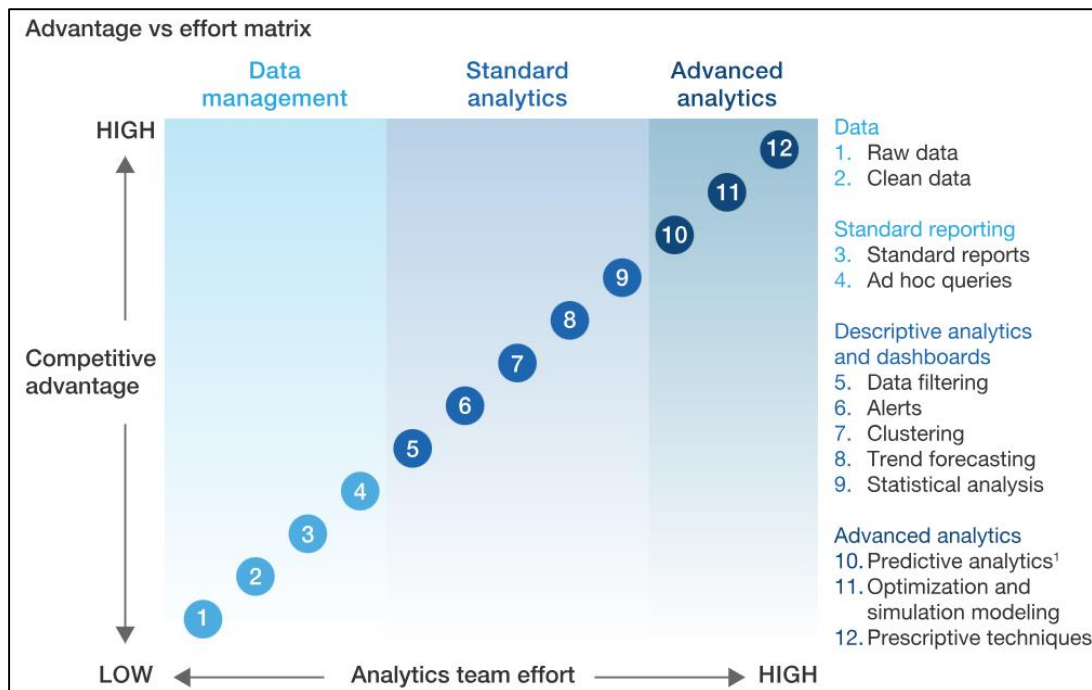


Figure 7.7: How companies can create an analytics strategy that generates value [Levene et al., 2018]

Traditionally, diagrams, schemes, histograms and tables are used to visualize the results, together with numerical formats, in a static format, or the results are presented in discussions and meetings, supported by numerical or graphical presentations, in an oral conversation. But data visualization has undergone major changes in recent times, and the proliferation of Big Data applications in Business Intelligence and the development of hardware that allows data visualization has facilitated the creation of many types of tools and visualization environments: platforms, web applications and other interactive online tools, smartphones, tablets or other agile, comfortable and well-integrated devices in our daily lives that allow the results of data analysis to be presented in a more attractive and understandable way.

Not only the way in which analyzes are visualized has been adapted to modern solutions, but also the structure of these presentations. The tendency is to integrate a multitude of formats for representing all types of data within the same presentation, solutions known as dashboards, which are included in numerous software platforms offered by various providers: Microsoft Power BI, Tableau, QlikView, SAS Visual Analytics, Pentaho, ESRI, Jaspersoft, Quadrigam, etc.

7.3 Advanced analytical features in Microsoft Power BI

As explained in section 6.5 "Create interactive visuals", the Analytics pane allows the addition of several dynamic reference lines, which can help to optimally visualize data and trends: X-Axis constant line, Y-Axis constant line, Min line, Max line, Average line, Median line, Percentile line, Symmetry shading.

Apart from these dynamic lines that can be added to visuals, the application also contains other functions for advanced data analysis, which can considerably improve the analytical capabilities of the reports.

► **What-if scenario analysis** offers users the flexibility to analyze a solution and the consequences that appear in the context of changes in the assumptions on the basis of which it was generated: decrease in product demand, increase in raw material prices, increase in fuel prices or other elements which can disrupt the company's activities. This technique is also called sensitivity analysis.

Taking into account the possible variables that affect a company's projects, based on historical data the company can compare different scenarios and their potential results to make predictions or forecasts of what might happen in the future. Thus, informed business planning decisions can be made. Also, this analysis is useful in identifying and evaluating potential risks or for the optimal allocation of the company's resources.

The scenarios can be simulated by using a what-if parameter, which will take the values entered by the user.

For example, a company wants to analyze sales scenarios when offering promotional discounts of 5%, 10%, 15%, 20%, 25% and 30% respectively.

To create a new what-if parameter, the following steps are performed:

1. The **Report view** → **Modeling** tab → **New parameter** (Parameters section) → **Numeric range**.

In the **Parameters** window that opened, one can define the parameter's properties, including the name, data type, minimum and maximum values:

Parameters

Add parameters to visuals and DAX expressions so people can use slicers to adjust the inputs and see different outcomes. [Learn more](#)

What will your variable adjust?
Numeric range

Name
forecast

Data type
Whole number

Minimum
0

Maximum
30

Increment
5

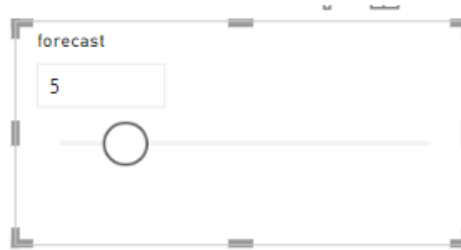
Default

Add slicer to this page

Create Cancel

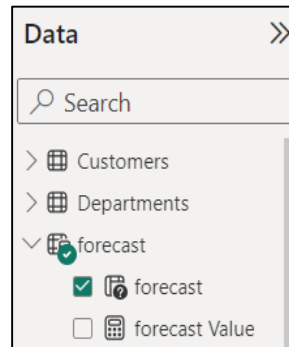
In the **Name** section, enter the parameter name; here the parameter name is **forecast**.

If the option **Add a slicer to this page** is selected, the application will display the slicer on canvas:



Using this slicer, it will be possible to dynamically adjust the rate of increase of the discount value.

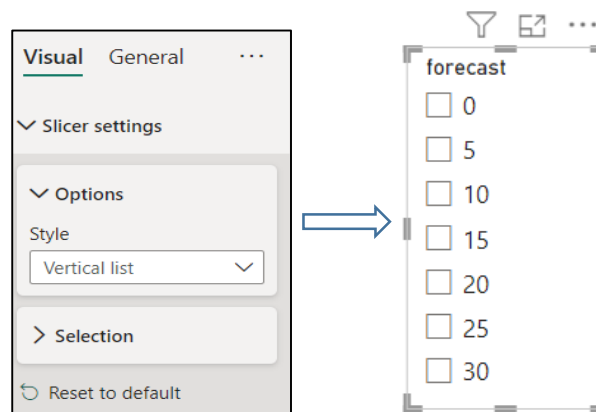
After creating the parameter, we will get in the **Fields pane** the new **forecast table**:



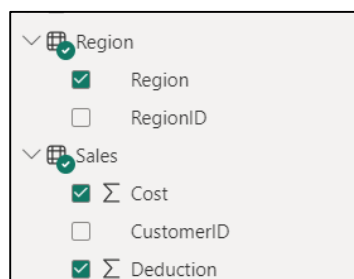
This parameter will create a column in the data table for all the individual values that were specified in the range.

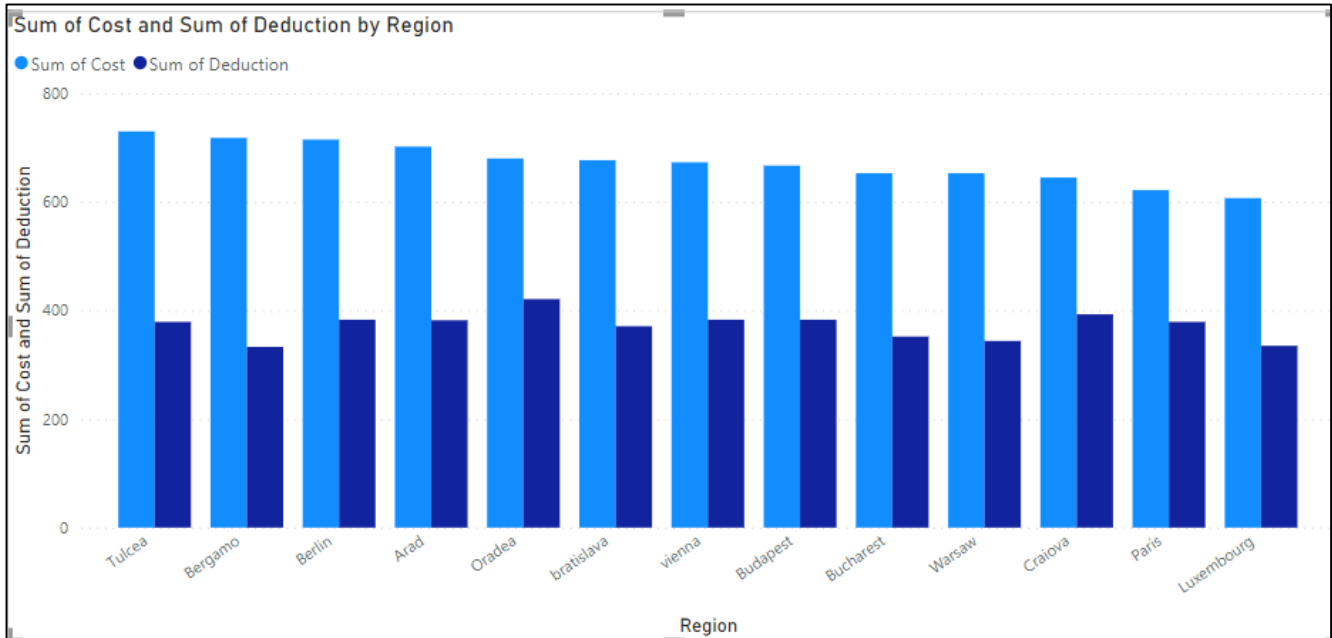
Note: one can change the pattern slicer from “Single value” to “Vertical list”:

In the **Visualizations** pane → **Slicer settings (Visual section)** → **Options** → **change the pattern slicer** from **Single value** to **Vertical list**:



To observe the effect on sales, a new "Clustered column chart" visualization is created with the fields:





To generate an interactive change value, a new DAX measure must be created:

```
forecast-Sales = SELECTEDVALUE('forecast'[forecast])
```

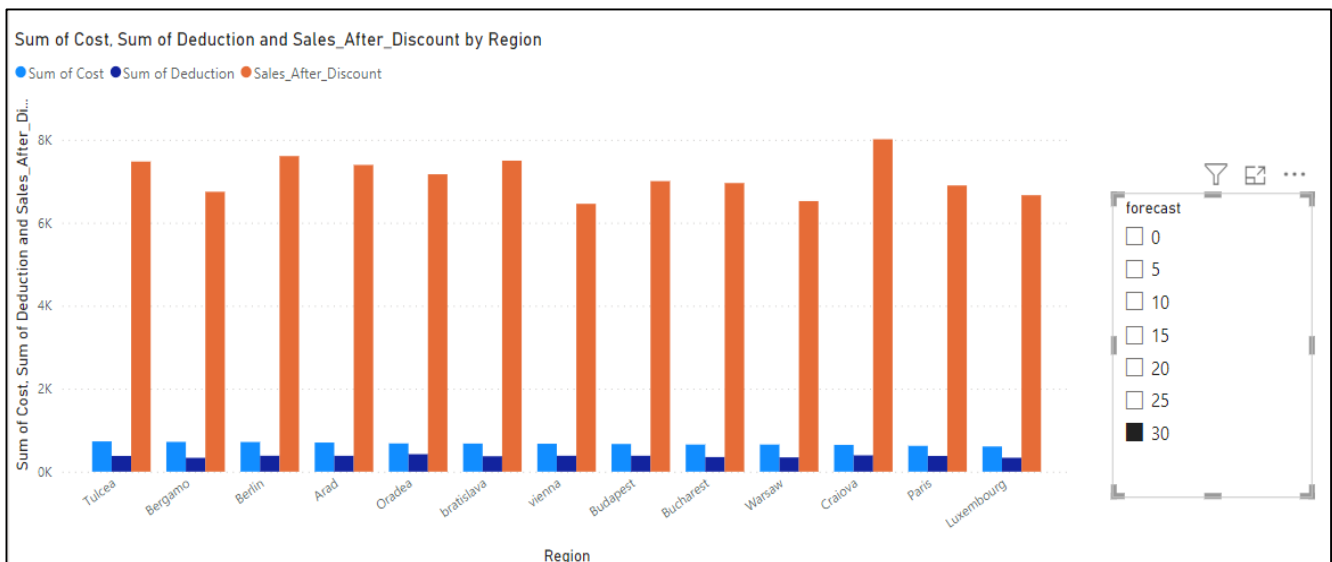
Based on this measure, a new measure, **Sales_Discount**, will be created, multiplying the **sum of Revenues** (from the **Sales** table) by **forecast – Sales** value (from the **forecast** table). Then we convert to percentage, dividing the result by 100:

```
Sales_Discount = ((sum(Sales[Revenue])*forecast[forecast-Sales])/100)
```

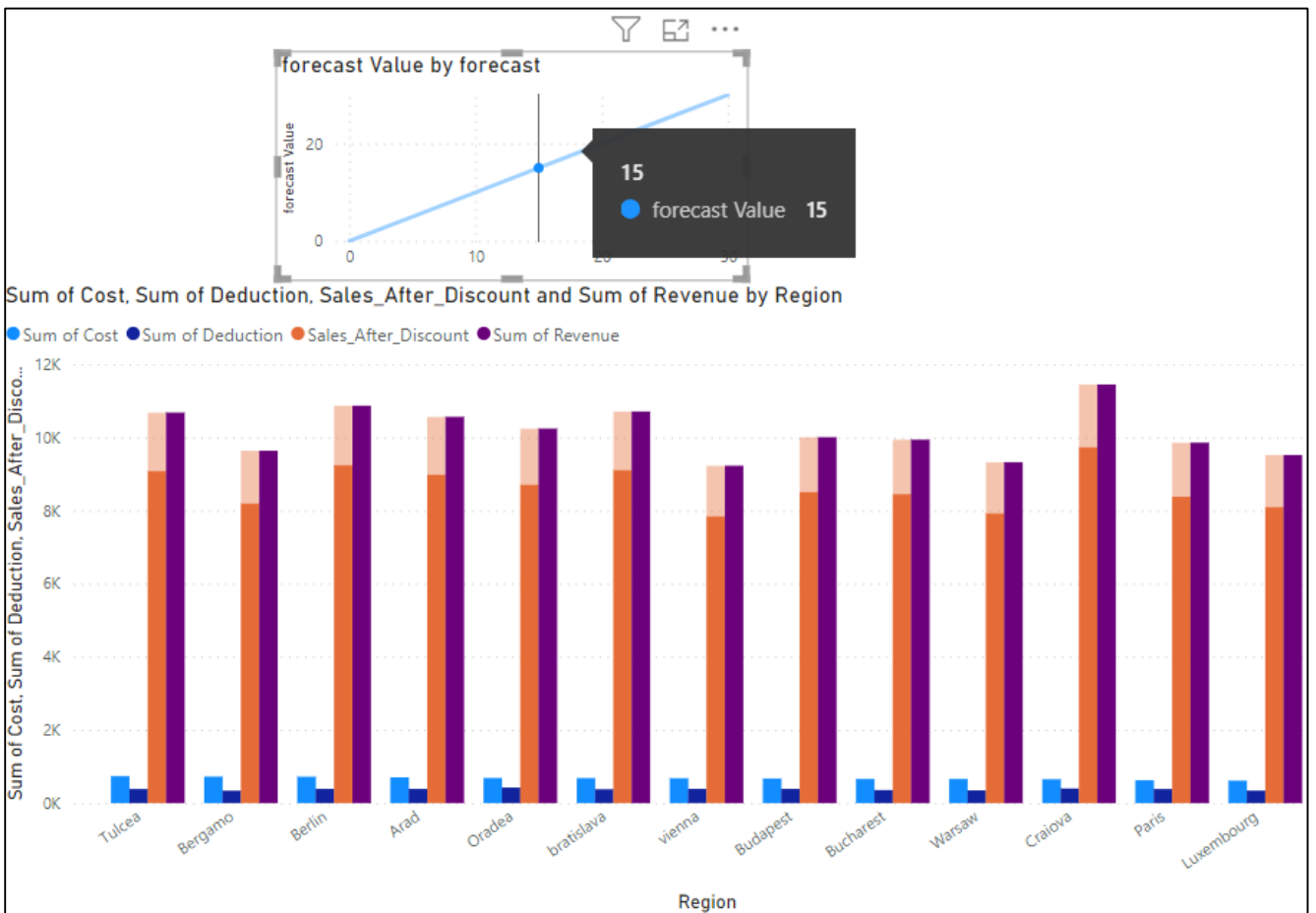
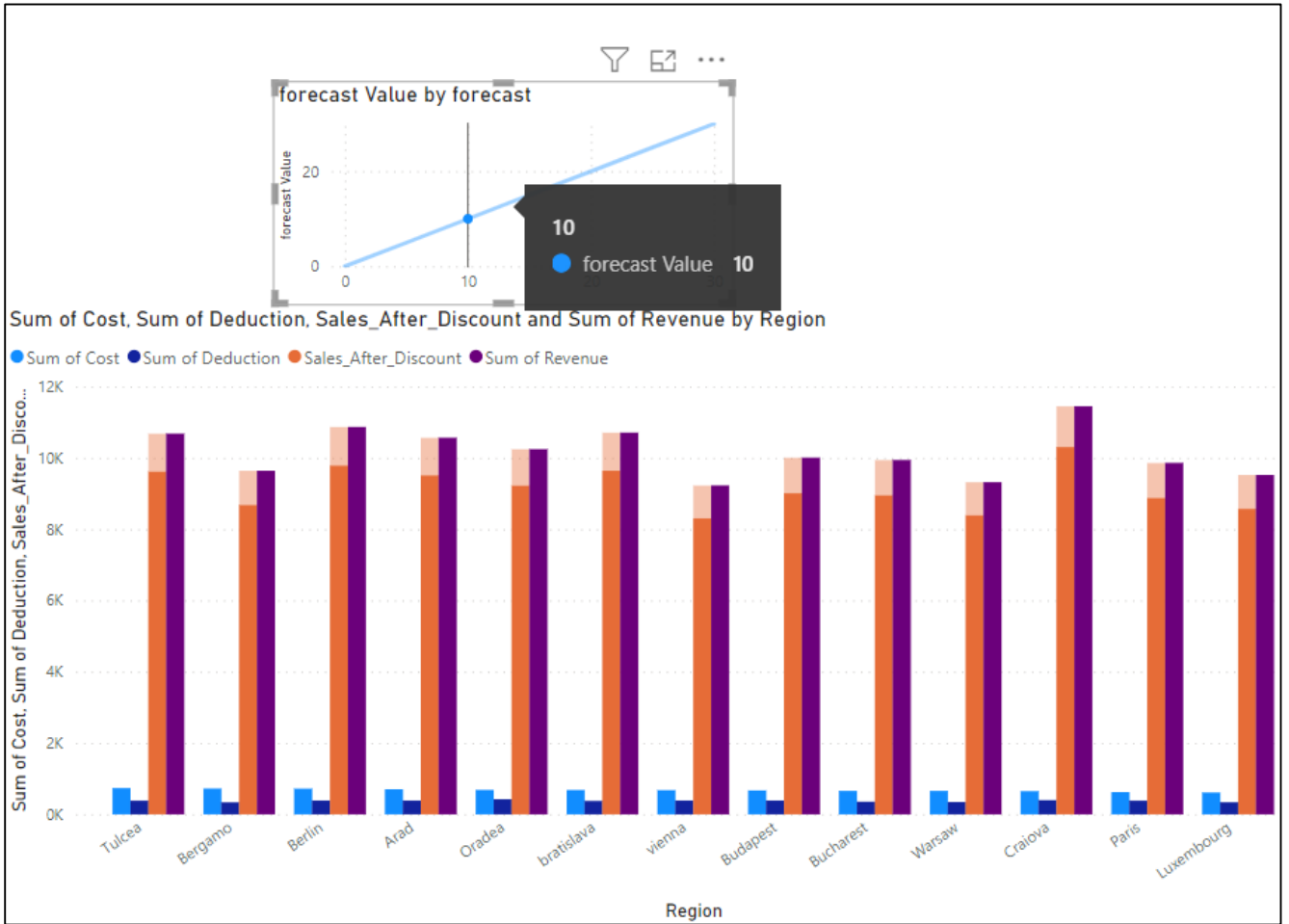
To show the final result after obtaining the discount, the following measure is calculated:

```
Sales_After_Discount = sum(Sales[Revenue]) - forecast[Sales_Discount]
```

Next, by adding the **Sales_After_Discount** field to the report, it can be seen that as the What-If parameter changes, Power BI dynamically updates the images to reflect the results of different scenarios:



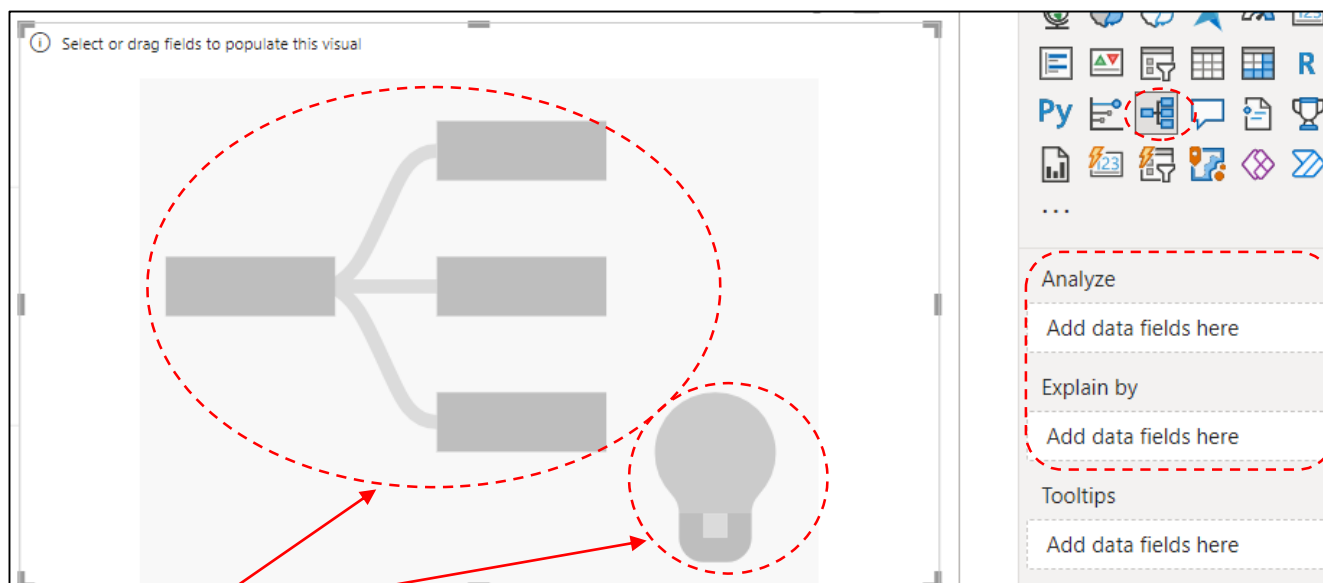
One can see how the forecast changes by dynamically adjusting the discount growth rate in the slicer:



► **Decomposition tree** is a multidimensional data visualization option that arranges features in a hierarchy for quick analysis. Through its Artificial Intelligence function, the visualization allows the user to find the following dimension data according to the defined criteria.

To create a decomposition tree, the following steps are performed:

1. The **Report view** → Visualizations pane → Select the **decomposition tree icon**:

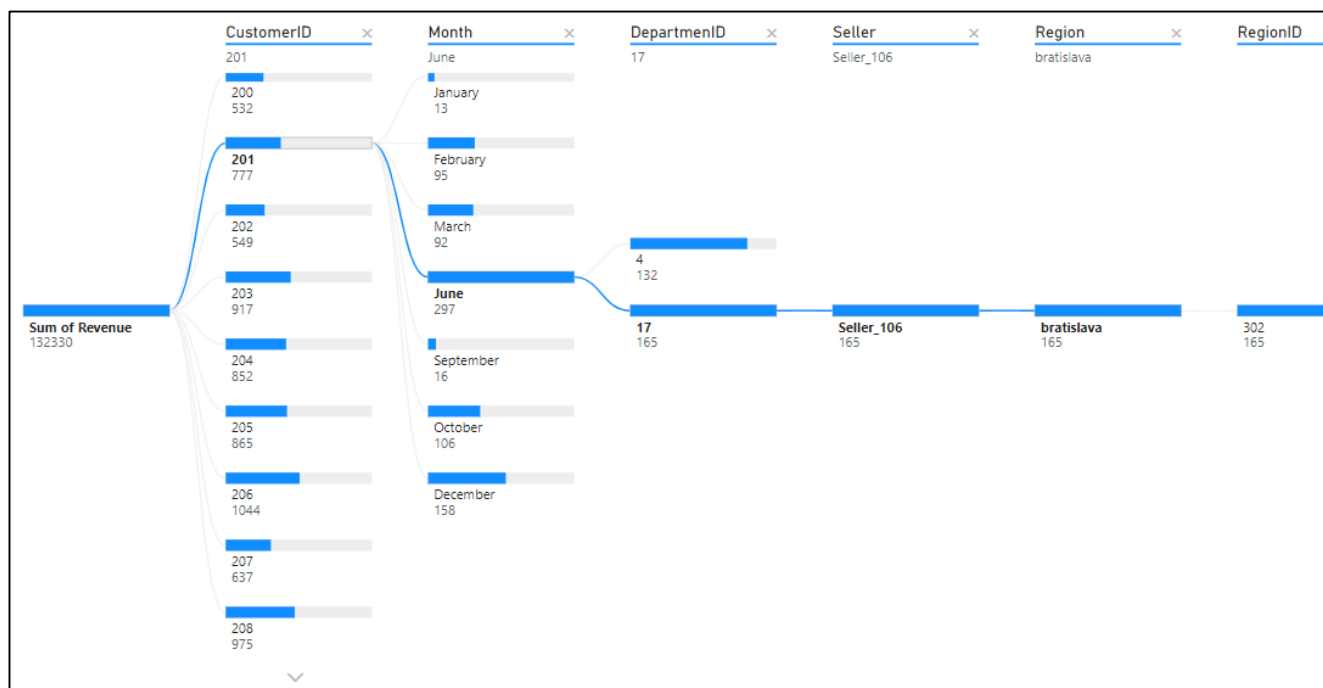


The **Decomposition tree** visual consists of two parts: the visual shape and a bulb representing the Artificial Intelligence and requires two inputs in the Visualizations pane:

Analyze: specify the metric that must be analyzed.

Explain by: one or more dimensions - the drill down columns.

For example, the image below begins with the Sum of Revenue which is broken down into CustomerID, Month, DepartmentID, Seller, Region and then RegionID:

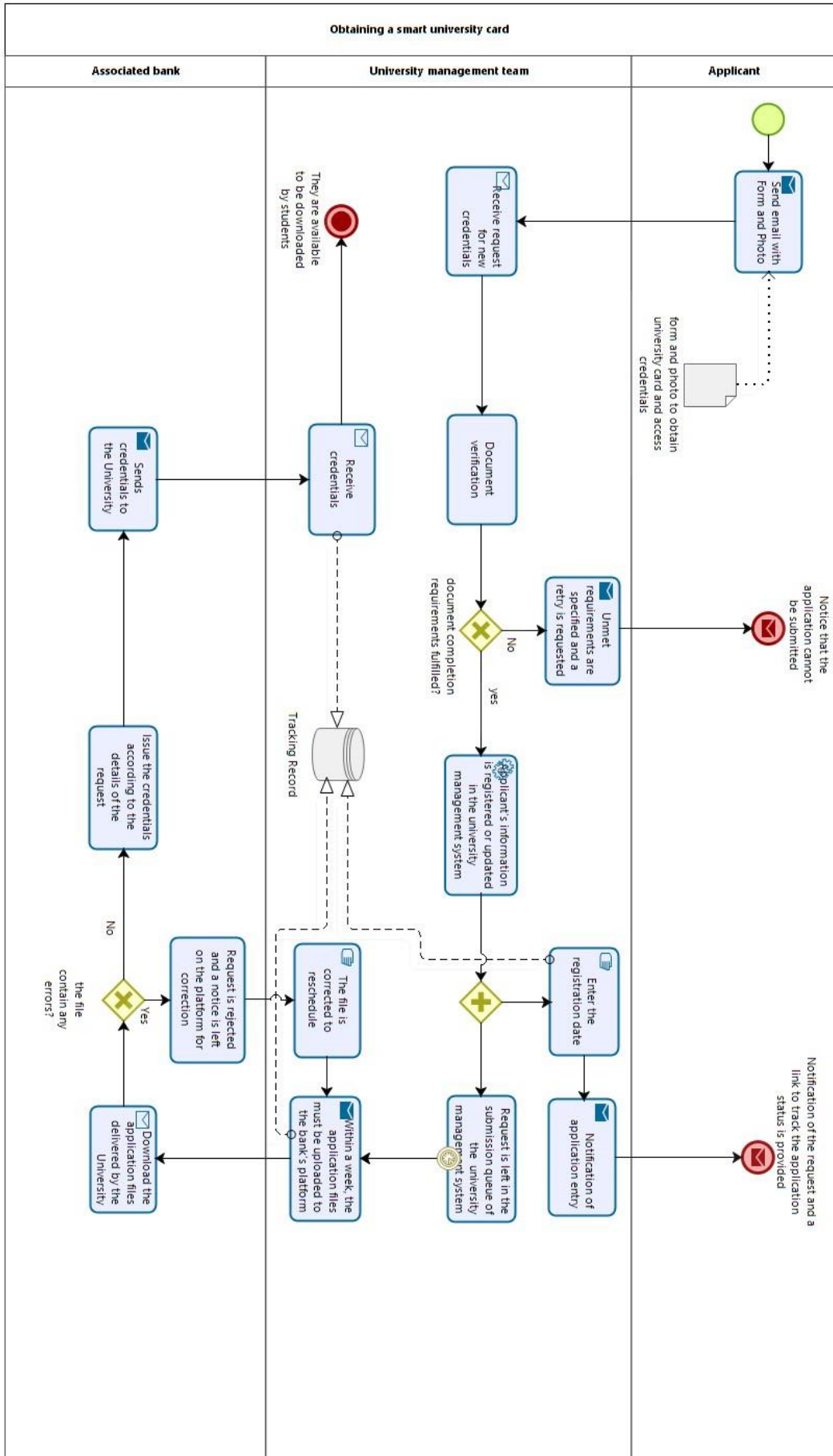


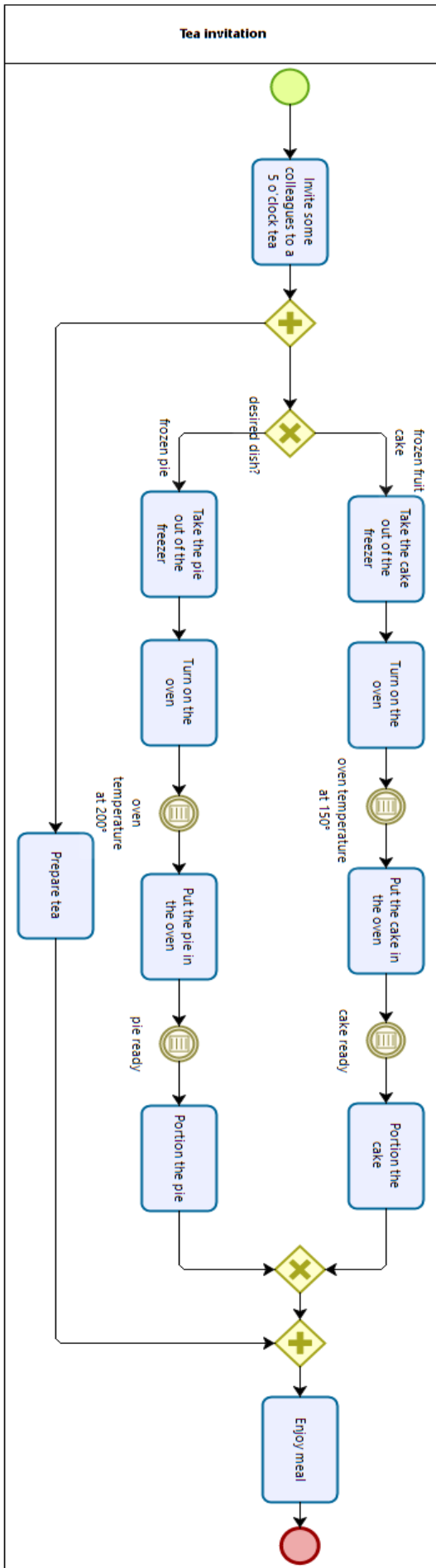
Bibliography

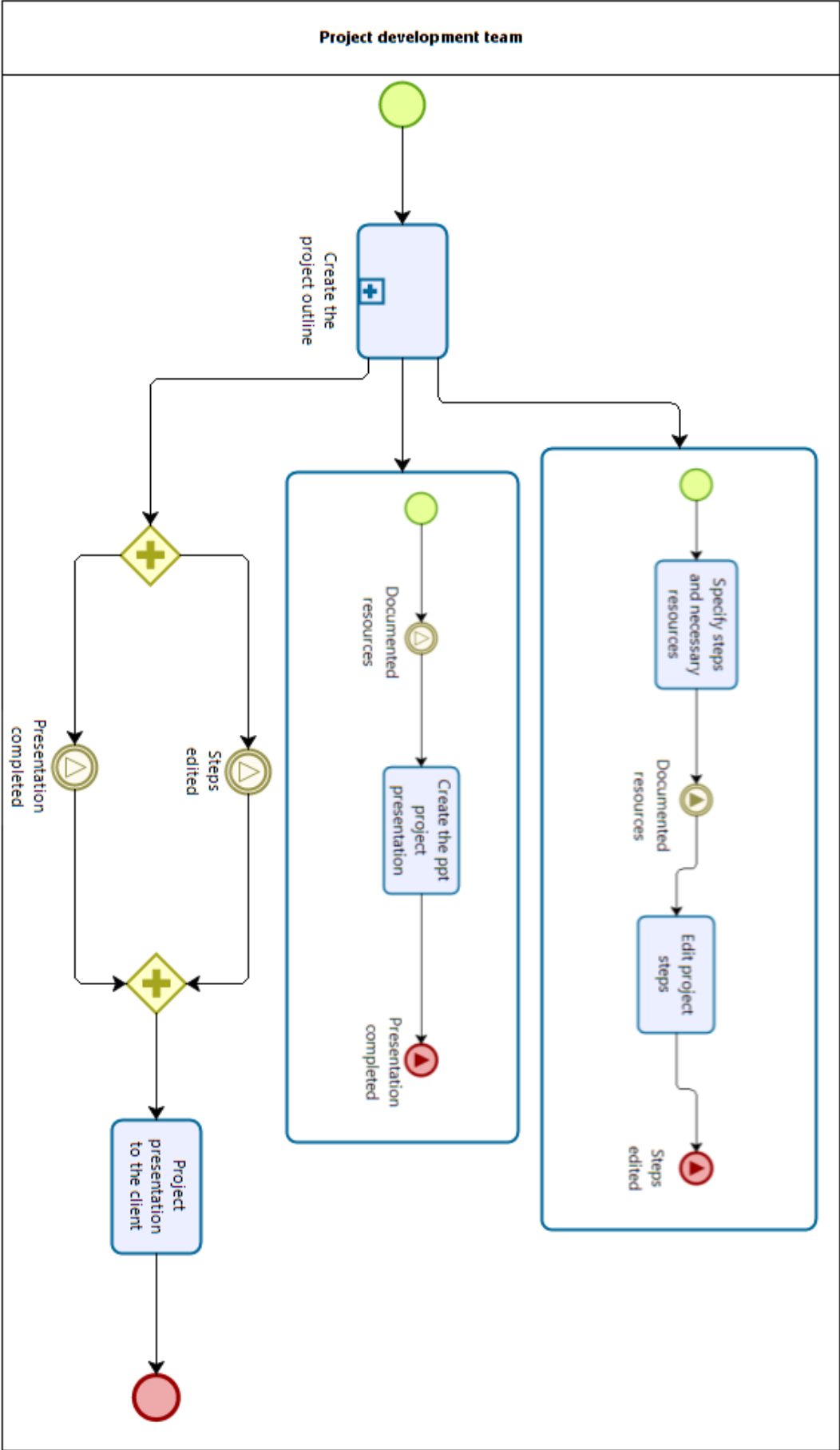
- [1] Ángel, M.M. "Process Management: An Effective Management Approach", *Visión de Futuro*, Vol. 1(13), 2010.
- [2] Aspin, A. "Pro Power BI Desktop Self-Service Analytics and Data Visualization for the Power User-Third Edition", Ed. Apress, 2020.
- [3] Association of Business Process Management Professionals-ABPMP, 2013.
- [4] Bell, B.S., Kozlowski, S.W.J. "A Typology of Virtual Teams: Implications for Effective Leadership", *Group and Organization Management*, Vol. 27(1), pp. 14-49, 2002.
- [5] Benedict, T., Bilodeau, T.N., Vitkus, P., Powell, E. Morris, D. et al. "Guide to the Business Process Management Body of Knowledge (BPM CBOK)" Version 3.0, Ed. CreateSpace Independent Publishing Platform, 2013.
- [6] Brewer, E. "Towards robust distributed systems", *Proceedings of the Nineteenth Annual ACM Symposium on Principles of Distributed Computing*, Portland, Oregon, USA, July 16-19, 2000.
- [7] Carley, K. M., Gasser, L. "Computational Organization Theory" in: "Multiagent systems: A modern approach to distributed artificial intelligence", Ed. Cambridge, Mass. MIT Press, 1999.
- [8] Codd E.F., Codd S.B., Salley C.T. "Providing OLAP to user-analysts: An IT mandate", Technical Report, E. F. Codd & Associates, 1993.
- [9] Elliot, T. "#GartnerBI: Analytics Moves to the Core." *Business Analytics*, February, 2013.
- [10] Ferrer, A.G., Fdez-Olivares, J., Castillo, L. "From Business Process Models to Hierarchical Task Network Planning Domains", *The Knowledge Engineering Review*, Cambridge University Press, 2010.
- [11] Inmon, W.H. "Building the Data Warehouse - 4th Edition", Ed. Wiley, 2005.
- [12] Kimball, R., Ross, M. "The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling, 3rd Edition", Ed. Wiley, 2013.
- [13] Levene, J., Litman, S., Schillinger, I. and Toomey, C. "How advanced analytics can benefit infrastructure capital planning" 2018, <https://www.mckinsey.com/capabilities/operations/our-insights/how-advanced-analytics-can-benefit-infrastructure-capital-planning#/>
- [14] Loshin, D. "Business Intelligence – The Savvy Manager’s Guide", Ed. Elsevier, 2013.
- [15] Manjunath, T.N., Ravindra, S.H., Ravikumar, G.K. "Analysis of Data Quality Aspects in Data Warehouse Systems", *International Journal of Computer Science and Information Technologies*, Vol. 2(1), pp. 477-485, 2010.
- [16] Marz, N., Warren, J., "Big Data - Principles And Best Practices Of Scalable Real-Time Data Systems", Ed. Manning Publications, 2015.
- [17] Meadows, A., Pulvirenti, A.S., Roldan, M.C." *Pentaho Data Integration Cookbook-Second Edition*", Ed. Packt Publishing, 2013.
- [18] Muller, G. "Process Decomposition of a Business", <http://www.gaudisite.nl>, 2010.

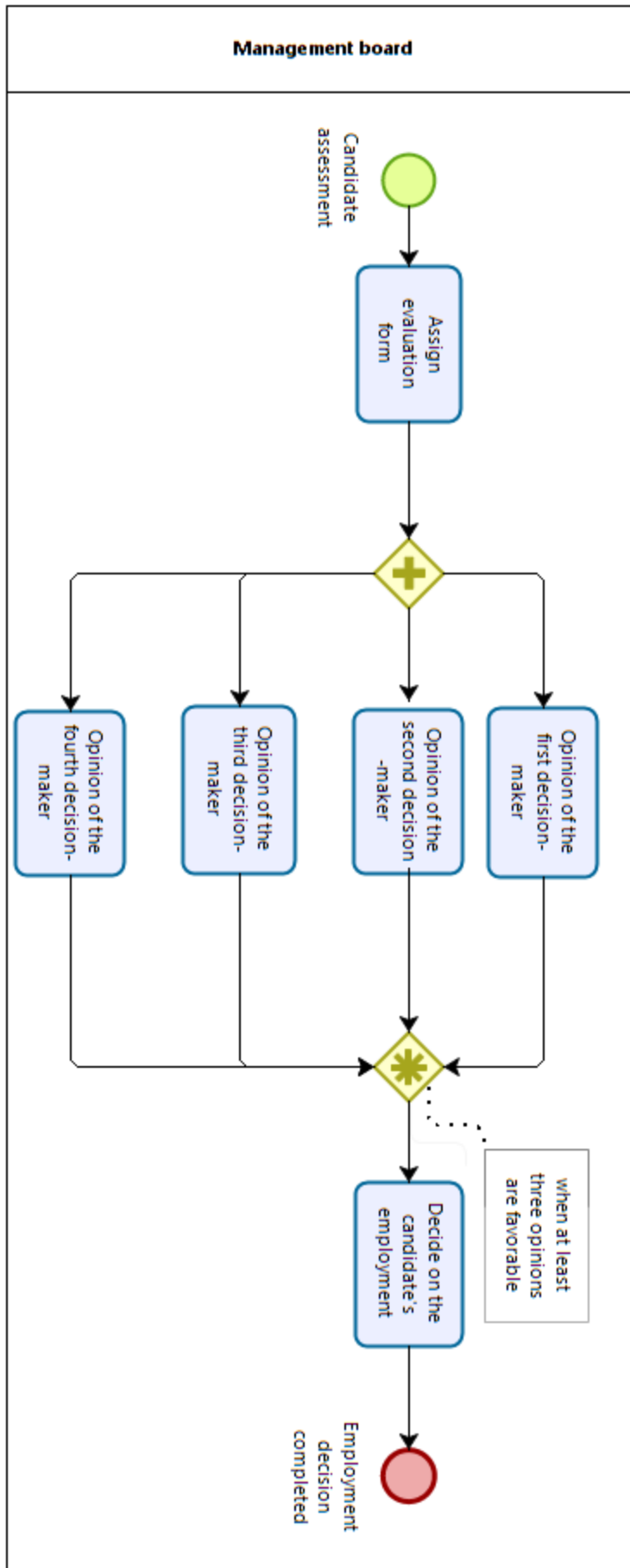
- [19] Oracle "22 Big Data Use Cases You Want to Know", 2nd edition, Oracle Corporation, USA, 2018.
- [20] Rikhardsson, P., Yigitbasioglu, O. "Business intelligence and analytics in management accounting research: Status and future focus", International Journal of Accounting Information Systems, 29, pp. 37-58, 2018.
- [21] Silberschatz, A., Korth, H.F., Sudarshan, S. "Database System Concepts - Sixth Edition", Ed. McGraw-Hill Companies, 2011.
- [22] Stedman, C. "Tableau vs. Power BI vs. Qlik: Comparing BI software choices", <https://www.techtarget.com/searchbusinessanalytics/feature/Tableau-vs-Power-BI-vs-Qlik-How-the-BI-rivals-stack-up>
- [23] Szymczyk, K., El Emary, I.M.M., "Advanced Trends in ICT for Innovative Business Management", Ed. CRC Press, 2023.
- [24] Wang, R.Y., Strong, D.M. "Beyond Accuracy: What Data Quality Means to Data Consumers", Journal of Management Information Systems, Vol. 12(4), pp. 5-33, 1996.
- [25] Weske, M. "Business Process Management - Concepts, Languages, Architectures", Ed. Springer, 2007.

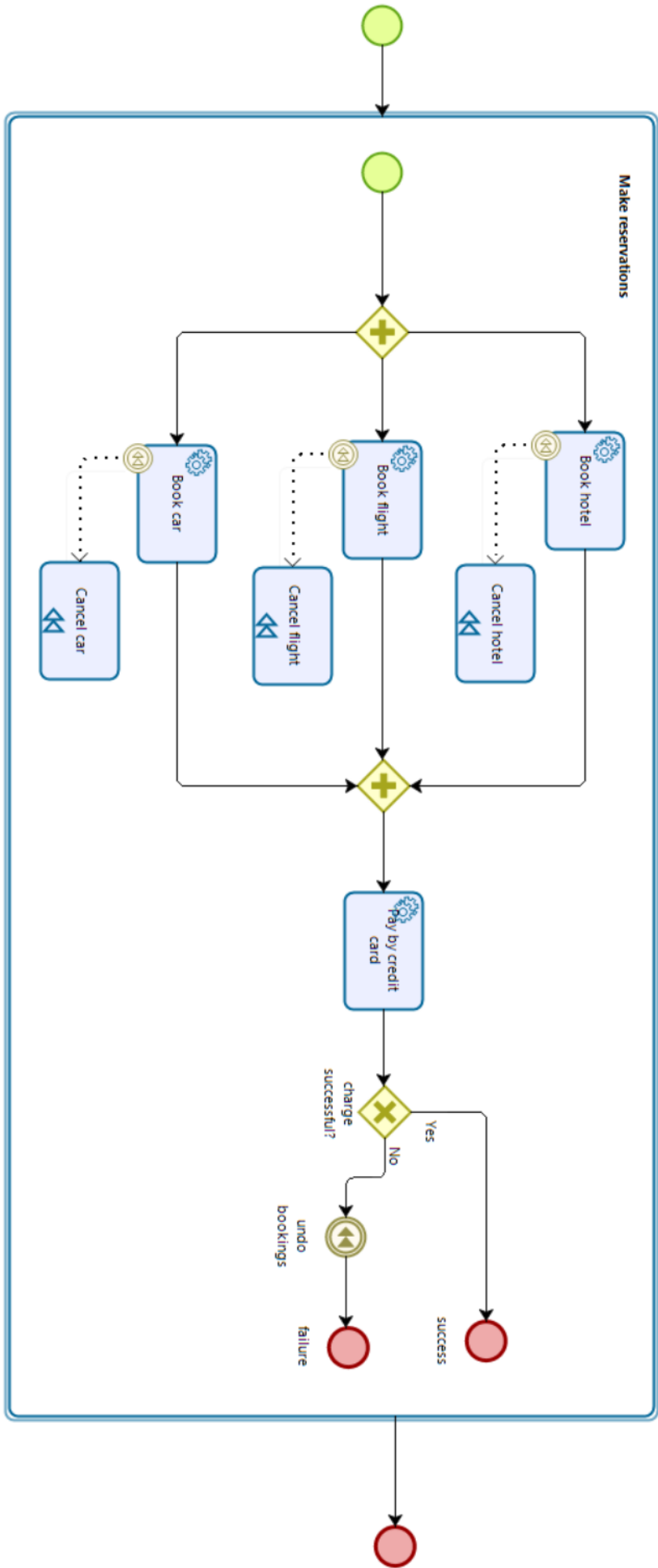
Appendix 1

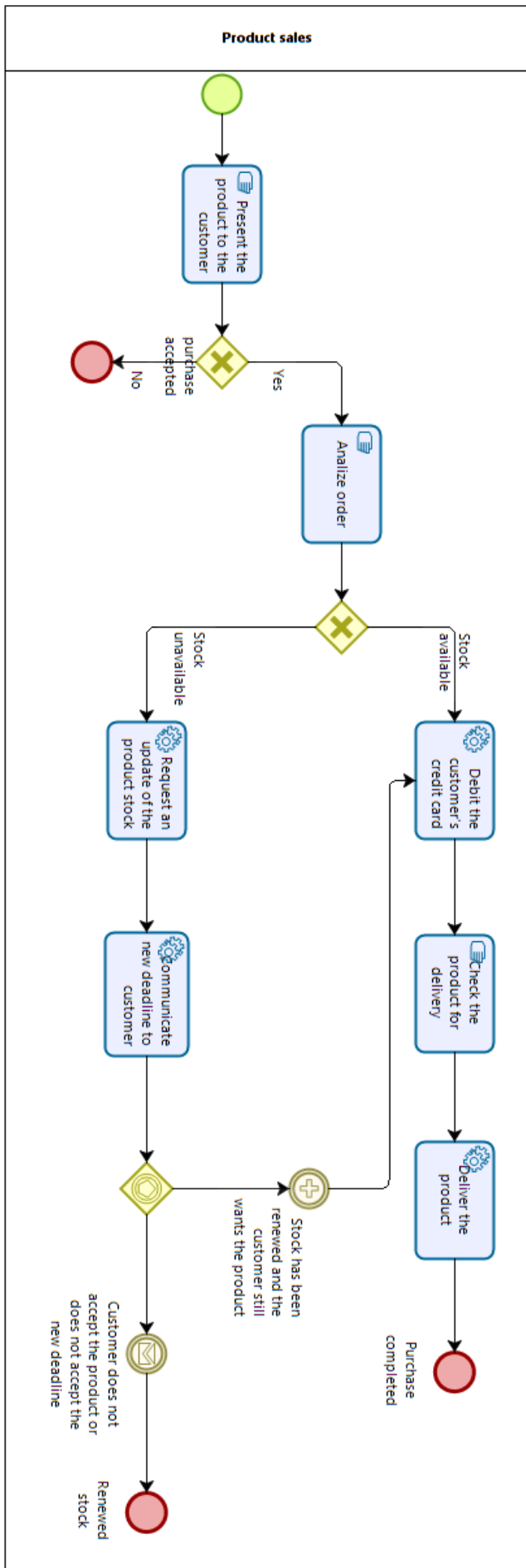


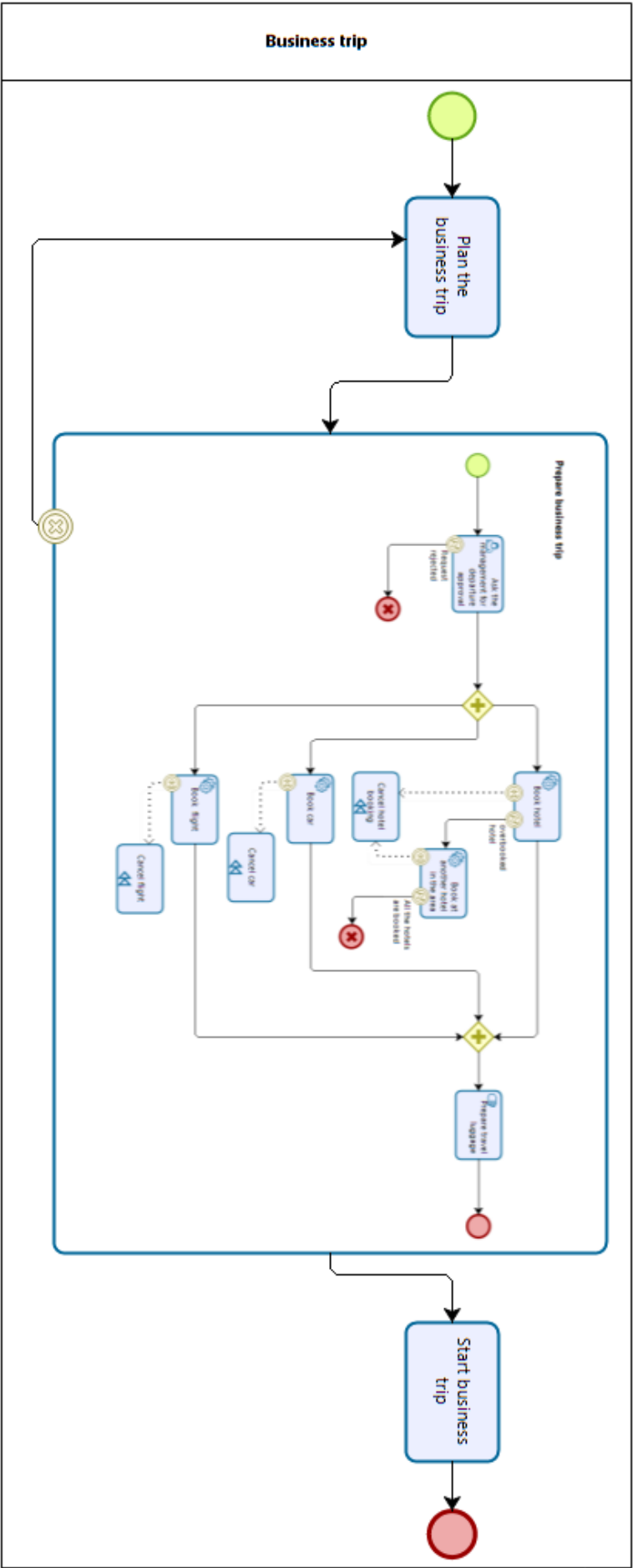




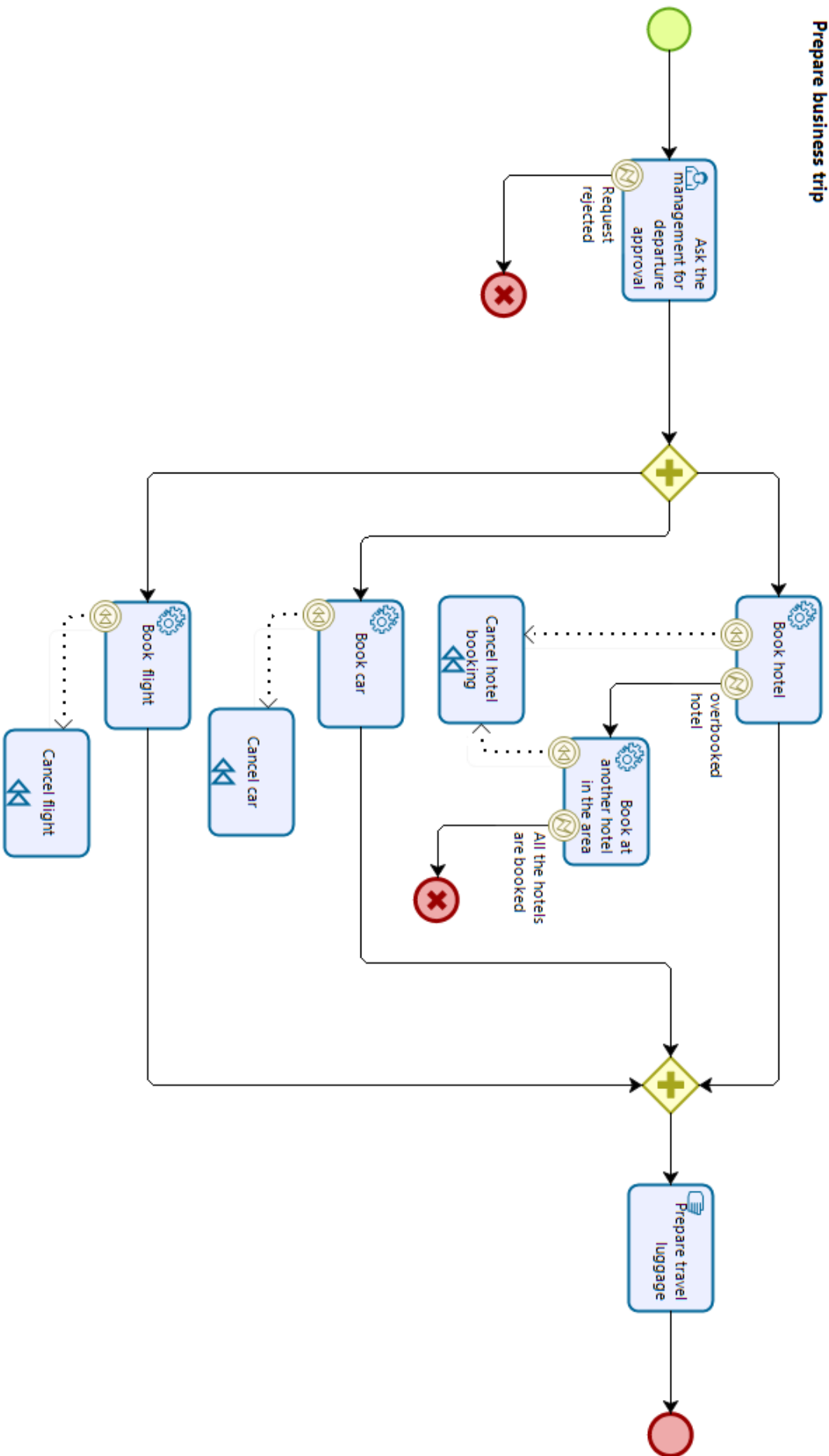


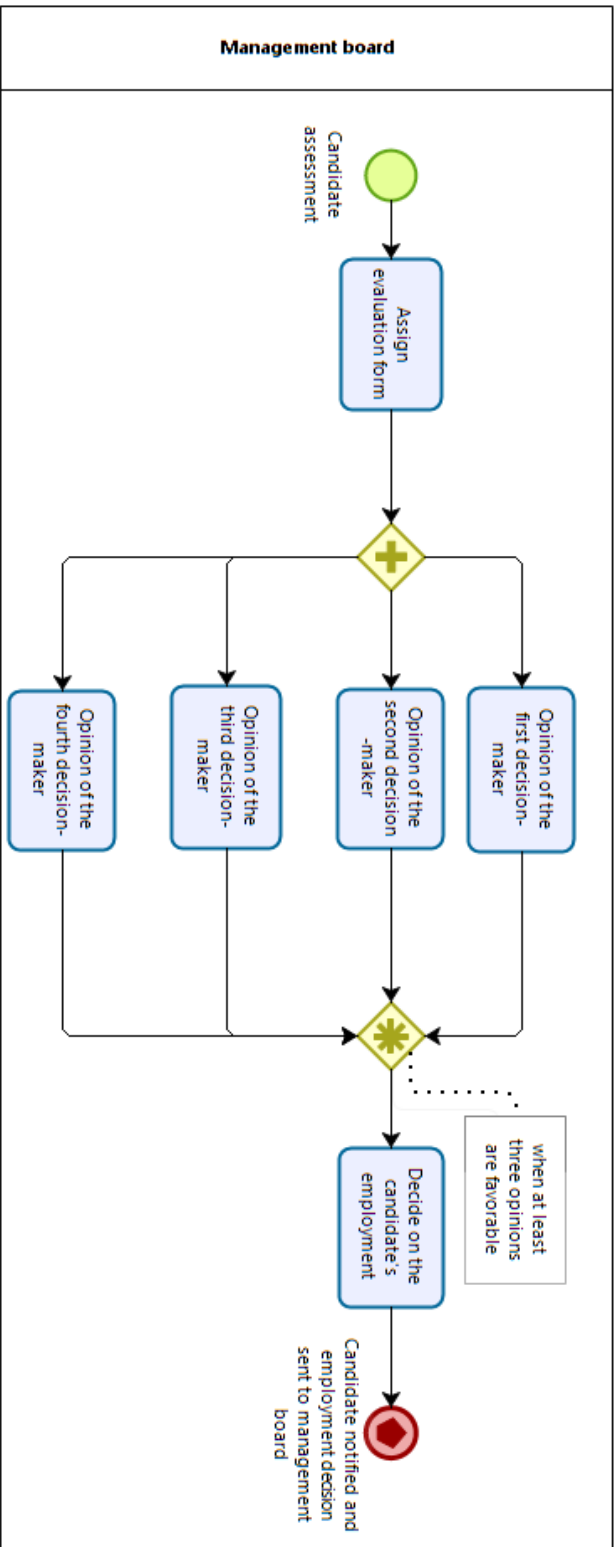


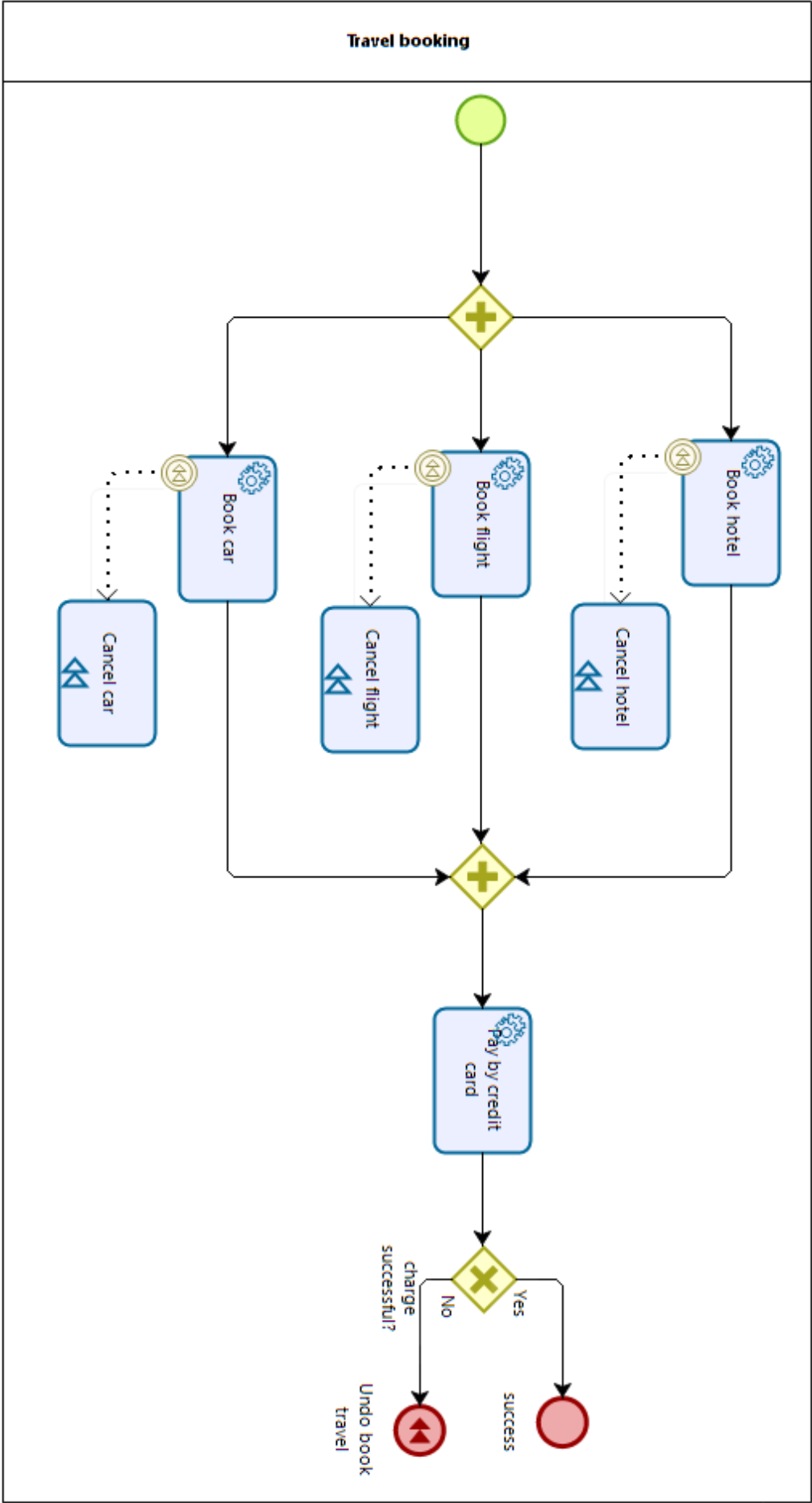




Prepare business trip







Appendix 2

Query Editor Home Ribbon Options	
Option	Description
Close & Apply	Finishes the processing steps; saves and closes the query.
New Source	Lets you discover and add a new data source to the existing query set.
Recent Sources	Lists all the recent data sources that you have used.
Enter Data	Lets you add your own specific data in a custom table.
Data Source Settings	Allows you to manage and edit settings for data sources that you have already connected to.
Manage Parameters	Lets you view and modify any parameters defined for this Power BI Desktop file.
Refresh Preview	Refreshes the preview data.
Properties	Displays the core query properties.
Advanced Editor	Displays the “M” language editor.
Manage	Lets you delete, duplicate, or reference a query.
Choose Columns	Lets you select the columns to retain from all the columns available in the source data.
Remove Columns	Lets you remove one or more columns.
Keep Rows	Keeps the specified number of rows at the top of the table.
Remove Rows	Removes a specified number of rows from the top of the data table.
Sort	Sorts the table using the selected column as the sort key.
Split Column	Splits a column into one or many columns at a specified delimiter or after a specified number of characters.
Group By	Groups the table using a specified set of columns and aggregates any numeric columns for this grouping.
Data Type	Applies the chosen data type to the column.
Use First Row as Headers	Uses the first row as the column titles.
Replace Values	Carries out a search-and-replace operation on the data in a column or columns. This only affects the complete data in a column.
Merge Queries	Joins a second query table to the current query results and either aggregates or adds data from the second to the first.
Append Queries	Adds the data from another query to the current query in the current Power BI Desktop file.
Combine Files	Adds the data from a series of similarly structured text files into a single table

Query Editor Transform Ribbon Options	
Option	Description
Group By	Groups the table using a specified set of columns; aggregates any numeric columns for this grouping.
Use First Row as Headers	Uses the first row as the column titles.
Transpose	Transforms the columns into rows and the rows into columns.
Reverse Rows	Displays the source data in reverse order, showing the final rows at the top of the window.
Count Rows	Counts the rows in the table and replaces the data with the row count.
Data Type	Applies the chosen data type to the column.
Detect Data Type	Detects the correct data type to apply to multiple columns.
Rename	Renames a column
Replace Values	Carries out a search-and-replace operation inside a column, replacing a specified value with another value.
Fill	Copies the data from cells above or below into empty cells in the column.
Pivot Column	Creates a new set of columns using the data in the selected column as the column titles.
Unpivot Columns	Takes the values in a set of columns and unpivots the data, creating new columns using the column headers as the descriptive elements.
Move	Moves a column.
Convert to List	Converts the contents of a column to a list. This can be used, for instance, as query parameters.
Split Column	Splits a column into one or many columns at a specified delimiter or after a specified number of characters.
Format	Modifies the text format of data in a column (uppercase, lowercase, capitalization) or removes trailing spaces.
Merge Columns	Takes the data from several columns and places it in a single column, adding an optional separator character.
Extract	Replaces the data in a column using a defined subset of the current data. You can specify a number of characters to keep from the start or end of the column, set a range of characters beginning at a specified character, or even list the number of characters in the column.
Parse	Creates an XML or JSON document from the contents of an element in a column.
Statistics	Returns the Sum, Average, Maximum, Minimum, Median, Standard Deviation, Count, or Distinct Value Count for all the values in the column.
Standard	Carries out a basic mathematical calculation (add, subtract, divide, multiply, integer-divide, or return the remainder) using a value that you specify applied to each cell in the column.
Scientific	Carries out a basic scientific calculation (square, cube, power of n, square root, exponent, logarithm, or factorial) for each cell in the column.

Query Editor Transform Ribbon Options (continued)	
Option	Description
Trigonometry	Carries out a basic trigonometric calculation (Sine, Cosine, Tangent, ArcSine, ArcCosine, or ArcTangent) using a value that you specify applied to each cell in the column.
Rounding	Rounds the values in the column either to the next integer (up or down) or to a specified factor.
Information	Replaces the value in the column with simple information: Is Odd, Is Even, or Positive/Negative.
Date	Isolates an element (day, month, year, etc.) from a date value in a column.
Time	Isolates an element (hour, minute, second, etc.) from a date/time or time value in a column.
Duration	Calculates the duration from a value that can be interpreted as a duration in days, hours, minutes, and so forth.
Expand	Adds the (identically structured) data from another query to the current query.
Aggregate	Calculates the sum or product of numeric columns from another query and adds the result to the current query.
Extract Values	Extracts the values of the contents of a column as a single text value.
Scripts	Runs scripts from languages such as “R” or Python.

Query Editor Add Column Ribbon Options	
Option	Description
Column From Examples	Lets you use one or more columns as examples to create a new column.
Custom Column	Adds a new column using a formula to create the column’s contents.
Invoke Custom Function	Applies an “M” language function to every row.
Conditional Column	Adds a new column that conditionally adds the values from the selected column.
Index Column	Adds a sequential number in a new column to uniquely identify each row.
Duplicate Column	Creates a copy of the current column.
Format	Modifies the text format of data in a new column (uppercase, lowercase, capitalization) or removes trailing spaces.
Merge Columns	Takes the data from several columns and places it in a single column, adding an optional separator character.
Extract	Creates a new column using a defined subset of the current data. You can specify a number of characters to keep from the start or end of the column, set a range of characters beginning at a specified character, or even list the number of characters in the column.
Parse	Creates a new column based on the XML or JSON in a column.
Statistics	Creates a new column that returns the Sum, Average, Maximum, Minimum, Median, Standard Deviation, Count, or Distinct Value Count for all the values in the column.
Standard	Creates a new column that returns a basic mathematical calculation (add, subtract, divide, multiply, integer-divide, or return the remainder) using a value that you specify applied to each cell in the column.
Scientific	Creates a new column that returns a basic scientific calculation (square, cube, power of n, square root, exponent, logarithm, or factorial) for each cell in the column.
Trigonometry	Creates a new column that returns a basic trigonometric calculation (Sine, Cosine, Tangent, ArcSine, ArcCosine, or ArcTangent) using a value that you specify applied to each cell in the column.
Rounding	Rounds the values in a new column either to the next integer (up or down) or to a specified factor.
Information	Replaces the value in the column with simple information: Is Odd, Is Even, or Positive/Negative.
Date	Isolates an element (day, month, year, etc.) from a date value in a new column.
Time	Isolates an element (hour, minute, second, etc.) from a date/time or time value in a new column.
Duration	Calculates the duration from a value that can be interpreted as a duration in days, hours, minutes, and seconds in a new column.

Query Editor View Ribbon Options	
Option	Description
Query Settings	Displays or hides the Query Settings pane at the right of the Power BI Desktop window. This includes the Applied Steps list.
Formula Bar	Shows or hides the formula bar containing the M language code for a transformation step.
Monospaced	Displays previews in a monospaced font.
Show whitespace	Displays whitespace and new line characters.
Column quality	Shows column quality characteristics.
Column distribution	Shows column distribution characteristics.
Column profile	Shows column profile characteristics.
Go to Column	Allows you to select a specific column.
Always allow	Allows parameterization in data source and transformation dialogs.
Advanced Editor	Displays the Advanced Editor dialog containing all the code for the steps in the query.
Query Dependencies	Displays the sequence of query links and dependencies.



UNIVERSITARIA

Tipar executat în

Tipografia

Universității Maritime din Constanța



978-606-681-180-4