

Universidade do Minho
Escola de Engenharia

Mariana Reimão Queiroga Valério de Carvalho

**Enhancing the Process of View Selection
in Data Cubes using What-If analysis**

março de 2019



Universidade do Minho
Escola de Engenharia

Mariana Reimão Queiroga Valério de Carvalho

Enhancing the Process of View Selection in Data Cubes using What-If analysis

Tese de Doutoramento em Informática

Trabalho efetuado sob a orientação do
Professor Doutor Orlando Manuel de Oliveira Belo

março de 2019

STATEMENT OF INTEGRITY

I hereby declare having conducted my thesis with integrity. I confirm that I have not used plagiarism or any form of falsification of results in the process of the thesis elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

University of Minho, March 29th 2019

Full name:

Marcia Pereira Queiroz Veloso de Carvalho

Signature:

Marcia Veloso de Carvalho

To André, my parents and my sister

Acknowledgements

The realization of this thesis would not have been possible without the support of several people.

First, I would like to express my gratitude to my supervisor, Dr. Orlando Belo, for all the support and guidance. For helping me whenever I needed to and made me grow on a scientific and professional level. Thank you for the advices and patience over these last years.

I wish to extend my deepest gratitude to André, who accompanied me throughout the development of this thesis and was relentless with me, always helping me when I came across difficulties, giving me advice and strength to never give up.

I would like to express my most sincere gratitude to my parents and sister, because without them, none of this would have been possible. Thank you so much for all the support and strength. I am very grateful to them.

Finally, I extend my thanks to all the people, friends and colleagues who, even if not mentioned here, have contributed to my personal and professional growth. Thank you!

Resumo

Melhorar o Processo de Seleção de Vistas em Cubos de Dados usando Análise What-If

Para competir na sociedade atual é necessário que os responsáveis de negócio consigam lidar com os desafios que o mercado lhes coloca no seu quotidiano. A elevada competição e o aumento na quantidade de informação eletrónica envolvida nestes processos implicam novos desafios relacionados com aquilo que designamos por processos de tomada de decisão. A recolha de informação relevante e o uso de ferramentas de Business Intelligence são dois fatores determinantes nos processos de tomada de decisão, e consequentemente na aquisição de vantagem competitiva das empresas. Apesar disto, recolher e armazenar informação relevante pode não ser suficiente. O processo de simular cenários hipotéticos de um negócio pode ser a vantagem que as empresas necessitam para sobreviver no mercado. As técnicas de análise What-If podem ajudar nesta vertente. O processo de análise What-If permite aos utilizadores criarem modelos de simulação para explorarem o comportamento de um dado sistema, analisando os efeitos causados pela alteração de um dado conjunto de variáveis que, usualmente, não podem ser descobertas através de um processo manual de análise de um qualquer conjunto de dados históricos, permitindo, assim, analisar as consequências dessas mesmas alterações. O sucesso de um processo de análise What-If depende crucialmente da experiência do utilizador, do seu conhecimento relativo à informação disponível e, obviamente do próprio processo What-If. Na ausência destes, podemos ter que encarar um processo de análise longo e difícil, especialmente na escolha dos parâmetros de entrada da análise. Nesta tese de doutoramento, é proposta uma metodologia híbrida, que integra preferências OLAP no processo convencional de análise What-If. Esta integração visa descobrir as melhores recomendações para a escolha dos parâmetros de entrada dos vários cenários de análise que considerem um conjunto de preferências OLAP, com vista a ajudar o utilizador a ultrapassar algumas das dificuldades que normalmente surgem durante um processo de análise What-If convencional. A metodologia desenvolvida ajuda a descobrir

informações mais específicas, orientadas e detalhadas, que não poderiam ser descobertas usando o processo de análise What-If convencional.

Palavras chave: Análise What-If, Business Intelligence, On-Line Analytical Processing, Preferences, Mineração de dados em sistemas OLAP, Bases de dados multidimensionais, Sistemas de suporte à Decisão, Métodos Formais, Alloy

Abstract

Enhancing the Process of View Selection in Data Cubes using What-If analysis

To compete in today's society, enterprise managers need to be able to deal with the arising challenges of the competitive market. The increasing competition and the amount of electronic information imply new challenges related to decision-making processes. Collecting relevant information and using Business Intelligence tools are determining factors in decision-making processes and in gaining competitive advantage.

However, gathering and storing relevant information may not be enough. The possibility of simulating business hypothetical scenarios could be the advantage that companies need. What-If analysis can help to achieve this competitive advantage.

What-If analysis allows to create simulation models to explore the behavior of a system, by analyzing the effects of changing values of parameters, which cannot otherwise be discovered by a manual analysis of historical data, and so, allowing the analysis of the consequences of those changes.

A successful What-If analysis process depends mainly on the user experience, his/her knowledge about the business information and the What-If analysis process itself. Otherwise, it can turn into a long and difficult process, especially in the choice of input parameters for the analysis.

In this doctoral thesis, a hybridization methodology is proposed that integrates OLAP preferences in the conventional process of What-If analysis. This integration aims to discover the best recommendations for the choice of input parameters for the analysis scenarios using OLAP preferences, helping the user to overcome the difficulties that normally arise in conventional What-If analysis process. The developed methodology helps to discover more specific, oriented and detailed information that could not be discovered using the conventional What-If analysis process.

Keywords: What-If Analysis, Business Intelligence, on-Line Analytical Processing, usage preferences, OLAP mining, Multidimensional Databases, Decision Support Systems, Formal Methods, Alloy

Table of Contents

Introduction	23
1.1 Context Overview	23
1.2 Motivation.....	25
1.3 Main Goals	27
1.4 Main Contributions	28
1.5 Thesis Structure	30
Related Work	33
2.1 What-If analysis	33
2.1.1 Main theory.....	33
2.1.2 What-If analysis improvements	34
2.1.3 What-If analysis applications – Business Intelligence	35
2.1.4 What-If analysis applications – GIS tools and others	37
2.1.5 What-If analysis applications – Contributions to existing tools	38
2.1.6 What-If analysis applications – Optimization Source Code Management Tools	39
2.1.7 Classification of the studies surveyed	42
2.2 OLAP Preferences.....	44
2.2.1 Personalization frameworks	45
2.2.2 OLAP Preferences theory and applications.....	46
2.2.3 Classification of the studies surveyed	47
What-If analysis: process and discussion	49
3.1 What-If Analysis Process.....	49

3.1.1	The 'What-If Question'	49
3.1.2	The Simulation Model	50
3.2	A What-If Case Example	53
3.3	Methodologies using What-If.....	55
3.4	Summary	58
OLAP Preferences.....		59
4.1	Contextualization of OLAP preferences.....	59
4.2	OLAP Personalization	64
4.2.1	Personalization in OLAP systems.....	66
4.2.2	A formal definition of Preference	69
4.2.3	Extracting OLAP Preferences using OLAP mining.....	70
4.3	OLAP Preferences Extraction Examples	74
4.4	Summary	76
A Hybridization Process Based on Preferences.....		77
5.1	Methods and Methodologies.....	77
5.2	Integrating OLAP Preferences	78
5.3	Overview of the Hybridization Process	81
5.4	The Methodology	83
5.5	A Formal Specification	85
5.6	Describing the Key Phases	86
5.6.1	Selecting Views in the Data Warehouse.....	87
5.6.2	Construction of the OLAP cube	89
5.6.3	Extraction of Association Rules of the OLAP cube.....	92
5.6.4	Extracting Usage Preferences	97
5.6.5	Using Preferences on What-If Analysis	102
5.7	Formal Validation of the Hybridization Process	104
5.8	Summary	109
A Case Study		111
6.1	Analysing Data	111
6.2	A Software Platform for Receiving the Methodology.....	115
6.3	Example Case Study	116

6.3.1	Conventional What-If analysis	116
6.3.2	Itemsets and Association Rules	119
6.3.3	The Hybridization Process	121
6.4	Comparative Analysis.....	126
6.4.1	Conventional What-If analysis Results.....	126
6.4.2	Hybridization Process Results	127
6.4.3	Conclusions	130
Conclusions and Future work.....		131
7.1	Final Remarks	131
7.2	Lessons Learned and Knowledge Acquired	134
7.3	Future Work.....	137
References		139

List of Figures

Figure 1 – The importance of a What-If question.....	25
Figure 2 – Representing a simulation model in What-If analysis.....	51
Figure 3 – Relating historical and prediction scenarios.....	52
Figure 4 - Performing What-If analysis - Example of a PivotTable scenario.	54
Figure 5 - Performing What-If analysis - Change of variables' values.....	54
Figure 6 - Performing What-If analysis - Accepting changes.	55
Figure 7 - The lattice of a data cube.	60
Figure 8 - Example of a multidimensional schema.	61
Figure 9 - Roll-up operation over a data cube.	62
Figure 10 - Drill-down operation over a data cube.	63
Figure 11 - A strategical issue: using preferences obtained with OLAP mining.	80
Figure 12 – The hybridization process.	81
Figure 13 – A general overview of the simulation model.....	82
Figure 14 - A general overview of a What-If analysis process.	82
Figure 15 - Methodology for the hybridization process.	83
Figure 16 - An example of a star schema.	88
Figure 17 – Alloy specification: Definition of Table (Fact Table and Dimension) and Field (Measure and Attribute).....	89
Figure 18 – Construction schema of an OLAP cube.	89
Figure 19 - Concepts of dimensions and data cells in a multidimensional structure.	90
Figure 20 – Alloy specification: Definition of Cube parameters, OLAP Cube and predicate ConstructCube	90
Figure 21 - Representation of a data warehouse's view.	91
Figure 22 - Example of a multidimensional database – a cube.	92

Figure 23 - Alloy specification: Definition of Mining structure, Mining model, Rule, Performance and SubsetFields	96
Figure 24 - Alloy specification: Predicate ConstructRules	97
Figure 25 - The extraction process of OLAP preferences.	98
Figure 26 - Filtering association rules.	98
Figure 27 - Alloy specification: Definition of PrefParams	100
Figure 28 - Alloy specification: Predicate ConstructStrongRules	100
Figure 29 - Alloy specification: Definition of Preference.	100
Figure 30 - Alloy specification: Predicate ConstructPrefs.	100
Figure 31 - Example of filtering association rules.	101
Figure 32 - Overview of the What-If analysis process.	102
Figure 33 - Overview of the simulation model.	103
Figure 34 - Overview of the hybridization process.	104
Figure 35 - Alloy specification: Predicate RulesCorrect.	105
Figure 36 - Alloy specification: Assertion CheckRulesBad.	105
Figure 37 - Alloy specification: Check of the assertion CheckRulesBad.	105
Figure 38 - Alloy specification: Predicate GoodCubeParams.	105
Figure 39 - Alloy specification: Assertion CheckRulesGood.	106
Figure 40 - Alloy specification: Predicate StrongRulesCorrect.	106
Figure 41 - Alloy specification: Assertion CheckStrongRulesBad.	106
Figure 42 - Counter-example for <i>StrongRulesCorrect</i> found by the Alloy Analyzer.	107
Figure 43 - Alloy specification: Predicate GoodPrefParams.	107
Figure 44 - Alloy specification: Assertion CheckStrongRulesGood.	108
Figure 45 - Alloy specification: Run command to generate a valid instance.	108
Figure 46 - An example of a consistent instance obtained by the Alloy Analyzer.	108
Figure 47 - Selected data warehouse's view - "Sales" schema.	112
Figure 48 - Example 1 of a view of the schema "Sales".	112
Figure 49 - Example 2 of a view of the schema "Sales".	113
Figure 50 - Extracted Itemsets.	113
Figure 51 - Extracted Association Rules.	115
Figure 52 - Overview of the software platform UI - WIF tab.	117
Figure 53 - WIF tab - Choice of the set of business variables to perform the analysis.	118
Figure 54 - WIF tab - Change variables' values.	118

Figure 55 - WIF tab - Prediction scenario.	118
Figure 56 - Overview of the application UI - MiningStructure tab.	119
Figure 57 - MiningStructure tab - explore the "Products" mining structure.	119
Figure 58 - Itemsets of the Products' mining structure.	120
Figure 59 - The association rules of the Customers' mining structure.	120
Figure 60 - Overview of the application UI - HybridizationModel tab.	122
Figure 61 - HybridizationModel tab - Choice of the mining structure.	122
Figure 62 - HybridizationModel tab - Choice of the goal analysis business variable.	122
Figure 63 - HybridizationModel tab - Change support value to 500 and confidence value to 80%.	122
Figure 64 - Selection of the Top association rules.	123
Figure 65 - Recommendations made to the user.	124
Figure 66 - HybridizationModel tab - changing the variables' values.	124
Figure 67 - HybridizationModel tab - Prediction scenario.	125
Figure 68 - Conventional What-If Analysis - Historical scenario.	126
Figure 69 - Conventional What-If Analysis - The prediction scenario.	127
Figure 70 - Hybridization process - Historical scenario.	128
Figure 71 - Hybridization process - Prediction scenario.	128

List of Tables

Table 1 - Classification of the studies surveyed – What-If analysis43

Table 2 - Classification of the studies surveyed - What-If analysis (continued).....44

Table 3 - Classification of the studies surveyed – OLAP preferences48

Table 4 - Example of 1-itemset.....114

Table 5 - Example of 2-itemsets.114

Table 6 - Example of 3-itemsets.114

Chapter 1

Introduction

1.1 Context Overview

The incredible growth in the gathering of electronic data and the increasing competitiveness in business environments are important factors to consider in any knowledge-based society. Companies need to make better use of analytical information systems, techniques and models for data exploration and analysis in order to try to gain competitive advantages from a better use of the knowledge they own. An increasing number of companies has been having the need to obtain relevant information using tools and business data, in order to reduce redundant information, increase profits and save time, reducing waste and optimizing decisions. Such as, there has been a noticeable increase in the number and quality of data retrieving and handling processes created, developed or used by companies.

Business Intelligence (BI) has been assuming a leading role, and is one of the most important tools responsible for companies' development in data support systems. In general terms, we may say that BI consists of the entire process of transformation of information, which includes a wide range of applications, practices, and technologies for the analysis, extraction, integration, loading and presentation of data for supporting decision making. More precisely, it refers to a set of tools and

techniques that allows for a company to transform its business data into knowledge that will be useful for supporting its decision-making processes. BI is used mainly for presenting business data, helping the decision makers for obtaining the information they need to make daily business decisions. Usually, BI uses multidimensional data structures or data cubes for storing information (Fouché and Langit, 2011).

On-line analytical processing (OLAP) is one of the most interesting working area in BI arena and in decision-support systems. Navigating in a multidimensional data using OLAP operators allows for exploring and analyzing data from cubes (Gray et al., 1997). OLAP tools provide a multidimensional view over business data and means for business analytics, which is a very efficient logical way for analyzing businesses activities. A decision-support analysis process is an interactive exploration of multidimensional databases, often performed in *ad hoc* manner that allows for users to see data from different perspectives of analysis. OLAP has taken a leading role within BI. It provides excellent analysis methods and tool for supporting decision processes, although it is not capable of anticipating future trends. What-If analysis (Golfarelli et al., 2006) technology helps to fill this gap. Due to its characteristics, an OLAP cube is probably the most adequate data structure for supporting a What-If simulation scenario. It is a well-known data structure for supporting information analysis, being capable for representing historical trends and supporting information at different abstraction levels.

What-If analysis allows for changing the values of some variables with the goal for analyzing how those changes will affect other variables. What-If analysis can be described as a data simulation whose goal is to inspect the behavior of a complex system under some given hypothesis. The user starts by choosing a set of variables that he wants to change, and the What-If process uses a simulation model for calculating the effects of that change, which usually cannot otherwise be discovered by a historical data manual analysis process (Koutsoukis et al., 1999). In a real business system, creating a simulation model through What-If analysis enables the user to implement changes in characteristics of the business, to test hypothesis and to analyze its consequences without endangering the business. In other words, What-If analysis helps decision makers to assess beforehand what can happen in complex systems as result of changing what can be consider a normal business behavior. Decision makers can use What-If analysis scenarios to test and validate business hypothesis to support their decisions and make decisions according to the results without risking business and avoiding possible risks. What-If analysis allows for decision

makers to manipulate parameters for building hypothetical scenarios and for analyzing them to make better decisions, testing and validating hypothesis and supporting their decisions. What-If analysis can be the safer methodology for the decision maker to tackle some doubt or issue, in order to ensure, if possible, that subsequent decisions will be successful. Moreover, it allows for analyzing different scenarios and perspectives of business, anticipating some solutions - these concepts will be detailed in chapter 3.

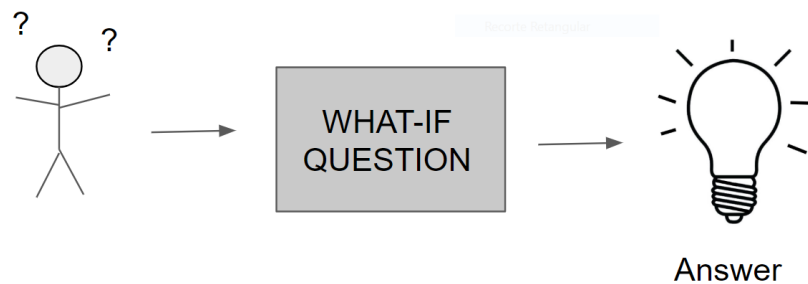


Figure 1 – The importance of a What-If question.

Decision makers usually resort to What-If analysis when doubts about business decisions arise (Figure 1). Uncertainties can be translated into a What-If question and What-If analysis outcome is used for answering a What-If question. For example, a example of a What-If question can be "What-if we want to increase the overall profit in 2014, 2015 and 2016 by 10% of the products with black, blue, red and white colors?" - this example of What-If analysis that involves a typical application scenario is presented in chapter 3. One possible simulation scenario is to analyze the profit during the referred years and increasing the profit values of the product X by 10%. With this scenario, one can analyze if the increasing price can influence the profit of the entire company.

1.2 Motivation

The lack of expertise of a user during a What-If design and implementation solution is one of the pitfalls of a What-If analysis process. A user who is not familiar with the What-If process, or even the business data, may not choose the most correct parameters in an application scenario, leading

to poor results and inadequate outcomes. Thus, the main research questions we addressed in this thesis were the following:

- Can we optimize the decision process in data cubes and especially for improving What-If scenarios using prediction models?
- Can we improve a view selection process (and consequently query time response and memory consumption) by restructuring automatically What-If scenarios?
- Can we get more oriented results by integrating prediction models into a conventional What-If analysis process?

There are a few papers in the literature that present different tools and techniques for improving What-If analysis processes. We aim at improving What-If scenarios using prediction models, optimizing the decision process, for providing appropriate views to the user. Nevertheless, it is necessary to define a general methodology, and to develop a hybridization model with the ability for combining prediction models and What-If analysis models and techniques. A hybridization model is an implementation of the methodology determined by appropriate choices of tools and techniques. We claim that combining OLAP preferences with conventional What-If analysis models and techniques may enhance significantly the conventional What-If analysis. OLAP preferences are widely used as recommendations, and we chose them for this work. This combination will certainly help inexperienced users, and may even be helpful for experienced users.

Working as a recommendation system, OLAP preferences help to filter data during the What-If process, giving a more oriented outcome to the user. In OLAP platforms, when performing complex queries, it is likely that the outcome will be a huge volume of data that may be quite difficult to analyze. With OLAP usage preferences, it is possible to filter the volume of data we have to deal. The returned data is adjusted to the users' needs and to the business requirements without losing data quality. The extraction of OLAP usage preferences according to each analytic session promoted by a user may come as an advantage to decision-makers, since it provides a very effective way to personalize the outcome of queries of analytical sessions and multidimensional data structures, acting as a useful and effective decision-making support. Additionally, OLAP preferences can recommend axis of analysis that are strongly related to each other, introducing helpful and useful information to the application scenario under construction. OLAP preferences are discussed at length in chapter 4.

1.3 Main Goals

In this thesis, we propose and discuss a hybridization methodology that integrates OLAP usage preferences in conventional What-If scenarios applications. The hybridization process can suggest OLAP preferences, providing the user with the most adequate scenario parameters according to the needs and making What-If scenarios more valuable. This methodology helps the user by suggesting new axes of analysis to the What-If analysis scenario. These new axes of analysis are discovered through OLAP mining. They cannot be discovered using a manual analysis. These axes happen to be strongly related to the goal analysis, pre-defined by the user in the What-If question. In the end, this integration helps the user by adding new relevant information to the What-If scenario.

During this thesis the What-If analysis process will be presented in detail. In addition, an in-depth study of preferences, having a special attention to OLAP preferences, will be conducted for helping choose the most adequate mining technique for the recommendation process to be included in the What-If analysis. An application of a What-If analysis with the integration of OLAP preferences will be conducted, to better assess the merits of the hybrid methodology we propose here.

The proposed hybridization process, which integrates prediction models and What-If analysis, was developed mainly for helping the enhancement of conventional What-If analysis, and it will help to improve the following points:

- Optimize the decision-making process using What-If analysis in OLAP environments: the hybridization process, which integrates What-If analysis and OLAP preferences, aims essentially to improve the decision-making process using the conventional What-If analysis.
- Provide automatic restructuring of What-If scenarios: With a recommendation engine, the experience of the What-If analysis gets easier. The user can choose from the recommended set of input parameters and ends out with more refined and oriented scenarios.

- Improve query time response and memory consumption when dealing with very large databases: Using OLAP preferences as the basis of the hybridization methodology, we can filter the relevant information for the user, by suggesting recommendations for the input parameters of the analysis scenario, and proposing a new structure for data cubes using a set of selected views based on user preferences. With this filter process, we can decrease memory consumption (with less information to process) and consequently decrease query time response.

- Reduce processing and data materialization time: as the data is already materialized, the user does not need to materialize the data each time he wants to create an analysis scenario. With this optimization of the conventional What-If analysis process, the processing and data materialization time are significantly reduced, due to the fact that the system has the ability to propose a new set of data cube views based on user preferences.

- Prefetching data systems and caching: With this hybridization process, before creating the hypothetical scenarios using the What-If analysis process, a set of data cube views is suggested to the user to be added in the scenario. This set of views (input parameters) is defined using the OLAP preferences extracted from the OLAP cube.

- Quality of results more effective, oriented and less dispersed: By integrating OLAP preferences in the conventional What-If analysis, we aim at discovering the best recommendations for the analysis scenario, helping the user during the What-If analysis process. Using OLAP preferences, we get a more oriented outcome. The outcome data is filtered, avoiding huge amounts of information that it is not useful for the user.

1.4 Main Contributions

The main contributions of this thesis are divided in three principle topics:

1) Conventional What-If analysis process.

An extended study analysis of the conventional What-If analysis was made. This study analysis led to the discovery of some drawbacks in the process. Despite the many advantages and usefulness of the What-If analysis, the lack of experience of the user is

one of the pitfalls of this process and may lead to poor results and low-quality decisions. This work (section 3.1 and 3.3 of Chapter 3) was published in:

- Carvalho, M., Belo, O., (2011). "Exploração de Cenários What-If em Plataformas de Processamento Analítico de Dados", CAPSI'2012, Actas da CAPSI'2012 - 12^a Conferência da Associação Portuguesa de Sistemas de Informação, Guimarães, Portugal, 7 Setembro.

2) The Hybridization process.

To overcome the pitfalls of the conventional What-If analysis, a hybridization model is proposed. This hybridization model consists of integrating OLAP usage preferences in conventional What-If scenarios applications. This work (Chapter 5) was published in:

- Carvalho, M., Belo, O., (2016). "Enriching What-If Scenarios With OLAP Usage Preferences", In Proceedings of The 8th International Conference on Knowledge Discovery and Information Retrieval (KDIR'2016), Porto, Portugal, November 9-11.
- Carvalho, M., Belo, O., (2017a). "Conceiving Hybrid What-If Scenarios Based on Usage Preferences", In Proceedings of EWG-DSS 2017 International Conference on Decision Support System Technology (ICDSST' 2017), Namur, Belgium, May 29–31.

3) Formal specification and validation of the hybridization process.

To verify if the proposed hybridization model meets its critical requirements and provides the desired functionality, it is imperative to use formal methods. With the formal specification and verification, we can create an abstract representation of the hybridization model process in order to detect possible inconsistencies of the hybridization model.

This work (sections 5.4 and 5.5 of Chapter 5) was published in:

-
- Carvalho, M., Belo, O., (2017b). "Using Alloy for Verifying the Integration of OLAP Preferences in a Hybrid What-If Scenario Application", In Proceedings of 9th International KES-IDT Conference (KES-IDT'2017), Vilamoura, Algarve, Portugal, June 21–23.
 - Carvalho, M., Belo, O., (2017c). "Inception and Specification of What-If Scenarios Using OLAP Usage Preferences", In Proceedings of The 11th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO'2017), Leon, Spain, 6-8 September.
 - Carvalho, M., Macedo, N., Belo, O., (2017d). "Checking the Correctness of What-If Scenarios", In the 11th IFIP WG 8.9 Working Conference – CONFENIS 2017, Crowne Plaza Shanghai Fudan, Shanghai, China, October 18th - 20th.

1.5 Thesis Structure

Apart from this chapter, this thesis is structured and organized as follows:

- Chapter 2 – In this chapter a literature review about What-If analysis, OLAP preferences and their application is presented. Concerning What-If analysis, we overview some research papers that explain how this process is used in several areas of application. Literature about OLAP preferences is also presented. We overview how OLAP preferences have been used, and their importance.
- Chapter 3 – This chapter contains a detailed explanation of the What-If analysis process and its components, followed by a case example that demonstrates how What-If analysis works. After presenting the example, we describe in more details some methodologies that use What-If analysis, and discuss them at length, addressing some of the drawbacks that justify the development of a new methodology to improve the traditional What-If analysis.
- Chapter 4 – Our methodology involves the combination of What-If analysis and OLAP preferences. We implemented our methodology using OLAP preferences, which are reviewed in this chapter. We briefly refer to OLAP Personalization, and how personalization

has evolved in the past years. A formal definition of preference and some techniques for extracting preferences in OLAP environments are also presented.

- Chapter 5 – Following the choices made to implement our methodology, whose main components are addressed in the previous chapters, a methodology of the hybridization process is presented. The chapter starts with a reference to the choices made to implement the methodology, and then explains the advantages of the integration. This is followed by a description of the proposed methodology and a detailed explanation of key phases of the integration process. In the description of each phase, a formal specification and validation are presented, to guarantee that the hybridization process is free of failures and inconsistencies.
- Chapter 6 –The hybridization process is illustrated using a software platform to address a case study. The example analysis is used to show that the hybridization process outperforms the traditional What-If analysis, due to the fact that relevant information that otherwise could not be obtained becomes available. We show how the What-If scenarios are created and enhanced using the extracted OLAP preferences, describing all the steps between the extraction of the association rules until the definition of the What-If scenario.
- Chapter 7 – In this chapter, the results achieved with the presented hybridization process are summed up, some conclusions are drawn, and finally some possible future lines for future researching and applications are suggested.

Chapter 2

Related Work

2.1 What-If analysis

2.1.1 Main theory

What-If analysis evolution over the last decade is remarkable, as evidenced by several papers that were published during this period. Golfarelli et al. (2006) are a landmark in What-If analysis. The authors present What-If analysis as a solution methodology for the resolution of problems in a BI context. They presented an analysis and a discussion of some of the lessons learned and experience obtained after using What-If projects in real business processes, where they found immature technology, complexity of design and lack of design methodology. They also suggest several tools that present What-If features that help to ease the users' problems. In the following years, there is a group of research papers concerning the What-If process. Most papers present studies about the behavior of this process in different environments. Let see some interesting and pertinent cases.

Kottemann et al. (2009) addressed unaided decision support systems and decision support systems aided with What-If analysis. The authors presented a formal simulation approach, comparing unaided and aided decision-making performance. They verified that the performance differences

between the two cases are significant, concluding that the effectiveness of the decision making strategies is dependent on the environmental factors and on the supporting tools; and though What-If analysis is very helpful and popular in decision making, it is not 100% effective. Being dependent on human hand, it is very easy to alter or diverge from the optimal outcome for decision making. This is where simulation helps in overcoming this problem in any circumstance. In the same year, Zhou and Chen (2009) addressed What-If analysis in Multidimensional OLAP environments. Special attention was paid to storage and organization of the hypothetical modified data, when dealing with What-If analysis, because some cells of the data cube may be modified. The solution proposed by Zhou and Chen (2009) consists in storing the new hypothetical modified data into a HU-Tree data structure (variant of r^* -tree). This allows for storing and managing hypothetical modified cells using the hypothetical cube, instead of modifying the original cube directly. When a What-If analysis is processed, the original cube and the What-If cube are manipulated simultaneously. In the next year and following their previous work, Golfarelli and Rizzi (2010) focuses on the resolution of a particular problem of a real case study using the What-If methodology they proposed before. In this paper, they mainly focused on getting a precise formalism for expressing conceptually the simulation model. They achieve a simulation model that satisfies several issues; for instance, with their methodology, they can model static, functional and dynamic aspects in an integrated fashion, combining use cases, class and activity diagrams, build specific What-If constructs using the UML stereotyping mechanism, and get multiple levels of abstraction using YAM2.

2.1.2 What-If analysis improvements

Gavanelli et al. (2012) suggested improving the traditional What-If analysis process, typically based on a “generate and test” paradigm, by integrating a combinatorial optimization and decision-making component, which helps in enriching and identifying the most interesting What-If scenarios, which are then used when performing the simulation. The authors apply their methodology in social policy making. In the next year, Xu et al. (2013) presented a specialized work that mainly focused on improving the performance of What-If query processing strategies for Big Data in an OLAP system. They aimed to improve the classical delta-table merge algorithm in the process of What-If, taking advantage from the MapReduce framework. Also, they explain a What-If algorithm of BloomFilterDM (Bloom filter-based delta table merging algorithm) and What-If algorithm of DistributedCacheDM (distributed cache-based delta table merging

algorithm). Later, Hung et al. (2017) presents a work that aims to address the What-If analysis process when there are conflicting goals, *i.e.*, multiple goals that are contradictory between each other. The authors propose the use of data ranges for the input scenario parameters in the What-If simulation, to limit the number of scenarios explored. They present several ways of optimizing the input parameters to get a What-If analysis outcomes that balances the conflicting goals.

2.1.3 What-If analysis applications – Business Intelligence

An increased use of What-If techniques in several areas has been noticed. What-If analysis has been widely used in several areas, and it proved to be a useful technique for BI, like data warehouses, relational databases and in OLAP cubes. In the following we review some papers, including some of the earlier contributions to What-If analysis, published long before the work of Golfarelli et al. (2006). In 1998, Chaudhuri and Narasayya (1998) applied What-If analysis for exploring the possibility of assessing the impact of an index on the performance of a system using a What-If analysis for determining the usefulness of creating a given index, asking questions like "Which queries implemented in the last days will be slower in the future because of performed modifications?". Ten years later, they presented other developments of their work in Chaudhuri and Narasayya (2007).

Going back a little bit in the decision modelling field, we found the work of Koutsoukis et al. (1999), emphasizing the importance of OLAP for decision modelling. The authors described the tasks of a Decision Support System, like the data and symbolic modeling and the What if analysis phases. They proposed a paradigm for analyzing data, applying Decision Support System tools and show how the information chain impacts on decision making. They also analyze the impact of operations of aggregation (roll-up) and breakdown (drill-down) models' typical decision support in decision models. Almost ten years later, Papastefanatos et al. (2008) presented a tool, called Hecataeus, which uses What-If analysis for changes in data warehouse. They focused mainly on the changes that occur in data warehouse schemas and how they affect dependent applications and data stores. The Hecataeus tool aims for representing a database schema and helping the user to create hypothetical scenario evolution when a database schema occurs, and to analyse its effects. Also in 2008, Jouini and Jomier (2008) addressed the problem of efficiently indexing data in a Multiversion Data Warehouse (MVDW), which is a framework that allows for the user to separate and identify various versions of a data warehouse, corresponding to different time

periods; running queries on one or several versions of the data warehouse, creating and managing hypothetical virtual scenarios using What-If analysis. They aimed to take a MVDW and analyze data sources using BI applications and study the evolution of the analysis context and versioning facts and dimensions, for analyzing the impact of alternative hypothetical scenarios using What-If analysis.

In 2009, Xiao et al. (2009a) analyzed the problem of multiple versions of What-If views (or scenarios). More specifically, the problem that arise from various versions of data processing and incremental computation, and propose a strategy to solve to process those. This strategy consists in storing What-If views in a delta table and computing the delta data cube. They also proposed incremental computation data cube for MAX and MIN aggregation functions. This latter proposition allows for reducing costs and accessing times of the base table, and consequently the incremental computation of the MAX and MIN aggregation functions. In the same year, Xiao et al (2009b) works on additional studies in this area, proposing a solution for incremental computation data cube for MEDIAN function on What-If analysis. Four years later, Deutch et al (2013) described the Caravan system, which was developed for performing What-If analysis. With this system it is possible to users to get a personalized session, oriented to their needs, displaying only relevant data and exploring different answers within computed views. The novelty of this system is the use of Provisioned Autonomous Representations (PARs) to maintain the necessary information of the What-If scenarios instead of preserving the entire source database. In the same year, Saxena et al. (2013) aimed to use in-memory What-If analysis using a query system to introduce new values - this was considered an extension of Balmin et al. (2000) and Li et al. (2004). They aimed essentially maintain intact the real data cube, not changing it, by introducing new values for dimensions and measures and storing them as scenarios. Whereas, Balmin et al. (2000) concerned about What-If queries in OLAP platforms. These authors focused mainly on a spreadsheets' lack of data integration and ad-hoc OLAP tools' lack of flexibility and performance, and how to deal with these problems. Balmin et al. (2000) proposed the SESAME system for modeling hypothetical scenarios as a list of modifications on views of data warehouses and factual data, providing both a concrete syntax and semantics to describe their scenarios, and analyze the behavior of queries What-If in environments analytical processing of data. They also define a formal syntax and semantics for hypothetical scenarios. More recently, Hartmann et al., (2018) focuses on predictive analytics, also known as What-If analysis. The authors focused mainly in extract temporal models from current and past historical facts with the intention of creating predictions of the future. Their

intent was to solve the problems inherent to predictive analytics, like the complexity and the diversity of the data models, using novel data model to support large scale What-If analysis on time-evolving graphs, called *Many-World Graph*.

2.1.4 What-If analysis applications – GIS tools and others

Some authors that develop Geographic Information Systems (GIS)-based and other type of traffic management tools also resort to What-If analysis for improving their work. Tsunokawa et al. (2006) study concerns about roads' problems. The authors developed a system to find the best schedule for road maintenance. The authors aim specially to overcome the difficulties that arise from the use of certain What-If models in this task, such as explore infinite number of options. To accomplish this, the authors used a What-If model, called highway development and management tool (HDM-4), among optimization algorithms to find the best maintenance options. Six years later, Moreno et al (2012) proposed a GIS-based simulation tool. The proposed system was conceived for helping individual brigades and fire fighters. The system aims to give them relevant information, so they can prompt a rapid response with minimal damage. The authors also used What-If analysis in this solution, providing to users a simulation of hypothetical scenarios so decision makers can analyze the impact of different actions on a virtual scenario. Thus, users can explore with the outcome of the scenarios before they go into action and avoid possible hazards. In the following year, Golfarelli et al. (2013) and Wickramasuriya et al. (2013) also focused in the domains of GIS and BI. The former authors proposed a Geo-BI solution, called Lily, to overcome the difficulty of users when handling with spatial data. Lily provides an interactive interface integrating BI and geospatial dimensions, and also some analytical features like What-If analysis and data mining. The analytics features have the purpose of enhancing the capability of the system for discovering patterns, simulating scenarios and exploring some hypothetical questions using territorial information. In the same year, Wickramasuriya et al (2013) reinforced the research efforts in BI and GIS. These authors proposed an integration of BI and GIS, offering a solution called SMART Infrastructure Dashboard. This solution aimed to perform spatio-temporal analysis, allowing for the user to discover relationships between utility usage, demographics and weather patterns in regions. This solution also allows the user to perform What-If analysis and explore future planning. The authors consider What-If analysis an important part of the solution, because it allows for analyzing the effects of changes in variables, like expected utility; and then provide anticipation of future trends. Two years later, Asharani et al. (2015) presented a work that

concerns about pattern classification systems and their weakness in classifier security. They showed one of many issues related to this subject, evaluating the performance degradation towards potential attacks, and proposed a framework for classifier security evaluation. What-If analysis was used to simulate a set of hypothetical potential attack scenarios that could occur during the operation and analyze the impact of the effects and identify the most critical weaknesses.

More recently in 2017, Timar et al. (2017) was presented a What-If Analysis tool for planning airport traffic. The authors presented an implementation and application of a prototype What-If Analysis decision support tool for airport traffic planning, focusing mainly on the start and end times of a Department Management Programs for airport-wide departures, and evaluating the resulting traffic performance. They used What-If analysis to predict the airport traffic performance and to design a DMP to ease negative impacts. The What-If Analysis tool was used to predict airport traffic performance during a future time horizon with forecast operating conditions and to design and decrease the negative impacts of predicted demand and capacity imbalances. The authors also demonstrated how the prototype works using a historical traffic and weather scenario at Charlotte Douglas International Airport.

2.1.5 What-If analysis applications – Contributions to existing tools

In 2010, What-If analysis starts to be used in other areas with Brenner et al (2010). In this work was presented a simulation study conducted in the emergency department at the University of the Kentucky Chandler Hospital. The authors had the need of study the normal behavior of the emergency department and find a way to improve the process. They decided to run a simulation using What-If analysis to achieve this goal by predicting the impact of specific changes in the process. With What-If analysis is possible compare various scenarios, study when and in what stations the bottlenecks happen and discover how many human and equipment resources are needed. Four years later, Krishnamoorthy et al. (2014) reflected about manufacturing processes and integration with What-If analysis. The authors talk mainly about manufacturing processes that include physical or virtual inventories. There was the need to analyze very careful to calculate the optimal operational settings. The authors proposed a framework, called temporal Manufacturing Query Language (tMQL), that aims to allow for the composition and manipulation of process

models and perform What-If analysis. In this context, What-If analysis is used to determine metric optimization and computation queries.

In 2016, Rome et al. (2016) developed a work that concerned mostly on the management of crisis situations and how What-If analysis can improve crisis situations. The authors presented a tool, called 'CIPRTrainer', which allows for simulating and exploring different application scenarios. In this work, What-If analysis was used to explore and analyze a possible and different course of action in a crisis situation, mainly for training purposes. The authors also explained how the crisis models are created, the level of detail to be realized and how the user can handle with missing data. Still in 2016, Rozema (2016) showed how an independent IT service provider, called 'ShipitSmarter', takes advantage using What-If analysis. The main aim was to provide a data analysis tool to analyze carriers' performance to help customers and to create hypothetical scenarios using What-If analysis in the ShipitSmarter's processes. What-If analysis was specially used to create scenarios of switching carriers or using a different service level and analyze its effects. In the following year, Van Cauwelaert et al. (2017) presented a web tool that allows for What-If analysis on performance profiles. This tool helps to build and export Performance profiles, which is a great addition to the Operational Research community. This approach uses What-If analysis for helping to simulate the effect of the computation time.

2.1.6 What-If analysis applications – Optimization Source Code

Management Tools

As seen before, most of the papers found in the literature about What-If analysis centers mainly in how the integration of a What-If analysis process could improve an existing tool. The following papers focus mainly on optimizing source code management related tools using What-If analysis. For instance, Van den Akker et al (2008) presented an optimization tool for helping software salespersons to determine when is the best time to release a new version of a software product. The authors used integer linear programming in the tool and chose several parameters to influence the outcome. The goal was to find the best outcome (or date) that results in the higher revenue giving a certain period of time. The authors also integrated some mechanisms that integrates What-If analysis in the tool – in this case, What-If analysis helps to explore how changing

parameters can influence the outcome scenario. Another interesting work was presented by Zarras and Vassiliadis (2008) concerning Business Process Execution Language (BPEL) and their reliability analysis. The authors aimed to provide methods that can handle What-If analysis to BPEL, in order to predict the risk of failure during its execution, which can be very useful when dealing with systems with several execution paths. Additionally, What-If analysis helps to analyse the consequences and behaviour of each one of the execution paths. In the next year, Datta and Roy (2010) presented another interesting work focused on cost estimation models. These authors aimed to improve cost estimation models of service contracts. They intended to identify areas that can be improved using What-If analysis and study how different cost estimation techniques, life cycle costing, maintenance and service cost can influence these models. With What-If analysis, these authors created multiple scenarios, with specific parameters, like usage, and explore its effects. Three years later, Bird and Zimmerman (2012) explored the use of branches in Source Code Management Systems and how developers deal with large industrial projects when face some difficulties with the branches. The authors introduced What-If analysis for assessing isolation and liveness of branches, emphasizing that the integration of What-If analysis was one of the main contributions of the project. Additionally, they also showed how What-If analysis can support branches decision by creating hypothetical scenarios and analyzed its behavior, especially in terms of isolation and liveness of the branches.

Another very interesting research initiative was done by Singh et al. (2013), which focused their works on cloud computing, especially in the workload in cloud computing application. They propose a workload-based tool that uses What-If analysis to predict the impact of workload changes on the behavior of cloud computing applications. The authors suggested integrating What-If analysis in these applications, focusing on workload in cloud computing applications and the benefit of using What-If analysis. In this case, What-If analysis allows for users to simulate hypothetical scenarios with specific parameters and analyze what may happen. Then, it is possible to reconfigure the parameters and see what the impact in the workload was in terms of the performance of the application. In the same year, Herodotou and Babu (2013) presented a What-If Engine aiming to ease the experience of MapReduce users and applications, following the studies made by Herodotou and Babu (2011), but avoiding all the complexity of a MapReduce system. The authors adopted a "profile-predict-optimize" approach. The first phase profile is responsible for record the information about run-time behavior of MapReduce workloads. What-If analysis is used in the second phase and aims to estimate a number of tuning knobs. This phase includes tasks like

discovering the best choice of query execution plans in MapReduce, the degree of task-level parallelism of Map Reduce jobs, the choice tasks scheduling policies, among others. The last phase consists in choosing the best parameters for improving workload performance. Two years later, Bulyonkov and Filatkina (2015) focused on economic modeling and how What-If analysis can improve a model map-based information system. These authors build up a set of subsystems of the MIX project, which aims to display information environment about management of regions. What-If analysis was implemented in the system and then used for solving the problem of transportation based on the creation of hypothetical changes in parameters, like freight services and handling in transport hubs. Angelini et al. (2016), in the following year, presented a tool, called Visual Analytics Tool for Experimental Evaluation and how it can turn more efficient by integrate What-If analysis. The authors use What-If analysis to help determining the possible effects of modification of an Information Retrieval system and to ease and make more effective the experimental evaluation process. In the same year, Feldman (2016) presents a work about "Decision Model and Notation"-based decision models and What-If analysis. The authors presented then a web-based graphical tool, which support What-If analysis simulation models according to DMN. In this context, What-If aims to allow for the user to modify the status of business rules of the decision model and analyze the changes in the decision variables. Therefore, What-If analysis helps the user to find optimal decision, either the user wants to minimize or maximize his goal. Also, in the same year, Dogan (2016) focused on scientific workflows and the effect of using What-If analysis for improving those models. Using scientific workflows is possible to illustrate complex steps of experiments using large datasets taken to produce scientific papers. The authors aim for integrating What-If analysis and scientific workflows. This allows users for changing and analyzing results without running the workflow steps. So, users can get an insight without running complex experiments, making a prediction of what could happen in certain circumstances. Also, in 2016, Jiang et al. (2016) showed the impact of What-if analysis on cloud-hosted web applications, presenting a system, called 'WebPerf', which aims to explore hypothetical scenarios in Web applications. What-If analysis was used for helping developers to choose the right configuration and tiers for their performance needs, helping to know how the changes of service tiers and runtime load affect the page load time. In the end, the system presented a low percentage of error when estimating the distribution of cloud latency of the request. And finally, in 2016, Meurice et al. (2016) approached data applications that need to access to databases in a dynamic way. Specially, when occurs a change in a database schema, there is the need to adapt the source code concerning the database schema modifications. Meurice et al. used What-If analysis for analyzing

the evolution of a database schema and identifying those program inconsistencies and simulating future database schema modifications to determine how they would affect the application source code. In the following year, Ke et al. (2017) presented a Provenance-based What-If analysis approach for data mining processes. This approach uses provenance data to help to identify the data mining results that are affected by hypothetical business, rerun the affected portions and refresh the data mining results - provenance data is similar to metadata, it provides information about the data and its historical record. More Recently, Bourini et al., (2018) used What-If simulation to avoid poor design and unsuitable handling equipment, like bottlenecks and longer production time, which leads to higher production costs. The authors show that with a What-If tool for simulation, using Delmia Quest software, it is possible to enhance significantly the production line and reduce the risk associated with decision of handling system.

2.1.7 Classification of the studies surveyed

In order to provide a more clear view, we resumed the works described before (Table 1 and Table 2), grouping them based on their characteristics. The first column references the paper, the second column ("WIF") identifies the papers that address subjects related to What-If theory, like methodologies. The "WIF improvement" column indicates the contributions that show how to enhance the What-If process. The last group of columns ("WIF Application") indicates the papers that describe tools that use What-If analysis as an extra feature for improving their functionality, with further information concerning the area of application, in BI or in GIS and other application areas, or whether they include optimization tools.

By analyzing Table 1, we show how the considered papers spread among the several topics. The "WIF" column, which refers to papers that address What-If methodologies, shows that Golfarelli et al. 2006, and as said before, are innovative in this area. Following this work, some studies arise showing some particularities of the What-If analysis process, like, for example, Zhou and Chen (2009) that shows how What-If analysis works in Multidimensional OLAP environments. In the following years, and as illustrated in the "WIF Improvement" column, only a few papers show how to enhance the What-If analysis process, which shows that this is an unexplored topic. Most of the works described before fit in the "WIF application" column. These works describe tools that use What-If analysis to improve their functionality. In the end, the What-If analysis process is mostly used in BI and GIS areas and as an optimization tool.

Table 1 - Classification of the studies surveyed – What-If analysis

	WIF	WIF Improvement	WIF Application			
			BI	GIS and others	Optimization tool	Contribution to existing tools
Koutsoukis et al. (1999)			X			
Golfarelli et al., 2006	X					
Tsunokawa et al., 2006				X	X	
Chaudhuri and Narasayya (2007)			X			
Van den Akker et al., 2007					X	
Papastefanatos et al., 2008			X			
Zarras and Vassiliadis, 2008					X	
Jouini and Jomier, 2008			X			
Xiao et al., 2009a			X			
Xiao et al., 2009b			X			
Kottemann et al., 2009	X					
Zhou and Chen., 2009	X					
Brenner et al., 2010						X
Datta and Roy, 2010					X	
Golfarelli and Rizzi, 2010	X					
Bird and Zimmerman, 2012					X	
Gavanelli et al., 2012		X				
Moreno et al., 2012				X		
Deutch et al., 2013			X			
Saxema et al., 2013			X			
Xu et al., 2013		X				
Singh et al., 2013					X	
Golfarelli et al., 2013				X		
Wickramasuriya et al., 2013				X		
Herodotou and Babu, 2013					X	

Table 2 - Classification of the studies surveyed - What-If analysis (continued).

	WIF	WIF Improvement	WIF Application			
			BI	GIS and others	Optimization tool	Contribution to existing tools
Krishnamoorthy et al., 2014						X
Asharani et al., 2015				X		
Bulyonkov and Filatkina, 2015					X	
Angelini et al., 2016					X	
Feldman, 2016					X	
Dogan, 2016					X	
Jiang et al., 2016					X	
Rome et al., 2016						X
Meurice et al., 2016					X	
Rozema, 2016						X
Timar et al., 2017				X		
Hung et al., 2017		X				
Ke et al., 2017					X	
Van Cauwelaert et al. (2017)						X
Bourini et al., 2018					X	
Hartmann et al., 2018			X			

2.2 OLAP Preferences

The main purpose of using preferences in information systems is to try to adapt querying results to user preferences as much as possible, by either avoiding irrelevant information or overflowing information. Preferences can express the most interesting data for decision-makers, so that they can obtain a more goal-oriented information. The contributions reviewed are presented in two parts, respectively, Personalization frameworks and OLAP Preferences theory and applications.

2.2.1 Personalization frameworks

In general terms, preferences are used for improving decision-making processes in data warehouse or OLAP environments. Personalization frameworks can be used for helping users during OLAP analysis sessions by suggesting recommendations based on the analysis of former OLAP sessions. In 2005, Bellatreche et al (2005) proposed a framework for personalizing OLAP queries, to avoid overflowing information in querying sessions in OLAP systems. The authors proposed a framework that helps to filter the information considering the preferences of users. This framework requests the user his preferences and also a visualization constraint, and then computes the answer that respects both. Later, in 2008, Jerbi et al. (2008) proposed a query personalization framework, based on user context-aware preferences. This framework has the ability for extracting preferences related to the query context, and then suggests an enhanced query, where preferences are integrated dynamically into the original query. Following a similar research path, Giacometti et al. (2009) proposed as well a framework for recommending OLAP queries that leverages former users' investigations to enhance discovery driven analysis. The framework discovers similar queries made by the user in former OLAP sessions and suggests them to the user, making the user more aware of the querying process during the interactive analysis. Next, in 2009 and following their previous work, Jerbi et al. (2008), Jerbi et al. (2009a) presented a recommendation methodology to assist a user during decision-support analysis. They aimed at helping users to query multidimensional data and to suggest to the user how to discover interesting patterns. Continuing their research, Jerbi et al. (2010) proposed also an OLAP Content Personalization framework to derive a personalized content of a multidimensional database based on user preferences. This new framework allows for extracting a personalized content for a specific user, which leads to include the user profile parts into the qualification of the user queries to further restrict the data content that generates querying results. The two steps of personalized content extraction are preference selection and integration. In this work, the authors also discuss the performance of the framework through a set of experiments. After the work of Jerbi et al. (2010), in 2011, Kozmina and Solodovnikova (2011) proposed an OLAP reporting tool. This reporting tool was conceived for helping users during OLAP sessions, by suggesting recommendations based on historical activity records or on other similar and helpful reports in an implicitly way. Four years later, Kozmina (2015) continues this last work, by describing the method for generating recommendations reports, using explicitly stated user preferences, instead of implicit preferences, developing a tool for converting user preferences to OLAP schema elements. Also, in 2011, Biondi et al. (2011) developed a Java-based tool, called MyOLAP. This tool allows for

helping OLAP users suggesting user preferences, using soft query constraints during OLAP sessions. The authors focused on three aspects to make the approach more efficient, namely formulation, analysis and navigation. Aligon and Marcel (2012) presented a framework for summarizing former analyses to assist the user in the exploration of a data cube. These authors also evaluated the proposed framework by testing it with respect to a query personalization technique based on mining a query log.

2.2.2 OLAP Preferences theory and applications

Usually, preferences are used to improve the decision-making process in data warehouse or OLAP environments. OLAP preferences reflect the most interesting data that decision-making agents selected and analyzed based on information collected in past OLAP sessions, using a specific set of data cubes during certain periods of time. According to this, Rizzi (2007) focused on describing the main research issues that arise when developing a system that handles user preferences on OLAP cubes, focusing in particular several aspects regarding context-awareness, user interface and query optimization and processing. After this, Xin and Han (2008) proposed the P-Cube, a data cube for preference queries. These authors presented a study of the complete life cycle of processing P-Cube, including signature generation, compression, decomposition, incremental maintenance and usage for efficient on-line analytical query processing. Xin and Han also proposed a signature-based progressive algorithm that allows for pushing boolean and preference constraints in query processing, in a simultaneous manner. Additionally, they made a performance study that showed that the proposed method achieves at least one order of magnitude speed-up over existing approaches. Later, Golfarelli and Rizzi (2009) presented a very interesting approach for computing queries with OLAP preferences. These authors developed a preference algebra for OLAP considering some peculiarities, namely: preferences can be expressed on both numerical and categorical domains; preferences can also be expressed on the aggregation level of facts; and the space on which preferences are expressed includes both elemental and aggregated facts. Also, in 2010, Rizzi (2010) focused on distribution and personalization. The author considers these two aspects a new trend in BI. Both are relevant to support business scenarios, especially when dealing with a distributed net with multiple associates. The author also describes a peer-to-peer architecture that allows collaborative decision-making functionalities, including OLAP query reformulation. A year later, Aligon et al (2011) proposed another approach for expressing OLAP preferences. This approach is mainly based on personalization. It consists on mining a MDX query

log for extracting OLAP preferences, following a specific methodology: the log of past MDX queries is mined for extracting a set of association rules related to a set of frequent query fragments; then, given a specific query, a subset of pertinent and effective rules is selected; finally, the selected rules are translated into a preference that is used for annotating the user's query. Researching using preferences in OLAP continued with Ahmed et al. (2012), which mainly focused on preference summarization. These authors introduced an approach for defining OLAP user profiles. This approach aimed for discovering user preferences by analyzing historical OLAP query logs. More specifically, it preprocesses the all the information collected, then it divides the information in several categories (consensual, semi-conflicting and conflicting preferences), and finally it derives the user profile. At the same year, Marcel (2012) centered on personalization and recommendation in OLAP contexts. This author summarized several contributions for developing user-centric OLAP. Especially, he focused on the use of former queries logged by an OLAP server for enhancing subsequent analyses.

2.2.3 Classification of the studies surveyed

To resume the works described before about OLAP preferences, we summed all up in Table 3. The first column references the paper, the second column ("OLAP query personalization frameworks") indicates papers that address OLAP query personalization frameworks and the third column "OLAP Preferences Theory and Applications" identifies the papers that address studies about theory and research approaches involving of OLAP preferences.

Therefore, in Table 3, we can analyze how the several studies fit in the different topics. The "OLAP query personalization frameworks" column is divided into three sections, "User preferences", "User context-aware preferences" and "Past OLAP sessions". These sections refer to the functionality of the developed frameworks described in the papers. For example, Giacometti et al. (2009) presents a framework for recommending OLAP queries that discovers similar queries made by the user in former OLAP sessions. The "OLAP preferences theory and applications" is divided into "Expressing OLAP preferences" and "Personalization". These sections refers to the papers content. For example, Aligon et al. (2011) proposes an approach for expressing OLAP preferences.

Table 3 - Classification of the studies surveyed – OLAP preferences.

	OLAP query personalization frameworks			OLAP preferences theory and applications	
	User preferences	User context-aware preferences	Past OLAP sessions	Expressing OLAP Preferences	Personalization
Bellatreche et al., 2005	X				
Rizzi, 2007				X	
Jerbi et al., 2008		X			
Xin and Han, 2008					X
Giacometti et al., 2009			X		
Golfarelli and Rizzi, 2009				X	
Jerbi et al., 2009a		X			
Rizzi, 2010					X
Jerbi et al., 2010	X	X			
Aligon et al., 2011				X	
Kozmina and Solodovnikova, 2011	X		X		
Biondi et al., 2011	X				
Ahmed et al., 2012					X
Marcel, 2012					X
Aligon and Marcel, 2012			X		
Kozmina, 2015	X				

Chapter 3

What-If analysis: process and discussion

3.1 What-If Analysis Process

What-If analysis (Golfarelli et al., 2006) allows for helping managers and executives, persuading the whole decision-making process. A What-If analysis process starts showing the intention of executives or managers to take future steps, having some doubts or questions to be answered. Then, decision makers are responsible for creating hypothetical scenarios about the specific business situation to explore and help them to take business decisions. Running the simulation model enables the user to get a better understanding of the business and to explore different outcomes that are likely to occur under different scenarios.

3.1.1 The 'What-If Question'

Usually a What-If analysis starts with the definition of a What-If question - 'What-if ...?' With a What-If analysis, it is possible to explore a scenario to get some information for answering the stated What-If question. Therefore, a What-If question represents a question that denotes the intention on exploring the effects of changes on business-related variables, revealing what will

happen if a user changes values of a set of variables. This analysis allows for the user to get information for answering the What-If question. For example, if an analyst wants to explore what will be the effects of the change of the sales value of a specific product of a particular store of a specific year, translated as: "What (should be the values of the sales and costs) if we want to increase the overall profit in 2014, 2015 and 2016 by 10% of the products with black, blue, red and white colors?". So, he starts by selecting a set of business-related variables as parameters in the What-If scenario included in the What-If question. Next, he changes the sales' value to the wanted value (increasing by 10%) and performs the What-If process. Finally, the new scenario, usually referred as the prediction scenario, is created. In the end, the user can compare the current situation with the calculated prediction and get some new information that will help to answer the What-If question previously made. With this, the decision maker can deal with the best guidance in specific situations.

There are several advantages of using a What-If simulation. It can help decision makers to predict the outcome of daily business decisions, helping to see how different changes could affect various characteristics of the business. It makes possible to decision makers to decide which decisions are beneficial or harmful to the business, without putting the business at risk (Kellner et al., 1999). When the What-If simulation model is built, variables' values can be changed, and the simulation model can be used repeatedly to analyze different scenarios. What-If simulation not only can help decision makers to make better and more informed decisions by changing assumptions, observing or estimating results but also can help to make daily business decisions in a quicker and easier way.

3.1.2 The Simulation Model

When performing What-If analysis, a simulation model is the main focus of the whole process (Figure 2). The simulation model is a representation of a real system reflecting the operations and relationships between variables of the real business. Thus, it is through the simulation model that the user or analyst can verify and analyze an ordinary behavior of a real business system. The simulation model is composed by various analysis scenarios. Each scenario is constructed based on the user's choice for the forecast he intends to seek, taking into consideration historical business data. Each of these scenarios is composed by a set of business variables and a set of scenario

parameters. The former set is related to the business domain, the latter is a set of variables technically related to the simulation.

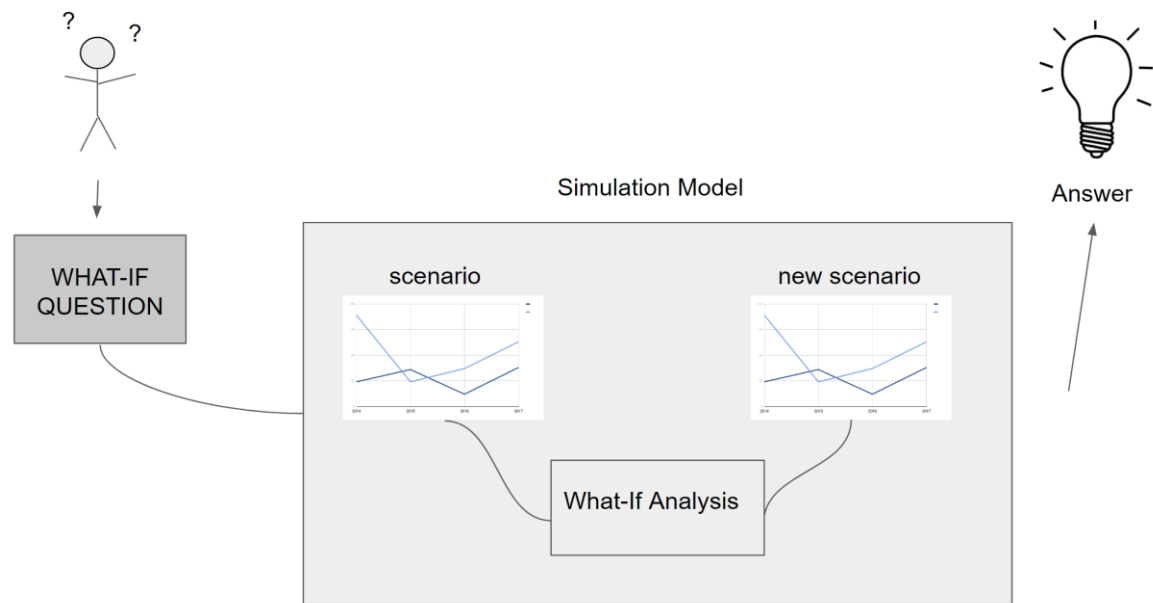


Figure 2 – Representing a simulation model in What-If analysis.

The simulation may be run several times to analyze different scenarios. Each scenario is defined by a set of business variables and a set of additional settings, called scenario parameters or configuration parameters. Scenario parameters usually refer to the algorithm configurations and some other additional parameters of the chosen What-If analysis tool. Examples of additional parameters are, for instance, filters of PivotTables in Microsoft Office Excel. After performing What-If analysis, we get new values and consequently a new (changed) scenario – we got a prediction. It is the user's responsibility to edit the variables and accept or reject the alteration of values and the prediction (the new scenario) as shown in Figure 2.

In databases, the concept of business variables is used to denote the attributes of a given entity. In simulation business variables are classified in different types, according to their role. There are dependent and independent variables. The dependent variables are those that modify their values as a consequence of changing a given business variable value. There are also independent

variables, which are those that do not change, in any case, when the value of a business variable changes.

In this thesis, we only do analysis with a single business variable. For example: Let A, B, C and D be the set of attributes of a dataset, A be the attribute that we intent to alter to perform the simulation, and B and C be the set of dependent variables of A, for example, let $B=A+10$; $C=A-10$ and $D=2$. If we intend to raise the value of A by 10%, the What-If question will be 'What (will happen to the business variables B, C and D) if we increase the value of A by 10%'? After performing the What-If analysis process, the A value will be increased by 10%. The B and C values will be altered too, because they are dependent on A: if A changes, then B and C change too. The value of D remains the same. In this case, as D is an independent and constant, including D as input parameter in the scenario depends on its contribution in the analysis and therefore should be the user responsibility.

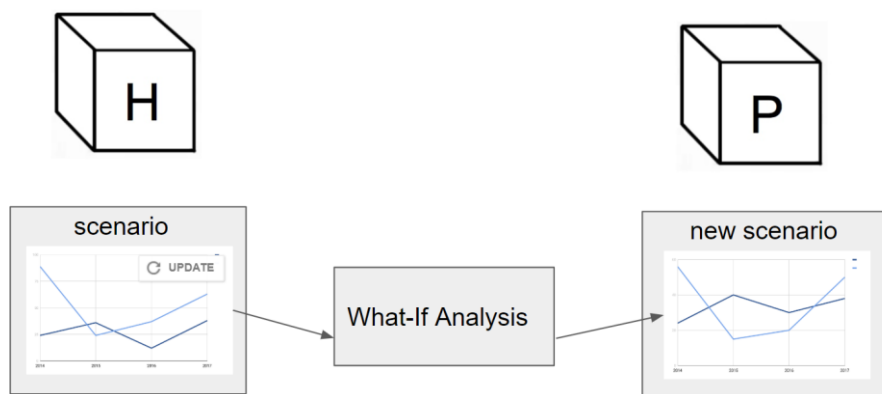


Figure 3 – Relating historical and prediction scenarios.

What-If analysis allows for decision-makers to simulate changes in historical data, creating hypothetical scenarios and helping for predicting the future. To do this, the data is altered to be able for assessing the effects of the changes. The user is accountable to change the value of one or more business variables and set the scenario parameters in a specific scenario, taking into consideration the analysis goals. The What-If process then calculates the effect of the impact of the change of the business variables, presenting the user a new changed scenario, the prediction

scenario (Figure 3). It is the responsibility of the user for accepting or recalculating this last scenario (Golfarelli et al., 2006).

3.2 A What-If Case Example

To perform a What-If case example for showing how What-If analysis works, we chose the Microsoft Office Excel Tool (Products.office.com, 2019). Microsoft Office Excel allows for creating PivotTable reports based on OLAP source data (Support.office.com, 2019). OLAP PivotTable Extensions is an Excel add-in, which extends the functionality of PivotTables on Microsoft Analysis Services multidimensional structures. As an OLAP analytical tool, Microsoft Office Excel allows to analyse data extracted from an OLAP source in a quickly and easily manner. With a PivotTable What-If Analysis it is possible to modify data in PivotTable cells easily, recalculating those values and if the user is satisfied with the changes, it is possible to publish them in the OLAP data source.

The example case study to show in a simple example how What-If analysis works on Microsoft Office Excel and using the Wide World Importers (SQL Server Blog, 2016) database, a Microsoft product sample. In this case example, we use an OLAP data cube as input data and to perform What-If analysis, Microsoft Office Excel has available the PivotTable Writeback feature. An example of a What-If question could be "What-if we want to increase the overall profit in 2014, 2015 and 2016 by 10% of the products with black, blue, red and white colors?". One possible answer that we can probably take by creating this scenario is how the profit values among the years would vary if the total profit value increase by 10%.

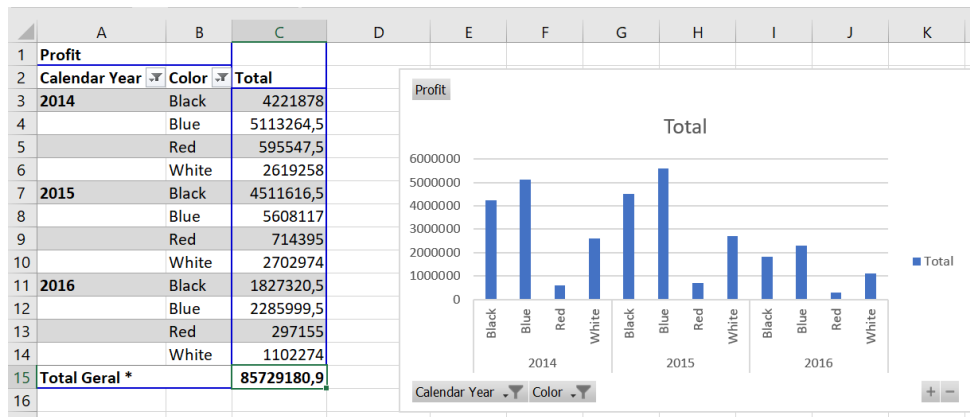


Figure 4 - Performing What-If analysis - Example of a PivotTable scenario.

To create the simulation model shown in Figure 4, we select "Calendar Year", "Color" and "Profit" as business values; and information about profit values among the years 2014, 2015 and 2016, and by color of available products for sale: products with black, blue, red and white colors. Therefore, to create the scenario we select "Calendar Year", "Color" and "Profit" as business values. As shown in Figure 4, we have detailed information about the profit values of "Calendar Year" in column A, "Color" in column B and the profit values in column C. The overall profit is represented in the data cell C15.

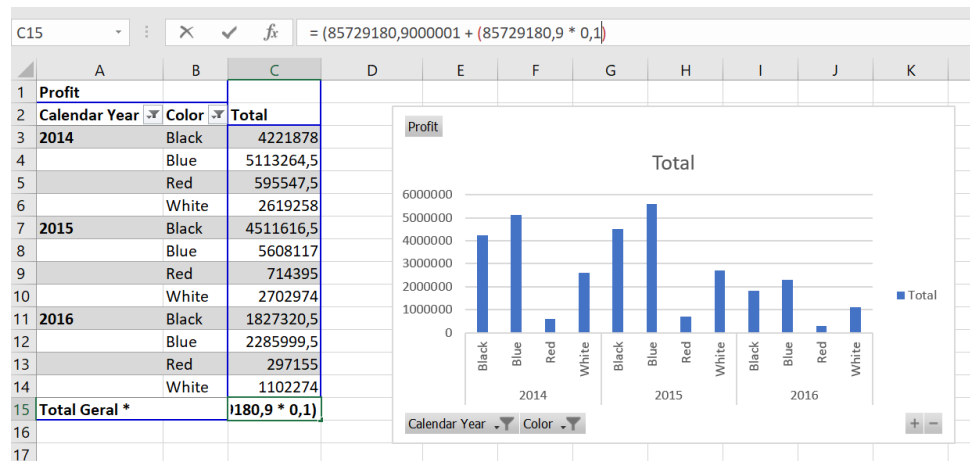


Figure 5 - Performing What-If analysis - Change of variables' values.

We want to analyze the effects of the change of the C15 data cell (the overall profit value). Bearing in the mind the pre-defined What-If question, we intend to increase this data cell value by 10%. In Figure 5, we select the C15 data cell and increase its value by 10% by altering the C15 data cell formula: $(85729180,9000001+(85729180,9*0,1))$. In Figure 5, the What-If analysis has not been performed yet. To perform the PivotTable Writeback (What-If analysis) is needed to "Calculate PivotTable with Changes" in the Microsoft Office Excel tool.

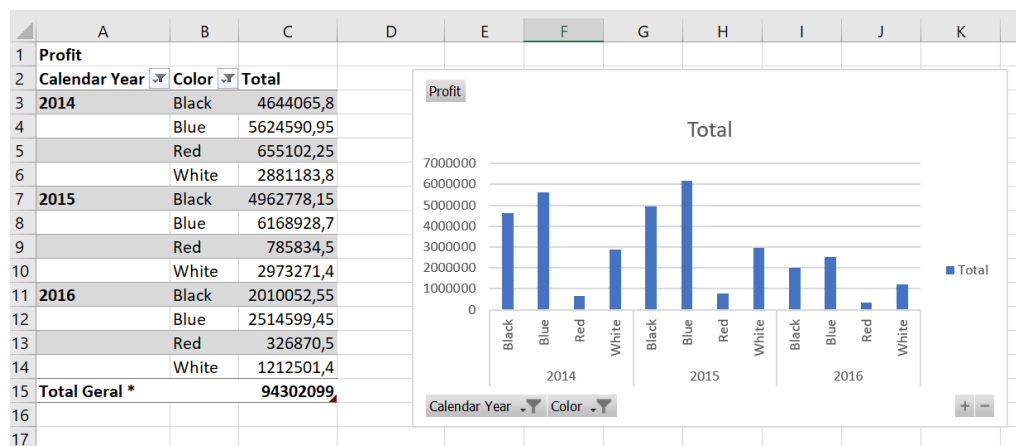


Figure 6 - Performing What-If analysis - Accepting changes.

After we perform What-If analysis, we can analyze the effects after increasing the C15 data cell by 10% (Figure 6). As we can see all the profit values of all the colors of all years represented in the column C were increased by 10%, which means that if we want to increase the overall profit value by 10%, all the profit values should also increase 10%. In this case example, using the traditional What-If analysis, it is possible to obtain information about the profit values of products by color and to take appropriate decisions. However, there is information that is not displayed to the user. Later this example will be explored using What-If analysis combined with OLAP preferences and more relevant information is provided to the user enabling better decision-making process. For larger examples, the advantages of the combined methodology are even greater.

3.3 Methodologies using What-If

Previously, we presented a few papers with many kinds of improvements of What-If analysis. In this section, we focus and discuss papers that present some related What-If methodologies. Therefore, papers like Xu et al. (2013) that only present improvements in some algorithms of the What-If process are not addressed.

When we state that we focus on papers that present methodologies, we mean that we will address papers in which What-If analysis is used within a methodology. That does not mean that all of them present methodologies that improve What-If processes. In some papers, What-If analysis is used to improve other methodologies. In fact, there are not many papers that present a methodology that aims at improving the traditional What-If analysis. To the best of our knowledge, only Gavanelli et al. (2012) does it. To make the contributions clearer, we start by the paper that set the basic What-If analysis methodology.

Golfarelli et al. (2006) is a basic What-If analysis methodology and a seminal paper: describes a methodology that describes how to use the What-If analysis in an OLAP environment. The authors suggest a methodology to be followed when dealing with problems that need the creation of hypothetical scenarios for answering questions during decision-making processes, or in other words, resorting to What-If analysis. This methodology is composed by six phases:

1. Definition of the main goal analysis and which scenarios to perform in the simulation.
2. Business analysis and identify the business variables involved in the defined scenario and the relation between them.
3. Analysis of the data source and its content.
4. Definition of the multidimensional data structure, considering the variables involved with the simulation and the relationships between them.
5. Creation of the What-If analysis simulation using the pre-defined multidimensional data structure as basis to the prediction.
6. Implementation and validation of the simulation model.

Herodotou and Babu (2013) also suggested a methodology that incorporates a What-If engine for predict performance values of a MapReduce system. The authors use a What-If analysis to help estimating what will be the performance of a MapReduce job according to several variables. A MapReduce job can be characterized by a set of components: program, input data, cluster resources and job configuration settings. Their proposed methodology is composed by four phases:

-
1. Definition of the What-If questions. These questions mainly translate in some changes in the MapReduce jobs executions, like increasing the size of input data, add nodes to the cluster resources.
 2. Definition of the MapReduce job and other configurations settings of the hypothetical job to perform in the What-If simulation.
 3. What-If Engine, which consists in estimating the virtual job profile and simulate the MapReduce job execution using the pre-defined parameters.
 4. Validation of the predictions of the MapReduce workflow performances.

Also Gavanelli et al. (2012) presented a methodology that can be followed if one wants to integrate What-If analysis with an optimization-simulation hybridization approach for improving conventional process What-If analysis for policy making. Firstly, the authors create a Decision Optimization Support System (DOSS) to replace the human interaction and then integrate machine learning in the process for avoiding the "generate and test" approach of the traditional What-If analysis process. In this case, machine learning helps to synthesize constraints for the decision-making support system from the created simulation results using the What-If simulator. Their process is composed by several elements, each one having its own function, namely:

- Definition of the hypothetical scenarios for policy making and possible solutions;
- What-If simulator. The simulator takes the scenarios as input and performs What-If analysis. As output the simulator generates a set of tuples, which are composed by decisions and observables.
- Machine learning. The set of tuples is stored as training set for the learning component.
- Decision Optimization Support System. The DOSS receives as input the set of possible decisions and the output of the machine learning and returns the optimal scenarios.

All the described methodologies differ in several aspects. The first one, Golfarelli et al. (2006), describes the basic What-If analysis methodology. One of drawbacks of the conventional What-If analysis is the possible lack of expertise of the user, which may lead to poor results and an inadequate outcome. Herodotou and Babu (2013) suggested a methodology that uses What-If analysis for predicting performance values of MapReduce jobs. This is an example of a methodology that uses What-If analysis to ease the handling of an already developed tool. And

finally, Gavanelli et al. (2012) presented a methodology that aims at improving the traditional What-If analysis. There are some issues related to this methodology. Machine learning requires a large number of simulations before attaining a mature and effective level. Also, the methodology lacks universality in the sense that it is necessary for developing an optimization model, tailored to a particular problem, to be embed in the process. In this paper, authors considered an application in a case study in regional energy planning, and used an optimization technique, called Benders decomposition, for addressing the optimization problem.

In this work, we aimed at establishing a methodology to improve traditional What-If analysis processes that overcome the issues mentioned above. In the following chapter, we review OLAP preferences, because the proposed tool combines What-If analysis with OLAP preferences.

3.4 Summary

In this chapter was presented in detail the explanation of the What-If process and its components. The What-If analysis allows for the user to create hypothetical scenarios mainly to explore the effects of changing some variables. For example, with this process is possible to create a simulation for analyzing the effects of "What if we want to increase the price of the product A by 10%?". Thus, the need of using What-If analysis comes up arising a doubt. The doubt can be translated into a What-If question, and in turn the What-If question can be translated into a specific simulation. What-If simulation is composed by a set of scenarios, which need to be configured using a set of scenario settings: the scenario parameters, which depend on the chosen tool, and on the business variables. The user needs to be aware of the business data to know which data to select and add to the simulation. This step can be a drawback to the user when using What-If analysis. If the simulation input data is inadequate, the outcome can be poor and lead to low quality decisions.

Chapter 4

OLAP Preferences

4.1 Contextualization of OLAP preferences

Over the years, OLAP (On-Line Analytical Processing) (Chaudhuri and Dayal, 1997) systems have been helpful in decision-making processes. OLAP may be seen as a BI technique, providing some of the most predominant and relevant tools for decision-support systems. These tools have the ability to manipulate and analyze a large volume of data from multiple perspectives. Due to its characteristics, OLAP systems allow for performing complex analytical and ad-hoc analytical queries in order to optimize multidimensional data structures with a quick execution time. OLAP applications are used in several business areas, such as sales, forecasting, finance and marketing, just to name a few. OLAP techniques refers mainly to data analysis techniques developed to analyze data stored in data warehouses. A data warehouse (Kimball and Ross, 2011) is a subject-oriented, integrated, time-variant and non-volatile collection of data in support of the management decision-making process. Data warehouses are specially centered on companies' specific concepts, in other words, they are business subject-oriented. They store detailed information about the several entities: customers, products and stores, for example; and detailed facts of sales, as, for

instance, date, the products bought, the name of the employee who made the sale and the name of the customer.

A data warehouse allows for gathering data from several data sources, integrating all the information in only one place. This eases to access to specific information, turning the process of decision making simpler and faster. Data warehouses are time-variant, which means that all the data warehouse data is dated, allowing for to analyze data from a specific period, like, for example, analyzing product sales of 2017. The use of data warehouses also provides a structured and organized way of analyzing data. And finally, data warehouses are non-volatile, which means that data is consistent and stable. New data can be added to the data warehouse, but it can be never removed. All these data warehouses' properties help companies to get a more consistent view of the business.

A data warehouse data can be represented as a multidimensional view of data, which can be materialized as multidimensional views (data cubes) and used in further inquiry. A data cube structure, also called multidimensional database, is composed by a set of data cells. Each data cell in the data cube stores information about the corresponding entity values in a multidimensional space. A lattice (Figure 7) is a representation of a data cube.

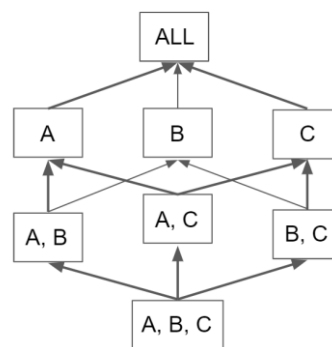


Figure 7 - The lattice of a data cube.

A typical multidimensional schema of a data warehouse (in Figure 8) is usually composed by a central table, called fact table and a set of tables linked to the main table, called dimensions. Following the formal specifications in Jerbi et al. (2009b), we can define a multidimensional schema and its components:

Definition 1. A multidimensional schema can be defined as $\langle FT, D_j \rangle$, where FT is a fact table, D is a finite set of dimensions $\{d_1, \dots, d_j\}$; in this case example, we focus in multidimensional schemas with one fact table and a finite set of dimensions.

Essentially, a fact table can have two types of columns: keys and measures. Fact Tables contain foreign keys (FK) from the dimension tables. Measures are numeric values that can be operands in mathematical operations and are used to express business metrics. Foreign keys are responsible for linking fact table's rows to the correspondent dimension table data. Dimensions have primary keys and attributes.

Definition 2. A fact table noted FT can be defined as $\langle M_k \rangle$, where M is a finite set of measures $\{m_1, \dots, m_k\}$, each defined on a numerical domain ($\text{Dom}(m_i)$, $0 < i \leq k$).

Definition 3. A dimension noted D can be defined as $\langle A_n \rangle$, where A is a finite set of attributes $\{a_1, \dots, a_n\}$, each defined on a categorical domain ($\text{Dom}(a_i)$, $0 < i \leq n$).

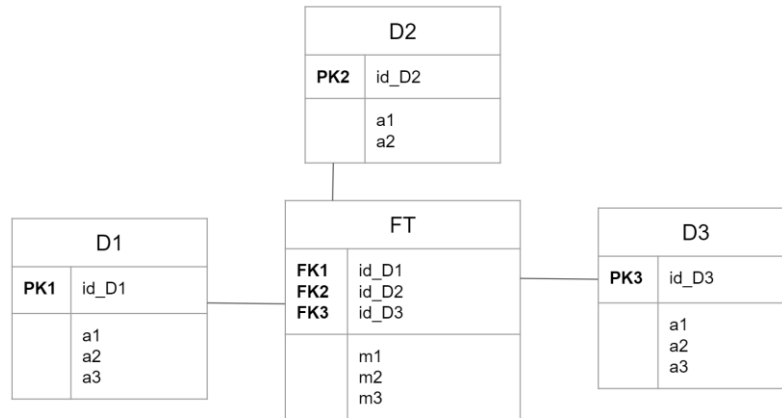


Figure 8 - Example of a multidimensional schema.

Dimensions are usually organized in hierarchies, which allow to support different levels of data aggregation.

Definition 4. A Hierarchy of a dimension is defined by $\{Au, Di\}$ where H is a set of attributes. each defined on a categorical domain ($Dom(ai), 0 < i \leq u$)

OLAP tools should be capable of performing operations that enable users to analyse multidimensional data interactively. A user can perform several OLAP operations (Chaudhuri and Dayal, 1997), such as drill-down (decreasing the level of aggregation), roll-up (increasing the level of aggregation), pivot (re-orienting the multidimensional view of data), slice and dice (selection and projection), which allow for interacting and exploring a data cube. For example, a user can analyse sales values by month, and then see them rolled up by year (Figure 9).

	All	1	2	3	4	5
All	1906136334	110100921	101219485	115214311	132451931	154618272
Alabama	42990211	2630827	2440801	2584165	3185286	3300306
Alaska	35787246	1829468	1607062	2569305	2076409	2905531
Arizona	29061888	2007662	1394770	1736845	1537657	2186867
Arkansas	22848241	1367370	1499129	1245550	1728514	2061026
California	98520376	5916590	5003852	6280838	6628194	7520942
Colorado	52922422	2749539	2956259	2818581	4142578	4510259
Connecticut	7909936	543184	277828	868772	586392	851925
Florida	62235376	3420382	3362878	3569970	4760494	5300842

↓

	All	2013	2014	2015	2016
All	-559752293	573135887	1906136334	-686396343	1942339125
Alabama	186529431	12455305	42990211	76526858	54557057
Alaska	145775135	10152029	35787246	65370469	34465391
Arizona	123638848	10219592	29061888	59306626	25050742
Arkansas	103443969	7320759	22848241	44410076	28864893
California	414441675	30235982	98520376	189773018	95912299
Colorado	217461472	13767444	52922422	99992553	50779053
Connecticut	46644650	2644136	7909936	20142228	15948350
Florida	263030011	19302855	62235376	122455257	59036523

Figure 9 - Roll-up operation over a data cube.

The drill-down, on the contrary, allows the user to navigate through the details, a user can analyse sales values by month, and then see them drilled down by day (Figure 10). Slicing and dicing allow the user to select (slicing) a specific set of data and project (dicing) that same set of data. The

workload of an OLAP application is measured by the user's navigation analysis task, or, in other words, the workload caused by the manipulation of the results by the user using OLAP operations.

	All	1	2	3	4	5
All	1906136334	110100921	101219485	115214311	132451931	154618272
Alabama	42990211	2630827	2440801	2584165	3185286	3300306
Alaska	35787246	1829468	1607062	2569305	2076409	2905531
Arizona	29061888	2007662	1394770	1736845	1537657	2186867
Arkansas	22848241	1367370	1499129	1245550	1728514	2061026
California	98520376	5916590	5003852	6280838	6628194	7520942
Colorado	52922422	2749539	2956259	2818581	4142578	4510259
Connecticut	7909936	543184	277828	868772	586392	851925
Florida	62235376	3420382	3362878	3569970	4760494	5300842

↓

	All	2014-01-01	2014-01-02	2014-01-03	2014-01-04	2014-01-05	2014-01-06
All	1906136334	5833527	3325001	5327643	1844632	(null)	3813060
Alabama	42990211	263527	94468	246584	57093	(null)	38088
Alaska	35787246	93993	56632	37916	95110	(null)	76316
Arizona	29061888	150468	94458	75912	57015	(null)	(null)
Arkansas	22848241	112899	56718	170576	(null)	(null)	(null)
California	98520376	357389	94451	322410	95037	(null)	305238
Colorado	52922422	94152	151204	56985	57027	(null)	76296
Connecticut	7909936	37690	(null)	56829	57030	(null)	(null)
Florida	62235376	263508	188793	132546	(null)	(null)	(null)

Figure 10 - Drill-down operation over a data cube.

An OLAP analysis consists in exploring interactively the Multidimensional Database. An OLAP analysis session can be defined as an interactive session during which a user performs a set of OLAP operations to find relevant data for decision making. OLAP should support *ad hoc* analytical queries. OLAP queries are more complex than queries used in relational databases. OLAP queries usually consist of multidimensional operations, like aggregation and grouping operations. An *ad hoc* analysis can be for example: querying how was overall sales profit in New York last year. If it was lower than expected, the analyst might want to know how the profit values may vary by month by performing another query. By analysing the profit values by months, the analyst may want to know which regions present the lower profit values. One way to do this is to analyse the

profit values by month and by regions and maybe add the product type sales values to analyse which products present the lower sales values.

The amount of data to be analysed and the complexity of the OLAP queries to be answer are two factors that can influence the query time response. When the execution of a query takes more than a few minutes, the company productivity can be impaired. Therefore, knowing which data should be materialized is a relevant step during the analysis process. A full materialization of the cube uses a huge amount of memory and storage. Although this approach is considered to be the "best" approach with best query time response, precomputing and storing the whole cube is not a practicable solution for large databases. On the other hand, if there is no cube materialization at all, it is necessary to compute the cube during the execution of the query, and therefore a longer time is required. This approach consists in performing the query, accessing to the data on request and computing it (Harinarayan et al., 96). It is necessary to find the right balance.

The outcome of complex OLAP queries may be a huge volume of data, which may contain a low percentage of interesting information to the user. Due to large volumes of data, typical OLAP queries performed via OLAP operations can make data explorations a large burden or even impractical. Using personalization can help the user by assisting him during the OLAP analysis by suggesting the next step or even by helping the user to choose which information is the most interesting. Thus, integrating preferences can be valuable in OLAP analysis.

4.2 OLAP Personalization

In a general context, personalization helps users to focus on the most interesting data. First, we address general issues concerning personalization, and then focus on OLAP personalization. To do personalization we use a specific piece of software, usually designed as recommender (or personalization) systems. They are software tools and techniques to discover and recommend items that are more appropriate to a specific user. To do that, the system must be able to deduce what are the needs or requirements of the user. Recommendations help the users to discover what he/she is looking for, filtering and obtaining the best outcome based on his/her preferences, as for instance, likes, dislikes or orientation. A recommender system must have a way to predict what will

be considered interesting or useful to the user. To perform such task, recommender systems need to collect users' interests, which can be explicitly expressed or inferred by interpreting users' past choices. The suggestions are provided by the system to the user as ranked lists of items. To gather this list, recommender systems recur to techniques and methods to predict which are the most suitable items based on the user's preferences.

Personalization systems are already widely used. These systems differ essentially on the chosen approach for discovering which will be the recommendations for the user (Lu et al., 2015; Ricci et al., 2015; Beel et al., 2016), namely:

- Content-based filtering – the content-based systems aim is to recommend items that are similar to items that a specific user has already expressed as interesting. Interesting information is stored in a user profile. The system analyzes the description of those items, in order to determine the common preferences or set of features used to characterize the set of items. The personalization system is then responsible for interpreting these defined preferences, comparing them with unrated items and discovering which unrated items could be suggested as interesting to the user. The system opts to choose the unrated items that are similar to the ones on the user profile.
- Collaborative Filtering – these systems suggest recommendations to the user that are based on items that other users, who share similar interests considered interesting. More specifically, these systems aim to match the rating system for objects of a specific user with the rating system for objects of similar users, which means users that have similar interests or even characteristics, in order to produce recommendations for items not yet rated by the user. These systems use a mining classification algorithm that compares a user's profile with historical profiles of other users to identify which ones have similar tastes or interests.
- Knowledge-based – this recommender system offers items to users based on knowledge about the users, items and their relationship. These systems aim at recommending items to a user based on a specific knowledge domain about how a specific item is useful for a particular user. In other words, how the item features adapt or meet the users' needs or preferences.
- Hybrid Recommender Systems – this type of recommender systems is based on the combination of the above described systems. A hybrid system, for instance, combines a

technique A with a technique B, and tries to use the advantages of the technique A to fix the disadvantages of the technique B.

- Computational intelligence-based – some recommendation tasks can be solved with the help of techniques developed in the field of computational intelligence, like data mining algorithms: clustering techniques, association rules, Bayesian techniques and artificial neural networks.

4.2.1 Personalization in OLAP systems

More recently, some personalization techniques have been developed in a multidimensional databases context. Personalization and recommendation are used to make the OLAP experience less confusing to the user. The main goal is to help the user by orienting him during the OLAP session analysis when navigating huge amounts of multidimensional data. Recommendations are made to guide the user through the OLAP session, by filtering irrelevant results so that the user can focus on the most relevant data. OLAP personalization helps the user to deal with either too many or too few results, formulate a query corresponding to a specific objective that the user can't express or even to suggest new queries to pursue the navigation. To do this, the integration of preferences is useful. Preferences are used to get the irrelevant results filtered or even rank the results to get the most relevant first.

The main goal of using personalization in an OLAP context is to ease the entire user experience. Personalization helps the user to get more refined information in its analysis, by delivering relevant information to the user. OLAP personalization can fit into each of the following approaches:

- Query recommendation. The system recommends queries based on the current query and on information of past sessions. This approach aims essentially to help the user to navigate the cube by easing this process in an OLAP session analysis.
- Personalized visualization. The user specifies a set of constraints that are used to determine a preferred visualization.
- Result ranking. The system is responsible for ranking the results of a specific query. The ranking process consists of using a total or partial order to organize the data and display the most relevant data first.

-
- Query contextualization: based on a previous analysis of the context of the OLAP session, suggestions are made to the user to enhance the current query by adding context based preferences predicates.

OLAP personalization and recommendation approaches can be categorized using the following criteria (Golfarelli et al., 2011):

- Formulation effort. This criterion refers essentially to the level of involvement of the user regarding the user preferences during OLAP sessions. In some approaches, the user needs to manually specify his preferences, while in others, the user preferences are inferred from the context of the analysis and the information of the user profile.
- Proactiveness. This criterion allows for evaluating how some approaches react during OLAP sessions. Suggesting new queries based on past navigation is an action of low proactiveness. Changing the current query or posting an outcome before returning them to the user is an example of an approach with increased proactiveness.
- Prescriptiveness. An approach with a high prescriptiveness uses profile elements as hard constraints, adding them to a query. On the other hand, an approach with a low prescriptiveness uses profile elements (preferences) as soft constraints, adding them to a query: tuples that satisfy as many profile criteria as possible are returned even if no tuples satisfy all of them.

Taking in consideration the above descriptions, the user's experience in an OLAP session can be improved by i) decreasing the formulation effort by decreasing the user involvement in the definition of the preferences; ii) increasing the proactiveness by limiting the interference of the user by changing the current queries and anticipating the query results; and iii) providing low prescriptiveness by annotating queries with soft constraints.

According to (Kozmina and Niedrite, 2011), the following topics describe OLAP personalization types presented in the literature:

- The *Dynamic personalization* consists in creating an adapted OLAP cube during the execution time according to the needs and performed actions of the user approach (Garrigós, et al., 2009).

-
- *Visual personalization of an OLAP cube* consists in easing the process of composing queries for the user in a database language, like SQL or MDX. An interface for formulating queries with graphical OLAP schema was provided by Ravat and Teste (2009).
 - In a *User Session Analysis* approach, the main goal is to recommend patterns detected in query logs of past OLAP sessions for helping users in a current session. The discovered information from past sessions delivered to the user aims to help the user navigate the cube, when facing the same unexpected data as the current sessions (Giacometti et al., 2009).
 - In a *User Preference Analysis* approach, the recommendations suggested to the user are based on his preferences. User preferences can be inferred from context-based method of his OLAP session analysis and are used to help the user on further analysis. User preferences are stored in a user profile and ranked with a degree of importance, and in a posterior phase are used to generate recommendations - the recommendation with the highest degree of importance is displayed to the user (Jerbi et al., 2009).
 - *Preference Constructors*. This approach consists of an algebra that allows for expressing preferences on queries. In other words, an algebra that allows for formulating preferences on attributes, measures and hierarchies. Preferences can be expressed on both categorical (attributes) and numerical (measures) domains and can be formulated on the aggregation level of data (Golfarelli and Rizzi, 2009).

In summary, OLAP personalization has been significantly useful in decision making processes. It helps users reducing the struggle of the analysis process, helping them to find the most interesting information. The time spent by the user in finding the wanted outcome is significantly reduced. Even in an OLAP session analysis, the time spent in the analysis can be minimized due to the reduction of the number of queries necessary to retrieve "manually" the results that best match user preferences. We overviewed some of the methods of OLAP personalization and how they have been used. In short, the main goal of the OLAP personalization is to help the user providing recommendations in the analysis based on his/her preferences.

4.2.2 A formal definition of Preference

Preferences arise naturally on a daily basis. The definition of preferences is quite common in our daily life, which means that people can intuitively express their preferences. A person can spontaneously communicate what she thinks or feels about a specific thing or object. Personal preferences can come from subjective feelings, influences or past experiences. When facing two objects, a person is used to express her preferences in a declarative way. For example, someone can automatically say "I like A better than B", and this kind of interpretation of preferences is universally understood. All the user preferences should be considered, and fulfilled as much as possible. There is no guarantee that preferences can be always satisfied. In case the system cannot find the perfect match suggestion, people must be prepared to accept other alternatives as suggestions or to compromise. The system recommends to the user the most similar suggestion that fits best the user preferences. In this case, the best possible match is suggested to the user.

A formal definition of preference was proposed by Ore (1962), Kießling (2002), and Kießling (2005). Preferences can be modelled mostly using strict partial orders. Let see how we can describe formally a user preference.

Definition 5. Let $A = \{A_1, A_2, \dots, A_k\}$ be a non-empty set of attribute names A_i associated with domains of values $\text{dom}(A_i)$, $1 \leq i \leq k$:

a) A preference P on a set of attributes A is a strict partial order defined as $P = (A, <P)$, where $<P \subseteq \text{dom}(A) \times \text{dom}(A)$. Thus, $<P$ is:

- Irreflexive (not $(x <P x)$), which means that the elements need to have some degree of comparison, the same element cannot be higher or lower than itself.
- Transitive ($x <P y \wedge y <P z \rightarrow x <P z$), which means that if Y is preferred to X and Z is preferred to Y , Z will be preferred to X .

These two properties imply asymmetric ($x < y$ and $y < x$ implies $x = y$). A function is asymmetric if X is preferred to Y and Y is preferred to X , then X and Y are the same element.

Further: $\text{range}(<P) := \{x \in \text{dom}(A) \mid \exists y \in \text{dom}(A):$

$$(x, y) \in <P \text{ or } (y, x) \in <P \}.$$

If $X <P Y$, then 'Y is preferred to X'. A preference $P = (A, <P)$ is an irreflexive, transitive and asymmetric binary relation $<P$ on the domain of values of attributes set A;

- b) The unordered (synonym incomparability) relation $||P \subseteq \text{dom}(A) \times \text{dom}(A)$ is defined as:
 $x ||P y \text{ iff } \neg(x <P y) \wedge \neg(y <P x);$
- c) A preference P is a chain (synonym total order) is defined as:
 If and only if for all $x, y \in \text{dom}(A)$, $x \neq y: x <P y \vee y <P x;$
- d) A preference P is an anti-chain iff $<P \emptyset$. The antichain on an attribute A is denoted as $A \leftrightarrow;$
- e) A preference P is a weak order if negative transitivity holds. A weak order is a binary relation $<P$ over a set of Attributes A is defined as: if $\neg(x <P y) \wedge \neg(y <P z) \rightarrow \neg(x <P z)$ which means that if a first element is not related to a second element and in turn, that element is not related to a third element, then the first element is not related to the third element;
- f) The maximal values of $P = (A, <P)$ are defined as: $\text{max}(P) := \{v \in \text{dom}(A) \mid \neg \exists w \in \text{dom}(A): v <P w\}.$

4.2.3 Extracting OLAP Preferences using OLAP mining

As seen before, in an OLAP context, there are several ways for extracting preferences. Performing On-Line Analytical Mining, also called OLAP mining (Han, 1997) is one of them. OLAP mining is a mechanism which integrates OLAP and data mining, which means that a data mining technique is applied to a part of the multidimensional structure containing historical data at different levels of abstraction. The choice of a data mining technique is done according to the user's needs. Data mining aims for helping to discover non-trivial, unknown and interesting knowledge (or patterns) in the data stored in large historical databases.

OLAP mining techniques are promising due to its own characteristics. OLAP mining provides the user with the flexibility of choosing the desired mining function and the possibility of switching mining tasks dynamically and consequently ease the process of extracting and achieving the required outcome. Dealing with data mining, it is possible to work on consistent, integrated and cleaned data. If the user wants to improve the quality of data, he can perform some techniques of pre-processing data, like data cleaning or data integration. This pre-processing data step is essential for achieving the best data quality and consequently, better quality of the mining outcome. And finally, OLAP operations brings to the OLAP mining the advantage of interactive exploratory data analysis. Through the OLAP operations, roll-up, drill-down, slice and dice and pivoting, it is possible to select portions of interesting data, analyse data at different levels of abstraction, and display knowledge in different formats.

Knowing beforehand what type of knowledge to extract from a data cube is quite a difficult task to the user. With the integration of OLAP and data mining, it is possible to interleave cubing and mining functions to perform flexible mining and discover interesting knowledge in data cubes in a highly interactive way (Han, 1997):

- Cubing then mining. One can perform cubing operations to select the portion of the cubing to be mined and then apply a data mining technique. The data mining can be applied to any portions of the data cube.
- Mining then cubing. Data mining can be first performed on a data cube and then the result can be analysed further by performing cubing operations.
- Cubing while mining. By performing cubing operations during mining, the mining operations can be performed at different abstraction levels or on any portion of the cube.
- Backtracking. It should be possible to backtrack one or more steps in the mining process so the user can explore alternative mining paths and ease the interactive mining process. By jumping back a few steps in the mining process, one can consider other mining operations and analyse the alternative results.
- Comparative mining. One should allow comparative data mining, which means, comparing alternative data mining processes. It should be possible to compare side by side the mining techniques results, efficiency and other aspects.

The known set of OLAP mining techniques were developed based on the traditional data mining methods. These OLAP mining techniques consist in including features from analytical processing into mining techniques, so they can be applied to multidimensional data structures. Next, we will see how Han (1997) described OLAP mining techniques.

OLAP-based characterization

Data Characterization is used to generalize a set of task-relevant data based on data generalization. This technique is used to extract different kinds of rules of the data cube. The application of this method leads to the extraction of set of characteristic rules that summarizes the set of general characteristics of a set of user-specified data. This technique is useful to characterize a target class, like, for example, frequent customers. It allows for defining the characteristics of specific customers that can be summarized by a characteristic rule.

The characterization technique can be integrated with OLAP techniques, such as drill-down (progressive deepening) and rollup (progressive generalization). These operations help to discover the set of characteristics in different levels of abstraction. Progressive deepening (drill-down) starts with a high-level cuboid and then progressively specializes attributes to lower abstraction levels. This approach is considered to be the best one, since it starts by finding the general data characteristics at a high abstraction level, and then follows to interesting paths to drill down to specialized cases. This is the opposite of what happens with progressive generalization (roll-up), which, starting with a more conservative generalization process, first generalizes the data to a level higher than the existing in the primitive data cube.

OLAP-based comparison

Comparison is used to mine the set of discriminant rules that summarize the general features of a target class in order to distinguish that class from other classes. For example, mining a discriminant rule that summarizes the characteristics of a customer to distinguish that one customer from others. This technique is similar to the OLAP-based characterization, described before. However, OLAP-based comparison uses comparative measures to be easier to distinguish classes. It is implemented as follows. The set of relevant data in the database is collected and partitioned into a target class and one or more other classes. Then, an attribute-oriented induction is performed on the target class to extract a main cuboid. Then the set of contrasting classes are generalized to the same level as those in the main cuboid, forming the main contrasting cuboid.

Finally, the information contained in these two cuboids is used to generate quantitative and qualitative discriminant rules.

OLAP-based association

Association aims to extract, from a set of relevant data, a set of association rules at multiple levels of abstraction. Facing an OLAP environment, it is important to consider the dependencies between attributes within the same dimension and between dimensions. Therefore, there are two kinds of associations: inter-attribute association and intra-attribute association. The intra-attribute association is the association within one or a set of attributes formed by grouping of another set of attributes. On the other hand, inter-attribute association is the association among different attributes. For a better understanding, consider the example shown in Han (1997). Let the "course taken" relation in a university database be the following schema:

course taken = (student id; course; semester; grade).

For example, consider that one wants to know the associations created between each student and his/her course performance. The course performance consists of the grouped set of attributes "course", "semester" and "grade", creating a nested relation. This kind of associations is intra-attribute association. The set of extracted association rules will be of type:

course taken = (student id; course history)
course history = (course; semester; grade).

On the other hand, consider that one wants to know the association between course and grade, which are attributes in the same relation. This kind of association is an inter-attribute association.

OLAP-based classification

Classification aims to analyse a set of training data with a known class label and constructs a model for each class based on the data characteristics. A set of classification rules is generated and used for classifying future and unknown data. There are many classification methods. The most used include decision tree methods, like ID-3 or C4.5 (Quinlan, 1993). In an OLAP environment, the classification methods consist on four phases:

- i) collecting the relevant data and partitioning the data into training and testing data sets;

-
- ii) analysing the relevance of the attributes in the training set, which is made by determining how much an attribute is relevant to the class attribute; a generalization operation is also performed at this phase, allowing to classify the objects in the different levels of abstraction;
 - iii) applying the mining algorithm and creating the classification tree (also called decision tree) and finally;
 - iv) testing the effectiveness of the created model using the testing data set.

OLAP-based clustering analysis

Clustering consists in grouping a selected set of relevant data into a set of clusters. The clusters must ensure that the similarity between two different clusters is low and the similarity within the same cluster is high. The clustering process is based on a known mining methods, the k-means algorithm. OLAP-based clustering is performed using a k-means based methods. This kind of methods is considered promising due to their efficiency in processing large data sets. However, they are limited to numeric data. To overcome this difficulty, a method was implemented to encode concept hierarchies. With this, it is possible, for the adapted methods, to deal with large data sets with both numeric and categorical attributes. OLAP-based clustering can also be applied at the different levels of abstraction.

To conclude, when dealing with an OLAP environment and applying a data mining technique to a data cube, it is possible to extract preferences. Through the interpretation of the outcome of a mining technique, it is possible to discover which are the sets of "preferred" or "non-preferred" variables. This is an advantage of using OLAP preferences: it is possible to control the returned information. They provide access to relevant information and eliminate the irrelevant one. Consequently, extracting and using preferences leads to more focused and refined results.

4.3 OLAP Preferences Extraction Examples

Most of the research in OLAP preferences discusses how to help the user during the OLAP analysis session. OLAP preferences proved to be useful in guiding users in OLAP sessions, especially when they are not familiar with the database content, or even in guiding experienced users by controlling queries outcome, by filtering the returned information. OLAP preferences can help the user by

suggesting queries based on the past user sessions, or even by suggesting enhancing the current query. Next, we describe some situations in which some authors used the extraction of OLAP preferences to improve the OLAP cube navigation in OLAP sessions.

Aligon et al. (2011), for example, focused on deriving OLAP preferences using a mining technique. These authors present a proactive approach that integrates a MDX-based language and a mining technique for automatically deriving OLAP preferences. A log of past MDX queries is mined for extracting a set of association rules. The set of association rules is filtered with the support and confidence threshold values, to obtain a set of relevant association rules. The frequent query fragments are extracted. Based on a specific query made by the user, a subset of interesting rules is selected and translated into a preference: given a query q , the set of rules which has antecedent matches with q , is selected. The selected rules are then translated into a preference that is used to annotate the user's query.

In its turn, Jerbi et al. (2010) proposed a framework to suggest recommendation for OLAP analysis, presenting a context-aware preference model for helping users during OLAP analysis by proposing the forthcoming analysis step. These authors focused on context preferences. Context preferences are used to generate recommendations for the user. The set of user preferences are used to suggest relevant patterns to the user during analysis enhancing when possible the current analysis context. The user analysis is described by a graph, where a node represents an analysis context and the graph edges represent the operations from one context to another. The authors aim to recommend an analysis node within the user analysis graph. The recommendation process consists of two stages:

- recommendations building, where the recommendation system generates analysis nodes;
- recommendations ranking, where the candidate nodes are ranked considering the preferences scores, being only the top scored delivered to the user.

In 2011 Kozmina and Solodovnikova (2011) approached the generation of recommendations on reports using an OLAP reporting tool. The main goal was to determine and process user OLAP preferences explicitly formulated by users of the OLAP reporting tool. The process of processing the user preferences is divided in five phases:

-
1. Initial description of the preferences by the user: the user describes his/her preferences, amongst which the user chooses from a set of available terms in the OLAP reporting tool.
 2. Preference normalization by the system: the set of selected terms by the user is translated into concepts.
 3. Preference classification and re-formulation: the system is responsible for detecting the type of the user preference.
 4. Indication of preference importance: the user selects the degree of interest that should be assigned to each defined OLAP preferences, from very low to very high interest.
 5. Preference processing and generation of reports recommendations: after all the OLAP preferences are processed, the recommendations are finally suggested to the user.

The described OLAP preferences extraction processes differ in several aspects. The first one uses a mining technique for extracting preferences to help the user during an OLAP analysis session. The second focuses mainly on extracting preferences based on the analysis context. The last one describes how to extract preferences for generating recommendations on potentially interesting reports. These are some of many ways we have for extracting OLAP preferences.

4.4 Summary

This chapter presented the concepts of OLAP personalization and more specifically, OLAP preferences. OLAP personalization has been increasingly used, due to its importance. When dealing with large databases, it is important to guide the user in OLAP analysis session, helping him finding the most interesting information according to his needs or historical sessions. There are several ways for extracting preferences from multidimensional structures. OLAP mining is one of them. In this chapter, an overview of several techniques used in OLAP mining is provided. We can now aim at integrating the process of extracting preferences in a conventional What-If analysis.

Chapter 5

A Hybridization Process Based on Preferences

5.1 Methods and Methodologies

Etymologically, the Greek word "methodos" comes from "meta-" + "hodos". The element "meta-" means a search or a pursuit, and the element "hodos" literally means a path, track or road. Method has been used to mean a search of a way of accomplishing an objective. It has also been used to denote a particular procedure for accomplishing or approaching something, especially a systematic way. On the other hand, a methodology consists of a set of methods used in a particular area of study.

In this thesis, we use the term "methodology" in the sense defined in the Cambridge Dictionary, as "a system of ways of doing, teaching, or studying something", attending in particular to the meaning "a system of ways of doing something" (Cambridge Dictionary | English Dictionary, T., 1999). The methodology of integrating OLAP preferences in What-If Analysis is a general methodology, and can be implemented through various methods and using different technologies and tools, such as the tool for performing the simulation, or the technique or way preferences are extracted. Clearly other choices of tools and techniques are also possible.

To present our methodology, we selected Data Mining as a technique for extracting preferences. Within Data Mining techniques, we also opted for an Apriori-based algorithm (Agrawal and Srikant, 1994), which is an association rule technique, for discovering preferences from a multidimensional structure. We claim that this algorithm is the most adequate mining technique to extract preferences from a multidimensional structure. We explore the potential of data mining in generating OLAP preferences for designing a tool that combines What-If analysis with OLAP preferences. Furthermore, it is imperative to run a formal specification and validation of each phase of the process looking for defects or inconsistencies. Along with the detailed description of the process, we demonstrate how a formal specification and the consequent formal verification of the hybridizing process is made using formal methods - we specify our hybridization methodology as an abstract model in Alloy (Alloytools.org., 2019).

5.2 Integrating OLAP Preferences

In order to perform a What-If analysis it is important to be familiar with the business process or at least to know the content of the database in question. This is so, because one of the most critical issues when using What-If analysis is the choice of the set of business variables used as scenario (application) parameters to perform the What-If analysis. The outcome data depends on the input data. If the input data is irrelevant or not oriented to the goal analysis, the outcome may not be as valuable and suitable as if the input data was correctly chosen. For example, if one is willing to analyze the sales of product A of city X, the user should not select as input of the What-If simulation the sales of the stores of city Y, obviously.

The user lack of expertise can be an impediment during the What-If analysis process design and implementation. If a user is not familiar with the business, or even does not choose the most correct parameters in a particular application scenario, the outcome provided may not be the most adequate. Inadequate input data may result from:

- i) circumstances when the user selects data with noise;
- ii) the user selects the wrong set of business variables to include in the scenario, like for example, when the user forgets to add a needed business variable;

-
- iii) when the user selects a set of business variables that would not bring valuable information to the scenario.

Hence, there is a need to use some kind of technology for helping users to identify and select the most adequate scenario parameters according to their analysis goals. OLAP preferences are appropriate in this situation. Therefore, we developed a hybridization process as a solution approach, which will help to overcome the lack of user expertise. The main difference between our approach and the conventional What-If analysis is the introduction of a process of extraction of preferences of a multidimensional database and their usage before the simulation of the model, which allows for having the basis to predict the behavior of a given scenario. We use OLAP preferences to improve the conventional What-If analysis process and as a recommendation mechanism to help the user select the scenario parameters. OLAP preferences consist on information (patterns or knowledge) derived from the application of a data mining algorithm over a data cube. Preferences suggest to the user business variables that are strongly related to the previously defined goal analysis business variable in the What-If question.

As described in the previous chapter, the OLAP mining process, which consists in applying a data mining technique on a multidimensional structure, is one of many possible solutions that can be used to extract preferences and find out which are the “preferred” and “non-preferred” business variables. In our case, using an association rules algorithm as the basis of the mining process we can discover which are the business variables that are related to the goal analysis business variable. Through the interpretation of the outcome of the mining algorithm, it is possible to discover the “preferred” set of parameters by pointing out the business variables that are strongly related to the goal analysis business variable.

OLAP preferences help figure out which other attributes are strongly related to the main goal analysis attribute, providing exactly the relevant and useful information to each specific case, depending on the goal scenario analysis. This is an advantage of using OLAP preferences: it is possible to control the returned information. Recommendations help to introduce valuable information to the scenario analysis, which otherwise may not happen, providing access to relevant information and eliminating irrelevant one. One may not know the proportions of the outcome; it may be an empty result, or an information flooding. Consequently, in our process, we get more

focused, refined results. It helps both a user who is not familiar with business analysis and an analyst who is familiar with the business modeling data.

As seen before, OLAP preferences can be obtained using On-Line Analytical Mining (also called OLAP mining) (Han, 1997). The multidimensional structure that contains the historical data, which is used to be analyzed using What-If analysis, is mined. OLAP mining is a mechanism which integrates OLAP and data mining. This means that a data mining technique is applied to a part of the multidimensional structure, in this case, leading to the choice of an association rules technique to perform the mining task. This will allow to discover hidden patterns in the historical data cube, revealing relationships within the data. This strategical component of the hybridization process is drafted in Figure 11.

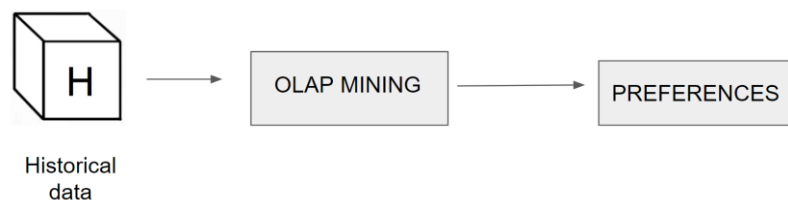


Figure 11 - A strategical issue: using preferences obtained with OLAP mining.

The set of association rules, which is the result of the application of the mining algorithm to the OLAP cube, will form the set of preferences. These preferences correspond to a set of business values strongly related to the goal analysis business variable. With this recommendation, the user only has to choose parameters that are strongly related to the goal analysis business variable, avoiding selecting and inserting all the business values as parameters into the What-If scenario and eventually uninteresting data. With this, the returned scenario is a more refined scenario, including less (noisy) data and more valuable information. With improved What-If scenarios, we may improve query time response, provide appropriate views, and help the user deal with large databases. The main purposes of these systems are to improve decision-making and to manage knowledge. They help to obtain a solution for problem reasoning about knowledge in a large database. Consequently, we can get a significant reduction of the cube processing time, computation costs and memory usage derived using OLAP preferences. Only needed information is returned as outcome, which means that cube materialization and computation times are reduced.

The multidimensional structure is a very complex structure, and it may be difficult for the analyst to acquire the needed information. With a simple interface that recommends queries based on the past data, the whole process is much easier and less complex for the user. Due to this, query runtime can be enhanced against processes that do not use preferences.

5.3 Overview of the Hybridization Process

The hybridization process is shown in Figure 12. An OLAP data cube with historical data is used as input. An OLAP mining process is performed for extracting the set of OLAP preferences and discover the set of business variables strongly related to the goal analysis business variable. This step is taken according to the content of the What-If question and the goal analysis business variable. The set of variables is then suggested to the user and it is the user responsibility to choose which variables should be added to the simulation model. Next, the user chooses the scenario input data from the suggested set of business variables and sets the scenario parameters.

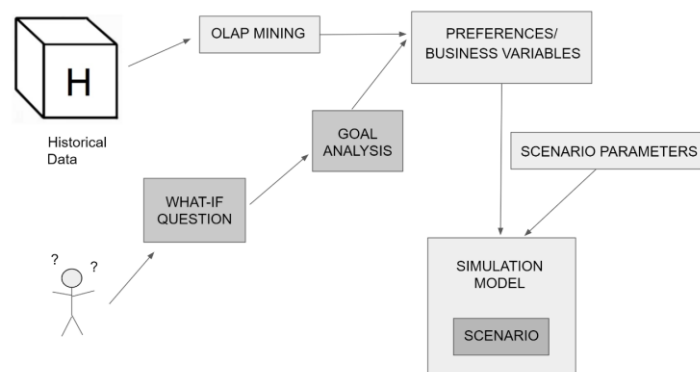


Figure 12 – The hybridization process.

Next, the user chooses the scenario parameters, which is essential in the What-If simulation process: select the axis of analysis and the set of values to analyze and change and change them according to previously defined goals. Next, the application processes the What-If analysis, changing the variables values of the historical scenario and get the new scenario (prediction scenario) as seen in Figure 13. It is required to have an appropriate tool (a What-If scenario analysis tool) to run a simulation model based on What-If analysis.

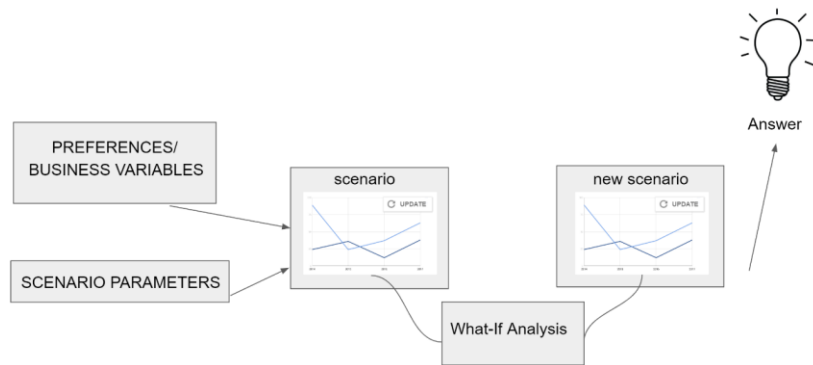


Figure 13 – A general overview of the simulation model.

The What-If analysis scenario tool calculates and allows for the user to explore and analyze the impact of the changed values on the entire scenario. Using OLAP in this process comes as an advantage. Decision makers acquainted with the navigation of multidimensional data within OLAP cubes can interactively try different scenarios and compare predictions, mixing navigation of historical data and simulation in a single session of analysis. It is the user's responsibility to accept the new data cube or to return to change the scenario setting, and to repeat the previous steps and make new changes in the variables, as shown in Figure 14.

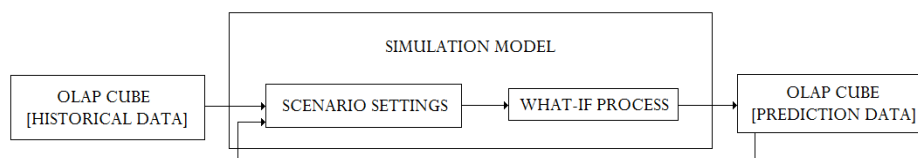


Figure 14 - A general overview of a What-If analysis process.

Summarizing, we want to discover the best recommendations for What-If analysis scenarios based on past analysis. This process consists in integrating OLAP and data mining, also called OLAP mining, which consists in apply an association rules algorithm to an OLAP cube; then, defining preferences using the retrieved association rules, and suggesting them to the user as What-If scenario parameters.

5.4 The Methodology

After the overview of the hybridization process we propose, it is time for describing a methodology we suggest that should be followed when dealing with What-If-based problems.

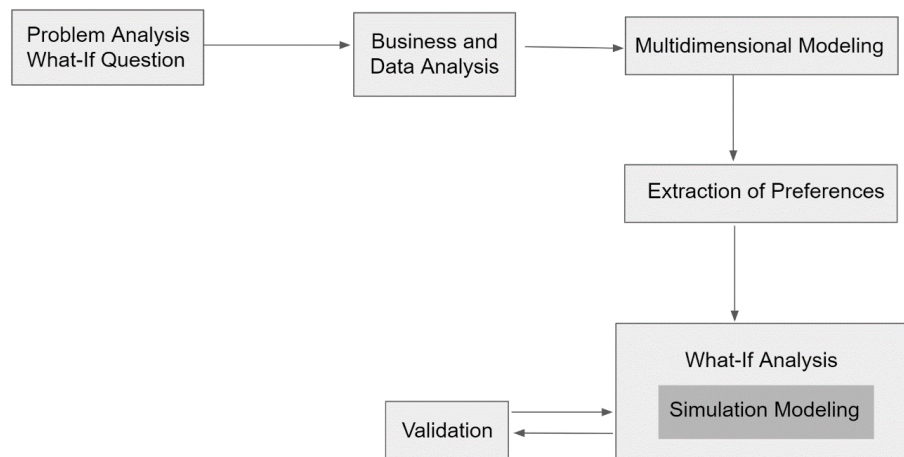


Figure 15 - Methodology for the hybridization process.

Our methodology is composed by six phases, as shown in Figure 15:

1. **Problem Analysis and Definition of the What-If question.** It starts when a doubt arises, forming a What-If question. A What-If question translates in a question about what can happen in a specific hypothetical scenario and the consequences of changing variables. In this phase, it also needed to define the goal of analysis and the set of business variables to add in the simulation. It is necessary to know the context of the problem to create the What-If question. For example, if an analyst wants to explore what will be the effects of the change of the profit value (increased by 10%) of red and blue products of 2016. The What-If question should be "What if we want to increase the overall profit in 2016 by 10% of the products with blue and red colors?". The goal analysis in this case is product color and the analyst also need to add to the simulation the parameters year and profit values.
2. **Business and Data analysis.** The user needs to perform an analysis of the business and data. One should know which is the set of business variables to be included in the

simulation model and their associations, identifying the dependent and independent ones. The relevant data sources need to be analysed to understand which set of data needs to be added to the simulation. One should take into consideration the quality of the data: if the simulation data has noise, the outcome of the simulation could not be the most adequate.

3. **Multidimensional Modelling.** In this phase, the data structure is prepared to extract the preferences. The multidimensional structure is constructed based on the information collected in the previous phase and the goal analysis defined in the first phase.
4. **Extraction of Preferences.** There are several ways to extract preferences, in this case, a mining technique is applied to the created multidimensional structure - this process is called OLAP mining. Then, a filter process is applied to the outcome of the association rules technique. This filter process consists in filtering the data that is interesting to the user and should be included in the simulation. To do this, it is necessary to filter the set of association rules and return only the set of strong association rules that contain the goal analysis business variable. In the end, this process suggests the user a set of variables, which are strongly related to the goal analysis, to introduce in the simulation model.
5. **What-If Analysis simulation.** Here, the user performs the What-If simulation. To perform the simulation the user needs to use an appropriate tool. The user introduces a set of scenario settings: source or business variables and scenario parameters. The set of business variables includes the goal analysis business variable (the focus of the analysis defined in the first phase) and a set of recommended parameters (which are derived from the extracted preferences of the fourth phase). The set of scenario parameters, as seen before, depends on the tool. The set of scenario parameters that are introduced according to the chosen tool, like the algorithm and additional parameters.
6. **Validation and Implementation of the decisions.** Finally, the user evaluates how credible and practicable is the simulation model created. The user needs to compare the results of the simulation model with the real business model outcome and to evaluate if the behaviour of the simulation model is adequate. If the simulation outcome is irregular or unacceptable, the user needs to go back and to redefine the simulation model.

5.5 A Formal Specification

Before describing the hybridization process in detail, we address the issue of validating the process. It is imperative to run a formal specification and validation of each phase of the process to check for defects or inconsistencies.

Formal methods (Clarke and Wing, 1996) are mathematically-based techniques that provide an environment and tools that enable users to specify, verify and analyze models. Formal methods reveal ambiguities, incompleteness and inconsistencies in a system. Formal specifications are not the system implementation, but they describe what a system should do, not how it should do it. Given a formal specification, it is possible to use techniques to demonstrate that a system design is correct according to its specification. In the following section, along with the detailed description of key phases of the process, we demonstrate how a formal specification and the consequent formal verification of the hybridizing process is made using formal methods.

We chose Alloy (Alloytools.org., 2019) to specify our hybridization process as an abstract model in Alloy. Alloy allows for producing an abstract model of a system, which is a representation of the real system and makes it easier to evolve or expand on in the future. Alloy is a formal object-oriented modeling language based on first-order logic, which makes it analyzable and gives a mathematical notation for specifying objects and their relationships. An Alloy model may contain signatures, relations, facts or predicates. Alloy allows creating models that can be automatically checked for correctness using its own analyzer, Alloy Analyzer.

The Alloy Analyzer has been built as a model finder built upon a boolean satisfiability (SAT) solver (Moskewicz, et al., 2001). This language was chosen due to its ability for generating an initial model, which becomes more robust and complex as the project evolves. We can use Alloy as a modelling language for specification and the Alloy Analyzer, which provides graphic instant feedback, for verification. We can run the specification model, or we can check an assertion by looking for counterexamples. The main commands in Alloy are defined as follows:

- **Definition 6.** Signature (**sig**) represents one or more sets of atoms and their relations to other sets.
- **Definition 7.** Function (**fun**) represents a way of getting a relation (or set, or atoms). It can take one or more parameters and produces a parameter.

-
- **Definition 8.** Facts (**fact**) defines a formula that is valid (always true).
 - **Definition 9.** Predicate (**pred**) command is similar to a fact: defines a formula that describes something true but is only verified when invoked (unlike facts which are always true in returned instances).
 - **Definition 10.** Run (**run**) command is used to invoke a predicate in a specified scope and finds out for which the predicate is true. If it finds an example, then the predicate is valid; if it finds no examples, the predicate may be invalid, or may be valid but not within the chosen scope.
 - **Definition 11.** Assertion (**assert**) defines a formula that is consider valid (always true). Unlike facts, assertions are checked to verify if they are true for all examples in a specified scope.
 - **Definition 12.** Check (**check**) command is used to invoke an assertion and possibly find out an counter-example.

With the defined commands, the Alloy Analyzer returns an example of an instance consistent with the defined specifications. We start by defining an empty predicate, which is often a useful starting point to determine whether the model is consistent or not, in other words, if the Alloy Analyzer can find an instance of the model that satisfies the specified facts.

5.6 Describing the Key Phases

The key phases of the methodology we propose involve the extraction of OLAP preferences, and their use for enriching What-If scenarios. They are the phases 3, 4 and 5, Multidimensional Modelling, Extraction of Preferences and What-If Analysis simulation, respectively. In this section, we describe in more detail these phases, focusing in 5 underlying steps that belong to the phases as follows:

- Multidimensional Modelling:
 1. Selection of the Data Warehouse's view.
 2. Construction of the OLAP cube.
- Extraction of Preferences:
 3. Extraction of Association Rules of the OLAP cube.

-
4. Extraction of usage Preferences of the Association Rules.
 - What-If Analysis simulation:
 5. Performing What-If analysis using suggested preferences.

We used Microsoft SQL Server Management for importing the database and Microsoft Visual Studio 2017 for selecting the Data Warehouse view, constructing the OLAP cube, creating the mining structure and finally extracting the association rules. Microsoft Visual Studio is also used to create the filter process to extract the OLAP preferences from the association rules. To support and perform What-If analysis process we choose Microsoft Office Excel which will be approached in the next chapter.

5.6.1 Selecting Views in the Data Warehouse

We start with a view selection process over the data warehouse we chose to support our work. In this step, we select the data, meaning that we select the tables that contain the information that is relevant to the goal simulation. A data warehouse (Kimball and Ross, 2011) is a repository that aims to store data in an integrated and consistent, subject-oriented, time-variant and non-volatile collection of data in support of decision-making, which makes it an ideal foundation to support decision-making processes. Data Warehousing consists of a set of decision support technologies that help to make better and faster decisions. It has demonstrated its importance over the years, providing reliable and consistent information to companies. Companies create their own data warehouses with business data with the intention of providing managers and decision makers with a global view of the organization and helping them in the decision making process. Data Warehousing technologies have been successfully deployed in many areas, like manufacturing, retail, financial services and others.

A typical Data Warehouse schema for representing a multidimensional data model is represented in Figure 16. It is a typical star schema, the most regular organization of data elements in a data warehouse. This schema is composed by a central table, called fact table, and a set of tables linked to the main table, called dimensions. A fact table can have two types of columns: keys and measures. Fact table's keys are foreign key (FK), represented by *FK1*, *FK2* and *FK3*. Instead of dimensions that have primary keys (PK), *PK1* in *Dimension 1*, *PK2* in *Dimension 2* and *PK3* in *Dimension 3*. FKs links fact table's rows to the correspondent dimension table data, which means,

FK1 in the Fact Table is linked to *PK1* in *Dimension 1*, *FK2* in the Fact Table is related to *PK2* in *Dimension 2*, and so on. Usually, measures are numeric values that allow mathematical operations and are used to express business metrics. Dimensions also contain attributes.

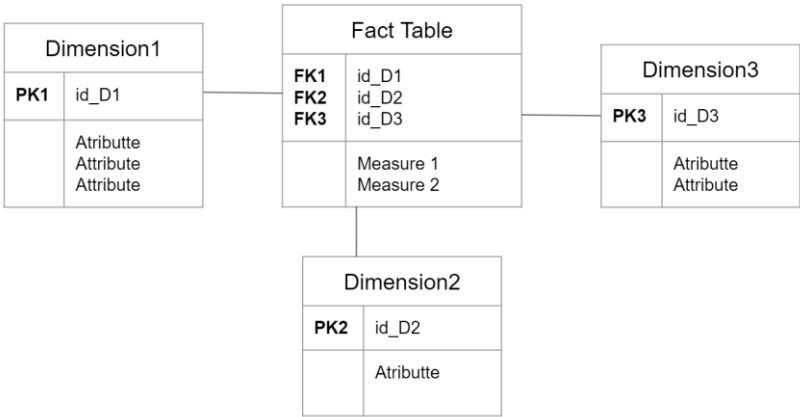


Figure 16 - An example of a star schema.

We start by formally specifying a view of a Data Warehouse (Figure 17) describing elements like fact tables, dimensions, measures and attributes, according to the relation' properties between each other. Tables (abstract signature `Table`) can either be fact tables (`FactTable`) or dimensions (`Dimension`), and the assigned fields (`flds`) represent their records (`Field`, each one forced to belong to exactly one table by the signature constraint). A Fact Table receives the numerical performance measurements of the business and is composed by primary keys (usually a set of foreign keys that are related to dimensions) and numeric values, here represented by signature `Measure`. Each row in a fact table corresponds to a measurement event and every foreign key in the fact table has a match to a unique primary key in the respective dimension. A dimension contains the textual descriptors of the business, here represented by signature `Attribute`. We specify that a fact table can be related (`rels`) with the other existing dimensions (but not with itself).


```

abstract sig Table {
  rels: set Table, flds: some Field
} { this not in rels }
sig FactTable extends Table {} { flds in Measure }
sig Dimension extends Table {} { flds in Attribute }
abstract sig Field {} { one this.~flds }
sig Measure, Attribute extends Field {}

```

Figure 17 – Alloy specification: Definition of Table (Fact Table and Dimension) and Field (Measure and Attribute)

5.6.2 Construction of the OLAP cube

Data warehouse supports OLAP. OLAP consists of a set of techniques developed for analyzing data in data warehouses. Therefore, we aim to create a data cube structure, also called OLAP cube or multidimensional cube, to analyse the data. Using the data view described above, "Selection of the Data Warehouse's view", we create and analyze the multidimensional data cube according to the schema presented in Figure 18.

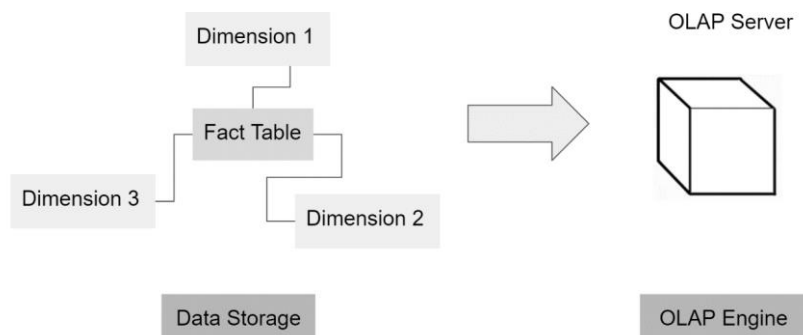


Figure 18 – Construction schema of an OLAP cube.

The data cube is a multidimensional database, in which each cell within the cube structure contains measures (numerical values). Each one of the cube axes represents the values of each of the available dimensions (Figure 19).

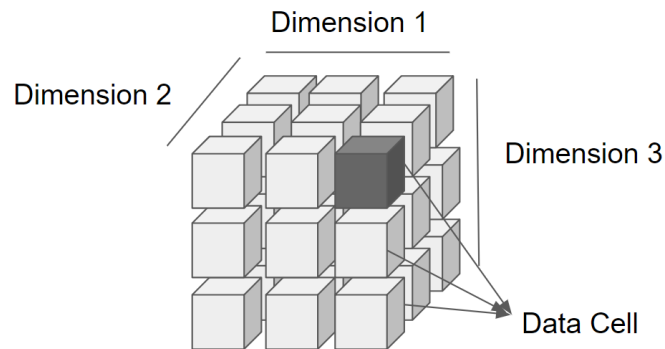


Figure 19 - Concepts of dimensions and data cells in a multidimensional structure.

To specify the multidimensional structure of our abstract model, we need to formally specify several elements (Figure 20). The cube (`OLAPCube`) is constructed using the elements of the data warehouse, and defined by `fields` that are either measures or attributes. The cube construction is specified using the predicate `ConstructCube`, which constructs an `OLAPCube` provided a representation of user parameters for its creation (`CubeParams`), which are the tables selected to generate the cube. The fields of every selected tables are assigned to the fields of the cube.

```
one sig CubeParams { tabs: set Table }
one sig OLAPCube { fields: set Field }
pred ConstructCube[c: OLAPCube, p: CubeParams] {
  c.field = p.tabs.flds }
```

Figure 20 – Alloy specification: Definition of Cube parameters, OLAP Cube and predicate `ConstructCube`

To illustrate all the concepts we use, we chose an example of a cube with three dimensions, in order to be able to represent the cube structure graphically and to accommodate some complexity that does not exist in a data warehouse with just two dimensions. Figure 21 represents an example of a view of a data warehouse, where the granularity can be represented by 'sales information about one specific product that was sold in one specific city at a specific month in 2018'. Let Dimensions 1, 2 and 3 be dimensions "Time", "Product" and "Location", respectively. The fact table

holds information about the sales of the products sold in three different cities in the EUA during three months. The dimension "Time" contains information about the calendar to register when the sales are made. Dimension "Location" contains information about the stores' location and finally Dimension "Product" contains information about the available products for sale. The Fact table contains information about the sales transactions that were made, for example, the first fact table record means that in 'February 2018', a 'tape dispenser' was sold in 'New York' by US\$ 23, the second record means that in 'March 2018', a 'bubble wrap dispenser' was sold by US\$ 50 also in 'New York'.

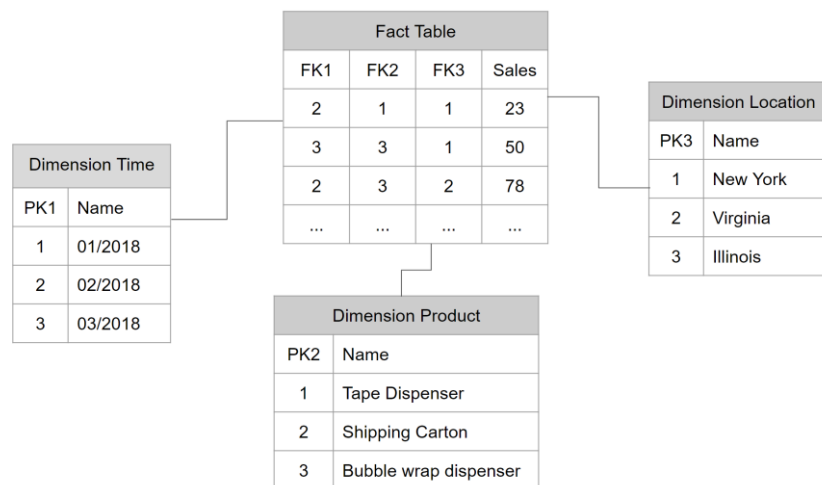


Figure 21 - Representation of a data warehouse's view.

In the OLAP server, the information represented in the Figure 21 will be used for building a data cube similar to the data cube represented next in Figure 22. The values of the three dimensions "Time", "Product" and "Location" are represented in the three axes of the cube and the data cells contain information about the sales values (represented by the measure value in the Fact table). The built cube can also be a sparse cube (Beyer and Ramakrishnan, 1999), which means that there can be data cells that do not contain data which means that for a given combination of dimension values there is no data.

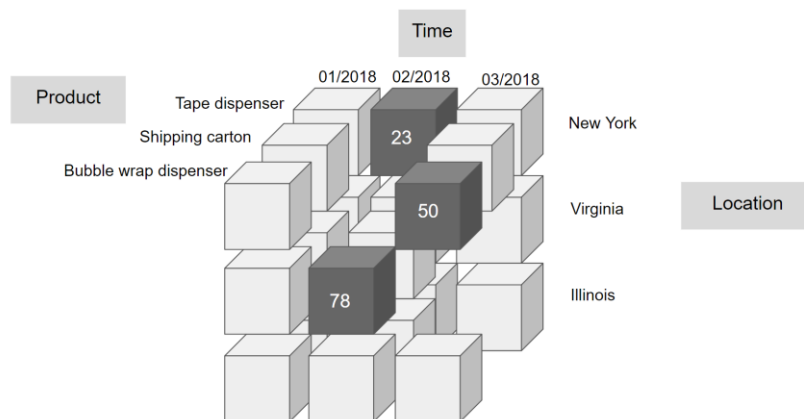


Figure 22 - Example of a multidimensional database – a cube.

The use of multidimensional database is more advantageous comparatively to using traditional relational data bases in this analysis; it is possible to analyze data in different levels of abstraction, to see the output of the queries in different formats, to perform operations like navigation of data, and, as stated before, many OLAP operations including roll up (decreasing detail or increasing the level of aggregation), drill down (increasing detail or decreasing the level of aggregation), slice and dice (selection and projection) and pivot (re-orienting the multidimensional view of data). Roll up and drill down are OLAP operations that allow to analyze data in less or more detail, respectively. Also, another advantage is to be able to apply mining to the data cube through OLAP mining. OLAP mining is a mechanism with integrates on-line analytical processing with data mining so that mining can be performed in different parts portions of databases or data warehouses and at different levels of abstraction (Han, 1997).

5.6.3 Extraction of Association Rules of the OLAP cube

The next step is to perform OLAP mining in order to extract the set of association rules. As seen before, OLAP mining consists in applying a mining technique to an OLAP Cube, aiming at finding correlations between the variables in the data base. In the methodology we proposed, we chose to use an Apriori-based algorithm (Agrawal and Srikant, 1994), which is an association rule technique, for extracting OLAP preferences from the multidimensional structure. This algorithm is the most adequate mining technique to identify OLAP preferences from the multidimensional structure. It fits well on mining process that involves recommendation engines or processes for finding

correlation between attributes in a dataset, meaning, in our case, between business variables. In our methodology, we suggest to the user a set of items that are most likely to appear together in a particular What-If scenario search. The outcome of the application of the association mining techniques is a set of rules: $X \rightarrow Y$ (i.e., if X happens, then Y is likely to happen in the same transaction).

A mining structure is a data structure that contains the data in which mining models are built. The mining structure contains information about the data like, which attributes were chosen for the mining structure and their type of data. The mining structure is defined using the existing data source view and it can support multiple mining models. The mining models from the same mining structure can use different attributes from the structure, for example, from the same mining structure we can create two separate clustering and associate models.

In our methodology, an Association Rules algorithm is the most adequate data mining technique: apart from Classification-based algorithms, more specifically Microsoft Decision Tree Algorithm, which the main goal is to predict class labels in datasets using classification or regression. This algorithm creates a structure, called a decision tree, which is a predictive model. This model makes predictions based on the relationships between the input data. This type of algorithms is useful when one has a particular label (or target value) and wants to create a model that helps to predict the value of a label based on the characteristics on the data features. And apart from Clustering-based algorithms that are responsible for grouping elements of the same dataset into different clusters based on the similarity of the elements. This kind of algorithms are usually used for exploring data. We chose an association-based algorithm for helping to refine our recommendation engine, suggesting business variables that are most likely to appear together in a specific analysis session.

Before describing how the algorithm works, it is necessary to define formally a transaction and a transaction database:

- **Definition 13.** Let $I = \{a_1, a_2, \dots, a_n\}$ be a set of items in the database. A **transaction** T is a set of items such that $T \subseteq I$.

-
- **Definition 14.** Let I be a set of items and T a transaction. A **transaction database DB** is a set of transactions $T_i, i \in [1..n]$.

Association analysis is a rule-based machine learning method which aims for discovering correlations between variables in large databases. The discovering process of the association rules can be divided in two distinct phases:

1. **Frequent Itemset Generation**, in which the main goal is to find the combination of available items that satisfies the minimum support threshold value.
2. **Rule Generation**, in which occurs the extraction all the high value confidence rules from the discovered frequent itemsets found in the previous step.

First, the Apriori algorithm (Agrawal and Srikant, 1994) generates and then counts candidate itemsets. Let I be a set of all available items in the database. A set with zero items is called a null or empty set. A k -itemset is a set of k items. The algorithm aims to find the list of k -itemsets that are frequently found together. For each itemset, the algorithm calculates its support. The support count, $\sigma(X)$, is a measure of interest which refers to the number of transactions that contain the same itemset. Mathematically, the support count (1), $\sigma(X)$, for an itemset X can be stated as follows:

$$\sigma(X) = \{t_i | X \subseteq t_i, t_i \in T\} \quad (1)$$

The k -itemsets that have support values above the minimum support threshold value (minsup), pre-defined by the user, are called frequent k -itemsets. So, to discover the frequent itemsets, the algorithm calculates the support value of a k -itemset, and if it is higher than the pre-defined minimal support threshold value, then it is considered a frequent itemset. For example, if the support value of the 3-itemset $\{A, B, C\}$ is higher than the pre-defined minsup, then this 3-itemset is a frequent itemset. The Apriori principle says that if an itemset is frequent, then all of its subsets, $\{A\}$, $\{B\}$ and $\{C\}$ must also be frequent.

The second phase of the algorithm consists in deriving the association rules and discovering the strong rules. Strong rules are association rules with meaning and interest to the user based on his

analysis our research. The set of strong rules is filtered from the discovered association rules using the pre-defined thresholds, minimal support (*minsup*) and confidence (*minconf*) defined by the user. The support value of the rule is calculated, and if it is higher than the pre-defined minimal support threshold value, then the rule is a strong rule. Otherwise, the rule is discarded. The same process happens to the confidence measure. The confidence of the rule is calculated, and then if it is lower than the pre-defined minimal confidence threshold value, the rule is discarded.

Let $I = \{i_1, i_2, \dots, i_n\}$ be a set of frequent items. Association rules are created using all the possible combinations of the all frequent items i in I . Each frequent k -itemset, X , can produce up to $2^n - 2$ association rules (Agrawal and Srikant, 1994). An association rule is an implication $X \rightarrow Y$, where X and Y are disjoint itemsets, which means that $X \cap Y = \emptyset$. X is the antecedent and Y is the consequent (i.e., if X happens, then Y is likely to happen in the same transaction). An association rule can be extracted by partitioning the itemset X into two non-empty subsets, X and $X - Y$ (i.e., Y minus X), such that $X \rightarrow Y - X$ satisfies the *minconf* threshold.

The strength of an association rule can be measured with support, confidence and lift values. The support determines the fraction of transactions that satisfy the rule. It is an important and relevant measure, especially in what concerns discarding uninteresting rules. If a rule has a low frequency and consequently low support, it means that rarely (or seldom) occurs in a data set and is likely to be uninteresting to the analysis. Confidence (also referred to as probability) measures the conditional probability of Y given X . In other words, it calculates the probability of having the itemset Y present in the transactions that contain the itemset X , in $X \rightarrow Y$. The probability describes how likely the result of a rule occurs. Finally, the importance (also known as lift) is calculated as the probability of the itemset divided by the compound probability of each item in the itemset. A rule's lift measure is calculated by the log likelihood of the right-hand side of the rule, given the left-hand side of the rule.

At this stage, all the rules and frequent itemsets extracted are stored in the mining model. An association mining model is a simple structure organized in two blocks:

- 1) the information about the mining model itself and its metadata;
- 2) 2) a flat list containing information about the frequent itemsets and the rules.

Itemset nodes include information of the itemset like, description of the itemset, number of cases that contains the itemset, and other diverse information for support. In turn, a rule node describes a general pattern for the association of items. Every node has detailed information about the itemset or the rule that will be relevant in the next steps of the process. All this information is used for defining OLAP preferences on the fourth stage of the methodology.

At this phase, we need to formally specify several elements (Figure 23). With the exception of the Apriori algorithm, simply because this is an already validated algorithm. Therefore, it is not crucial if the algorithm is not described in detail in the Alloy: only the main steps in the algorithm will be represented in the Alloy. We only describe the main steps in the algorithm in the Alloy specification: applying the mining rule algorithm to the cube structure and return a set of rules. The outcome of the Alloy Analyzer in these circumstances does not influence our final goal. We define the mining structure and the mining model to support a mining association process that runs over the cube and retrieves the rules. The `MiningStructure` defines the data from which mining models are built and the resulting `MiningModel` (`mdl`), created by applying an association rules algorithm to data. This model consists of a set of `rules`, each denoting a logical implication, a rule $X \rightarrow Y$ meaning that if X occurs, then it is likely that Y also occurs. Each rule can be either $(A \times A \rightarrow A)$ or $(A \rightarrow A)$. The antecedent can be one or more fields (`is`) and the consequent represent a single one (`o`). Each rule is related to a pair of (positive) performance measures (support `supp` and confidence `conf`), which help to identify which rules are relevant. To guarantee that every valid rule is created, signature `SubsetField` is defined to represent the powerset of all available fields.

```

one sig MiningStructure { cube: one OLAPCube, mdl: one MiningModel }
one sig MiningModel {
  rules: set Rule, strongRules: set rules, prefs: set Preference }
sig Rule {
  is: some Field, o: one Field, performance: one Performance }
sig Performance {
  supp: Int, conf: Int } { gte[conf,0] && gte[supp,0] }
sig SubsetFields {
  elems: set Field
} { all s : SubsetFields | s.@elems = elems => s = this }

```

Figure 23 - Alloy specification: Definition of Mining structure, Mining model, Rule, Performance and SubsetFields

The predicate `ConstructRules` (Figure 24) specifies the creation of the rules given a cube and its parameters by creating all the combination of the rules using the cube's fields of the tables in the parameters. The function `reach` helps finding all the fields that are reachable within the OLAP cube given the tables selected in the parameters.

```

pred ConstructRules[c: OLAPCube, p: CubeParams] {
  ConstructCube[c,p]
  all f: c.fields, s: SubsetFields |
    some s.elems && s.elems in related[f,p] =>
      some r: c.~cube.mdl.rules | r.o = f && r.is = s.elems
  all r: (c.~cube.mdl.rules) {
    some f: c.fields | r.is in related[f,p]
    all r': c.~cube.mdl.rules - r | r.is != r'.is || r.o != r'.o } }
fun reach[f: Field, p: CubeParams] : set Field {
  ((f.~flds&p.tabs).*(p.tabs <: rels :> p.tabs)).flds - f }

```

Figure 24 - Alloy specification: Predicate `ConstructRules`

5.6.4 Extracting Usage Preferences

The main goal of the extraction of preferences step is to discover the best recommendations for What-If Analysis scenarios based on the analysis of historical OLAP sessions. In this step, OLAP preferences, which are derived from the association rules, are used for defining recommendations to the users. The set of strong rules helps in understanding the co-relation between business variables and in understanding which business variables come along together in the analysis made by a specific user. With this, it is possible to recommend to the user the axes of analysis that are strongly related to each other, helping him to introduce valuable information in the application scenario he is building. This process of prediction involves finding the set of business variables strongly related to a goal variable or relevant to the user; or predicting the business value or values based on the set of data similar do the studied historical dataset.

The main goal in this step is to determine what is the set of business variables, which will be included in the What-If scenario. The process of extracting OLAP preferences runs as depicted in Figure 25. It starts with a What-If question defined by the user. As stated before, the What-If question is composed by a set of business variables and scenario parameters. Interpreting a What-

If question, we can extract different types on information relevant to the analysis: the goal analysis attribute value, which is a business variable used to filter the association rules to get the set of preferences; and the specific attribute, which is a business variable that the user intents to alter and explore the effects of that change. The goal analysis attribute is used in a filter process to filter the association rules extracted from the OLAP cube using an OLAP mining technique.

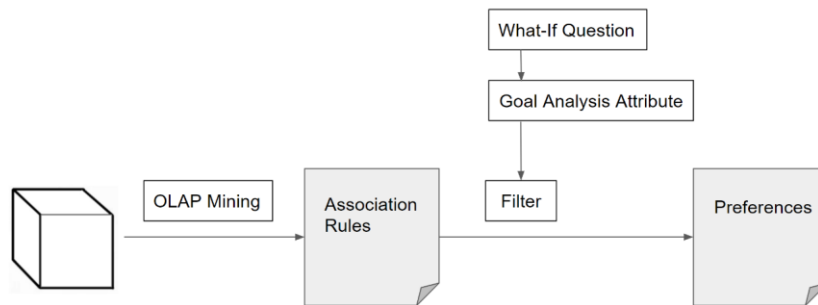


Figure 25 – The extraction process of OLAP preferences.

The goal analysis attribute is used as a filter in the set of extracted association rules. This filter process is divided in two phases (Figure 26), the former consists in extracting the set of association rules that contains the goal analysis attribute and order them to get the set of strong association rules; the latter consists in decomposing the return set of strong association rules and forming the set of preferences.

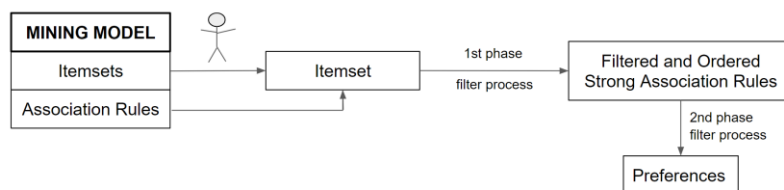


Figure 26 - Filtering association rules.

Firstly, the association rules technique returns a mining model with a list of item sets and association rules related to the user's main research focus the user intends to analyze. Then, the application displays to the user the mining model content obtained with the Apriori-based algorithm: the frequent itemsets and the extracted set of strong association rules. The user chooses the goal analysis attribute within the set of available itemsets, returning only an item set. The application takes the set of association rules of the mining model and returns only the association rules that contains the returned item set, which represents the chosen goal analysis attribute. This set of association rules that contains the goal analysis attribute is ordered by decreasing values of confidence and support. And according to the pre-defined by the user confidence and support threshold values, this set of filtered association rules is filtered again returning only the set of strong rules. With this, it is easier to see which rules are frequent and more relevant to the user.

Next in a second phase of the filtering process, the filtered and ordered strong association rules set is divided into 1-itemset and used to form the set of OLAP preferences to the user. The set of rules is partitioned and divided into sets of 1-itemsets, more specifically, the 1-itemset of each one of the filtered and strong association rules are suggested to the user as preferences to become parameters of the What-If scenario. It may happen that the application returns, in a first phase of the filter process, strong association rules (meaning that they are relevant and frequent according to the support and confidence values), but not relevant to the user analysis. Therefore, and accordingly to the business and his/her needs, the user may choose the set of association rules that he/she thinks better fits the needs.

Due to the high support and confidence values of the strong association rules and the fact that all the remaining rules contain the goal analysis attribute, we end up with association rules with relevant content that will be useful to the scenario and may add up important information to answer the What-If question. Meanwhile, the association rules with irrelevant information were discarded in the previous steps. Decomposing the strong rules, we get the set of itemsets that is strongly related to the goal analysis attribute.

To formally specify the extraction of preferences in our abstract model, we start by specifying the strong rules. Strong rules must be selected from all the rules and item sets extracted from the OLAP cube through the OLAP mining process (Figure 28). To accomplish this, rules are filtered

using performance measures' thresholds and an attribute chosen by the user and stored in the mining model (`strongRules`). These user parameters are encoded by `PrefParams` (Figure 27).

```
one sig PrefParams {  
    conf: one Int, supp: one Int, attr: one Attribute  
} { gte[conf,0] && gte[supp,0] }
```

Figure 27 – Alloy specification: Definition of `PrefParams`

```
pred ConstructStrongRules[c: OLAPCube, p: PrefParams] {  
    c.~cube.mdl.strongRules = { r : c.~cube.mdl.rules |  
        p.attr in (r.is + r.o) && gte[r.performance.supp,p.supp] &&  
        gte[r.performance.conf,p.conf] } }
```

Figure 28 – Alloy specification: Predicate `ConstructStrongRules`

Strong rules allow us to acknowledge which attributes are strongly related with the chosen attribute. Preferences are built by merging the set of strong rules' attributes (Figure 29). A `Preference` is characterized by a set of fields (`atts`) and a source strong rule (`srcRule`). Preference generation is threefold (`ConstructPrefs`) (Figure 30).

```
sig Preference {  
    atts: set Field, srcRule: one Rule }
```

Figure 29 - Alloy specification: Definition of `Preference`.

```
pred ConstructPrefs[c: OLAPCube] {  
    all p: c.~cube.mdl.prefs | p.atts = p.srcRule.(is+o)  
    all p1,p2: c.~cube.mdl.prefs | p1.srcRule = p2.srcRule => p1 = p2  
    c.~cube.mdl.strongRules = c.~cube.mdl.prefs.srcRule  
}
```

Figure 30 - Alloy specification: Predicate `ConstructPrefs`.

Let us analyse how this works. We want to discover the strongly related attributes with the goal analysis attribute represented in the What-If question. The process of extracting OLAP preferences starts with the choice of a user preference itemset from a list of frequent itemsets. This means that the user chooses the goal analysis attribute according to the What-If question. For example, 'What (will happen to the business variable A_1) if we increase the total sales value by 10%?'. Here, the goal analysis attribute is the business variable A_1 . Then, the user chooses attribute A_1 and, consequently, A_1 is preferred to the attribute A_2 , A_3 , A_4 , A_5 , A_6 , and so on (Figure 31). All the association rules returned by the previous phase are filtered using the goal analysis attribute (A_1), represented by the "filter{ A_1 }" and a list of rules is created. All of the rules of this returned list contain the chosen attribute, attribute A_1 . With this list, we can show which attributes are strongly related with the chosen attribute, since this list only contains rules with the higher performance measures. This list is then used then to form the set of OLAP preferences for the user.

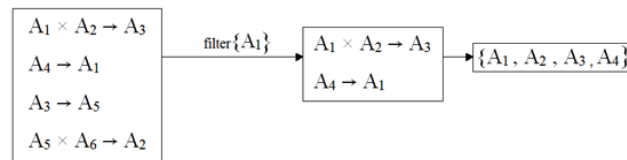


Figure 31 - Example of filtering association rules.

At this phase, we show how we can form OLAP preferences from the extracted set of association rules. The OLAP cube with the historical data is mined and we get the mining model with the set of frequent itemsets and association rules. The available frequent itemsets are presented and the user has the responsibility of choosing the itemset that corresponds to the goal analysis attribute. This step should be in accordance with the What-If question. Next, the application filters the set of association rules of the mining model and returns only the association rules that contain the itemset chosen by the user. This filtered set of association rules is ordered by support and confidence values, in order to find out among the set of association rules which ones are the strong rules. Using the pre-defined support and confidence threshold values, the filtered set of association rules is reduced to a set of strong rules. This set of strong association rules correspond to a set of rules that contains important and relevant information to the user.

5.6.5 Using Preferences on What-If Analysis

In this step, we explain how we can create a What-If scenario using the OLAP preferences extracted previously. OLAP preferences are suggested to user in form of recommendations. Then, the user chooses, according to the goal analysis, the set of business variables that are the most adequate to add in the simulation model. The What-If analysis process starts with a doubt about the way to take in future decision-making processes, and the consequent definition of a What-If question (Figure 32). As said, the What-If analysis process consists in changing variables and exploring the possible consequences of this change, to obtain relevant information to respond to What-If question. Resort to a simulation model helps in exploring this process.

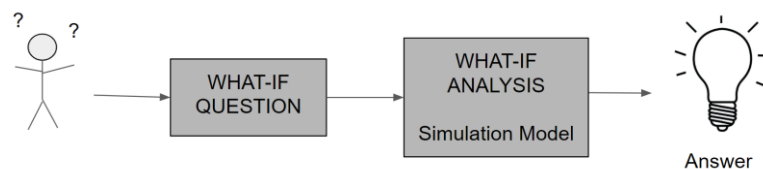


Figure 32 - Overview of the What-If analysis process.

A simulation model is the focus of a What-If application (Figure 33). Commonly, this model is a representation of a real business model and usually is composed of several application scenarios. Each scenario considers a set of business variables and a set of setting parameters (scenario parameters). It is the user responsibility to delineate the axis of analysis, the set of values for analyzing, and the set of values to change according to the goals defined previously. Then, the What-If process is performed with an appropriate tool. To run a simulation model, which is a scenario based on historical data, it is required to have a tool that can perform What-If scenario analysis, to get a prediction scenario. The What-If analysis tool calculates and lets the user to explore and analyze the impact of the changes in the setting values of the entire application scenario. It is the user who is responsible to accept the new data cube, or to return to change the settings of the application scenario and make the changes required over to the target data.

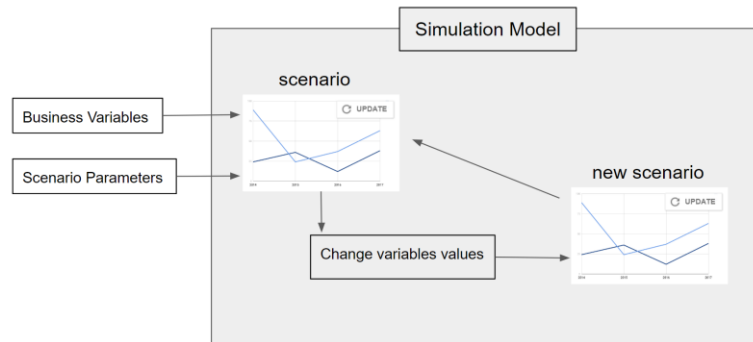


Figure 33 - Overview of the simulation model.

What-If Analysis allows for the user to try out different values and examine the changes and its impacts. What-If analysis can analyze the change effects by modifying a set of independent values and analyze the consequences in the values of the dependent values. The process analyzes the relation between attributes (if they are dependent on each other). If a set of attributes is dependent of the value that the user intent to alter, the set of attributes' values will be altered too. To perform this process successfully it is necessary to know the input data and know which data is relevant to add to the scenario to get the best possible outcome to make better decisions.

In a conventional What-If analysis process, the input data to the scenario is chosen according to the user requirements. It is required to understand and analyze the business content, to know which business variables to add in the What-If scenario. Identifying business variables that could be relevant and which ones could add significant information to the What-If scenario can take time for an inexperienced analyst or even a person which is not familiar with the business. With the methodology we proposed, the business understanding step is skipped. The user choses the scenario parameters from the set of extracted OLAP preferences in form of recommendations. The recommendations provide a set of business variables that add relevant information to the What-If analysis.

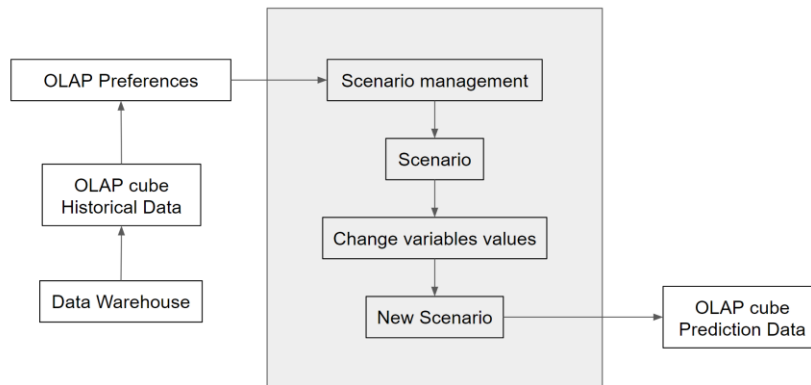


Figure 34 - Overview of the hybridization process.

The simulation model starts with the scenario management (Figure 34). This step consists in the user adding the business variables and scenario parameters to the scenario and all the features that are needed to perform the simulation model, which usually depend on the used tool. Business variables come from the OLAP preferences recommendation. Scenario parameters are usually dependent on the used tool to perform the What-If process. To perform a What-If process, the user must choose an appropriate tool that can create a What-If scenario and perform the calculations allowed in the What-If process. Then, after the scenario is created, the set of values to be changed in the scenario are shown to the user and he/she makes the intended changes. Then, the tool selected to perform What-If analysis makes the calculations and shows the new scenario to the user.

Regarding to the formal specification of the current step, it will not be necessary to specify the What-If process. This is a similar circumstance to the apriori algorithm specification. The What-If process is already a pre-validated algorithm and excluding it from the abstract model will not influence the outcome.

5.7 Formal Validation of the Hybridization Process

Once specified, the hybridization process must be validated. The first thing to do is to specify the properties that are expected to hold. For rule creation (`RulesCorrect`) (Figure 35), for instance, at least one rule must be created, their fields must belong to the cube and all the elements in a

specific rule must be unique. An assertion (`CheckRulesBad`) (Figure 36) is defined to test whether the construction of the rules guarantees their correctness. The check command instructs the Alloy Analyzer to check the assertion for a particular scope (Figure 37).

```
pred RulesCorrect[c: OLAPCube] {
  some c.~cube.mdl.rules
  c.~cube.mdl.rules.(is+o) in c.fields
  all r: c.~cube.mdl.rules | r.o not in r.is }
```

Figure 35 - Alloy specification: Predicate `RulesCorrect`.

```
assert CheckRulesBad {
  all c: OLAPCube, p: CubeParams |
    ConstructRules[c,p] => RulesCorrect[c] }
```

Figure 36 - Alloy specification: Assertion `CheckRulesBad`.

```
check CheckRulesBad
  for 8 but 4 Table, exactly 4 Field, exactly 16 SubsetFields
```

Figure 37 - Alloy specification: Check of the assertion `CheckRulesBad`.

In this case the Alloy Analyzer finds a counter-example that violates the assertion, because it is possible for the parameters to select tables for which no field is reachable from another, rendering the set of rules empty. Thus, an additional restriction must be imposed on the preference selection, which should also be enforced in the implementation of the process: that the selected tables contain reachable fields (`GoodCubeParams`) (Figure 38). Once the assertion (Figure 39) is fixed to consider this a pre-condition, no counter-examples are found.

```
pred GoodCubeParams[p: CubeParams] {
  some f: p.tabs.flds | some reach[f,p] }
```

Figure 38 - Alloy specification: Predicate `GoodCubeParams`.

```

assert CheckRulesGood {
  all c: OLAPCube, p: CubeParams |
    (GoodCubeParams[p] && ConstructRules[c,p]) => RulesCorrect[c] }

```

Figure 39 - Alloy specification: Assertion CheckRulesGood.

The creation of the strong rules must also be validated. Given user preferences, predicate `StrongRulesCorrect` (Figure 40) defines correct strong rule creation: there is at least one rule, all strong rules contain the preferred attribute, the strong rules are among the set of regular rules, and the performance measures of each strong rule is above the specified threshold.

```

pred StrongRulesCorrect[c: OLAPCube, p: PrefParams] {
  some c.~cube.mdl.strongRules
  p.attr in c.~cube.mdl.strongRules.(is+o)
  c.~cube.mdl.strongRules in c.~cube.mdl.rules
  all r: c.~cube.mdl.strongRules.performance |
    gte[r.suppl,p.suppl] && gte[r.conf,p.conf] }

```

Figure 40 - Alloy specification: Predicate StrongRulesCorrect.

```

assert CheckStrongRulesBad {
  all c: OLAPCube, p1: CubeParams, p2: PrefParams |
    (GoodCubeParams[p1] && ConstructRules[c,p1] &&
      ConstructStrongRules[c,p2]) => StrongRulesCorrect[c,p2] }

```

Figure 41 - Alloy specification: Assertion CheckStrongRulesBad.

Checking this assertion (Figure 41) with the Analyzer also generates a counter-example, which is illustrated in Figure 42. In this example instance, we have a `FactTable` and a `Dimension`, with certain measures and attributes assigned, respectively. Cube parameters select only the fact table and the preference parameters select `Attribute1`. No strong rule is created because `Dimension`, which contains `Attribute1`, is not part of the cube, so it could not be obtained through the application of the mining algorithm and consequently to be a preference suggested to the user. This counter-example violates the structure and correct function of our process. In order

to fix this issue, restrictions must be added in the alloy code to force the chosen attribute to be part of a dimension that belongs to the OLAP cube being mined.

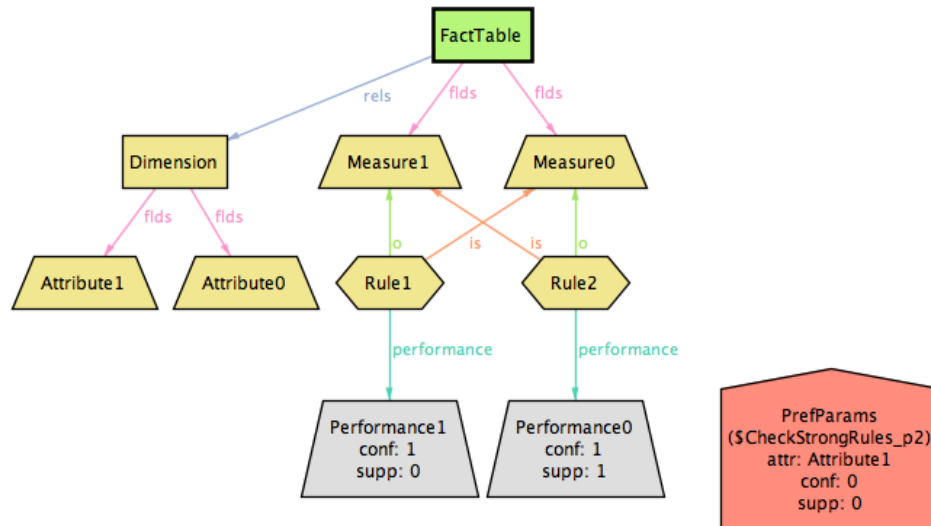


Figure 42 - Counter-example for *StrongRulesCorrect* found by the Alloy Analyzer.

The predicate `GoodPrefParams` (Figure 43) holds if the selected attribute belongs to the cube fields and if there is at least one strong rule that passes the given thresholds. Enforcing valid preference parameters, the assertion (Figure 44) no longer generates counter-examples, meaning that it is guaranteed to hold for the provided scope. This scope can be increased until the level of confidence in the design is high enough.

```
pred GoodPrefParams[c: OLAPCube, p1: CubeParams, p2: PrefParams] {
  p2.attr in c.fields
  some r: c.~cube.mdl.strongRules.performance |
    gte[r.supp,p2.supp] && gte[r.conf,p2.conf] }
```

Figure 43 - Alloy specification: Predicate `GoodPrefParams`.

```

assert CheckStrongRulesGood {
  all c: OLAPCube, p1: CubeParams, p2: PrefParams |
    (GoodCubeParams[p1] && ConstructRules[c,p1] &&
     GoodPrefParams[c,p1,p2] && ConstructStrongRules[c,p2]) =>
     StrongRulesCorrect[c,p2] }

```

Figure 44 - Alloy specification: Assertion CheckStrongRulesGood.

It is possible that no instance breaks the assertion due to over-restriction, i.e., by removing suitable instances for the search space. Thus, it is always useful to use run commands (Figure 45) to generate valid instances of the specification. Figure 46 represents one such example instance, which was generated with the following command.

```

run {
  some c: OLAPCube, p1: CubeParams, p2: PrefParams |
    GoodCubeParams[p1] && ConstructRules[c,p1] &&
    ConstructStrongRules[c,p2] && GoodPrefParams[c,p1,p2] &&
    ConstructPrefs[c] }
  for 30 but 2 Table, exactly 3 Field, exactly 8 SubsetFields

```

Figure 45 - Alloy specification: Run command to generate a valid instance.

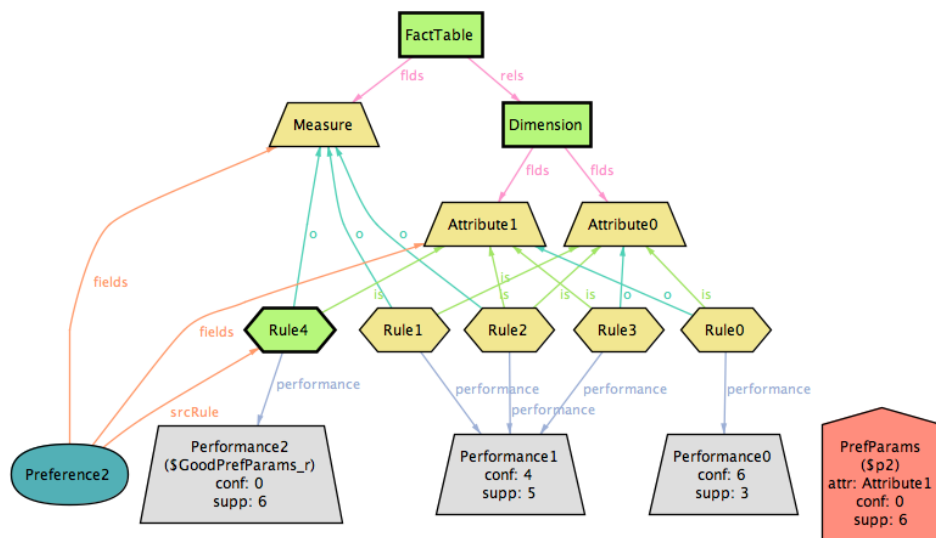


Figure 46 - An example of a consistent instance obtained by the Alloy Analyzer.

The instance contains a `FactTable` with a measure and a `Dimension` with 3 attributes, both selected by the cube preferences. A set of rules is extracted (`Rule0` through `Rule4`). Taking into account the user `PrefParams` (arbitrarily chosen by the Analyzer) only `Rule4`, from `Attribute1` to `Measure` and confidence 0 and support 6, is considered a strong rule (green bold rule), surpassing the thresholds (confidence 0 and support 6) and containing the selected `Attribute1`. Therefore, the attributes that compose `Rule4` are suggested to the user as a preference (`Preference2`).

5.8 Summary

In this chapter we presented in detail the approach we propose for overcoming the drawbacks of What-If simulation. Integrating OLAP preferences in the What-If analysis is one way to overcome the difficulties of inexperienced users when using the What-If analysis process. We explained why integrating OLAP preferences can help us improve the What-If analysis process. Next, we briefly explained how we integrate the process of extracting preferences in the conventional What-If analysis, followed by a proposed methodology based on the suggested process. Then, we explained in detail all the steps of the proposed hybridization process and proposed a formal verification and validation of the hybridization process using a formal object-oriented modeling language.

Summing up, the hybridization process proposed and discussed here is quite helpful during the execution of simulations when dealing with What-If based problems. The What-If analysis process consists in creating hypothetical scenarios in which is possible to analyze the consequences of changing business values. The difference between our developed process and the conventional What-If analysis is the introduction of the process of extraction of preferences. This set of preferences helps the user to select the most adequate parameters in the simulation process. To know which set of data to be included in the What-If simulation is a crucial step to achieve the best result during the whole process. The outcome of the simulation depends on the input data. If the input data to be added in the simulation is not adequate the outcome may come wrong. With

recommendations that fit best the user goal analysis, selecting the most adequate input data would be a much easier process, even for inexperienced users.

The referred hybridization process consists mainly on integrating the process of extraction of preferences into the conventional What-If analysis process. Firstly, as in the conventional What-If analysis process, there is the arising of the doubt or a problem. The data is stored in an OLAP cube, and it starts by using this multidimensional structure as input. Then, the OLAP mining is performed and extracted preferences from the outcome of the mining technique. The preferences are suggested to the user as form of recommendations and then, the user proceeds choosing a suitable tool and selecting the scenario settings of the simulation, delineating the axis of analysis, the set of values for analyzing, and the set of values to change according to previous defined goals. In the end, the user performs the simulation and gets the prediction OLAP cube. Adding the suggested scenario parameters, the user can have a simulation with more relevant information when comparing to the simulation with parameters randomly selected.

Chapter 6

A Case Study

6.1 Analysing Data

In order to illustrate our hybridization methodology, we selected a simple case study, from the Wide World Importers (WWI) (SQL Server Blog, 2016) data warehouse. The creation and analysis of the small data cube can clearly be generalized to larger complex cases.

The WWI database contains information about a fictitious company, which is a wholesale novelty goods importer and distributor. As a wholesaler, WWI's customers are mostly retail companies who resell to individuals. WWI has customers across the United States. WWI buys goods from suppliers including novelty and toy manufacturers, and other novelty wholesalers. The database schema of the case study "Sales" is presented in Figure 47. It contains a fact Table "Sale" and all the related dimension tables, namely: "Customer", "Employee", "Stock Item", "City" and "Date", each one containing the information about customers, employees, stock items', about cities of 49 states of EUA and date details between January 1, 2013 and December 31, 2016, respectively.

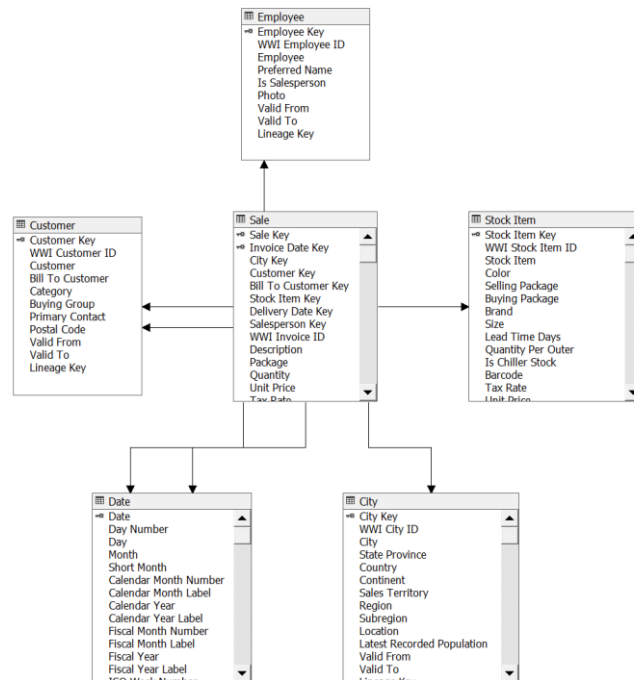


Figure 47 - Selected data warehouse's view – "Sales" schema.

In Figure 47 we can see a view of the data cube, data from the dimension "City" and the fact tables "Sale". We can see the sales data of some cities of 'Alabama', between 'January 1, 2013' and 'December 31, 2016'. "State Province" and "City" are attributes from the dimension "City", while "Profit", "Quantity", "Sale Count", "Tax Amount", "Total Chiller Items", "Total Dry Items", "Total Excluding Tax" and "Total Including Tax" are measures from the fact table "Sale".

State Province	City	Profit	Quantity	Sale Count	Tax Amount	Total Chiller Items	Total Dry Items	Total Excluding Tax	Total Including Tax
Alabama	Bazemore	141061.9	12655	313	42416.68	96	12559	283051.15	325467.83
Alabama	Belgreen	152717.35	13265	339	44781.44	96	13169	298816.4	343597.84
Alabama	Broomtown	120804.2	14212	367	37140.16	240	13972	248285	285425.16
Alabama	Coker	133529	14759	373	39282.89	48	14711	262022.55	301305.44
Alabama	Eulaton	117716.55	13576	339	36659.94	288	13288	245220.2	281880.14
Alabama	Flomaton	128217.9	11259	310	37674.75	408	10851	252327.7	290002.45

Figure 48 – Example 1 of a view of the schema "Sales".

The next figure (Figure 48) displays another possible view to the cube, where we can see data from the fact table "Sale", the Dimension "Date" and the Dimension "Customer". We can see sales

data organized by customer, in the month 'April, 2013'. The "Fiscal Year" and "Month attributes" are from Dimension "Date", and the "Customer" attribute is from Dimension "Customer". The remaining columns display measures from the Fact Tables "Sale".

Fiscal Year	Month	Customer	Profit	Quantity	Sale Count	Tax Amount	Total Chiller Items	Total Dry Items	Total Excluding Tax	Total Including Tax
2013	April	Tailsin Toys (Absecon, NJ)	3426.9	402	10	1048.83	0	402	6992.2	8041.03
2013	April	Tailsin Toys (Aceitunas, PR)	458.4	202	3	159.78	0	202	1065.2	1224.98
2013	April	Tailsin Toys (Airport Drive, MO)	3192.75	411	8	964.35	0	411	6429	7393.35
2013	April	Tailsin Toys (Alstead, NH)	4048.2	456	13	1004.25	0	456	6695	7699.25
2013	April	Tailsin Toys (Amanda Park, WA)	738.75	60	8	192.64	0	60	1284.25	1476.89
2013	April	Tailsin Toys (Andrix, CO)	4628.2	669	12	1207.59	0	669	8050.6	9258.19

Figure 49 – Example 2 of a view of the schema "Sales".

By performing OLAP Mining, we can extract a set of association rules. All the extracted frequent itemsets and associations rules are stored in the mining model. Figure 49 shows the information about itemsets, containing information about support, size and description of the Itemset.

Support	Size	Itemset
1	1	Employee = Eva Muirden
1	1	Employee = Unknown
1	2	Employee = Archer Lamble, Profit >= 5486.6337792
1	2	Employee = Archer Lamble, Quantity >= 227
1	2	Employee = Archer Lamble, Sale Count = 1 - 2
1	2	Employee = Archer Lamble, Total Dry Items >= 239
1	2	Employee = Archer Lamble, Total Excluding Tax >= 7705.4192984064
1	2	Employee = Archer Lamble, Total Chiller Items >= 156
1	2	Employee = Amy Trefl, Total Chiller Items >= 156
1	2	Employee = Sophia Hinton, Total Chiller Items >= 156
1	2	Employee = Taj Shand, Total Chiller Items >= 156
1	2	Total Chiller Items >= 156, Employee = Anthony Grosse
1	3	Employee = Archer Lamble, Total Chiller Items >= 156, Profit >= 5486.6337792
1	3	Employee = Archer Lamble, Total Chiller Items >= 156, Total Excluding Tax >= 7705.4192984064
1	3	Employee = Archer Lamble, Total Excluding Tax >= 7705.4192984064, Profit >= 5486.6337792
1	3	Employee = Archer Lamble, Total Dry Items >= 239, Sale Count = 1 - 2
1	3	Employee = Archer Lamble, Total Excluding Tax >= 7705.4192984064, Sale Count = 1 - 2
1	3	Employee = Archer Lamble, Total Excluding Tax >= 7705.4192984064, Quantity >= 227
1	3	Employee = Archer Lamble, Total Excluding Tax >= 7705.4192984064, Total Dry Items >= 239

Figure 50 - Extracted Itemsets.

As we can see (Figure 50), the mining model contains itemsets of 1, 2 and 3 items (1-itemset in Table 4, 2-itemsets in Table 5 and 3-itemsets in Table 6, respectively).

Table 4 - Example of 1-itemset.

1-itemset
Employee = Eva Muirden
Employee = Unknown

Table 5 - Example of 2-itemsets.

2-itemset	
Employee = Archer Lamble	Profit >= 5486.6337792
Employee = Archer Lamble	Quantity >= 227

Table 6 - Example of 3-itemsets.

3-itmset		
Employee = Archer Lamble	Total Chiller Items >= 156	Profit >= 5486.6337792
Employee = Archer Lamble	Total Chiller Items >= 156	Total Excluding Tax >= 7705.4192984064

To perform the association mining technique, it is necessary to discretize the numerical continuous attributes for improving the quality of the induced rules (Moreno, et al., 2007). Discretization is a data pre-processing technique which transforms continuous functions, models, variables, and equations into discrete counterpart, which is required to perform the data mining technique. The numerical continuous attributes need to be divided in ranges as attribute-value pairs. The numerical (continuous) values of "Profit", "Quantity", "Sale Count", "Total Dry Items", "Total Excluding Tax" and "Total Chiller Items" need to be discretized. "Employee" corresponds to the employee name, and it does not need discretization, because it is a discrete data type string (or text).

Figure 51 shows the association rules and its details, as information about probability, importance and rule. The probability describes how likely an association rule occurs. The importance represents the usefulness of a rule. The Rule column presents the set of itemsets of each association rule. Taking in consideration two association rules:

-
- {"Employee" = 'Sophia Hinton' -> "Sale Count" = '1-2'}
 - {"Employee" = 'Hudson Onslow' -> "Total Excluding Tax" >= '7705.4192984064'}

These rules have the same probability values (equal to 1, or 100%) but different importance values: 0.308 and 0.280, respectively. The greater the value, the more important is the rule. We can conclude that, despite having the same probability of occurrence, the first one is more relevant or important than the second one.

↓ Probability	Importance	Rule
1.000	0.308	Employee = Sophia Hinton -> Sale Count = 1 - 2
1.000	0.280	Employee = Hudson Onslow -> Total Excluding Tax >= 7705.4192984064
1.000	0.280	Employee = Hudson Onslow -> Total Dry Items >= 239
1.000	0.280	Employee = Hudson Onslow -> Sale Count = 1 - 2
1.000	0.280	Employee = Hudson Onslow -> Quantity >= 227
1.000	0.280	Employee = Hudson Onslow -> Profit >= 5486.6337792
1.000	0.308	Employee = Sophia Hinton -> Quantity >= 227
1.000	0.308	Employee = Sophia Hinton -> Profit >= 5486.6337792
1.000	0.308	Employee = Sophia Hinton -> Total Excluding Tax >= 7705.4192984064
1.000	0.308	Employee = Sophia Hinton -> Total Dry Items >= 239
1.000	0.358	Employee = Hudson Onslow -> Total Chiller Items < 84
0.889	0.336	Employee = Sophia Hinton -> Total Chiller Items < 84
0.857	0.264	Employee = Lily Code -> Sale Count = 1 - 2
0.857	0.264	Employee = Lily Code -> Profit >= 5486.6337792
0.857	0.264	Employee = Lily Code -> Quantity >= 227
0.857	0.264	Employee = Lily Code -> Total Excluding Tax >= 7705.4192984064
0.857	0.264	Employee = Lily Code -> Total Dry Items >= 239
0.833	0.242	Employee = Hudson Hollinworth -> Total Excluding Tax >= 7705.4192984064
0.833	0.242	Employee = Hudson Hollinworth -> Total Dry Items >= 239
0.833	0.242	Employee = Hudson Hollinworth -> Sale Count = 1 - 2

Figure 51 - Extracted Association Rules.

6.2 A Software Platform for Receiving the Methodology

For receiving and support the application of the methodology we proposed, we designed and implemented a specific software platform, which we named as "OPWIF" (meaning, OLAP Preferences What-If analysis integration). This platform allows for the user to:

- i) create What-If scenarios choosing the available attributes of his choice (conventional What-If analysis);
- ii) consult the mining models' item sets and association rules;
- iii) use the hybridization process, described in the previous chapter.

Each functionality described above is associated to a specific tab in the main environment of the platform. Figure 31 represents the User Interface (UI) of the WIF tab, which allows the user to perform the conventional What-If analysis. The other functional tabs (MiningStructure and HybridizationModel) will be described in detail later. To illustrate the several functionalities of the software platform we start by describing a case study involving an analysis of the products' mining structure.

6.3 Example Case Study

The analysis example selected was one want to use What-If analysis to explore the effects of increasing the sales profit values by 10% of the profitable products of a specific store. Considering this scenario context, we formulate the following What-If question: "What if we want to increase the sales profit by 10% focusing mainly on the most profitable products' color?". Next, we need to define the goal analysis and a set of business variables (included in the What-If question) to add to the analysis scenario. The goal analysis is "color" from the products' mining structure, because the analyst wants to know how the profit values may vary according to the products' color, more specifically, the most profitable products' color. The set of variables to be added to the scenario would be "sales profit", because it is the attribute that we aim at altering (increasing 10%) and also, it would be useful and interesting to analyze the scenario data by year or month.

Next, we take the described analysis example and use the conventional What-If analysis (section 6.3.1) and the proposed hybridization process (section 6.3.3) to get the needed information to answer the What-if question.

6.3.1 Conventional What-If analysis

The developed software platform allows to perform the conventional What-If analysis. Figure 52 represents the application UI of this tab, the WIF tab. In the application first tab, we can create a typical What-If scenario using the conventional What-If analysis. The user chooses the parameters that he wants to introduce in the scenario (according to the pre-defined What-If question) and creates the graphic to analyze the profit values.

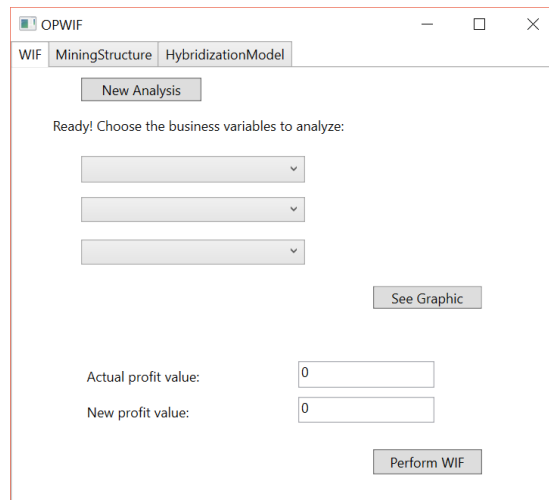


Figure 52 - Overview of the software platform UI - WIF tab.

Considering the analysis example described before and to perform the conventional What-If analysis, we start to choose the business variables to create the What-If scenario. Figure 53 show how we use the application UI to do that. As seen before, the set of parameters to be chosen are: "Calendar Year" and "Calendar Month" from the Dimension "Invoice Date" and "Color" from the Dimension "Stock Item", as we want to know which is the most profitable products' color. We opt to choose "Calendar Year" and "Calendar Month" to analyze the scenario data by month. Then, after 'See Graphic', the application shows the Historical Scenario and automatically calculates the profit values as shown in Figure 54 in the "Actual profit value". As we want to analyze the effects of changing the profit value by 10%, we set the new value in the "New profit value". After performing the What-If analysis, the application returns the Prediction Scenario (Figure 55).

Figure 53 - WIF tab - Choice of the set of business variables to perform the analysis.

Figure 54 - WIF tab - Change variables' values.

The prediction scenario (Figure 55) that shows in the Y axis: the attributes "Profit" with a range from '-200 000' to '1 600 000', and represented by the X axis: "Calendar Year" ('2013' to '2016'), "Month Number of Year" with a range of '1' to '12' which represents the months of a year, from 'January' to 'December'; and "Color" which can be 'Black', 'Red', 'Grey', 'Yellow', 'Blue', 'White', 'Light Brown' and 'N/A' (not available). The 'N/A' values could be derived from missing data, several colors on the product or software errors.

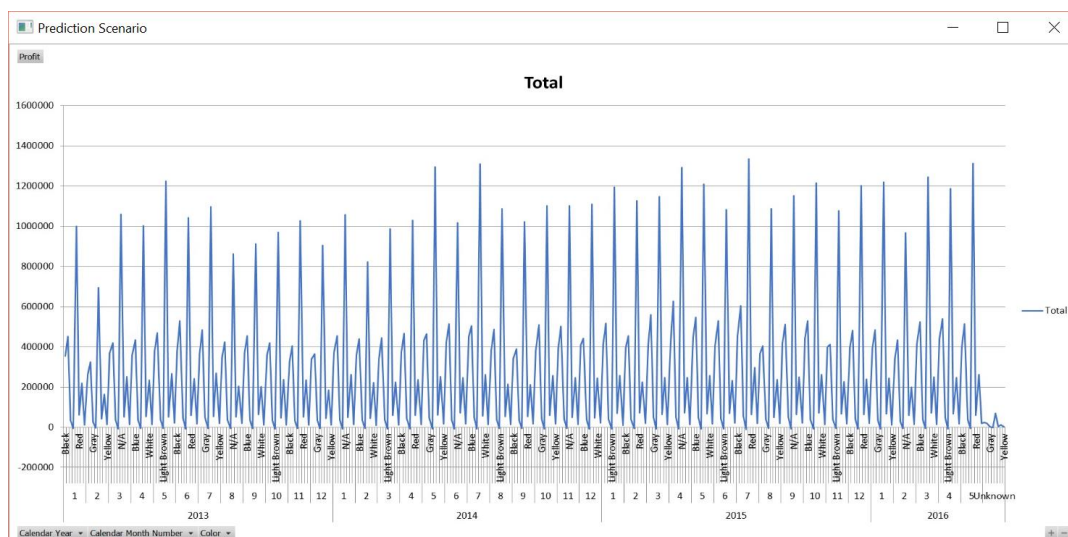


Figure 55 - WIF tab - Prediction scenario.

Although the content of the Prediction scenario is illegible, we can pass on potential knowledge for decision making to managers, such as: products with 'N/A' color are the most profitable and it is the product color that earn more money, especially in 'May', 'July' and 'August' in '2015' (near to '1,400,000'). Followed by the products' color 'Blue', which are the most profitable in 'April' and in 'July' of '2015' (over '600,000') and the products' color 'Black', which are most profitable in 'July' '2015' (around '400,000'). And finally, 'Light Brown' is the products' color the less profitable, especially in '2015' with negative values (between '-10,000' and '-4,000').

6.3.2 Itemsets and Association Rules

Additionally, the developed software allows to consult the mining models' item sets and association rules. In the application UI of the Mining Structure tab (Figure 56) is possible to analyze results from applying mining to the OLAP cube.

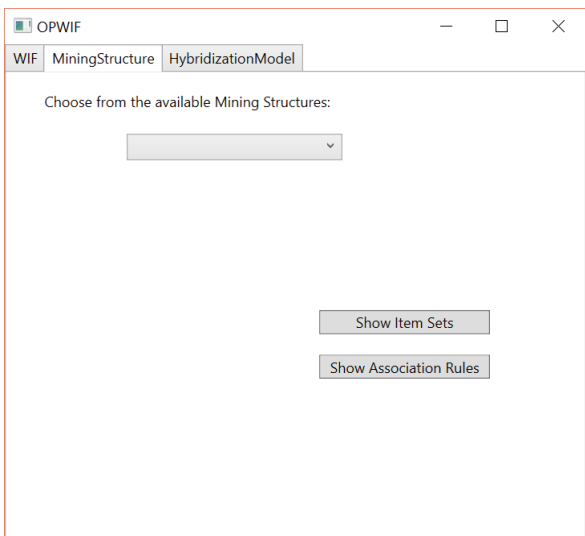


Figure 56 - Overview of the application UI - MiningStructure tab.

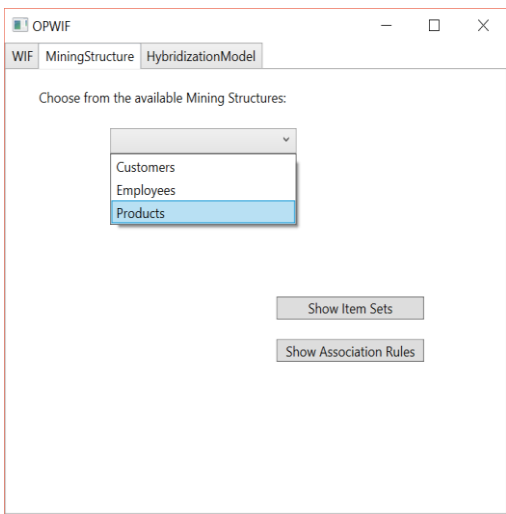
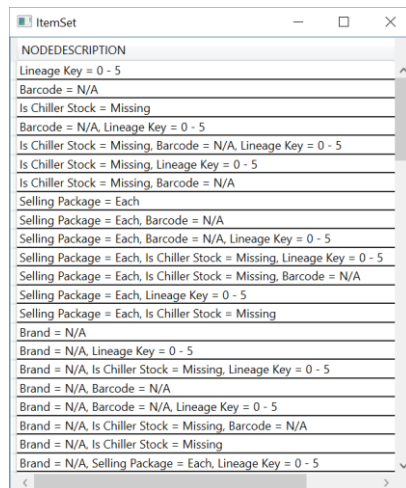


Figure 57 - MiningStructure tab - explore the "Products" mining structure.

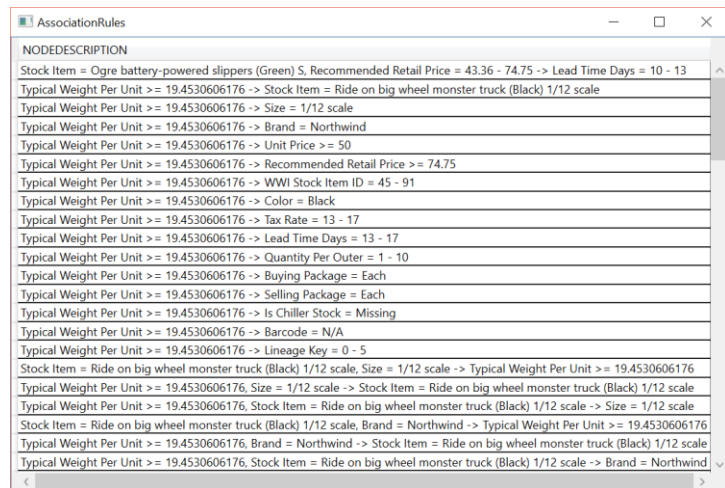
The user chooses one of the available mining structures and analyses their content, frequent itemsets and association rules. There are three mining structures: customers, employees, and products. Customers' mining structure has information about correlate business variables of

customers, frequent itemsets and association rules, and employees' mining structure has information about correlate business variables of employees, and so on. One can analyse which business variables of customer are more likely to appear together with business variables of product.



NODEDESCRIPTION
Lineage Key = 0 - 5
Barcode = N/A
Is Chiller Stock = Missing
Barcode = N/A, Lineage Key = 0 - 5
Is Chiller Stock = Missing, Barcode = N/A, Lineage Key = 0 - 5
Is Chiller Stock = Missing, Lineage Key = 0 - 5
Is Chiller Stock = Missing, Barcode = N/A
Selling Package = Each
Selling Package = Each, Barcode = N/A
Selling Package = Each, Lineage Key = 0 - 5
Selling Package = Each, Is Chiller Stock = Missing, Lineage Key = 0 - 5
Selling Package = Each, Is Chiller Stock = Missing, Barcode = N/A
Selling Package = Each, Lineage Key = 0 - 5
Selling Package = Each, Is Chiller Stock = Missing
Brand = N/A
Brand = N/A, Lineage Key = 0 - 5
Brand = N/A, Is Chiller Stock = Missing, Lineage Key = 0 - 5
Brand = N/A, Barcode = N/A
Brand = N/A, Barcode = N/A, Lineage Key = 0 - 5
Brand = N/A, Is Chiller Stock = Missing, Barcode = N/A
Brand = N/A, Is Chiller Stock = Missing
Brand = N/A, Selling Package = Each, Lineage Key = 0 - 5

Figure 58 - Itemsets of the Products' mining structure.



NODEDESCRIPTION
Stock Item = Ogre battery-powered slippers (Green) S, Recommended Retail Price = 43.36 - 74.75 -> Lead Time Days = 10 - 13
Typical Weight Per Unit >= 19.4530606176 -> Stock Item = Ride on big wheel monster truck (Black) 1/12 scale
Typical Weight Per Unit >= 19.4530606176 -> Size = 1/12 scale
Typical Weight Per Unit >= 19.4530606176 -> Brand = Northwind
Typical Weight Per Unit >= 19.4530606176 -> Unit Price >= 50
Typical Weight Per Unit >= 19.4530606176 -> Recommended Retail Price >= 74.75
Typical Weight Per Unit >= 19.4530606176 -> WWI Stock Item ID = 45 - 91
Typical Weight Per Unit >= 19.4530606176 -> Color = Black
Typical Weight Per Unit >= 19.4530606176 -> Tax Rate = 13 - 17
Typical Weight Per Unit >= 19.4530606176 -> Lead Time Days = 13 - 17
Typical Weight Per Unit >= 19.4530606176 -> Quantity Per Outer = 1 - 10
Typical Weight Per Unit >= 19.4530606176 -> Buying Package = Each
Typical Weight Per Unit >= 19.4530606176 -> Selling Package = Each
Typical Weight Per Unit >= 19.4530606176 -> Is Chiller Stock = Missing
Typical Weight Per Unit >= 19.4530606176 -> Barcode = N/A
Typical Weight Per Unit >= 19.4530606176 -> Lineage Key = 0 - 5
Stock Item = Ride on big wheel monster truck (Black) 1/12 scale, Size = 1/12 scale -> Typical Weight Per Unit >= 19.4530606176
Typical Weight Per Unit >= 19.4530606176, Size = 1/12 scale -> Stock Item = Ride on big wheel monster truck (Black) 1/12 scale
Typical Weight Per Unit >= 19.4530606176, Stock Item = Ride on big wheel monster truck (Black) 1/12 scale -> Size = 1/12 scale
Stock Item = Ride on big wheel monster truck (Black) 1/12 scale, Brand = Northwind -> Typical Weight Per Unit >= 19.4530606176
Typical Weight Per Unit >= 19.4530606176, Brand = Northwind -> Stock Item = Ride on big wheel monster truck (Black) 1/12 scale
Typical Weight Per Unit >= 19.4530606176, Stock Item = Ride on big wheel monster truck (Black) 1/12 scale -> Brand = Northwind

Figure 59 – The association rules of the Customers' mining structure.

In Figure 58 it is possible to see the frequent itemsets of the products' mining structure. We can see that the first three frequent itemsets are "Lineage key" with values between '0' and '5'; "Barcode" not available (with 'N/A' value) and "Is Chiller Stock" with 'Missing' values. In Figure 59, it is possible to see the association rules of the products' mining structure. The association rules are sorted by decreasing probability values. The first association rule ["Stock Item" = 'Ogre battery-powered slippers (Green) S', "Recommended Retail Price" = '43.36 – 74.75' -> "Lead Time Days" = '10-13']. This rule means that a specific product called "Ogre battery-powered slippers (Green) S" with the recommended retail price between '43.36' and '74.75' are often related to lead time days between 10-13 days. Another interesting rule is ["Typical Weight Per Unit" >= '19.4530606176' -> "Brand" = 'Northwind']. This rule means that products with typical weight per unit higher than '19.45' are usually brand 'Northwind'.

6.3.3 The Hybridization Process

To support the proposed methodology, we developed the HybridizationModel tab (Figure 60). The hybridization methodology is possible to be followed using this tab. In this tab occurs the filter process, where the association rules extracted are filtered, deriving the recommendation to the user. Step 1 is responsible to extract the frequent itemsets of a chosen mining structure. In step 2 the user chooses the frequent itemset of its choice (according to the What-If question). Finally, step 3 is responsible of showing the user the filtered association rules. These association rules are an association rules' subset that contain the chosen goal analysis attribute in step 2. Finally, preferences derived from the association rules are shown to the user.

Next, and considering the analysis example described before (in section 6.3) and following the steps of the hybridization process, we start by selecting the mining structure most adequate to answer the What-If question in the combo box in the right in the application UI. In this case, we selected the Products' mining structure (Figure 61). Then, in step 1, we have the possibility of accepting the default minimum support and probability values or altering them according to the needs. This step filters the set of itemsets of the mining structure and returns the frequent itemsets. In other words, returns the set of frequent itemset that are above the support and probability values.

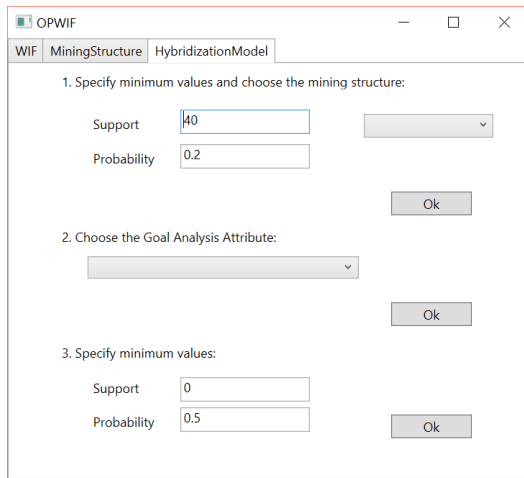


Figure 60 - Overview of the application UI - HybridizationModel tab.

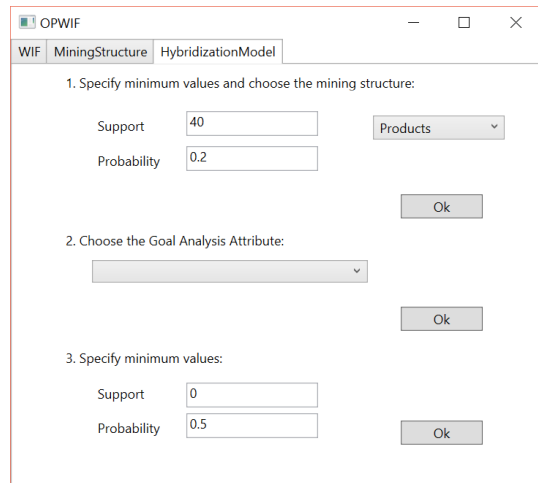


Figure 61 - HybridizationModel tab - Choice of the mining structure.

The list of frequent itemsets is displayed in the combo box of step 2. We select the most adequate itemset given the What-If question. In this case, we select the itemset "Color" as seen in Figure 62.

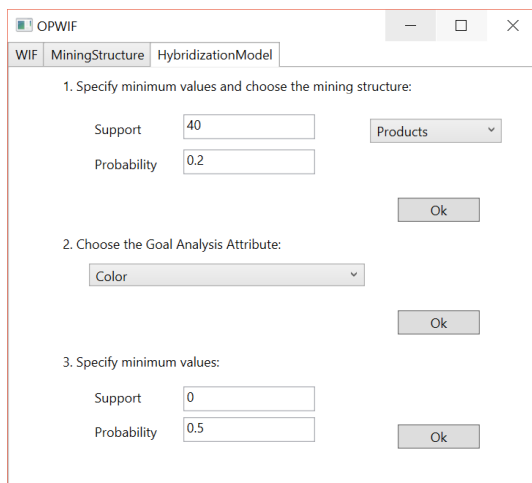


Figure 62 - HybridizationModel tab - Choice of the goal analysis business variable.

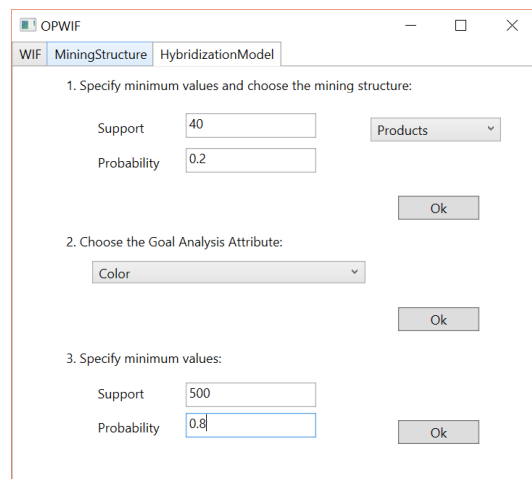


Figure 63 - HybridizationModel tab - Change support value to 500 and confidence value to 80%.

The step 3 consists in filtering the association rules that contain the itemset 'Color' and have support and probability values above '0' and '0.5' as default, respectively. Here we change the support value to '500' and confidence value to '0.80' (corresponds to 80%) (Figure 63). After filtering the association rules with minimum support and probability values, the application UI shows a new window, where no association rules were found matching the '500'-support value and the '80%' confidence value. As seen before, we have the possibility of accepting the default minimum support and probability values or altering them. Here, it is needed to return to the HybridizationModel tab and change the confidence values in order to get some association rules to proceed in the hybridization process.

After filtering the association rules with default minimum support and probability values, the application UI shows the window, represented by Figure 64, containing the final association rules' list ordered by probability of happening in the left. The three top rules 1) ["Brand" = 'Northwind', "Color" = 'Black' -> "Barcode" = 'N/A'], 2) ["Brand" = 'Northwind', "Color" = 'Black' -> "Buying Package" = 'Each'] and 3) ["Brand" = 'Northwind', "Color" = 'Black' -> "Is Chiller Stock" = 'Missing'] are chosen to form the OLAP preferences. The chosen rules are the association rules in the right. Next, the item sets contained in the filtered association rules will be suggested to the user as preferences.

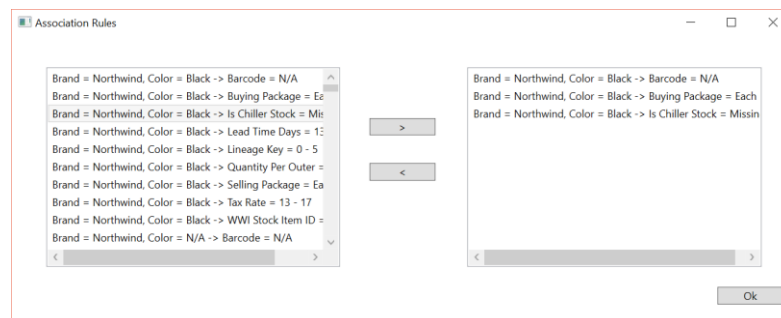


Figure 64 - Selection of the Top association rules.

Finally, the application UI shows a new window with the OLAP preferences, represented by Figure 65. We chose the ones to be part of the What-If scenario. The preferences are the itemsets of the chosen association rules "Brand", "Barcode", "Buying Package" and "Is Chiller Stock" in the left. "Calendar Year" and "Month Number of Year" are suggested too to be part of the scenario.

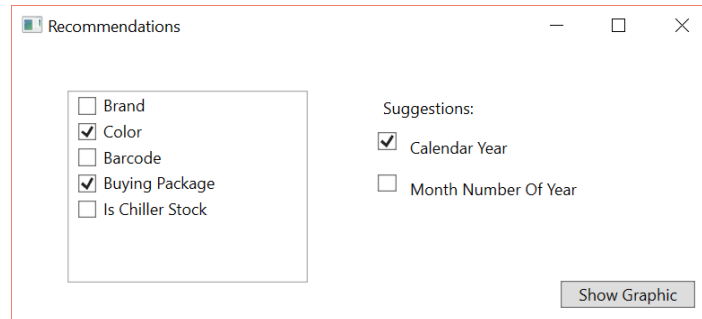


Figure 65 - Recommendations made to the user.

Then, the application UI creates a historical scenario with the chosen parameters and shows it to the user. Finally, the application UI shows a new window (Figure 66), in which the user can enter the desired final value. This step is similar to the one in the conventional What-If analysis, in which the user changes the value of the goal analysis variable to the wanted one. In other words, if the user wants to increase the profit value by 10%, we want to alter the profit final value by 10%.

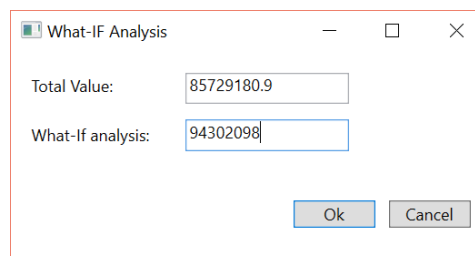


Figure 66 - HybridizationModel tab - changing the variables' values.

Then, the application performs What-If analysis and returns the new prediction scenario, represented by Figure 67.

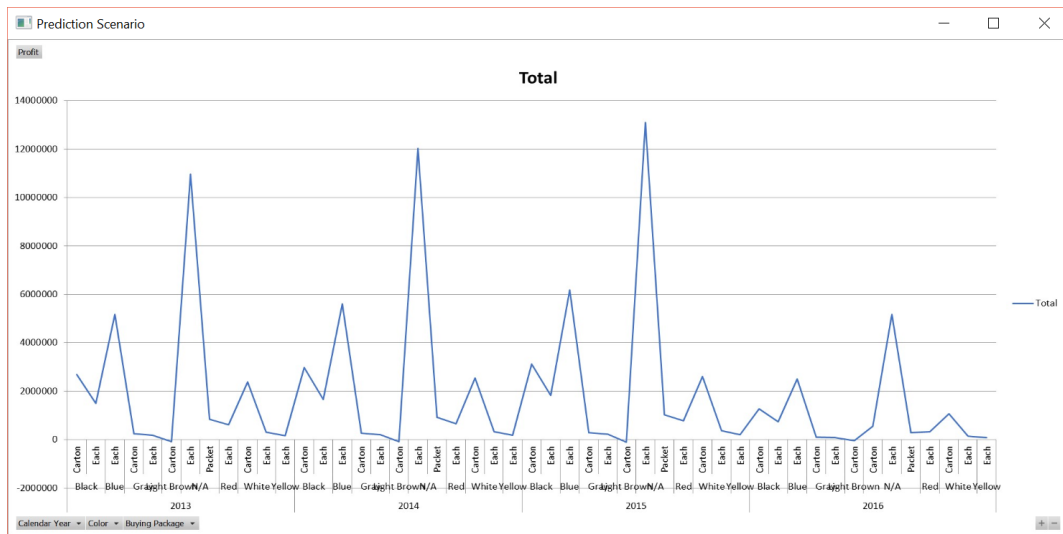


Figure 67 - HybridizationModel tab - Prediction scenario.

With the analysis of the Prediction Scenario, though the content of the Prediction scenario is illegible Figure 67, we can conclude that products with unknown color or not available ('N/A') with 'Each type' buying package are the most profitable products in '2015', followed by '2014', which resulting in a high profit in both years (over '12,000,000'). Followed by the products' color 'Blue' sold with the 'Each type' buying package in '2015' with total profit value over '6,000,000' and finally the products' color 'Black' sold with 'Carton' buying package with total profit value over '3,000,000' in '2014' and in '2015'. 'White' or 'Black' products with buying package made from 'Carton' are also profitable. Apart from these cases, products that are sold in 'Carton' and 'Packet' (regardless of color) generally have low profit values (less than '1,500,000' for year).

The new parameter "Buying Package" was suggested by the application and selected to be in the scenario by the user. This business variable could be 'Carton', 'Packet' and 'Each', meaning that the buying package is made from 'Carton' and 'Packet'. The introduction of this variable is the main difference between the two approaches, with or without the integration of preferences.

6.4 Comparative Analysis

Now, and considering the described example analysis, it is time to compare the outcome of both approaches shown, the outcome of the application of the conventional What-If analysis (section 6.3.1) and the outcome of the application of our proposed hybridization process (section 6.3.3). As the graphs obtained through both processes were unreadable, in this comparative analysis we take the obtained charts in both approaches and reduce the year range, which means that, for a clear analysis of data, only "Calendar Year" = '2016' is showed in all graphs.

6.4.1 Conventional What-If analysis Results

In this section, we consider the outcome of the application of a conventional What-If analysis, section 6.3.1. The historical scenario represented by Figure 69 and prediction scenario represented by Figure 70.

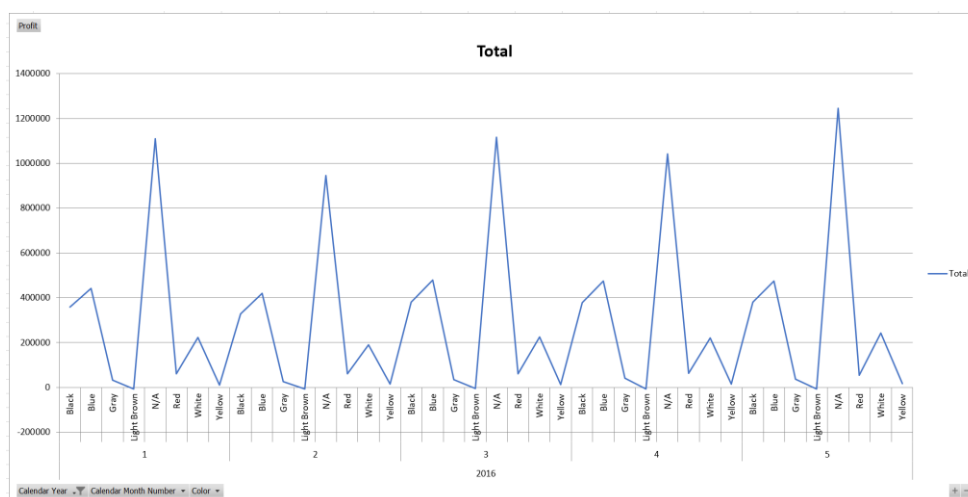


Figure 68 - Conventional What-If Analysis - Historical scenario.

In the Historical scenario (Figure 68) and in the prediction scenario (Figure 69) is possible to analyze the attributes "Profit" represented by the Y axis, with a range from '-200 000' to '1 600 000'; and represented by the X axis: "Calendar Year" ('2016'), "Month Number of Year" with a range of '1' to '5' which represents the months of a year, from 'January' to 'May'; and "Color" which can be 'Black', 'Red', 'Gray', 'Yellow', 'Blue', 'White', 'Light Brown' and 'N/A' (not available).

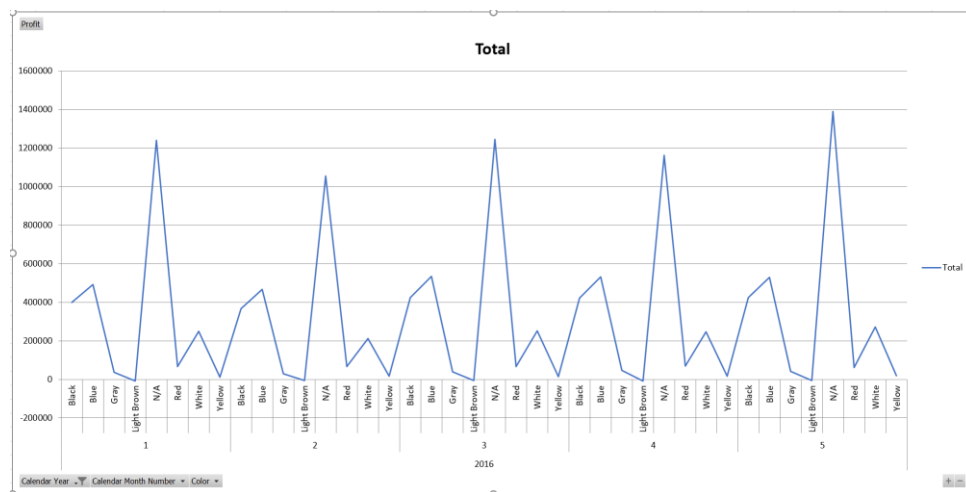


Figure 69 - Conventional What-If Analysis – The prediction scenario.

The prediction scenario shows that in '2016', products with 'N/A' color is the most profitable and it is the product color that earn more money, especially in 'May', 'March' and 'January', respectively; showing profit vales over than '1,200,000'. Followed by the products' color 'Blue', which is the most profitable in 'March', 'April' and 'May'; and finally, the products' color 'Black' are more profitable in the same months that the products' color 'Blue'.

Like the analysis made in the "Conventional What-If analysis" section, 'Light Brown' is the products' color less profitable, also with negative values in '2016'.

6.4.2 Hybridization Process Results

Now, we consider the outcome of the application of our hybridization process in section 6.3.3. When we analyze both scenarios, historical scenario (Figure 69) and prediction scenario (Figure 70), it is possible to verify that products with Light Brown shows negative profit. But this fact is not news, as we had already concluded this fact in previous subsection by analyzing the outcome of the conventional What-If analysis.

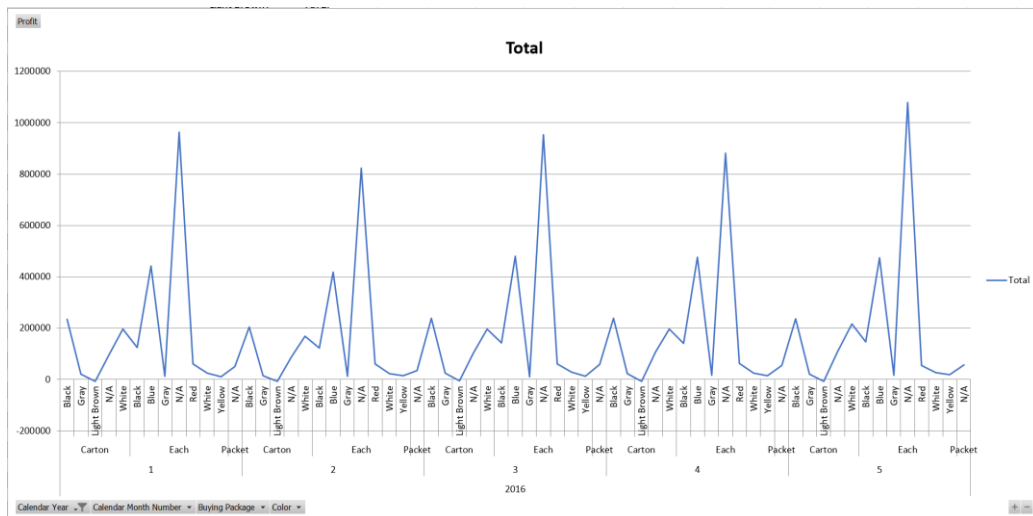


Figure 70 - Hybridization process - Historical scenario.

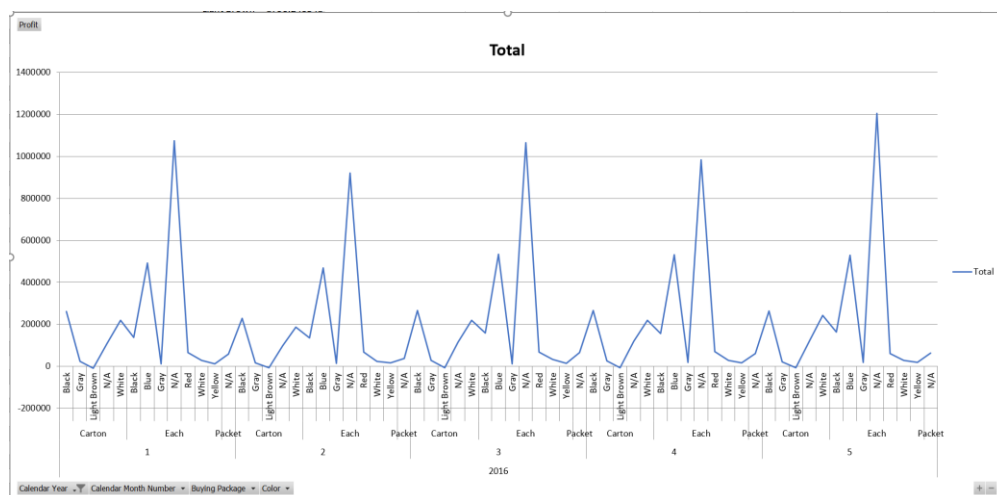


Figure 71 - Hybridization process - Prediction scenario.

The novelty using our hybridization process is the suggestion of the "Buying Package" parameter. With the addition of this new parameter it is possible to conclude more facts beyond what we previously conclude with the conventional What-If analysis.

Analysing the data of both charts presented, and similar to the conventional What-If outcome analysis, we can conclude that products with unknown color or not available information about color ('N/A') are the most profitable in 'May', 'March' and 'January'. The information that was

hidden from the user in the conventional What-If analysis and now it is possible to analyze that the most profitable products with ('N/A') color were sold with 'Each type' buying packages.

Products with unknown color or ('N/A') with 'Each type' buying packages are the most profitable products (with profit values over '1,000,000' in 'January', 'March' and 'May'), comparing to ('N/A') color products with 'Carton' and 'Packet' (less than '120,000'), which are less profitable.

Following the ('N/A') color products, the products' color 'Blue' are the second most profitable, especially in 'May', 'April' and 'March' (over '500,000'). This fact is already stated in the last analysis, in the conventional What-If analysis outcome. The novelty here is the fact that the most profitable 'Blue' products were sold with 'Each type' buying packages. Another fact that we can analyze is that 'Blue' products sold in 'Carton' and 'Packet' buying packages do not show any profit values.

The products' color 'Black', similar to the previous analysis, are the third most profitable products in 'May', 'April' and 'March' (over '250,000'). The novelty here is that the most profitable 'Black' products were sold with 'Carton' buying packages. This information is hidden in the conventional What-If outcome analysis. Also, 'White' products with buying package made from 'Carton' are also profitable (over '200,000' in 'May', 'April' and 'March'). Apart from these cases, products that are sold in 'Carton' and 'Packet' (regardless of Color) generally have low profit values (less than '150,000').

Thus, we can conclude that regardless the color, the buying package ('Carton' or 'Packet') influence the negatively the profit. Finally, and already known fact is that 'Light Brown' products have negative profit values. The new information that we can include in this last fact is that 'Light Brown' products have buying package made from 'Carton'.

One possible change to improve profit would then be to rethink the type of buying package on 'Light Brown' products and see if this change could really improve profitability. Other possibility is trying to discover why 'Light Brown' products have negative profit values: if the problem is on sales or on purchase them to the suppliers. Rethinking the cases of products with buying package made from 'Carton' or 'Packet' (independently of the product color) because of the lower profit values.

It will be helpful to discover which products are light brown and arrange a tactic to improve profits. This information is credible, but is very generic and abstract, and better decisions can be reached with more detailed information.

6.4.3 Conclusions

Comparing the outcomes of both approaches, the conventional What-If analysis and the hybridization process, we can conclude that when using the hybridization process, we get more refined and detailed results, leading to more accurate decisions. For example, in the conventional What-If analysis, the most profitable products' color was 'N/A'; on the other hand, in the hybridization process, the most profitable products' color was also 'N/A', but we also learned that the most profitable products with ('N/A') color were sold with 'Each type' buying packages. And additionally, products with the buying package ('Carton' or 'Packet') and regardless the color, influence negatively the profit.

The second most profitable products' color, in the conventional What-If analysis approach, was 'Blue'. In the hybridization process, by analyzing the scenarios, we conclude that the most profitable 'Blue' products were sold with 'Each type' buying packages (as in the 'N/A' colored products). Another fact that we conclude using the hybridization process is that 'Blue' products sold in 'Carton' and 'Packet' buying packages do not show any profit values.

Finally, in the conventional What-If analysis, 'Black' was the third most profitable products' color. In the hybridization process, we conclude that the most profitable 'Black' products were sold with 'Carton' buying packages. This information is hidden by the conventional What-If outcome analysis.

The presented example analysis represents a small case study and it demonstrates the potential of the methodology, which helps up to be helpful when dealing with more complicated cases. With this methodology we can add new relevant information to the analysis.

Chapter 7

Conclusions and Future work

7.1 Final Remarks

What-If analysis has been shown to be a useful tool in the BI area. It allows for creating hypothetical scenarios to analyze the behavior of a system under specific conditions. The What-If analysis starts with the definition of the What-If question, after a doubt about business arise. This What-If question translates into a specific scenario that is mounted by the user in an appropriate tool. The user chooses the input parameters to be added and set the configuration settings of the scenario and performs the What-If analysis, creating the new prediction scenario. The analysis of this new prediction scenario will help the user to answer the defined What-If question.

A successful What-If analysis process depends mainly on the user, the lack of expertise of a user during the What-If design and implementation is one of the disadvantages of this process. If the user is not familiar with the process, or even the business data, the What-If analysis can turn into a difficult experience, leading to inadequate outcomes and incorrect conclusions. Therefore, and based on the described aspects, in this thesis we demonstrate that we could overcome the pitfalls of the What-If analysis process. To do that, we studied the whole What-If analysis and its components, identified the faults and investigated how we could reduce the negative impacts in

the quality and effectiveness of a What-If analysis solution. In this research, we made efforts so that the following main research questions, addressed in this thesis, could be answered, namely:

- *Can we optimize the decision process in data cubes and in particular improve What-If scenarios using prediction models?*

What-If simulation allows for the user to create hypothetical scenarios to explore the consequences of changing variables. The user is responsible for choosing the scenario to be analysed in the simulation. Thus, it is important that the user is aware of the business data and how the What-If simulation proceeds. Otherwise, the simulation outcome may be inadequate and lead to bad decisions. The integration of the process of extraction of preferences into the What-If simulation can improve significantly the whole What-If simulation. Due to the discovery of the set of business variables strongly related to the goal analysis business variable, it is possible to the user to add valuable information to the scenarios and consequently get an outcome with information more oriented to the goal analysis. With the recommendations, the user ends up by saving time. Otherwise, the user may have to make several attempts until he gets a scenario that allows for extracting valuable information when compared to the outcome of the hybridization process. Thus, we can conclude that a decision process in a multidimensional structure can be improved using the proposed hybridization process, as seen in the detailed explanation of the hybridization methodology (Chapter 5) and with the example case study and the comparison of the two process results (Chapter 6), using the conventional What-If analysis and the hybridization process.

- *Can we improve the view selection (and consequently query time response and memory consumption) by restructuring automatically What-If scenarios?*

In the proposed hybridization methodology, the user experience using the What-If analysis is simplified due to the use of OLAP preferences. The system can propose a new set of data cube views based on user preferences, which means that the data is filtered and recommended to the user more oriented and refined scenarios. Otherwise, if the user creates the simulation model with all the information of the data cube, the query processing time and memory consumption will increase significantly. In the hybridization

process, with the integration of OLAP preferences and with more oriented information suggested to the user, the query time response and memory consumption decrease significantly. As seen in the example case study and the comparison of the two process results (Chapter 6), in the example case study using the hybridization process.

- *Can we get more oriented results by integrating prediction models into the conventional What-If analysis process?*

In the previous chapter, we shown an application of the What-If analysis without the integration of OLAP usage preferences (in the conventional What-If analysis section of the previous chapter) and an application in the same conditions with the integration of OLAP usage preferences (in the Hybridization process section of the previous chapter). Analysing and comparing the results, it is possible to conclude that the latter allows the user to get more richer and detailed information and consequently to accurate conclusions. An exception may occur on the application of the conventional What-If analysis, if an inexperienced user ends up by selecting and adding similar scenario parameters to the ones that would be suggested in the second case (where the recommendations are the outcome of the integration of the OLAP usage preferences), the outcome of these two processes would be similar. But we are assuming that, in the first process with the conventional What-If analysis, we are dealing with an inexperienced user, who restricts his selection of the scenario parameters to the What-If question content. If the user is not familiar with the business data, the user will not know what extra information to select as scenario parameters, which can lead to poorer results and may be not as useful when comparing to the outcome of the process with the integration of preferences. Thus, we can conclude that the hybridization process could effectively lead to a more oriented outcome, as we explain in the example case study with the comparison of the two process results (Chapter 6), using the conventional What-If analysis and the hybridization process.

7.2 Lessons Learned and Knowledge Acquired

The main research issue of the thesis is the enhance the process of view selection using What-If analysis. This hybridization process introduces a recommendation engine for assisting the user during a decision-support analysis process. A process combining OLAP preferences and What-If analysis tool is of interest to any researcher, company or entity that handles large amounts of data and intends to improve decision-making. This integration gives the user the best of both parts, namely:

- it renders possible to modify business variables to find an unexpected behavior of the system from What-If simulation
- it provides the user with the suggestion of the most adequate set of scenario parameters to add to the simulation from OLAP usage preferences.

What-If analysis allows for the user to inspect the behavior of a complex system. For just this reason, this process provides several advantages to the user. It makes possible to study the behavior of a system without building it or creating the circumstances to make it happen in a real-world system, clearly saving time and reducing costs. Another advantage is that it becomes possible to modify business variables to find an unexpected behavior of the system. With this, the business manager can be aware of the conditions that lead to an erratic behavior, and avoid them in the future.

Despite the advantages of using a What-If simulation, there is some drawbacks in this process. The integration of OLAP usage preferences will help to overcome the disadvantages that come with the use of What-If analysis. What-If analysis consists in creating hypothetical scenarios to analyze possible consequences of changing business variables. It is the user responsibility to choose the scenario parameters and if the user is an inexperienced user or even unaware of the business domain, it may become a difficult process, and lead to weak results.

Using usage preferences, the user does not need to know the business domain to choose the most adequate parameters for the What-If scenario. Another advantage that comes with using preferences is that preferences can also help to control the returned information, providing access to relevant information and eliminating the irrelevant one. Knowing beforehand usage preferences

can have a significant impact on the outcome results of the analytical system. It is possible to provide exactly the most relevant and useful information to each specific user in a specific analysis scenario.

The main differences between our approach and a conventional What-If analysis method is then to become possible to simulate a system behaviour based on past data extracted from OLAP sessions, in other words, our approach contains the process of extraction of usage preferences using association rules. Preferences can be defined based on historical data provided from a data mining system. Preferences have the ability to recommend to the user the axes of analysis that are strongly related to each other, helping to introduce valuable information in the application scenario being building.

Following this methodology, the user experience is facilitated. The choice of the scenario parameters is one of the phases that may be quite difficult to a user that is not familiar with the business data. A user that is not familiar with the data, may choose the wrong or inadequate scenario parameters. Instead of making the wrong choices or choosing only the scenario parameters included in the What-If question, our process finds and recommends the set of strongly related to the goal analysis attributes to the user. Thus, it is possible to the user to add relevant and important information to the scenario, which in a default or usual situation would not be done.

Due to this, query runtime can be enhanced against cases without preferences. There is a significant reduction of the cube implementation costs, processing time and memory usage. The cube will include in its structure only the data that match user preferences, and so it will return only the data that interest to user. Moreover, the entire analysis process can be improved. As already mentioned, a cube is a very complex data structure and it can be difficult for an analyst to acquire the information he wants. With a simple interface having the ability to recommend the right queries based on the history of past analytical sessions, the process of extracting information is much simpler. Consequently, in our process, we get more focused and refined results, which helps both a user who is not familiar with the business analysis and an analyst who is familiar with the business modelling data.

We believe that it is imperative to use formal methods to specify and validate the model due to the importance of the hybridization process when dealing with the conventional What-If scenarios process. The formal specification and validation of the model is useful to verify if the hybridization model process meets its critical requirements and provides the desired functionality without failures and inconsistencies. In this thesis, we presented and discussed the formal specification and verification of the process of extracting usage preferences in a hybrid model for enhancing What-If scenarios, in order to check for inconsistencies and prove the validity of the model.

Alloy was chosen to support such a task. We showed how to use this formal language to successfully specify the model, validate some properties of its syntax, and also illustrated an example of an instance of our hybrid model running all the Alloy specification. We also presented an example of a counter-example of a situation that comes out not following the correct behaviour of our syntax model and proposed a solution to correct the assertion that failed. After proposing additional restrictions, the new assertion was checked and no counter-example was found, meaning that our Alloy model is valid and correct within the specified scope. The main advantages of formal specification are to provide a more abstract specification of the process model and to allow the verification of the model. However, in our thesis, providing an abstract model is one of the disadvantages, because a formal specification might describe what the system can do, but cannot represent the knowledge extracted and data itself. In our case, the data transformation and recommendations extracted are the focus of our work.

Basically, our hybridization process suggests OLAP preferences to the user, providing more adequate scenario parameters to be included in the scenario in a What-If analysis process. In more detail, this process aims to discover axes of analysis that are strongly related to the user-defined goal analysis attribute, using an association rules mining algorithm, and suggest them to the user as parameters to be added to the What-If scenario. These axes of analysis are discovered using OLAP mining and cannot otherwise be discovered using a manual analysis. In the end, this integration helps the user by adding new relevant information to the What-If scenario, which means that we can enhance the process of selection of data cubes using What-If analysis.

7.3 Future Work

In this thesis, we focused on improving the What-If analysis process using prediction models. To overcome the pitfalls of the What-If process, we proposed the hybridization methodology, integrating OLAP preferences in conventional What-If analysis processes. This integration aims to ease the user experience as it can suggest a set of data cube views based on the user preferences. As the main goals of this thesis have been accomplished, there is still more that can be done.

In this work, we proposed a hybridization methodology and with prior investigation, we defined the set of techniques and tools to be used. There are several ways of extract or define user preferences, as we can see in Chapter 4. Integrating a different technique or tool to define and extract user preferences could be interesting. The same thing could be done with the choice of tool to perform the What-If analysis process. In this work, we chose to use Microsoft Office Excel and we could opt to use another tool, like Powersim Studio (Powersim.com. 2019), or other similar tool, and compare the results.

On the matter of the specification and validation of the Alloy, it is an aspect that we intend to continue improving. We showed how to use Alloy to specify the components and behaviour of the hybrid model, validating some properties of its syntax and illustrating with an example of a valid instance of our hybrid model running all the Alloy specification. The main advantages of using formal specification are to provide an abstract specification of hybridization process and allow to perform a formal verification and validation of the model. However, in our work, providing an abstract model is not enough. The formal specification might describe what the system can do but cannot represent or validate the knowledge extracted and the data itself. In our case, the data transformation and recommendations are the focus of our work. As future work, we can validate more properties of our hybrid model and we can investigate further tools in order to find a way to validate the data itself. Also, there are some aspects in the software platform we developed that can be improved and automated. For example, the user needs to manually define the minimum values to filter the association rules. If these steps are done wrong, the outcome may be inadequate. The application should analyse the extracted set of association rules and automatically define the minimum thresholds.

To the best of our knowledge, the hybridization process proposed in this thesis is a new approach (using OLAP preferences to improve the conventional What-If analysis). There is no similar framework that we can use to test and compare our results. Therefore, we focus essentially on comparing the results of the use of the hybridization process with the use of the conventional What-If analysis.

At the end, the long journey we have taken together, accomplishing what we have done in these years, allows us to hope for new horizons for the research and development of more effective and intelligent What-If analysis platforms.

References

- Agrawal, R. and Srikant, R., (1994). Fast algorithms for mining association rules. *In Proc. 20th int. conf. very large data bases, VLDB*. 1215, pp. 487-499.
- Ahmed, E. B., Nabli, A., and Gargouri, F. (2012). Building MultiView analyst profile from multidimensional query logs: from consensual to conflicting preferences. *arXiv preprint arXiv:1203.3589*.
- Aligon, J., Golfarelli, M., Marcel, P., Rizzi, S. and Turricchia, E. (2011). Mining preferences from OLAP query logs for proactive personalization. In *East European Conference on Advances in Databases and Information Systems*, pp. 84-97. Springer, Berlin, Heidelberg.
- Aligon, J. and Marcel, P. (2012). Summarizing former sessions for user-centric OLAP. In *EDA*, pp. 139-153.
- Alloytools.org. (2019). [Online] Available at: <http://alloytools.org/> [Accessed 25 Oct. 2018].
- Angelini, M., Ferro, N., Santucci, G. and Silvello, G. (2016). What-If Analysis: A Visual Analytics Approach to Information Retrieval Evaluation. In *IIR*.
- Asharani, V., Veerappa, B. N. and Rafi, M. (2015). Security evaluation of pattern classifiers in adversarial environments. *Int. J. Comput. Sci. Mob. Comput*, 4, pp. 768-774.
- Balmin, A., Papadimitriou, T. and Papakonstantinou, Y. (2000). Hypothetical queries in an olap environment. In *VLDB*. 220, pp. 231.
- Beel, J., Gipp, B., Langer, S. and Breiting, C. (2016). paper recommender systems: a literature survey. *International Journal on Digital Libraries*, 17(4), pp. 305-338.

Bellatreche, L., Giacometti, A., Marcel, P., Mouloudi, H. and Laurent, D. (2005). A personalization framework for OLAP queries. In *Proceedings of the 8th ACM international workshop on Data warehousing and OLAP*. pp. 9-18. ACM.

Beyer, K. and Ramakrishnan, R. (1999). Bottom-up computation of sparse and iceberg cube. In *ACM Sigmod Record*. 28(2), pp. 359-370. ACM.

Biondi, P., Golfarelli, M. and Rizzi, S. (2011). Preference-based datacube analysis with MYOLAP. In *Data Engineering (ICDE), 2011 IEEE 27th International Conference on*. pp. 1328-1331. IEEE.

Bird, C. and Zimmermann, T. (2012). Assessing the value of branches with What-If analysis. In *Proceedings of the ACM SIGSOFT 20th International Symposium on the Foundations of Software Engineering* (p. 45). ACM.

Bourini, I. F., al Hazza, M. H. F. and Taha, A. H. (2018). Investigation of Effect of Machine Layout on Productivity and Utilization Level: What If Simulation Approach.

Brenner, S., Zeng, Z., Liu, Y., Wang, J., Li, J. and Howard, P. K. (2010). Modeling and analysis of the emergency department at University of Kentucky Chandler Hospital using simulations. *Journal of emergency nursing*, 36(4), pp. 303-310.

Bulyonkov, M. A. and Filatkina, N. N. (2015). A research automation system for macroeconomic modeling. *Bulletin of the Novosibirsk Computing Center. Series: Computer Science*, (38), pp. 51-65.

Cambridge Dictionary | English Dictionary, T. (1999). Cambridge Dictionary | English Dictionary, Translations & Thesaurus. [online] Dictionary.cambridge.org. Available at: <https://dictionary.cambridge.org> [Accessed 28 Apr. 2018].

Carvalho, M., Belo, O., (2011). "Exploração de Cenários What-If em Plataformas de Processamento Analítico de Dados", *CAPSI'2012, Actas da CAPSI'2012 - 12ª Conferência da Associação Portuguesa de Sistemas de Informação*, Guimarães, Portugal, 7 Setembro.

Carvalho, M., Belo, O., (2016). "Enriching What-If Scenarios With OLAP Usage Preferences", In *Proceedings of The 8th International Conference on Knowledge Discovery and Information Retrieval (KDIR'2016)*, Porto, Portugal, November 9-11.

Carvalho, M., Belo, O., (2017a). "Conceiving Hybrid What-If Scenarios Based on Usage Preferences", In *Proceedings of EWG-DSS 2017 International Conference on Decision Support System Technology (ICDSSST' 2017)*, Namur, Belgium, May 29–31.

Carvalho, M., Belo, O., (2017b). "Using Alloy for Verifying the Integration of OLAP Preferences in a Hybrid What-If Scenario Application", In *Proceedings of 9th International KES-IDT Conference (KES-IDT'2017)*, Vilamoura, Algarve, Portugal, June 21–23.

Carvalho, M., Belo, O., (2017c). "Inception and Specification of What-If Scenarios Using OLAP Usage Preferences", In *Proceedings of The 11th International Conference on Soft Computing Models in Industrial and Environmental Applications (SOCO'2017)*, Leon, Spain, 6-8 September.

Carvalho, M., Macedo, N., Belo, O., (2017d). "Checking the Correctness of What-If Scenarios", In *11th IFIP WG 8.9 Working Conference – CONFENIS 2017*, Crowne Plaza Shanghai Fudan, Shanghai, China, October 18th - 20th.

Chaudhuri, S. and Dayal, U. (1997). An overview of data warehousing and OLAP technology. *ACM Sigmod record*, 26(1), pp. 65-74.

Chaudhuri, S. and Narasayya, V. (1998). AutoAdmin "What-If" index analysis utility. *ACM SIGMOD Record*, 27(2), pp. 367-378.

Chaudhuri, S. and Narasayya, V. (2007). Self-tuning database systems: a decade of progress. In *Proceedings of the 33rd international conference on Very large data bases*. pp. 3-14. VLDB Endowment.

Clarke, E. M. and Wing, J. (1996). Formal methods: State of the art and future directions. *ACM Computing Surveys (CSUR)*, 28(4), pp. 626-643.

Datta, P. P. and Roy, R. (2010). Cost modelling techniques for availability type service support contracts: a literature review and empirical study. *CIRP Journal of Manufacturing Science and Technology*, 3(2), pp. 142-157.

Deutch, D., Ives, Z. G., Milo, T. and Tannen, V. (2013). Caravan: Provisioning for What-If Analysis. In CIDR. ISO 690

Dogan, G. (2016). What-If Analysis and Debugging Using Provenance Models of Scientific Workflows. *International Journal of Engineering and Technology*, 8(6).

Feldman, J. (2016). What-If Analyzer for DMN-based Decision Models. In RuleML (Supplement).

Fouché, G. and Langit, L. (2011). What Is Business Intelligence?. Foundations of SQL Server 2008 R2 Business Intelligence, pp. 1-24.

Garrigós, I., Pardillo, J., Mazón, J. N. and Trujillo, J. (2009). A conceptual modeling approach for OLAP personalization. In *International Conference on Conceptual Modeling*. pp. 401-414. Springer, Berlin, Heidelberg.

Gavanelli, M., Milano, M., Holland, A. and O'Sullivan, B. (2012). What-If Analysis Through Simulation-Optimization Hybrids. In ECMS. pp. 624-630.

Giacometti, A., Marcel, P., Negre, E., Soulet, A. (2009). Query Recommendations for OLAP Discovery Driven Analysis. In *Proceedings of 12th ACM International Workshop on Data Warehousing and OLAP (DOLAP'09)*, Hong Kong, November 6, pp. 81-88.

Golfarelli, M., Rizzi, S. and Proli, A. (2006). Designing What-If analysis: towards a methodology. In *Proceedings of the 9th ACM international workshop on Data warehousing and OLAP*. pp. 51-58. ACM.

Golfarelli, M. and Rizzi, S. (2009). Expressing OLAP preferences. In *International Conference on Scientific and Statistical Database Management*. pp. 83-91. Springer, Berlin, Heidelberg.

Golfarelli, M. and Rizzi, S. (2010). What-if simulation modeling in business intelligence. In *Business Information Systems: Concepts, Methodologies, Tools and Applications*. pp. 2229-2247. IGI Global.

Golfarelli, M., Rizzi, S. and Biondi, P. (2011). myOLAP: An approach to express and evaluate OLAP preferences. *IEEE Transactions on Knowledge and Data Engineering*, 23(7), pp. 1050-1064.

Golfarelli, M., Mantovani, M., Ravaldi, F. and Rizzi, S. (2013). Lily: a geo-enhanced library for location intelligence. In *International Conference on Data Warehousing and Knowledge Discovery*. pp. 72-83. Springer, Berlin, Heidelberg.

Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D., Venkatrao, M., Pellow F. and Pirahesh, H. (1997). Data cube: A relational aggregation operator generalizing group-by, cross-tab and sub-totals. *Data mining and knowledge discovery*, 1(1), pp. 29-53.

Han, J. (1997). OLAP mining: An integration of OLAP with data mining. In *Proceedings of the 7th IFIP*. 2. pp. 1-9.

Harinarayan, V., Rajaraman, A. and Ullman, J. D. (1996). Implementing data cubes efficiently. In *Acm Sigmod Record*. 25(2), pp. 205-216. ACM.

Hartmann, T., Fouquet, F., Moawad, A., Rouvoy, R. and Traon, Y. L. (2018). GreyCat: Efficient What-If Analytics for Data in Motion at Scale. *arXiv preprint arXiv:1803.09627*.

Herodotou, H. and Babu, S. (2011). Profiling, What-If analysis, and cost-based optimization of mapreduce programs. *Proceedings of the VLDB Endowment*, 4(11), pp. 1111-1122.

Herodotou, H. and Babu, S. (2013). A What-if Engine for Cost-based MapReduce Optimization. *IEEE Data Eng. Bull.*, 36(1), pp. 5-14.

Hung, N. Q. V., Tam, N. T., Weidlich, M., Thang, D. C. and Zhou, X. (2017). What-if Analysis with Conflicting Goals: Recommending Data Ranges for Exploration. In *Proceedings of the VLDB Endowment*, 10(5).

Jerbi, H., Ravat, F., Teste, O. and Zurfluh, G. (2008). Management of context-aware preferences in multidimensional databases. In *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*. pp. 669-675. IEEE.

Jerbi, H., Ravat, F., Teste, O. and Zurfluh, G. (2009a). Applying recommendation technology in OLAP systems. In *International Conference on Enterprise Information Systems*. pp. 220-233. Springer, Berlin, Heidelberg.

Jerbi, H., Ravat, F., Teste, O. and Zurfluh, G. (2009b). Preference-based recommendations for OLAP analysis. In *International Conference on Data Warehousing and Knowledge Discovery*. pp. 467-478. Springer, Berlin, Heidelberg.

Jerbi, H., Ravat, F., Teste, O. and Zurfluh, G. (2010). A framework for OLAP content personalization. In *East European Conference on Advances in Databases and Information Systems*. pp. 262-277. Springer, Berlin, Heidelberg.

Jiang, Y., Sivalingam, L. R., Nath, S. and Govindan, R. (2016). WebPerf: Evaluating What-If scenarios for cloud-hosted web applications. In *Proceedings of the 2016 conference on ACM SIGCOMM 2016 Conference*. pp. 258-271. ACM.

Jouini, K. and Jomier, G. (2008). Design and analysis of index structures in multiversion data warehouses. *New trends in data warehousing and data analysis*, pp. 169-185.

Ke, J., Dong, H., Tan, C. and Liang, Y. (2017). PBWA: A Provenance-Based What-If Analysis Approach for Data Mining Processes. *Chinese Journal of Electronics*, 26(5), pp. 986-992.

Kellner, M. I., Madachy, R. J. and Raffo, D. M. (1999). Software process simulation modeling: why? what? how?. *Journal of Systems and Software*, 46(2-3), pp. 91-105.

Kießling, W. (2002). Foundations of preferences in database systems. In *VLDB'02: Proceedings of the 28th International Conference on Very Large Databases*. pp. 311-322.

Kießling, W. (2005). Preference constructors for deeply personalized database queries.

Kimball, R. and Ross, M. (2011). *The data warehouse toolkit: the complete guide to dimensional modeling*. John Wiley & Sons.

Kottemann, J. E., Boyer-Wright, K. M., Kincaid, J. F. and Davis, F. D. (2009). Understanding decision-support effectiveness: A computer simulation approach. *IEEE Transactions on Systems, Man and Cybernetics-Part A: Systems and Humans*, 39(1), pp. 57-65.

Koutsoukis, N. S., Mitra, G. and Lucas, C. (1999). Adapting on-line analytical processing for decision modelling: the interaction of information and decision technologies. *Decision support systems*, 26(1), pp. 1-30.

Kozmina, N. (2015). Producing report recommendations from explicitly stated user preferences. *Baltic Journal of Modern Computing*, 3(2), pp. 110.

Kozmina, N. and Niedrite, L. (2011). Research Directions of OLAP Personalization. In *Information Systems Development*. pp. 345-356. Springer, New York, NY.

Kozmina, N. and Solodovnikova, D. (2011). Towards introducing user preferences in olap reporting tool. In *International Conference on Business Informatics Research*. pp. 209-222. Springer, Berlin, Heidelberg.

Krishnamoorthy, M., Brodsky, A. and Menascé, D. A. (2014). Temporal manufacturing query language (tMQL) for domain specific composition, What-If analysis and optimization of

manufacturing processes with inventories. Department of Computer Science, George Mason University, Fairfax, VA, 22030.

Li, X., Han, J. and Gonzalez, H. (2004, August). High-dimensional OLAP: a minimal cubing approach. In *Proceedings of the Thirtieth international conference on Very large data bases*. 30. pp. 528-539. VLDB Endowment.

Lu, J., Wu, D., Mao, M., Wang, W. and Zhang, G. (2015). Recommender system application developments: a survey. *Decision Support Systems*, 74, pp. 12-32.

Marcel, P. (2012). *Leveraging query logs for user-centric OLAP* (Doctoral dissertation, université de Tours).

Meurice, L., Nagy, C. and Cleve, A. (2016). Detecting and preventing program inconsistencies under database schema evolution. In *Software Quality, Reliability and Security (QRS)*, 2016 IEEE International Conference on. pp. 262-273. IEEE.

Moreno, A., Segura, A., Zlatanova, S., Posada, J. and García-Alonso, A. (2012). Introducing GIS-based simulation tools to support rapid response in wildland fire fighting. *WIT Transactions on Ecology and the Environment*, 158, pp. 163-174.

Moreno, M. N., Segrera, S., López, V. F. and Polo, M. J. (2007). Improving the Quality of Association Rules by Preprocessing Numerical Data. In *II Congreso Español de Informática*, Zaragoza, 11 al 14 de septiembre.

Moskewicz, M. W., Madigan, C. F., Zhao, Y., Zhang, L. and Malik, S. (2001). Chaff: Engineering an efficient SAT solver. In *Proceedings of the 38th annual Design Automation Conference*. pp. 530-535. ACM

Ore, O. (1962). *Theory of graphs* (Vol. 38). American Mathematical Society.

Papastefanatos, G., Anagnostou, F., Vassiliou, Y. and Vassiliadis, P. (2008). Hecataeus: A What-If analysis tool for database schema evolution. In *Software Maintenance and Reengineering*, 2008. CSMR 2008. 12th European Conference on. pp. 326-328. IEEE.

Powersim.com. (2019). Powersim Studio | Powersim Software. [online] Available at: <https://www.powersim.com/> [Accessed 28 Oct. 2018].

Products.office.com. (2019). Software de Folha de Cálculo – Avaliação Gratuita do Excel – Microsoft Excel. [online] Available at: <https://products.office.com/pt-pt/excel> [Accessed 28 Oct. 2018].

Quinlan, J. R. (1993). *C4. 5: Programs for empirical learning*.

Ravat, F. and Teste, O. (2009). Personalization and OLAP databases. In *New Trends in Data Warehousing and Data Analysis*. pp. 1-22. Springer, Boston, MA.

Ricci, F., Rokach, L. and Shapira, B. (2015). Recommender systems: introduction and challenges. In *Recommender systems handbook*. pp. 1-34. Springer, Boston, MA.

Rizzi, S. (2007). OLAP preferences: a research agenda. In *Proceedings of the ACM tenth international workshop on Data warehousing and OLAP*. pp. 99-100. ACM.

Rizzi, S. (2010). New frontiers in business intelligence: distribution and personalization. In *East European Conference on Advances in Databases and Information Systems*. pp. 23-30. Springer, Berlin, Heidelberg.

Rome, E., Doll, T., Rilling, S., Sojeva, B., Voß, N. and Xie, J. (2016). The Use of What-If Analysis to Improve the Management of Crisis Situations. In *Managing the Complexity of Critical Infrastructures*. pp. 233-277. Springer International Publishing.

Rozema, L. (2016). *Extending the control tower at ShipitSmarter: Designing a tool to analyse carrier performance and perform What-If analyses* (Master's thesis, University of Twente).

Saxena, G., Narula, R. and Mishra, M. (2013). New Dimension Value Introduction for In-Memory What-If Analysis. arXiv preprint arXiv:1302.0351.

Singh, R., Shenoy, P., Natu, M., Sadaphal, V. and Vin, H. (2013). Analytical modeling for What-If analysis in complex cloud computing applications. *ACM SIGMETRICS Performance Evaluation Review*, 40(4), pp. 53-62.

SQL Server Blog. (2016). WideWorldImporters: The new SQL Server sample database - SQL Server Blog. [Online] Available at: <https://cloudblogs.microsoft.com/sqlserver/2016/06/09/wideworldimporters-the-new-sql-server-sample-database/> [Accessed 5 Mar. 2017].

Support.office.com. (2019). Create a PivotTable to analyze worksheet data. [online] Available at: <https://support.office.com/en-gb/article/create-a-pivottable-to-analyze-worksheet-data-a9a84538-bfe9-40a9-a8e9-f99134456576> [Accessed 28 Jan. 2019].

Timar, S., Peters, M., Davis, P., Lapis, M. B., Wilson, I., van Tulder, P. and Smith, P. (2017). A What-If analysis tool for planning airport traffic. In *Digital Avionics Systems Conference (DASC), 2017 IEEE/AIAA 36th*. pp. 1-9. IEEE.

Tsunokawa, K., Van Hiep, D. and Ul-Islam, R. (2006). True Optimization of Pavement Maintenance Options with What-If Models. *Computer-Aided Civil and Infrastructure Engineering*, 21(3), pp. 193-204.

Van Cauwelaert, S., Lombardi, M. and Schaus, P. (2017). A visual web tool to perform What-If analysis of optimization approaches. arXiv preprint arXiv:1703.06042.

Van den Akker, M., Brinkkemper, S., Diepen, G. and Versendaal, J. (2008). Software product release planning through optimization and What-If analysis. *Information and Software Technology*, 50(1), pp. 101-111.

Wickramasuriya, R., Ma, J., Somashekar, V., Perez, P. and Berryman, M. (2013). SMART Infrastructure Dash-board: A Fusion between Business Intelligence and Geographic Information Systems.

Xiao, Y., Zhang, Y., Wang, S. and Chen, H. (2009a). Efficient Incremental Computation of CUBE in Multiple Versions What-If Analysis. *Advances in Data and Web Management*, pp. 235-247.

Xiao, Y., Zhang, Y., Wang, S. and Chen, H. (2009b). Incremental Computation for MEDIAN Cubes in What-If Analysis. *Advances in Data and Web Management*, pp. 248-259.

Xin, D. and Han, J. (2008). P-cube: Answering preference queries in multi-dimensional space. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*. pp. 1092-1100. IEEE.

Xu, H., Luo, H. and He, J. (2013). What-if query processing policy for big data in OLAP system. In *Advanced Cloud and Big Data (CBD), 2013 International Conference on*. pp. 110-116. IEEE.

Zarras, A. V., Vassiliadis, P. and Issarny, V. (2008). Modelling and analysing reliable service-oriented processes. In *International Journal of Business Process Integration and Management*, 3(3), pp. 147-163.

Zhou, G. and Chen, H. (2009). What-if analysis in MOLAP environments. In *Fuzzy Systems and Knowledge Discovery, 2009. FSKD'09. Sixth International Conference on*. 2, pp. 405-409. IEEE.

