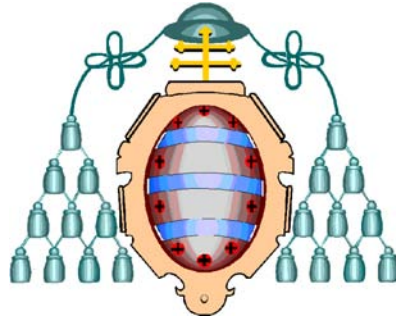


UNIVERSIDAD DE OVIEDO



DOCTORADO EN
AVANCES EN INFORMÁTICA

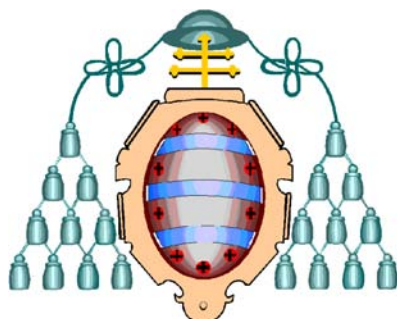
TESIS DOCTORAL

Energy Advisory Services at Households

by Carlos Menéndez González

May, 2014

UNIVERSIDAD DE OVIEDO



DOCTORADO EN
AVANCES EN INFORMATICA

TESIS DOCTORAL

Energy Advisory Services at Households

by Carlos Menéndez González

Advisors:

Luciano Sánchez Ramos– Universidad de Oviedo

Gracián Triviño Barros – European Centre for Soft Computing

May, 2014

To Alvaro and Sofia

*“Pity the poor speaker. All this information,
and more, must be packed into each clause!”*

E. Hovy, 1994

Abstract

The reduction of energy intensity is a prior objective to achieve a sustainable development. The company EDP, among other measures, launched the project "Ecofamilias", with the objective of providing useful information about energy consumption patterns to households.

Computational systems are able to store huge amounts of data from the environment. Nevertheless, their ability to extract the relevant knowledge from this data and express it in understandable language showing human-like behaviour remains limited. Current interface technology is still closer to the use of command language or click interaction rather than to conversational interactions.

For companies, having personalized contact with customers suppose a step forward to increase customer satisfaction and would like to move to more intelligent interaction with the users. Avatars are computer characters that show human-behaviour. Quite often are used to address online assistants able to answer questions and perform tasks through conversational dialogs with the users. There are some industrial applied virtual agents but they are closer to a string pattern matching process of questions and answers and lack the ability to deal with numeric information.

In the research line of Computational Theory of Perceptions has been defined the Granular Linguistic Model of a Phenomenon (GLMP). This data structure allows computational systems to generate linguistic descriptions of input data. It grounds on the fact that human cognition is based on the role of perceptions in order to perform physical and mental tasks without any traditional measurements and computations. It is argued, for example, that while parking a car we do not measure the distance in meters or degrees in which we have to move the car or turn the wheels, but instead we use perceptions of time, direction or speed, e.g., "*if we are close to the other car*" or "*if there is enough space*". While communicating in order to increase the effectiveness and acceptance speakers tailor text to the hearer and the actual situation. This enables to include more information than is contained in the literal meaning of the words for example including emotional contents.

With the objective of helping customers in the interpretation of the use of energy the aim of this Thesis is to define a Personalized Energy Advisor. It is a computational system able to assist users in achieving energy efficiency simulating the actions that would follow an expert in energy efficiency using a knowledge domain.

The approach developed in the thesis suppose a shift in the ability to dynamically

learn, extract, deduce and communicate energy information as requested by users from huge amounts of data available.

In the thesis are proposed different extensions to existing paradigms around natural language processing.

It is proposed a model for dynamically generating Granular Linguistic Models retrieving requested data from a database, and translating these data into linguistic expressions in a coherent and cohesive discourse. Linguistic capabilities include the ability to describe and identify relevant information, and to extract reasoned advice from input data. Expressiveness is extended by selection of the most suitable sentences, pruning non relevant perceptions and by aggregating related ones avoiding redundancy. Also a Question Answer environment based on perceptions is proposed to allow dynamically the interpretation of user questions and the dynamic computation of answers in a dialog with the user. The model includes the ability to process emotions, and therefore show emotional behaviour dynamically computed based on the conversation and the user behaviour.

Each of this proposed solutions placed together in a global architecture will be used to define the Autonomous Personal Energy Advisor.

It is defined a software to implement the solutions proposed. It consists in an application framework that allows defining, developing, executing and maintaining a natural language interface advisor for queering energy consumption knowledge. The framework allows to reduce design and implementation gaps and time of development.

A description of a software methodology is presented for covering the process involved in the development of the personal energy advisor system using the framework of the solution. To interpret energy consumption information is not trivial, the acquisition of knowledge from an expert advisor is a complex task. A relevant step in the methodology is the acquisition of knowledge from the environment, and the ability to translate this knowledge into an implementable design.

The results of the proposed extensions and the system framework are used to define the Personal Energy Advisor. Different scenarios of information requests and the resulting reports in a dialog interaction with the user are presented to show the functionality.

Preface

Over the years that this research has been carried out, I have kept in mind users, the energy efficiency and the development of computational abilities to deliver friendly human-like computer interaction.

The task of writing this thesis would not have been completed without the support of EDP and the European Center for Soft Computing.

I have to thank personally to Gracian Trivino Principal Researcher in ECSC, Luis Santos Moro head of I+D+i department in EDP Spain and Luciano Sanchez Ramos at Oviedo University.

Gracian has guided me over the years. Luis has supported the work. Luciano has made it possible.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Objectives	1
1.3	Scope	2
1.4	Energy Efficiency	5
1.5	Applied Technologies	7
1.6	Contributions	9
1.7	Contents	12
2	Energy Efficiency	15
2.1	Introduction	15
2.2	Efficiency in the European Union	16
2.3	The Smart Grid	17
2.3.1	Benefits	19
2.3.2	Applications	19
2.4	Household smart energy monitoring	22
3	Introduction to the applied Technologies	25
3.1	Introduction	25
3.2	Computational Theory of Perceptions	25
3.2.1	General GLMP architecture	26
3.2.2	Computational Perceptions (CP)	27
3.2.3	Types of CP	28
3.2.4	Perception Mapping (PM)	28
3.2.5	Granular Linguistic Model of Phenomena (GLMP)	29
3.3	On-Line Analytical Processing (OLAP)	30
3.3.1	Semantic Meta-data for OLAP Schema Definition	32
3.3.2	OLAP Schemas and Computational OLAP Model	32
3.3.3	Computational OLAP Model Operations	33
3.4	Affective Computing	34
3.4.1	Modeling emotions	34
3.4.2	Expressing emotions	36
3.5	Language generation	37
3.5.1	Gricean Maxims	39
3.5.2	Discourse Coherence and Cohesion	40
3.6	Question Answering Systems	41
3.6.1	Perceptions in Question Answering Systems	42

3.6.2	Question Interpretation	44
3.7	Virtual Assistants	45
3.8	Architecture Design	46
4	Extension proposed to the applied technologies	51
4.1	Introduction	51
4.2	Dynamic navigation through perceptions	54
4.2.1	GLMP OLAP Schema	55
4.2.2	GLMP OLAP Application	58
4.3	Dynamic discourse generation based on perceptions	64
4.3.1	Gricean Maxims	66
4.3.2	Computational Perceptions with relevance	66
4.3.3	Perception Mapping with relevance	67
4.3.4	Content Selection	68
4.3.5	Content Planning	70
4.3.6	Text Realization	74
4.3.7	Generation Process Application	74
4.4	Emotional perceptions	76
4.4.1	Emotional-GLMP	77
4.4.2	Expressing emotions	78
4.4.3	FFSM for modelling emotions	78
4.4.4	Modeling and Computing Emotions	80
4.4.5	Computational Perception with emotional states	81
4.4.6	Perception Mapping	82
4.4.7	Emotional-GLMP Application	83
4.5	Question Answering perceptions processing	86
4.5.1	Answer Generation	88
4.5.2	QA Application	89
4.6	Energy advisor architecture	91
4.6.1	Off-line information and On-line processes	91
4.6.2	Concepts, Procedures and Attitudes	92
4.6.3	Physical Level	93
4.6.4	Symbolic Level	94
4.6.5	Reactive Level	94
4.6.6	Structural Level	95
4.6.7	Conceptual Level	95
4.6.8	Procedural Level	96
4.6.9	Attitudinal Level	96
5	Energy Advisor Development Framework	97
5.1	Introduction	97
5.2	Objectives	98
5.3	Functional Components	99
5.4	Functional Architecture	101
5.5	Designer Interface. Knowledge domain definition	102
5.5.1	Metadata Definition of the knowledge domain	104

5.6	User interface to query the knowledge domain	108
5.6.1	Report Definition interface	110
5.6.2	Report Execution	113
5.6.3	Avatar interface	118
6	Energy Advisor Development Methodology	123
6.1	Introduction	123
6.1.1	Context Definition for the application	124
6.2	Methodologies in Natural Language Generation	125
6.3	Software Development and Knowledge Discovery Methodologies	125
6.3.1	Software Development Methodologies and Unified Process	126
6.3.2	Knowledge discovery Methodologies	129
6.4	Personal Energy Advisor methodology	130
6.4.1	Work Team	131
6.4.2	Phases in the Methodology for the Personal Energy Advisor	132
6.5	Methodology Application Personal Energy Advisor	134
6.5.1	Business Modeling	134
6.5.2	Requirements	135
6.5.3	Efficiency Analytical Modelling and Energy Advisor Development	136
7	Energy Advisor Efficiency Analytical Modelling	143
7.1	Introduction	143
7.2	Activities	143
7.3	Analytic Requirements	151
7.3.1	Report data matching	153
7.4	Data Identification and Understanding	154
7.5	Data Preparation	155
7.5.1	Data Evaluation	155
7.5.2	Data Cleaning	155
7.5.3	Data Transformation	156
7.5.4	Design the data model	156
7.6	Analytic modelling	159
7.6.1	Information Discovery	159
7.6.2	Customer behavior context analysis	162
7.6.3	Customer behavior segmentation	164
7.7	Report Refinement	167
7.8	Evaluation	168
7.9	Build the data model	169
8	Energy Advisor Development	171
8.1	Introduction	171
8.2	Activities	171
8.3	Analysis	172
8.3.1	Data Model Review	173
8.3.2	Use Cases Review	174
8.4	Design	176
8.5	Implementation	177

8.6	Test	177
8.7	Deployment	178
9	Functionality of the Personal Energy Advisor	183
9.1	Introduction	183
9.2	Users	184
9.3	Channels	184
9.4	Avatar functionality	185
9.4.1	Conversation about general topics	185
9.4.2	Advising on energy use	186
9.4.3	Communicating Emotions	187
9.4.4	Expressing extended contents	188
9.4.5	Dialogue through information	188
9.5	Report functionality	189
9.5.1	Report	189
9.5.2	Report Group	191
9.5.3	Predefined Reports	191
10	Concluding Remarks	197
10.1	Review	197
10.2	Contributions	198
10.3	Articles	199
10.4	Future work	200
	Bibliography	201

List of Figures

1.1	Innovation process and context for the development of energy advisory services “The market generates information and necessities that smartly transformed generates innovative services to meet customer claims” . . .	2
1.2	Process followed that transforms information into natural language iteration: Measure, Learn, Know and Act	3
1.3	Conceptual definition of the solution proposed for the user.	5
1.4	Electrical Grid.	7
2.1	Estimated energy consumption reduction potential in 2020 (Source: European Commission)	16
2.2	Households Demand in Spain (Source:IDAE)	17
2.3	Energy chain stages, technologies and applications of the Smart Grids. .	18
2.4	CO2 Emissions	19
2.5	Smart Grids Benefits	20
2.6	Smart Grid Application	21
2.7	PeakLoad in demand curve	22
2.8	Customer demand profile vs profile for energy price.	23
3.1	General architecture of computational systems for generating linguistic description of data	27
3.2	Example of Granular Linguistic Model of a Phenomenon.	29
3.3	Second example of Granular Linguistic Model of a Phenomenon.	30
3.4	Example of Granular Linguistic Model of a Phenomenon instantiation. .	31
3.5	OLAP multidimensional data model	31
3.6	Computational OLAP model operations	33
3.7	Simplified version of the Plutnik Wheel	35
3.8	States of the FFSM	36
3.9	Membership function example.	43
3.10	Fuzzy graph.	43
3.11	GDF, Scottish Power, YellowStrom and EDF Virtual assistants examples.	46
3.12	Architecture of the CPA7L	49
4.1	Information sample	52
4.2	Modules based on GLMP for the definition for the Personal Energy Advisor	54
4.3	From OLAP to GLMP	55
4.4	OLAP multidimensional schema definition, the data cube.	56
4.5	GLMP OLAP Schema definition. The perceptions data cube.	57

4.6	GLMP-OLAP data cube that allows to generate the sentence “The air conditioning, in summer, during midday, has quite high consumption”.	57
4.7	GLMP-OLAP scheme of the application prototype.	60
4.8	Graphical representation of the answers to queries in the application example.	60
4.9	Q1. Device YearsOld	62
4.10	Q2. Low Cost Consumption in 2010	63
4.11	Q3. Efficiency	64
4.12	Process followed for language generation through perceptions in the GLMP.	65
4.13	Example of Granular Linguistic Model of a Phenomenon.	68
4.14	GLMP Layout defined by dimensional constraints on perceptions.	70
4.15	Granulation in the GLMP dynamically.	71
4.16	Suitability Pruning Layout and Example of Sentence Instantiation.	72
4.17	Redundancy Pruning Layout and Example of Sentence Instantiation.	72
4.18	GLMP Aggregation Layout and example of Sentence Instantiation.	73
4.19	Linguistic Expressions Instantiation	76
4.20	Suitability assignment for perception instances.	77
4.21	Emotions inputs and outputs	78
4.22	States of the FFMSM	79
4.23	Example of computing emotions (Character by Hapteck).	80
4.24	GLMP Functional modules for computing emotions	80
4.25	Example of computational perception emotional states	82
4.26	Example of a computational perception that models the consumption efficiency	82
4.27	Example of GLMP for a high efficiency rate.	84
4.28	GLMP QA deduction process	87
4.29	Linguistic Expressions Instantiation	90
4.30	Functional architecture for the autonomous personal advisor.	92
4.31	CPA7L architecture elements for the definition of the Personal Energy Advisor.	94
4.32	Visual representation of the five emotional states of the virtual agent	95
5.1	Objective of improvement in the communication. From the bill to a closer customer interaction insights.	99
5.2	Functional definition of the application framework.	100
5.3	Information is transformed from numeric tables and charts, to a natural language description.	101
5.4	Application framework interfaces environments available for designer and users.	102
5.5	Advisory Report and Avatar interfaces for users to request information from the Personal Energy Advisor.	103
5.6	Application metadata for the physical layer definition.	104
5.7	Discourse Perceptions defined in the application framework.	106
5.8	Application metadata for the conceptual layer definition.	108
5.9	Application metadata for the Language layer.	109

5.10	Sample report generated by the application framework with contents expressed by charts, tables and natural language that answers user request about monthly consumption.	110
5.11	Interface for report definition.	111
5.12	Report processing stages, from its definition to the results.	112
5.13	Report definition and the execution generating a SQL to access the database.	114
5.14	SQL for data retrieval and report tree generation.	116
5.15	Discourse organization layout.	117
5.16	Discourse enrichment with complex perceptions.	118
5.17	Discourse pruning over perceptions.	119
5.18	Discourse tree instantiation in natural language.	120
5.19	Report results example. Consumption evolution per day period for different users.	121
5.20	QA Process, from the user natural language request, to the answer.	121
6.1	Four key principles for the definition of the Personal Energy Advisor.	124
6.2	Steps in UP	127
6.3	Overview of the activities to develop with the Personal Energy Advisor Framework.	133
6.4	Planning	139
6.5	Initial report definition.	140
6.6	Use case for efficiency report	141
7.1	Modeling stages, analytic Stages to build the analytical model.	145
7.2	Overall methodology with the analytic stages in red, namely detailed description of use cases, detailed description of subuse cases, data model definition and report redefinition.	146
7.3	Tariffs period.	147
7.4	Use case for efficiency report	148
7.5	Energy Watch (Registered by EDP)	149
7.6	From data to detailed information to support users decisions.	149
7.7	Membership definition of fuzzy values	150
7.8	Report decomposition into topics and association of analytical requirements.	151
7.9	Text Report topic for day period consumption.	152
7.10	Example of a report sentence decomposition	152
7.11	Detailed description of the use case for day period consumption.	153
7.12	Hourly consumption measure	154
7.13	Sociodemographic data related to the user. Print screen of the application framework.	155
7.14	Dimensional data model definition.	157
7.15	Modeling stages - Multidimensional data model.	158
7.16	Hourly consumption load curve	160
7.17	Hourly consumption load curve for a set of customers.	161
7.18	Hourly consumption load curve for a subset of customers.	161
7.19	Hourly consumption load curve by day.	162
7.20	Hourly consumption load curve by day and customer in detail.	163
7.21	Impact on behavior of external factors	164

7.22	Modeling stages.	165
7.23	Customer segmentation.	166
8.1	Overall methodology with the development stages in red, namely perceptions design, conceptual implementation, linguistic definition, testing and evaluation.	172
8.2	Methodology.	173
8.3	Use cases review.	174
8.4	Analysis definition for Consumption.	175
8.5	Analysis definition for Efficiency.	176
8.6	Design definition for Consumption.	178
8.7	Design definition for Efficiency.	179
8.8	Implementation for Consumption.	180
8.9	Implementation for Efficiency.	181
8.10	Detailed steps in perceptions generation for the content and discourse tree are facilitated in the framework for testing purposes.	181
9.1	Access to the Personal Energy Advisor and users profiles.	184
9.2	Personal Energy Advisor scenario for General knowledge.	186
9.3	Personal Energy Advisor scenario for natural language and emotions.	187
9.4	Personal Energy Advisor scenario for natural language and extended answers.	188
9.5	Personal Energy Advisor scenario of interaction for Angel, Matias and Javier.	189
9.6	Personal Energy Advisor report interface.	190
9.7	Personal Energy Advisor report group interface.	193
9.8	Efficiency report definition for Javier.	194
9.9	Efficiency report definition for Matias.	195
9.10	Efficiency report definition for Angel.	196

Chapter 1

Introduction

1.1 Motivation

The thesis is motivated by an environment that could be defined by three facts, (1) the reduction of energy intensity is a prior objective to achieve a sustainable development, (2) the spread of digital technology in the electricity grid provides the ability to measure more detailed information about consumption at households, and (3) for users, to interpret energy consumption information in order to be efficient and get savings is not trivial. Based on these facts the thesis proposes a model to use the data available about energy consumption in order to generate innovative services to the customer delivering personal advisory services to promote efficiency as showed on Fig. 1.1.

Information about consumption is affected by several internal and external factors. The company EDP launched different programs to promote energy efficiency. One of the initiatives is the project “*Ecofamilias*”. This project has the objective of providing useful information about consumption patterns to households.

In companies every customer plays a key role, and is integrated in the business model. One of the most common techniques that have been developed to include customers in the business model is Customer relationship management (CRM) (21) (5). It involves using technology to organize, automate and synchronize customer service activities. Companies would like to move to a more intelligent interaction with their customers.

1.2 Objectives

With the objective of helping customers in the interpretation of the use of energy, and in identifying paths to help to improve their efficiency the aim of this thesis is to design and develop a Personalized Energy Advisor. It is a computational system able to assist users in achieving energy efficiency simulating the actions that would follow an expert looking for logical solutions more appropriate for each household. The computational system should be able to generate advice reports personalized for customers or interact with users in a dialog using natural language and integrating emotional contents. The approach de-

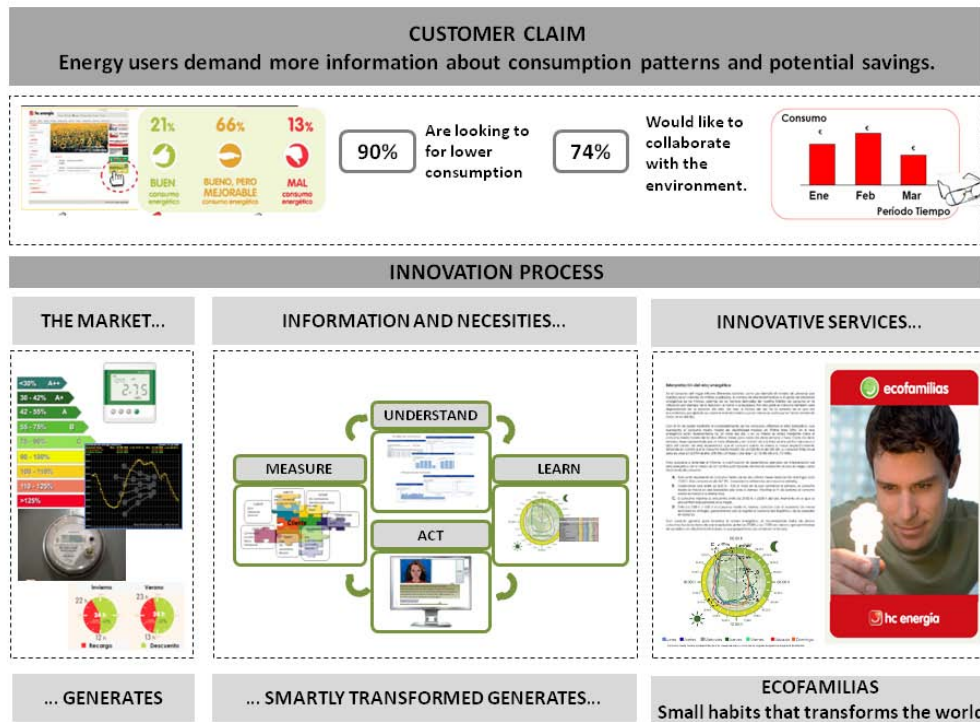


Figure 1.1: Innovation process and context for the development of energy advisory services “The market generates information and necessities that smartly transformed generates innovative services to meet customer claims”

veloped suppose a shift in the ability to dynamically understand, learn, extract, deduce and communicate information as requested by users from huge amounts of data available.

By means of natural language processing, a description of the current situation of the customer could be generated on easy terms to understand. Natural language is much better understandable while communicating than using a traditional table layout with numbers, or charts that need to be analyzed to get a description and where suggestions in most of the cases can be extracted just by an expert. While communicating in natural language the key is the ability to get the reaction of the customer. For customer reaction is required to identify the relevant advice to communicate using reasoning and to gain empathy showing closer human behaviour using emotions.

1.3 Scope

The thesis covers the design of the Personal Energy Advisor as a virtual agent to make recommendations for the customers, based on the hourly power consumption data. The personal advisor simulates the actions that an expert would do generating reports with advice contents, or implementing responses in an interactive dialog with the user. It implements the ability to transform the data into information to communicate in natural language. The design of the avatar interface and the voice generation technology is out

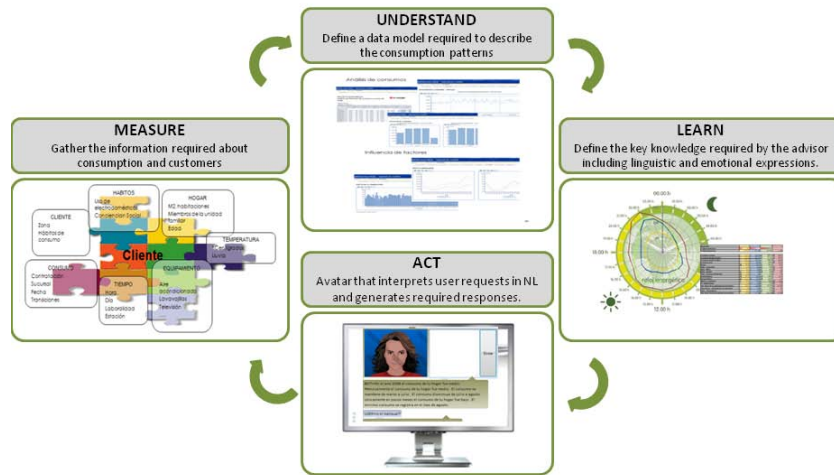


Figure 1.2: Process followed that transforms information into natural language iteration: Measure, Learn, Know and Act

of the scope of the thesis.

In the thesis is analyzed the actual energy efficiency context, the actual situation of the electrical system, and the developments expected in the electricity grid with introduction of digital technologies. Focusing in consumption efficiency different approaches are described to give feedback about domestic energy consumption, customer behaviour and communication of information.

Focusing in communication and tasks related to natural language generation and interpretation there is a review of different paradigms and studies that have been identified and used in the thesis. Based on existing technologies different extensions to previously referred existing paradigms are proposed.

Based on the extended paradigms an architecture and a computational system are developed to support the Autonomous Personal Energy Advisor. The advisor is able to deliver information to customers in a real environment. The thesis describes the software developed as a framework, based on concepts, procedures and attitudes using the referenced architecture *CPA7L*. The framework provides two different environments, one for the designer to define the knowledge domain and another for the user to execute user queries and get responses with personalized advice with an avatar interactive interface or due to predefined advice reports.

It is also defined a specific methodology to formalize the development of the Personal Energy Advisor, describing the analytic process to get the expert knowledge and develop the application over the framework to achieve desired results. To model the Personal Energy Advisor is required to identify the knowledge for the advisor. Secondly, it is required a system able to capture this knowledge, and a system able to process the knowledge and interact with a user being able to provide the required information dynamically and effectively as an expert would do.

The personal energy advisor learns with every new consumption measure and uses the overall users global knowledge to identify and adapt personal advices within a context of reference.

Process Covered

The advisory service required for the virtual agent is covered by a process of measuring the data, understanding the information, learning knowledge and interacting with the users as showed on Fig. 1.2.

- Measure data.
 - Reception and storage of the information gathered by Smart Meters.
 - Identification and collection of Customer data.
- Understand information.
 - Definition of the data model, storage and structure for exploration.
 - Identification of key elements and its influence on consumption behaviour and users savings.
- Learn Knowledge.
 - Knowledge definition for the energy advisor.
 - Computational linguistics and emotional content.
- Interaction.
 - Customer advisory services.
 - Advisory reports definition.
 - Dialog interaction.

Abilities Covered

The abilities of the virtual agent to generate responses based on user request as showed on Fig. 1.3 are:

- Describe in natural language the consumption patterns in order to lead the customer to a better use of the energy.
- Using a personal analysis, identify key patterns relevant to each user.
- Advice encouraging energy savings, reducing consumption and increasing device efficiency
- Perform the advisory tasks in a dynamic environment, with the ability to establish a dialog with the user using natural language in a human-like interaction showing emotional behaviour and generating personalized advisory reports.

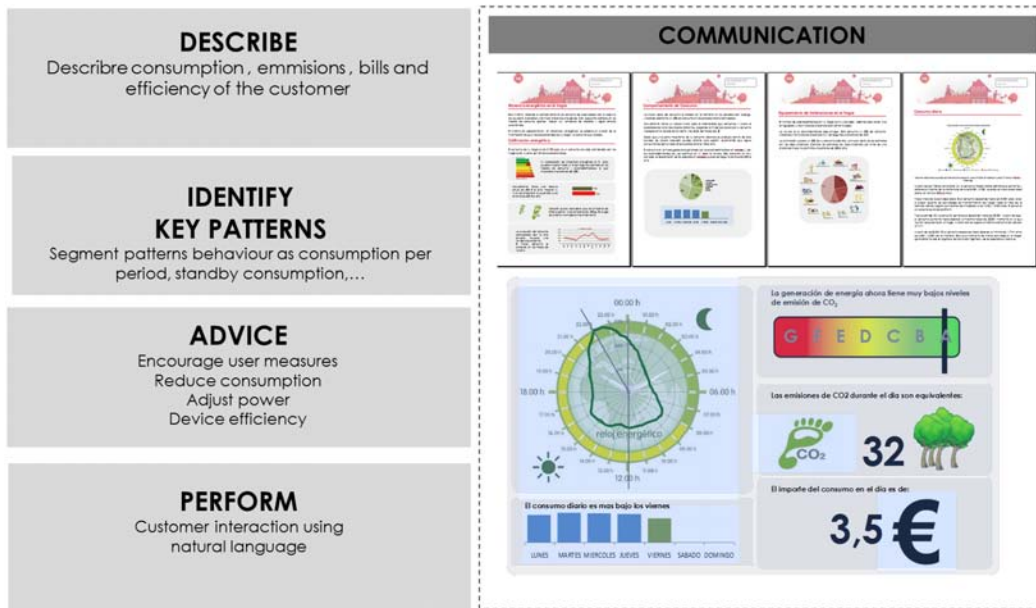


Figure 1.3: Conceptual definition of the abilities proposed for customer advice.

1.4 Energy Efficiency

The reduction of energy intensity is a prior objective to achieve a sustainable development. With this purpose measures and policies have been defined to improve the energy system and promote energy efficiency.

The electrical grid is an interconnected network that delivers electricity from its generation to its final consumption by users. The actual grid is experimenting a transformation, and faces the opportunity of improvement in many topics based on the movement from an analogical to a digital management, e.g., the integration of renewable energy or cogeneration in the grid.

The application of digital technologies to the electrical grid traditionally based on analog technology models the basis for the Smart Grid. The smart grid is not one specific technology, but a suite of technologies that can improve the performance and control over the grid. Smart Meters are an important part of the Smart Grid. They are advanced meters that measure energy consumption more accurately than conventional ones, being able to store the energy consumption with an hourly detail and transfer consumption in real time. This situation allows gathering data to know how the energy is used, rather than just provide the energy that was used.

One of the main focuses for energy efficiency is the consumption reduction, encouraging users to develop improvements in consumption patterns and therefore develop a rational use of energy maintaining the comfort level.

The company EDP launched different programs to promote energy efficiency. One of the initiatives is the project “*Ecofamilias*”. This project has the objective of providing

useful information about consumption patterns to households. “*Ecofamilias*” is an informative program to promote energy efficiency at households with the digital electricity smart meters installed, capitalizing the information gathered. It is intended to be a seed to improve understanding of our customers on energy consumption and raise awareness of their daily energy use. During a six months period, EDP conducted a pilot test on 30 families with the new meters in order to identify new sources of information, and to test how a higher level of information may affect their consumption habits.

It was also launched a program with an energy assessment test. The result for 14.000 evaluations, were that 13% of customers have a bad energy behaviour, 66% could improve energy efficiency and only 21% is performing right in energy efficiency. The findings of the study based on the information items should be useful to provide new saving tips to complement those known today.

According to the studies about energy consumptions habits, 90% of customers are willing to consume less energy and 76% would like to do more for the environment, but do not know what to do.

Energy is consumed through a variety of devices and consumers hardly know how do they use the energy, how can get savings, how can improve energy efficiency or how could they contribute to the energy system. New detailed data about consumption will be available with the smart grid. The ability to transform this data in easily interpretable natural language for users, extracting remarkable information, will be a key in the process of identifying paths to improve efficiency. New challenges for future development like electric vehicle or cogeneration will increase the user relation with the energy and will increase the requirement of this kind of advisory information systems.

To interpret energy consumption information is not trivial. Information about consumption is affected by internal and external factors. Due to the high number of factors that have influence, it is difficult for households to interpret their efficiency. It is therefore required systems able to structure high volumes of data and to identify a personalized advice.

Internal factors include for example sociodemographic characteristics of the household, like the number of members that live at home, the number of square meters, the number of electric devices and their efficiency or the way of usage.

External factors like the geographic area of the household, or the temporal period, season, month or even the week day determine the final consumption, e.g., it is consumed more in winter than in summer because of the less light hours during the day.

The main problems with consumption data is not just the lack of detail but also the lack of context around the information provided and therefore the difficulties for users to interpret and make sense of the information. Having the information of two households where, for one the consumption is 2.000 kWh per year and for the other 3.500 kWh, a basic analysis could make us fail concluding that the one with higher consumption is less efficient, e.g., if the one with higher consumption is a household of a family of five members and the other one is a household of a single.

Comparison of behaviour in time, and comparison with similar households in behaviour should be a way of positioning each individual pattern in a context that could

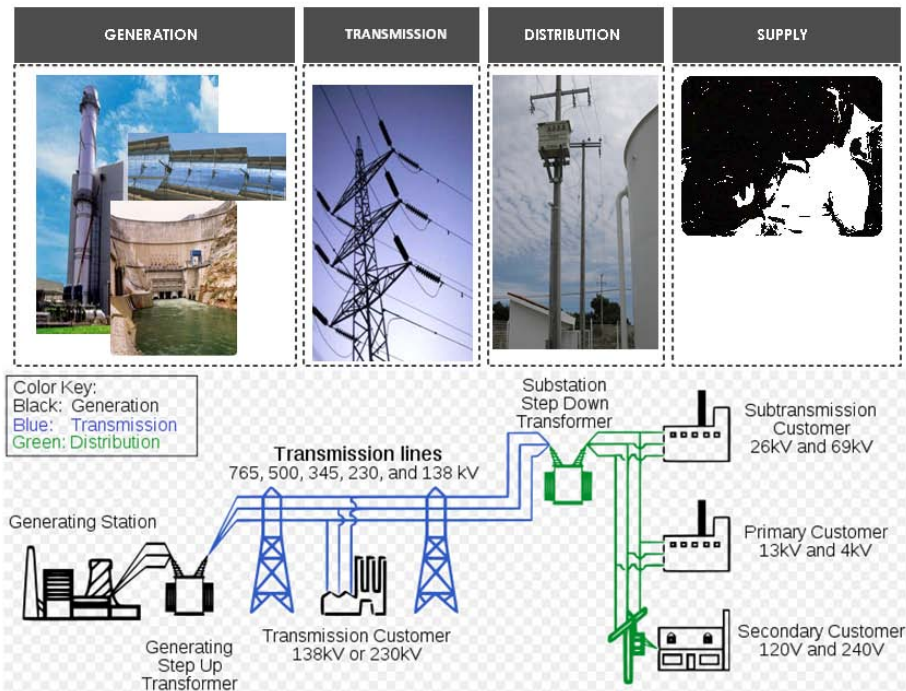


Figure 1.4: Electrical Grid.

be used as a reference for the users and make improvements.

Focusing on energy smart appliances, there are some studies about how users could interact with energy monitoring tools, or how can these tools help users to improve behaviour and get savings. Hargreaves (58) explored how UK householders interacted with feedback on their domestic energy consumption using real-time displays. Shove studied the sociology of consumption and technology to investigate the evolution and social meaning (110)(111).

Companies should move to more intelligent interaction with the users. Technological innovations should enable smart homes, moving users to a new relationship with their households (112). Smart meters that measure energy consumption, brings a step forward to Smart Homes.

1.5 Applied Technologies

Current interface technology is still closer to the use of command language or click interaction rather than to natural language conversational interactions. Computational systems are able to store huge amounts of data about the objects in our environment. Nevertheless, their ability to extract the relevant knowledge from this data and express it in understandable language showing human-like behaviour remains limited.

Avatars are computer characters that show human like behaviour in communication. Quite often are used to address online assistants able to answer questions and perform tasks through conversational dialogs with the users. There are several examples of research focused on virtual agents using natural language, but they are closer to a string patterns matching process of questions and answers and lack the ability to deal with numeric information.

Several works have been developed around interactive or conversational agents. In general they have had an increasing interest in the last years due to the advantages of this kind of interaction, easiness of interpretation and closer human-like communication. Most of them are based in technologies that use pattern matching for recognition and generation of questions-answers value pairs, e.g., A.L.I.C.E developed by R. Wallace (128) (129).

Automated question answering systems (QA) has been a topic of research and development since the earliest AI applications. A QA is a type of information retrieval environments, based on natural language, where input queries requires understanding and deductive reasoning over a background knowledge (131).

Zadeh underlies three key issues in QA systems. The first obstacle is that world knowledge is perception based. e.g., *“Most of your devices have a low efficiency”*, the second obstacle is the concept of relevance of the answer to the question proposed, and the third one is the deduction capabilities from perception based-information (137).

The Computational Theory of Perceptions (CTP) was introduced in the Zadeh’s seminal paper “From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions” (135) and further developed in subsequent papers (141). It grounds on the fact that human cognition is based on the role of perceptions, and the remarkable capability to granulate information in order to perform physical and mental tasks without any traditional measurements and computations.

It is remarkable the capability of humans to perform a wide variety of physical and mental tasks using fuzzy perceptions, without numerical measurements and computations. Zadeh argued, for example, that while parking a car it is not measured the distance in meters or degrees in which have to be moved the car or turn the wheels, but instead are used perceptions of time, direction or speed, e.g., “if we are close to the other car” or “if there is enough space”.

In the research line of CTP, has been defined by Trivino the Granular Linguistic Model of a Phenomenon (GLMP). It is a data structure that allows computational systems to generate linguistic descriptions of input data (122) (124) (119) (117). The linguistic description of a phenomenon is a summary of the available information where certain relevant aspects are remarked while other irrelevant aspects remain hidden.

Emotions are an important aspect of human intelligence and have been shown to play a significant role to present a comprehensive overview of a phenomenon. Emotions transmit meaning forwards the literal meaning of the words. Affective Computing techniques

study the simulation of emotions in computational environments, how emotions are defined and expressed. Affective computing builds an *emotion model* implementing the capability of perception and interpretation of human feelings as well as generating intelligent, sensitive, and emotional responses in computational environments. The emotion model proposed by Ortony (89) has often been the basis for the integration of emotions into cognitive architectures of embodied characters (45) (99), as it was designed as a computational model of emotions. Fuzzy Logic allows to model complex emotional states and behaviours (127) (126). Emotions in a face to face communication are incorporated to describe the situation including language with emotional contents, visual representation of face expressions or voice intonation.

The amount of data provided by computers about the objects in our environment is increasing. The intrinsic technical characteristics of databases have generated new models to facilitate the exploration of data and the access to information independently from the data structure and the query language. On-Line Analytical Processing (OLAP) environments support this independence. Multidimensional Data models (MD) and OLAP constitute a decision support system framework that provides users with flexible ad-hoc analysis in a multidimensional context.

The availability of a generic architecture is required when dealing with the project of designing a new computational system. The research and development in autonomous agents has produced different architectures (15). The CPA7L has been explained in detail by Trivino (120).

The distinction between Concepts, Procedures and Attitudes (CPA) has its origin in Cognitive Psychology and appears in the Theory of Elaboration introduced by Reigeluth (98).

Formal methods in software engineering are an increasingly important application area for intelligent systems. Research environments often produce prototypes that demonstrate the feasibility of a method and leave efficient and more complete implementation to industrial exploitation. Software projects, in order to meet requirements of the customers in time, are managed using a methodology to formalize the engineering process. Natural Language Generation (NLG), research results from software engineering and software technology are limited (61) (75). Even when there are factors that add complexity to the task beyond conventional software systems.

1.6 Contributions

The contribution of the thesis is to define a system to assist users in achieving energy efficiency, the Personal Energy Advisor. The extensions proposed to existing paradigms allow generating dynamically, based on perceptions, natural language interaction for users with a virtual energy advisor or access to advisory reports. The advisor is able to interpret user queries requested by users and reason answers and advice about the energy consumption patterns using a knowledge domain.

To interpret energy consumption information is not trivial, the acquisition of knowledge for an expert advisor is a complex task. A relevant step in the methodology, in order to develop the Personal Energy Advisor, is the acquisition of knowledge from the environment and the translation of this knowledge into an implementable design by a designer.

With the model proposed, the computer takes the role of a virtual assistant that interacts with the user, extracting relevant information from huge amount of data stored in a database, identifying dynamically based on users requests, key specific behaviours and describing them in natural language. Each interaction generated by the virtual agent show emotional behaviour, facilitating the understanding of the communication. A virtual advisor can be instantiated through different channels that would allow communicating with the customers. The possible final use could be through the web interface, the report generation with key aspects on the bill or the use in a smart home device for communication.

In the research line of Computational Theory of Perceptions, this thesis was developed based on the Granular Linguistic Model (GLMP), OLAP data access, Affective Computing and Virtual Interactive Agents.

The contributions can be organized in two main blocks, a) a set of proposed paradigms, namely OLAP-GLMP, Discourse OLAP-GLMP, Emotional OLAP-GLMP, and Question-Answer OLAP-GLMP and b) a software framework implementation of the paradigms, a methodology for development over the framework and the implementation of the Personal Energy Advisor and test of the solutions proposed.

a) The proposed paradigms are described as follows:

- OLAP-GLMP. Based on previous paradigms, it is proposed an On-line analytic processing model for the navigation through granular linguistic models in order to retrieve dynamically data from a database. The data is translated into linguistic expressions exploring possibilities of obtaining from natural language queries, reasoned responses using the GLM for modelling the knowledge base. Linguistic capabilities include the ability to describe, identify relevant information and extract reasoned advice from input data. This proposal "*OLAP Navigation in the Granular Linguistic Model of a Phenomenon*" has been published in (85).
- Discourse OLAP-GLMP. During text generation several tasks are performed, e.g., analyze the possibilities to automatically generate coherent and cohesive linguistic sentences based on natural language interpretation and generation, avoiding repetition and linking individual granular sentences. It is done using acquired constraints to reorganize the structure of an initial data obtained from a defined input domain. It is performed by a machine driven process of reorganization of the GLM. The reorganization is performed by pruning non relevant perceptions and by aggregating related ones avoiding redundancy. This proposal "*Selection of the Best Suitable*

Sentences in Linguistic Descriptions of Data” has been published in (83).

- Emotional OLAP-GLMP. In order to increase the effectiveness and acceptance while communicating information, it is needed to express a description in a natural way showing human-like emotions. When language is produced, as speakers, text is tailored to the hearer and the actual situation. This enables us to include more information than is contained in the literal meaning of linguistic expressions. Depending on the speaker, the hearer and the communicative contents the possibilities of obtaining emotional computation in the GLM are explored. This proposal “*Generating advices with emotional content for promoting efficient consumption of energy*” has been accepted for publication in the International Journal of Uncertainty, Fuzziness and Knowledge-based Systems (84).
- Question-answer OLAP-GLMP. Using perceptions is defined the procedures to interpret user questions and dynamically compute the answers in a dynamic dialog with the user.

b) The software framework, the methodology and the functionality of the Personal Energy Advisor is described as follows:

- In order meet the desired requirements and to test the proposed paradigms it has been developed a software tool to implement the energy advisor. It consists in an application framework for the definition of an agent based on requirements reducing design and implementation gaps and facilitating development. It allows developing, executing and maintaining a natural language interface tool for queering energy consumption knowledge. The basic benefit of the framework is that it allows designers and programmers to minimize the time-to-market, dedicating their time to meet enhanced functional requirements rather than dealing with the more standard low-level technical details.
- It is also proposed a methodology that using the software framework allows developing this kind of computational applications. Formal methods in software engineering are an important application area for intelligent systems to deal with the application of a systematic, disciplined, quantifiable approach to the design, development, operation, and maintenance of software. A description of software methodologies are presented, followed by the particularization of a methodology for covering the process involved in the development of an Avatar system using the framework of the solution. The methodology is developed step by step to define the Personal Energy Advisor.
- Finally the proposed extensions and the system framework are used to test the functionality of the Personal Energy Advisor. The resulting advisory reports and interaction between the Personal Energy Advisor and the user is presented showing the viability of the approach including a practical demonstration.

1.7 Contents

The contents of the thesis are organized as follows.

In chapter 1 “*Introduction*” is described an overview of the overall approach and the contributions of the thesis. The objective, goals and scope to develop the Energy Advisor are presented. An overview of existing paradigms and studies used in the thesis are presented. For the definition of the Personal Energy Advisor, extensions to existing studies are introduced in order to be able to support households in the path to achieve energy efficient behaviour.

In chapter 2 “*Energy Efficiency*” is presented the actual situation of the electrical system, and the developments expected in the Smart Grid with the introduction of digital technologies. Also are introduced the measures proposed to improve the energy efficiency within the European Union, and the existing smart energy monitoring tools at households.

In chapter 3 “*Introduction to the Applied Technologies*” there is a review of the different paradigms and studies that have been identified and used in the thesis. For the sake of brevity, the concepts required to understand the development of the Thesis are only introduced. For a more thoroughly description of concepts and details several references are included. Previous work considers the fields of Computational Theory of Perceptions and the Granular Linguistic Model, OLAP Data Processing, Natural Language Processing, Affective Computing, Question-Answering Systems and Autonomous Virtual Agents.

In chapter 4 “*Extensions proposed to applied technologies*” is described based on previous paradigms, the extensions proposed in the thesis and the final objective for each extension. Each proposed solution placed together in a global architecture will support the definition of the Personal Energy Advisor.

It is described the dynamic navigation through perceptions retrieved from a database. With this purpose it is defined a Granular Linguistic Model over an OLAP schema. The typical operations in OLAP, namely drilling, rolling and slicing/dicing are applied to navigate by hierarchical granular structures of fuzzy perceptions.

It is described the dynamic natural language discourse generation based on perceptions, where the resulting perceptions acquired through navigating the knowledge domain are arranged in a structured discourse. Here are explored the possibilities to reorganize the structure of a Granular Linguistic Model to deploy a coherent and cohesive linguistic discourse, avoiding repetition and linking individual granular sentences in a set of connected ones.

It is described the dynamic generation of perceptions with an emotional component. In this case the dynamic discourse generation will be able to express emotions based on user behaviour increasing the expressivity of the language.

It is described a Question Answer environment based on perceptions. This extension allow to dynamically interpret user questions and dynamically compute the answers in a dialog with the user.

In chapter 5 “*Energy Advisor Development Framework*” is presented a software framework used to implement the Personal Energy Advisor. It is based on concepts, procedures and attitudes. The framework implements the extended paradigms and works due to the separation of an on-line and off-line functionality. Two different environments with the associated interfaces are provided, one for the designer to define the knowledge domain for the off-line functionality and another for the user to execute queries and get responses using the on-line processes. The architecture (1) for designers, simplifies the development of a knowledge domain for the Energy Advisor by a visual and declarative environment that relies on metadata and (2) for users, provides out-of-the-box infrastructure for the execution of the Personal Energy Advisor.

In chapter 6 “*Energy Advisor Development Methodology*” is presented a methodology to analyze, design, develop and maintain the personal energy advisor using the software framework. In the chapter are reviewed the general characteristics and processes involved in existing methodologies for software development and knowledge discovery in data bases. In Natural Language Generation, research results from software engineering and software technology are limited. This chapter describes a methodology developed upon previous research to build this kind of systems. It is defined a specific methodology to formalize the development of the Personal Energy Advisor, describing the analytic process to get the expert knowledge and develop the application to achieve desired results in the software framework.

In chapter 7 “*Efficiency Analytical Modelling*” is described the analytic process involved in the design of the Personal Energy Advisor. Initial requirements are defined by experts in a detailed report that covers the contents and knowledge that will be managed by the personal advisor. This phase will reveal the gaps between initial requirements defined in the initial report independent of the technical details and an implementable report due to technical details. During this phase, the analytic capabilities to report to the customers are identified, validated and described. The data model required to store input information is built.

In chapter 8 “*Energy Advisor Development*” is developed the virtual energy advisor following the design, implementation and testing activities. Here is applied the methodology for the development phase based on the application framework and the knowledge defined in the previous analytical modelling phase.

In chapter 9 “*Functionality of the Personal Energy Advisor*” the developed Personal Energy Advisor is explored from the user perspective. Several scenarios of users interacting with the Personal Energy Advisor are described to show the resulting functionality.

In chapter 10 “*Concluding Remarks*” conclusions and next steps are presented.

Chapter 2

Energy Efficiency

2.1 Introduction

The objective of reduction of energy intensity is a priority to achieve a sustainable development. With this purpose measures and policies have been defined to improve the energy system and to promote energy efficiency.

The electrical grid is an interconnected network of stages that allows delivering electricity from its generation to the final user consumption. The stages consist of generation plants that produce electrical power, high-voltage transmission lines that carry power from its generation to substations through long distances, distribution lines that bring connection from substations to individual buildings and supply that provide the service to final consumers. The electrical grid is experimenting a transformation with the introduction of digital technologies. The application of digital technologies to the electrical grid is referred as the Smart Grid. It is a suite of technologies that would improve the performance based on a higher capacity for measurement, communication and control over the grid. Currently different applications of improvement like off peak demand reduction, consumption information management or cogeneration and electrical vehicle integration in the grid are faced.

Smart Meters are an important part of the Smart Grid. They are advanced meters that identify energy consumption more accurately than conventional ones. They measure and store the energy consumption with a detail per hour and make it available to users in real time. One of the focuses for energy efficiency is the rational use of the energy encouraging the households to develop improvements of the consumption patterns while maintaining the comfort level. With Smart Meters can be provided detailed information about when and how the energy is used, rather than just to provide the amount of energy used monthly for billing.

The importance of consumption reduction in households, related to the overall electricity system, relies in the increasing volumes of demand. The fact for consumers is that energy is consumed through a variety set of devices and it is hard to be aware about how do they use the energy, how they can improve energy efficiency and how they can get savings or how could they contribute to the energy system. A key element to achieve

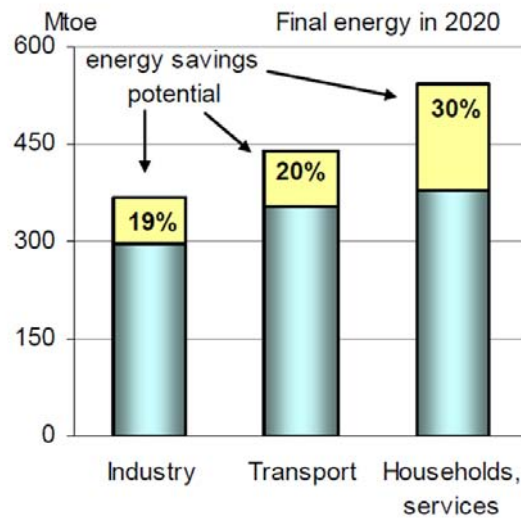


Figure 2.1: Estimated energy consumption reduction potential in 2020 (Source: European Commission).

energy efficiency at households relies on the capacity to acquire, interpret and communicate personalized advice about energy consumption patterns identifying the required information for improvement.

EDP, among other measures, has launched the project “*Ecofamilias*” to provide useful information about consumption patterns to households. Energy-monitoring technologies with persuasive interfaces could inspire sustainable energy lifestyles. These approach lets to provide more value to domestic energy consumers and support a positive change in users behaviour (90) (35) (81).

2.2 Efficiency in the European Union

The EU adopted the Directive 2012/27/EU on energy efficiency. This Directive establishes a common framework of measures for the promotion of energy efficiency within the Union. To ensure the increase of energy efficiency, the goal for year 2020 is to achieve the called “20-20-20”: saving a 20% of the primary energy consumption, reducing a 20% greenhouse gas emissions and using a 20% of renewable energies. Measures proposed remove barriers in the energy market and overcome market failures looking for efficiency in the supply and use of energy. As shown on Fig. 2.1 recent studies indicate that the opportunities for energy savings still remain significant (51).

Following the Directive, it brings forward legally binding measures to step up Member States to use energy more efficiently at all stages of the energy chain, from the transformation of energy and its distribution to its final consumption. Measures include the legal obligation to establish energy efficiency schemes or policy measures in all Member States. These will drive energy efficiency improvements in households, industries and transport sectors.

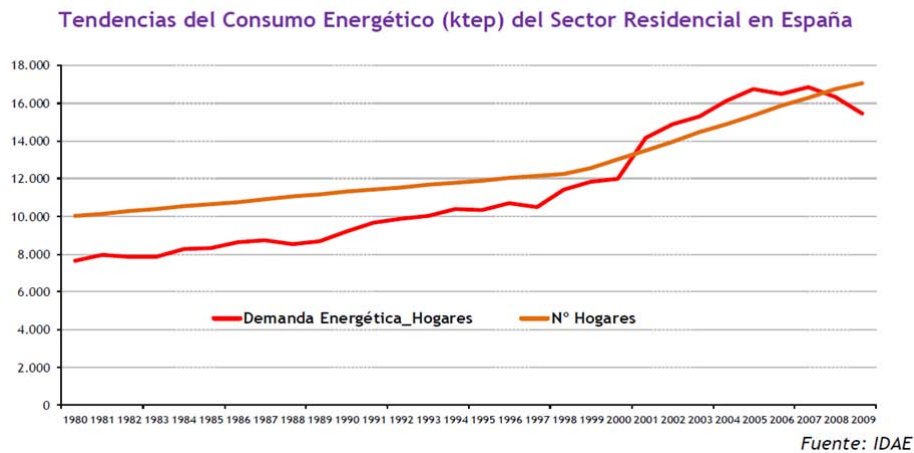


Figure 2.2: Households Demand in Spain (Source:IDAE).

The Directive argues that improving energy consumers efficiency will make possible to get potential cost-effective energy savings. The aim of this Directive is not only to continue promoting the supply side of energy services, but also to create stronger incentives for the demand side. Due to the lack of consumer awareness, measures include the right for consumers to know how much energy they consume.

At the moment, liberalization of energy markets has not led to significant competition in products and services which could have resulted in improved energy efficiency on the demand side. Member States should ensure the availability of energy advice, even in segments like households where energy audits are generally not sold commercially.

The residential sector is key in the actual energy context, due to the importance of their energy requirements, e.g., in Spain it is up to 25% of the total demand. It is also remarkable the growth in this consumption that is explained due to the increase of households, the increase in the comfort required and the increase in equipment used. The standard of living foresees this tendency as showed on Fig. 2.2.

Through Eurostat and member states was defined the elaboration of statistics under the project SECH (Development of detailed Statistics on Energy Consumption in Households) to measure detailed consumption in every state. The results have been detailed in the Spanish project *Sech-spahousec* explaining consumption by climate zone and type of housing, in line with the previous approximations from IDAE (IDAE) (MIT). The project results have described equipment and residential consumption, confirming the high level reached and the need of rational policies to influence at households.

2.3 The Smart Grid

The concept of Smart Grids makes reference to the application of digital technologies to the electrical grid.

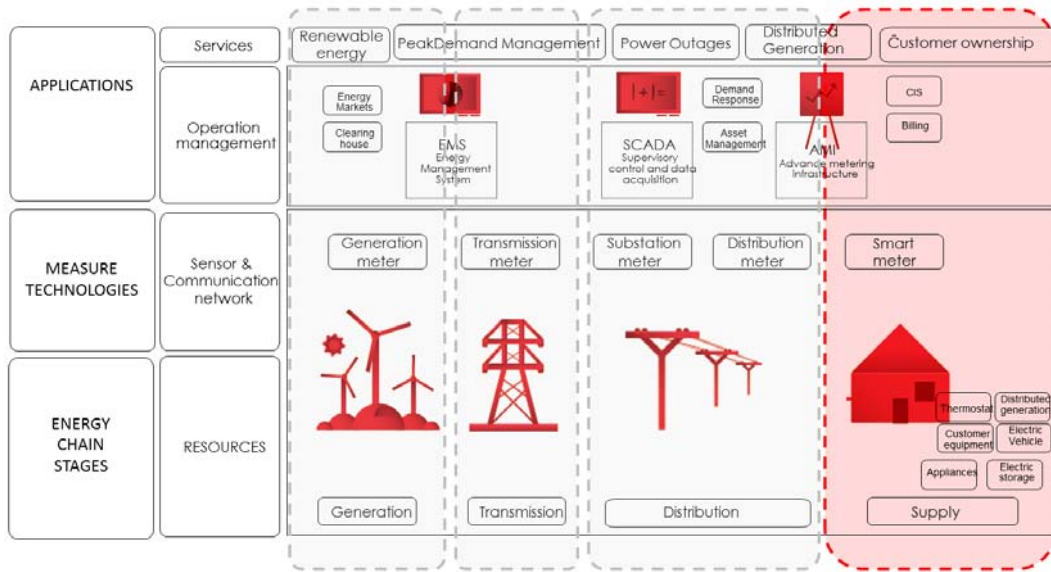


Figure 2.3: Energy chain stages, technologies and applications of the Smart Grids.

Four energy high level stages can be considered in the electrical grid: generation, transmission, distribution and supply. Each of these stages will incorporate new technologies based on digital metering and communication and new resulting applications will increase performance as showed on Fig. 2.3. Digital technologies have been integral to the modernization of different sectors of the economy achieving efficiency gains. The electric power sector, however, has lagged behind.

Smart Grid will develop a broad set of technologies focused in the improvement and performance control over the electrical grid. Smart Meters are an important part of the Smart Grid. They are advanced meters that identify energy consumption more accurately than conventional meters. They measure and store the energy consumption with a detail per hour and make it available to users in real time. Rather than just provide the amount of energy used monthly for billing, with Smart Meters can be provided detailed information about when and how the energy is used.

Currently are faced many areas of application improvement like management off peak demand, consumption information management or cogeneration and electrical vehicle integration in the grid.

The information flow between the different stages of the electrical grid is crucial to improve management. Even consumers have very little knowledge about how their electricity is priced or how much energy they are using at a given time. This limits the incentives for promoting energy efficiency and demand response. New challenges for the next years include the electric vehicle.

The reduction of greenhouse gas emissions is implicit in every improvement. Peak and valley demand periods and CO₂ emissions are showed on fig. 2.4.

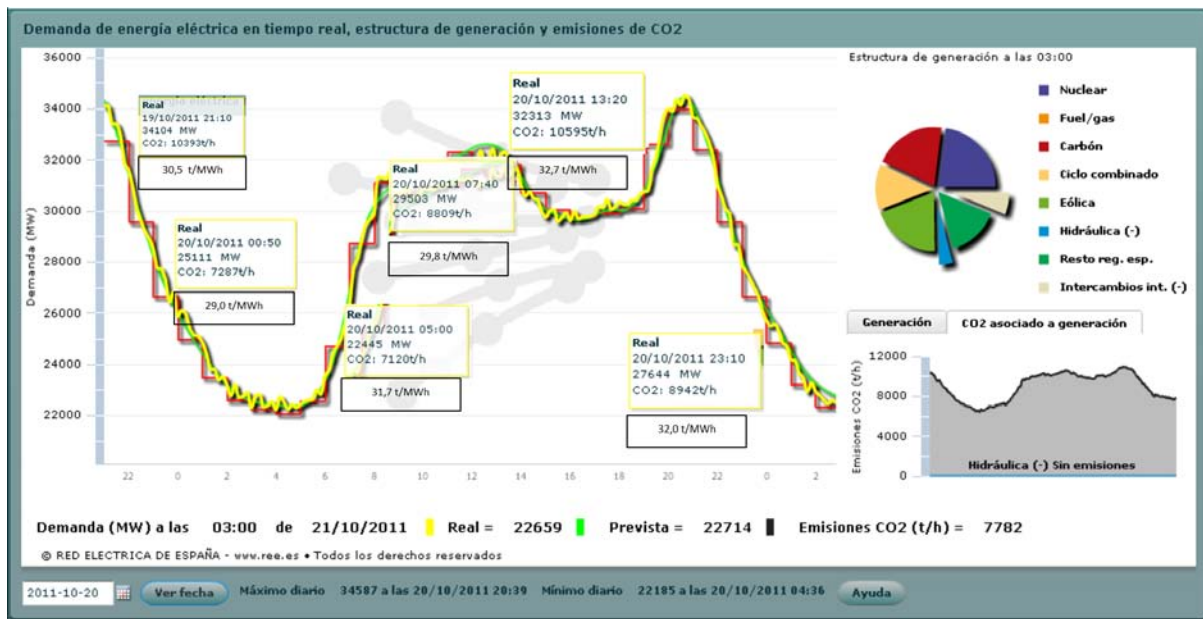


Figure 2.4: CO2 Emissions in the Spanish market. Source REE.

2.3.1 Benefits

Smart grid technologies offer potential economic and environmental benefits as showed on Fig. 2.5:

- Customer Satisfaction. Improve reliability measures of consumption and empower user management of energy. Provide tighter communication and additional product and services.
- Operational Efficiency. Increase operational productivity reducing capital and operating cost and improving security.
- Energy Efficiency. Optimizing energy usage and growing demand.
- Environmental Impact. Reduce CO2 emissions meeting regulatory requirements increasing environmental generation options.

2.3.2 Applications

The Smart Grid technologies enable new smarter applications as showed on Fig. 2.6, including:

- Power outages.

A power outage or blackout is the loss of electric power in a geographic area. If it is produced for a very short term it is defined as quality disruption. There are many causes of power failures in an electricity network, e.g., due to faults at power plants, damage to electric transmission lines or other parts of the distribution system. This situation usually means economic losses, e.g., power quality disruptions for

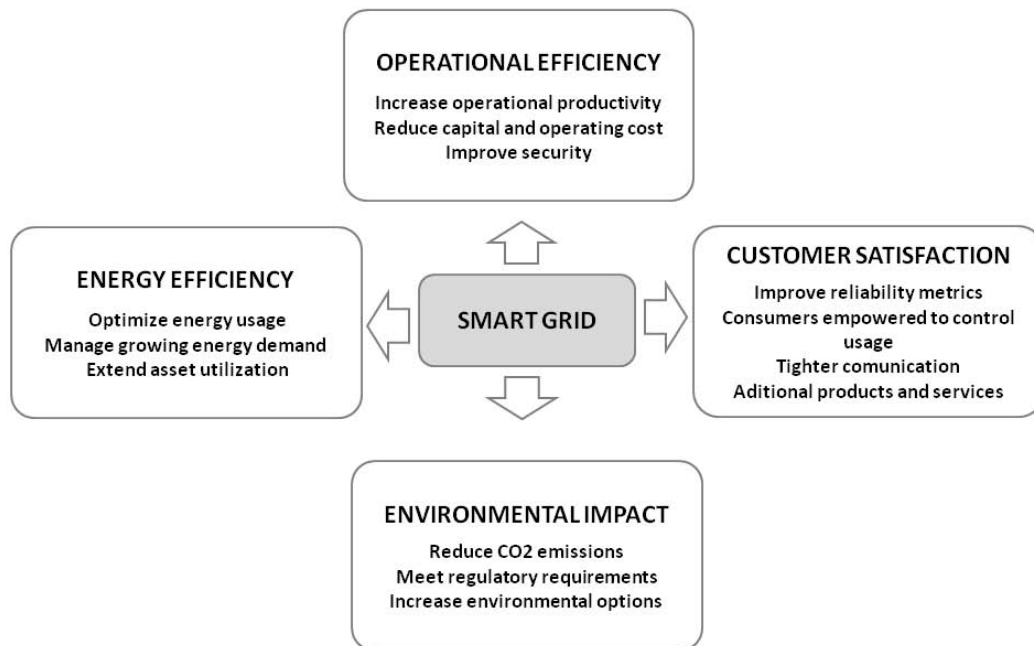


Figure 2.5: Smart Grids Benefits.

households may be just a light blinking, but for manufacturing plants these events can halt the manufacturing chain and cost millions. At the moment blackouts or power disruptions are not identified in advance but when consumers call to notify them.

- Peak demand management.

Peak demand is the period when electricity demand is at the highest level. Electricity cannot be stored, so once generated has to be consumed. Users demand electricity without a fix pattern, and the requirements of production, transmission and distribution varies in time, while the generation capability installed has to cover the peak even when this level of demand is not used most of the time. Peak and valley demand periods are showed on Fig. 2.7.

- Integration of renewable energy.

Generation in renewable technologies depends on the nature of resources used, e.g., the wind or sunlight. The power grid must constantly maintain a buffer of excess supply to adapt the changes in natural resources generation, which is primarily based on traditionally generation, resulting in lower efficiency and higher emissions. The ability of the existing grid to manage levels of variable renewable generation can be improved reducing the buffer required.

- Integrated distributed generation.

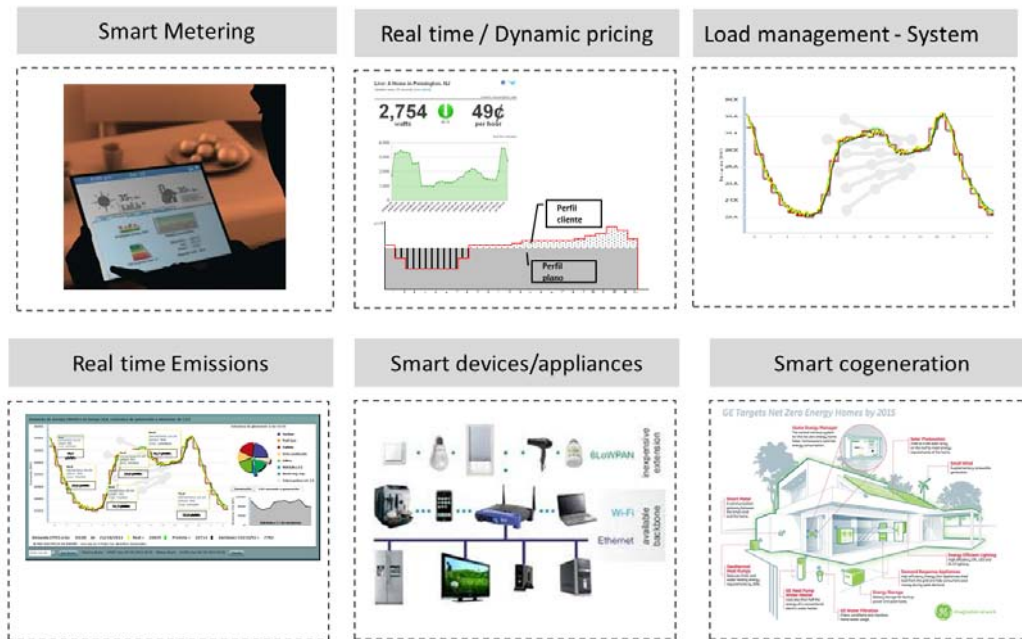


Figure 2.6: Smart grid Application.

Distributed generation is based on the generation of electricity from small energy sources at consumers supply points. The management of distributed generation with actual technology is difficult because of the lack of information to establish control. The grid was designed for a power flow from centralized power plants to end users. New requirements imply a two-way power flow that supports small distributed generators, e.g., rooftop solar or micro wind micro-generation.

- Smart Metering.

Smart metering means remotely read consumptions for management; as opposed to sending a meter-reader to each house every month. Digital technologies allow not only to retrieve measures remotely, but remotely interaction is allowed, e.g., customers demand for power increase can be managed remotely.

- Dynamic Pricing.

Dynamic pricing allows charging electricity prices dynamically to reflect the realities of the electricity market, e.g., charging more expensively electricity consumed at peak periods. This would allow to efficient consumers pay less for energy than inefficient ones. At the moment a plain cost of energy is applied for users as it is showed on Fig. 2.8.

- Distribution Management Automation.

Distribution management automation allows distribution systems to reconfigure themselves when a fault occurs, e.g., restricting the problem to a smaller area reducing the impact.

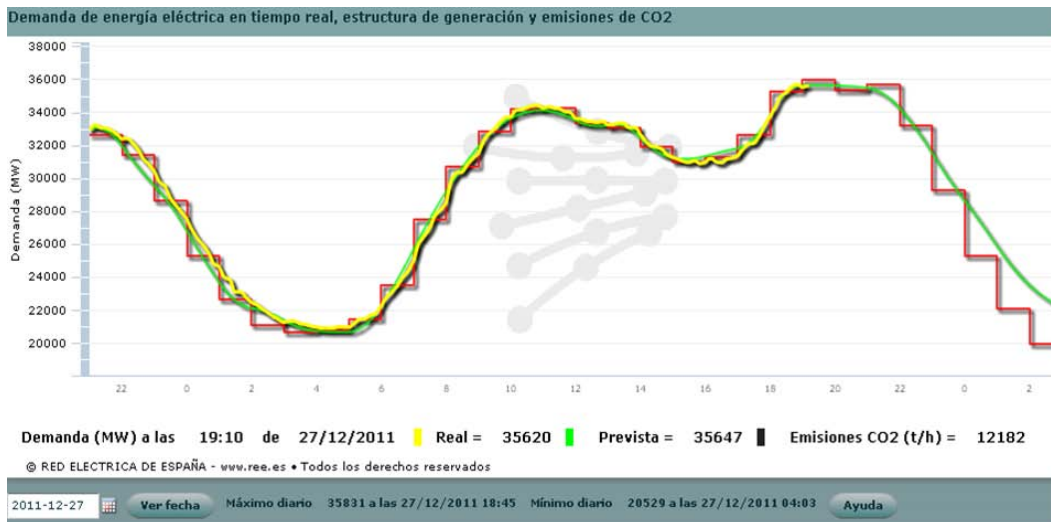


Figure 2.7: PeakLoad in demand curve in the Spanish market. (Source REE)

2.4 Household smart energy monitoring

Smart meters that measure energy consumption, brings a step forward to smart homes (112). Companies should move to more intelligent interaction with the users. Technological innovations should enable smart homes move us to a new relationship with users.

Smart homes include technology that makes life easier for users. Smart Homes were described as those *“equipped with computing and information technology which anticipates and responds to the needs of the occupants, working to promote their comfort, convenience, security and entertainment through the management of technology within the home and connections to the world beyond”* (32) (115).

Focusing on energy smart appliances, there are some studies about how users could interact with energy monitoring tools, or how can these tools help users to improve behaviour and get savings. Hargreaves (58) explored how UK householders reacted to energy consumption using real-time displays. Shove studied the sociology of consumption and technology to investigate the evolution and social meaning (110)(111).

To date little research has explored in-depth how householders actually interact with homes and the use of suites of different technologies and their results in terms of efficiency. Constanza studied an interactive visualization of domestic electricity consumption that leverages user-generated annotations to provide multiple views of the data (33). As expressed in (35) *“the norm is for savings from direct feedback (immediate, from the meter or an associated display monitor) to range from 5%-15% and from indirect feedback (feedback that has been processed in some way before reaching the energy user, normally via billing) savings have ranged from 0%-10%, but they vary according to context and the quality of information given”*.

Although users are interested in savings and in collaborating with the environment,

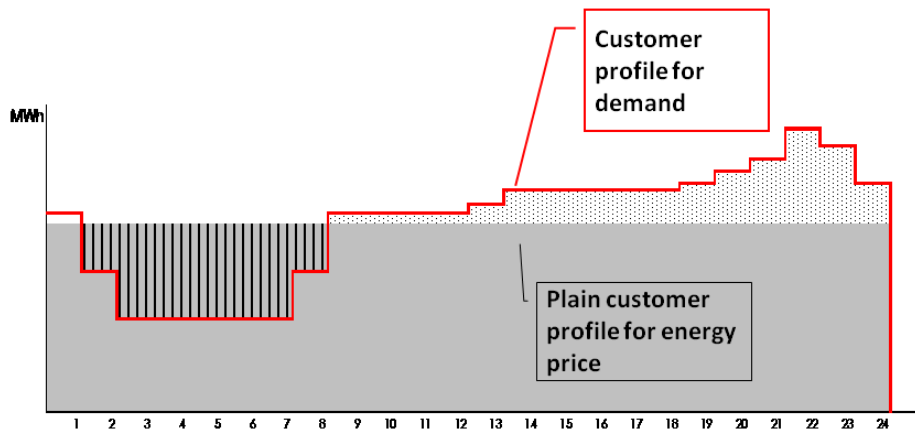


Figure 2.8: Customer demand profile vs profile for energy price.

they seem to be passive but in fact they have little information about what to do. Information provided to users and economic incentives to save energy should change habits. Users could be motivated by reward reflecting real prices of energy along the different hours of the day, but also by providing the required information to empower the users to take the right decisions. Incentives can be understood as premiums prices. Reports defined by expert advisors should be clear inputs for users, for analyzing behaviour and adapt consumption habits to get savings.

To interpret energy consumption information is not trivial. Information about consumption is affected by external influences like sociodemographic characteristics of the household, e.g., the number of members of the household, infrastructure of devices, their characteristics and use, the temperature or light hours available along the year seasons. All this external factors should be understood by advisors in order to provide correct feedback. Information provided to users identify behaviour, by describing efficiency, identifying key relevant patterns, reason recommendations and reflecting specific alerts or compromises where the user wants to be focused.

There are different levels for advisory services about energy consumption. It can be divided in two axis, level of information and level of interaction.

- Information level

At the information level, it is referred the ability to transform the information into useful clues for the customer and the granularity of this information, e.g., if it can be measured per second, hour, or on a monthly basis.

- Interaction level

At the interaction level, it has to be distinguished the capacity to update of information, e.g., from real-time to off-line periodically delivered reports.

Information about energy detailed consumption has many advantages, but also disadvantages have been identified.

Between others, could be underlined the following drawbacks:

- Overload of information

Information captured by smart meters could overload users with data meaningless. Measures of consumption could range from seconds to hours, and for a year an hourly based measure supposes 8.760 individual measures that out of context could be overwhelming. To manage this drawback it is clear that a more intelligent communication with the user is required in order to facilitate the interpretation of the data, managing information and communication abilities.

- Concerns about devices at households

The number of devices at households increases, as user's moves to more intelligent homes.

- Expenses

Expenses required to incorporate new technology to households should be lower than the benefits obtained from savings.

Between others, could be underlined the following advantages:

- Consciousness about expenses and energy is used

Users are conscious about consumption when they receive the bill, it is usually quite late to take measures and correct behaviour minimizing expenses. The ability to monitor more frequently and detailed information increase consciousness of the users.

- Empowerment for households to manage the energy use

The availability of information about consumption, and the ability to transform this information in useful recommendations easily understandable to improve consumption behaviour is the key of the process. Information context is requested at this stage, as time comparison e.g., if consumption increases from month to month, or comparison to other household with similar characteristics. It is possible to automate and schedule many routine functions of the home, for example moving consumption to cheaper period maximizing savings. Using domotics it is possible to automate the use of devices. Understand and analyze the devices and their efficiency, using habits that maximize their efficiency making a proper use of the devices, or change devices with low efficiency getting savings.

Chapter 3

Introduction to the applied Technologies

3.1 Introduction

In this chapter the different paradigms and studies that have been identified and used in the thesis are reviewed. Between others the fields of computational theory of perceptions, affective computing, data processing, natural language processing, question answering systems and autonomous virtual agents are considered.

For the sake of brevity, the concepts required to understand the development of the thesis are only introduced. For a thoroughly description of concepts and details several references are included.

3.2 Computational Theory of Perceptions

Human cognition is based on the role of perceptions. Human beings describe perceptions using natural language (NL). NL allows us to make imprecise representations of complex phenomena. The most adequate degree of granularity in each circumstance when using NL is chosen. It is to remark the relevant aspects and to hide the irrelevant ones.

A collection of granular perceptions is used while describing a phenomenon. Perception granulation abstracts from individual observations of a phenomenon to view the phenomenon as a whole, improving understanding for problem solving (140). Humans routinely make use of the granularity of information in order to take decisions, perform everyday tasks, and communicate with others.

The Computational Theory of Perceptions (CTP) was introduced in the Zadeh's seminal paper "*From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions*" (135) and further developed in subsequent papers (141). It grounds on the fact that human cognition is based on the role of perceptions, and the remarkable capability to granulate information in order to per-

form physical and mental tasks without any traditional measurements and computations.

Zadeh argued for example, that while parking a car we do not measure the distance in meters or degrees in which we have to move the car or turn the wheels, but instead we use perceptions of distance, direction or speed, “*if we are close to the other car*” or “*if there is enough space*”. CTP suggests a direction that might enhance the ability to deal with real world problems in which decision-relevant information is a mixture of measurements and perceptions.

In the research line of CTP, has been defined the Granular Linguistic Model of a Phenomenon (GLMP). It is a data structure that allows computational systems to generate linguistic descriptions of input data. GLMP is focused in developing computational systems able to provide users with meaningful linguistic descriptions of a phenomenon. The linguistic description of a phenomenon is a summary of the available information where certain relevant aspects are remarked while other irrelevant aspects remain hidden.

3.2.1 General GLMP architecture

Several works have been performed in the European Center for Soft Computing, for the definition of the Granular Linguistic Model of a Phenomenon (GLMP) (121) (123) (119) (117) (39) (107) (124). The GLMP is a data structure that allows computational systems to generate linguistic descriptions of input data.

Figure 3.1 shows the approach to the architecture of a computational system for generating linguistic description of data, indicating processes with ovals and data structures with rectangles.

- Off-line stage

During a preliminary off-line stage, due to the amount of data available, it is required to extract the most relevant knowledge from the data in order to generate relevant and useful information. The designer collects a corpus of NL expressions that are typically used in the application domain to describe the relevant features of monitored phenomena. The designer analyzes the particular meaning of each linguistic expression in specific situation types to build *Granular Linguistic Model of Phenomena* (GLMP) and *Report templates*. In this way, the designer creates a *tree of choices*, i.e., a structure of possible linguistic expressions to be applied depending on each different input data.

- On-line stage

During the on-line stage, data is retrieved for the specific phenomenon, the *Instantiation module* selects among these available choices the most suitable linguistic expressions to describe the current input data. Finally, the *report generator* combine the selected expressions using a *report template* to generate the output.

In the following sections, using examples taken from the domain of the application, the components of this architecture are described including several definitions aimed to establish a framework for developing practical applications.

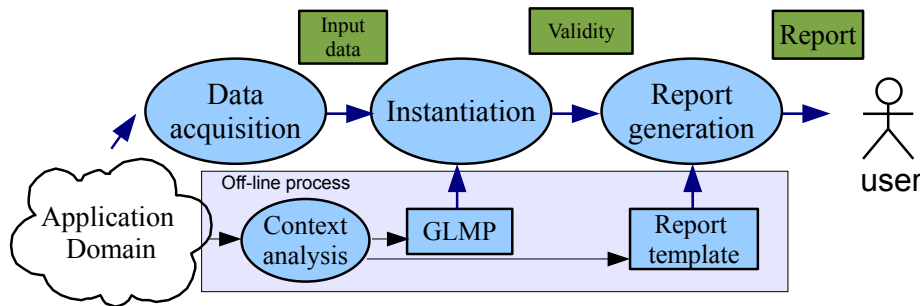


Figure 3.1: General architecture of computational systems for generating linguistic description of data

3.2.2 Computational Perceptions (CP)

The concept of Computational Perception (CP) is based on the concept of linguistic variable introduced by Zadeh (134). CPs are computational models of information units (granules) acquired by the designer about the phenomenon to be modelled, i.e., CPs correspond with specific parts of the phenomenon at certain degree of granularity. A CP is a tuple (A, W) described as follows:

A is a vector (a_1, a_2, \dots, a_n) of linguistic expressions (sentences in NL) that represents the linguistic domain of the CP. Each a_i describes the value of the CP in each situation with specific degree of granularity. These sentences can be either simple, e.g., $a_i = \text{“The consumption in the morning is quite high”}$ or more complex, e.g., $a_i = \text{“Some days you could save a lot of money”}$. During the preliminary off-line stage, these values are assigned by the designer extracting the most suitable sentences from a linguistic corpus of the application domain.

W is a vector (w_1, w_2, \dots, w_n) of validity degrees $w_i \in [0, 1]$ assigned to each a_i in the specific context. The value of validity depends on the application, i.e., it is a function of the suitability of each sentence to describe specific input data. During the on-line stage, these validity values are assigned (and updated) in function of the phenomenon current state. Typically, A is a strong fuzzy partition of the domain of existence of CP and therefore $\sum w_i = 1$.

For example, a computational perception that models the consumption during the morning $CP_1 = (A_1, W_1)$ could be instantiated as:

$$a_1 = \text{“The consumption in the morning is high”}, w_1 = 0.7$$

$$a_2 = \text{“The consumption in the morning is medium”}, w_2 = 0.3$$

$$a_3 = \text{“The consumption in the morning is low”}, w_3 = 0$$

Here, the first sentence has currently the highest validity $= 0.7$. Therefore is a good candidate sentence to be included in the final report.

3.2.3 Types of CP

Different kinds of perceptions can be defined based on the application context. In (39), as an example, three types of *CPs* are defined, namely, *assertive CP*, *derivative CP* and *integrative CP*.

- Assertive CP is the fundamental one. It represents the linguistic fuzzy model of the perception of the current state of a characteristic of the phenomenon, e.g., “*The consumption of energy is High*”.
- Derivative CP corresponds with trend analysis information and it gives insight into how the phenomenon is evolving in time. e.g., “*The consumption of energy is rapidly increasing*”. It helps contextualizing the information and it may be important for decision making.
- Integrative CP represents the accumulated perception of the phenomenon over a period of time. The text associated with these perceptions consists of summary sentences of historical event occurrences, and answers the question of “Which is usually the state of the variable *A*”.

The definition of the typical template for an *Integrative CP* could be:

{Never | A few times | Sometimes | Many times | Most of the time | Always}, the variable A was {Low | Medium | High}.

3.2.4 Perception Mapping (PM)

It is called perception mapping (PM) to a function that performs the constraint propagation process, i.e., the function that converts input perceptions in output perceptions (136).

The designer uses 1PM to define the maximum level of detail (the highest granularity) in the linguistic model of a phenomenon. 1-PMs correspond with the designer’s interpretation of input data (*u*), e.g., sensor data or a data base. Perceptions whose meaning is based on other subordinate perceptions are called *second-order perceptions* (2CP). 2CPs are outputs of 2PM.

PMs are used to create and aggregate *CPs*. A *PM* is a tuple (U, y, g, T) where:

U is a vector of input *CPs*, $U = (u_1, u_2, \dots, u_n)$, with $u_i = (A_{u_i}, W_{u_i}, R_{u_i})$. We call them first order perception mappings (*1PMs*) when *U* are values $z_i \in R$ being provided, e.g., either by sensors or obtained from a database.

y is the output *CP*, $y = (A_y, W_y)$.

$g()$ is an aggregation function employed to calculate the degrees of validity $W_y = (w_1, w_2, \dots, w_{n_y})$. It is a fuzzy aggregation of the input vectors $W_y = g(W_{u_1}, W_{u_2}, \dots, W_{u_n})$, where W_{u_i} are the degrees of validity of *U*. In Fuzzy Logic many different types of aggregation functions have been developed. For example *g* could be implemented using a set of fuzzy rules. In the case of *1PMs*, *g* is built using a set of membership functions as follows:

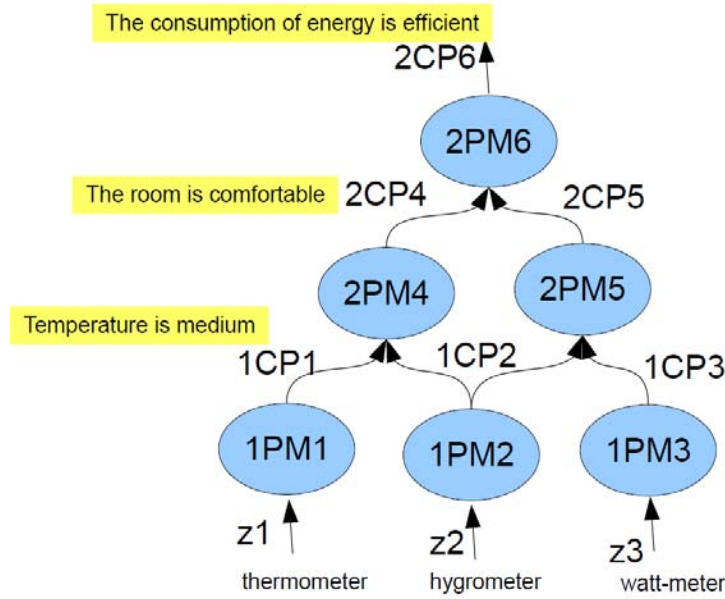


Figure 3.2: Example of Granular Linguistic Model of a Phenomenon.

$$W_y = (\mu_{a_1}(z), \mu_{a_2}(z), \dots, \mu_{a_{n_y}}(z)) = (w_1, w_2, \dots, w_{n_y})$$

where W_y is the vector of degrees of validity assigned to each a_y , and z is the input data.

T is a text generation algorithm that allows generating the sentences in A_y . In simple cases, T is a linguistic template, e.g., “*The efficiency in this building is {high | medium | low}*”.

It is worth remarking that A_y is defined by the designer during the off-line stage.

3.2.5 Granular Linguistic Model of Phenomena (GLMP)

GLMPs consist of networks of *PMs*. In each situation type, the designer uses a network of *PMs* to create a description of the monitored physical phenomenon with different levels of granularity that is called *GLM*.

Each *PM* receives a set of input *CPs* from its subordinate *PMs* and transmits upwards the output *CP*. Each output *CP* is *explained* by a set of input *CPs*.

In the network, each *PM* covers specific aspects of the phenomenon with certain degree of granularity. Each *PM* has associated a set of linguistic clauses, each one with an associated degree of validity that covers the whole of occurrences of the phenomenon from the designer’s perspective.

Using different aggregation functions and different linguistic expressions, the *GLMP* paradigm allows the designer to model computationally her/his own perceptions. Note that, after being instantiated with a set of input data, the *GLMP* provides a structure of valid sentences that in medium size applications could include a tree with hundreds of possible sentences.

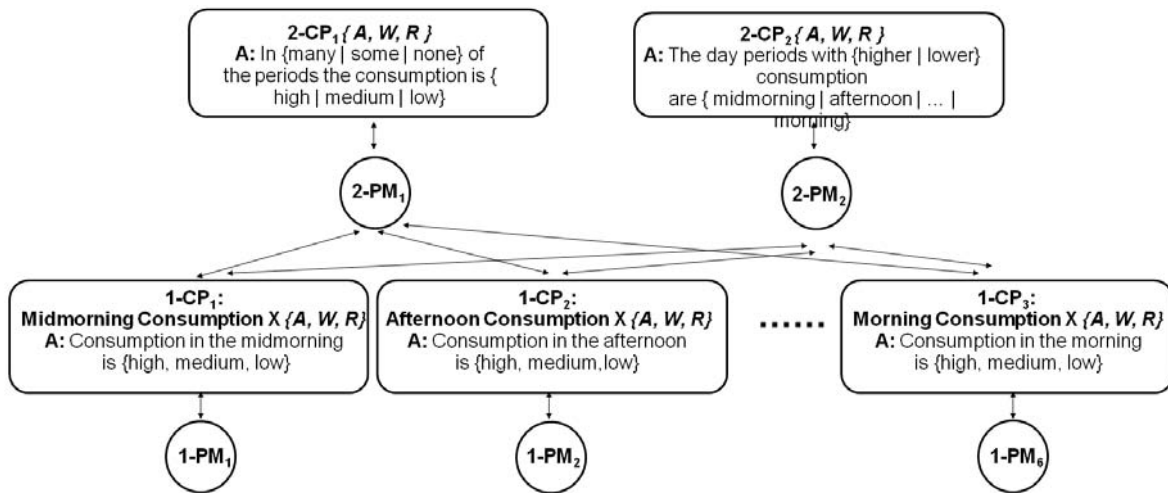


Figure 3.3: Second example of Granular Linguistic Model of a Phenomenon.

Figure 3.2 shows a GLMP where $1CPs$ are obtained from sensors data (temperature, humidity, and power consumption) and $2CPs$ are calculated based on subordinate CPs , providing information about the phenomenon at different levels of granularity. Here, the model generates sentences about the efficiency of an air conditioned system. An example is showed in the figure several possible sentences.

In the second example showed on Fig. 3.3, $1CPs$ are obtained from input data, $2CPs$ are calculated based on subordinate CPs , describing information at different levels of granularity. Figure 3.4 shows an example of a GLMP Linguistic expression instantiation, for simplicity only the sentences for each CP with the highest validity are showed.

3.3 On-Line Analytical Processing (OLAP)

The amount of data provided by computers about the objects in our environment is increasing. Multidimensional Data models (MD) and On-Line Analytical Processing (OLAP) constitute a decision support system framework that provides users with flexible ad-hoc analysis in a multidimensional context.

Databases constitute a model for information storage and recovery widely extended, in particular, relational databases proposed by E.F. Codd (30) has become a predominant choice. The intrinsic technical characteristics of databases have generated new models to facilitate the exploration of data and the access to information independently from the data structure and the query language. MD and OLAP environments support this independence. This model constitutes a decision support system framework which affords the ability to calculate, consolidate, view, and analyze data according to multiple dimensions (28) (29) (27).

OLAP techniques overcome the limitation of conventional database models, whose structures are not well suited for friendly querying of large amounts of historical data.

Usually, the final user of these technologies is a business analyst who interprets the

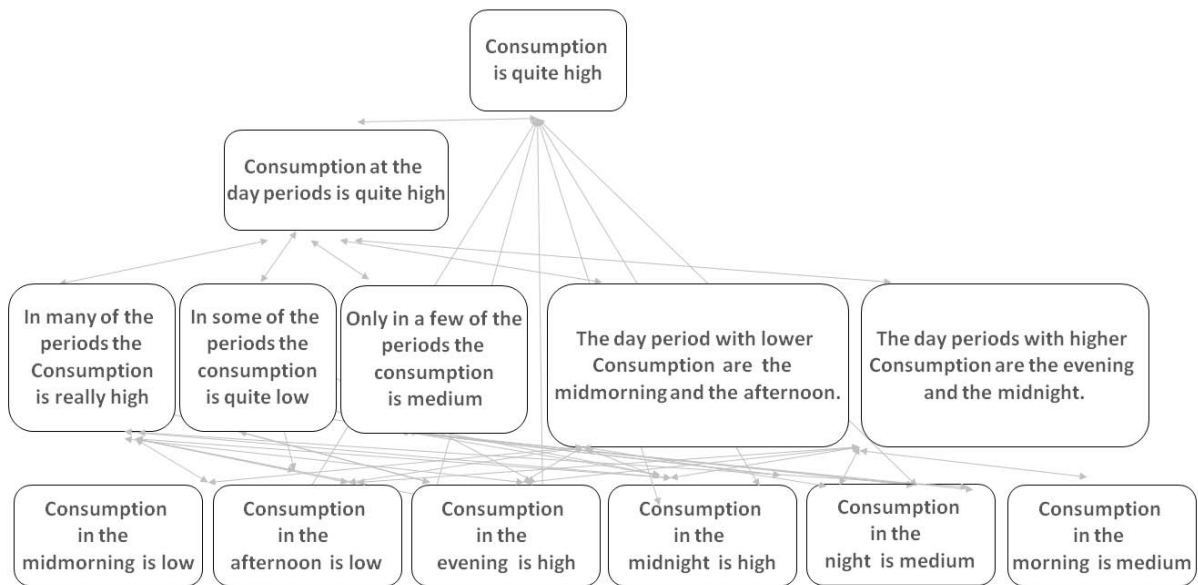


Figure 3.4: Example of Granular Linguistic Model of a Phenomenon instantiation.

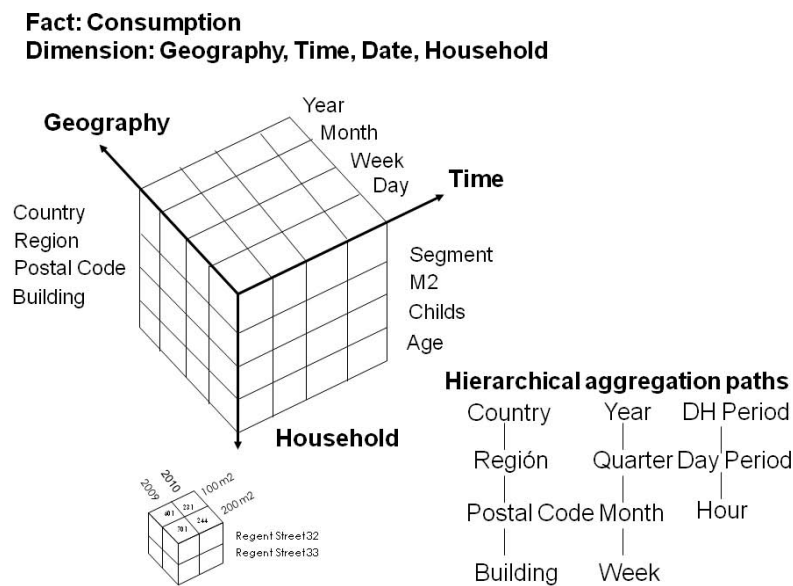


Figure 3.5: OLAP multidimensional data model

resulting numerical reports. OLAP provides an efficient search on a high dimensionality and historical data search space, describing facts according to different dimension perspectives.

There are several interesting works in which authors combine OLAP with Fuzzy Logic (36) (106) (67) (76) (37) (47). Related with this work, fuzzy summaries have been developed over multidimensional databases in order to accomplish appropriate linguistic summarization of data cubes containing historical information using fuzzy quantified

statements (73) (25). Also, fuzzy decision trees (74) and fuzzy association rules (4) extracted from OLAP have been developed to deal with numerical values in a natural way in mining processes.

3.3.1 Semantic Meta-data for OLAP Schema Definition

OLAP analysis is performed over a multidimensional data model with a specific data structure that constitutes a declarative memory about facts and events. This multidimensional information structure, defined over a relational database, is usually conceptualized as a data cube as showed on Fig. 3.5. The data cube conceptual definition is usually referred as a star schema, where the fact is the centre and dimensions are placed around the fact.

In a data cube, the different faces compose the dimensionality of the facts (22) (68). Facts are quantified, and the dimensions specify the quantification context defining what, which, where, who, when, etc. Each cell within a cube contains aggregated data related to elements along each of its dimensions.

Multidimensional data models are described by meta-data, a data semantic description that establishes a higher level of knowledge over data. The meta-data is defined by a designer to provide to the user an abstraction from technical issues, adding easier capabilities to understand and manipulate the data, independently from the technical structured query language. The designer adds a logical semantic definition that formalizes the information physically stored in the database.

The semantic modelling defines fact measures and dimensions hierarchical attributes over the physical database tables and fields. The user queries the database using the logical attributes and measures. Due to the hierarchical organization of dimensions, detail in the data cube can vary according to the selected level chosen. The meta-data contains relationships between both fact-dimension and hierarchical-dimension attributes.

3.3.2 OLAP Schemas and Computational OLAP Model

Analysis of information in OLAP is made up by “*what-if*” scenarios executed within the context of some specific historical basis and perspective. The treatment of information based in OLAP and relational algebra, has been studied in a different works (3) (116).

Agrawal proposed a data model and a few algebraic operations in order to provide formally a semantic foundation to multidimensional databases on which OLAP is based (3). This model provides support for multiple hierarchies along each dimension and support for ad-hoc aggregates. The different levels of the dimensions hierarchies allow accessing the facts at different levels of granularity. In order to do so, classical aggregation operators are needed (maximum, minimum, average, etc.).

The exploration of the information via OLAP is possible due to the logical semantic data definition that formalizes the information physically stored in the database. The

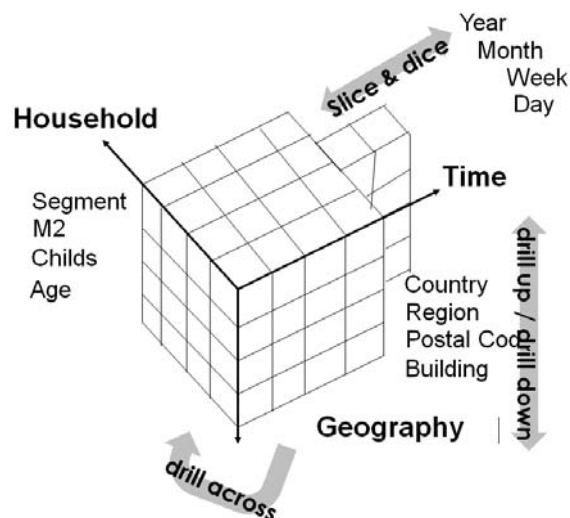


Figure 3.6: Computational OLAP model operations

OLAP engine accesses the relational database to provide responses to queries. Response is based on a table display showing a specific view of the cube. Each query returns a data view composed by attributes disposed in columns and rows, and measures as central values of the data view.

A multidimensional OLAP schema MD^n is a logical structure composed by data cubes as the basic unit of operation, e.g., an OLAP schema could be designed to organize the information about the *fact* “Consumption of energy in an household” (See fig. 3.5). An OLAP schema can be defined as a tuple (D, M, F, V) as follows:

- D is a finite set of dimensions $D = \{D_1, \dots, D_n\}$ with domains $dom(D) = \{dom(D_1), \dots, dom(D_n)\}$, e.g., $D = \{Geography, Time, Date, Household\}$. Dimensions are characterized by dimensional attributes that provide a hierarchical organization of data from the most detailed to the most general granularity, e.g., $Time = \{Hour, Day, Week, Month\}$. These attributes take numerical values, e.g., $Hour = \{1, 2, \dots, 24\}$.
- M is a set of measures of the characteristics of the *fact* $M = \{M_1, \dots, M_m\}$, e.g., $M = \{Consumption[kWh], Cost[Euros]\}$.
- F is the mapping function F based on OLAP operations (See below). $F(MD_q^n) : dom(D_1) \times \dots \times dom(D_n) \rightarrow V$. It maps a set of values of dimensional attributes to the corresponding measures.
- V is a vector of measures (m_1, \dots, m_m) obtained in a specific context defined by the values in $(dom(D_1) \times \dots \times dom(D_n))$.

3.3.3 Computational OLAP Model Operations

Navigation is a term used to describe the process of dynamic exploration of a cube. Navigation through scalar operations work “horizontally” along dimensions attributes and

navigation through aggregation operations work “vertically” along dimensions hierarchies as showed on Fig. 3.6.

By *drilling-across*, the cube can be rotated to show a particular dimensional face, by *slicing and dicing* some subset of the face is selected in order to restrict the set of data retrieved, and by *rolling-up* and *drilling-down* through a dimension hierarchy the information can be summarized or detailed extracting higher or lower granularity.

3.4 Affective Computing

Affective computing deals with the capability of perception and interpretation of human emotional content in computational environments. While computing with emotions is tried to assign computers the humanlike capabilities of observation, interpretation, and generation of emotions. Basically the ground of the process is based on an internal emotional model that manages an internal state and externalizes emotions (expressing emotions) due to the ability of triggering an outward appearance for each emotion. In embodied agents the emotional expressions are used to control gestures, voice and language, also the model of how the system should behave is used in order to convey the intended expression.

Researchers in areas such as cognitive science, philosophy, and artificial intelligence have proposed a variety of models of emotions. Emotions are an important aspect of human intelligence and have been shown to play a significant role in the human decision-making process (42). Arbib and Fellous (6) present a comprehensive overview of the attempts to improve human-computer-interaction through the simulation and expression of emotions.

The emotional model proposed by Ortony (89) has been the basis for the integration of emotions into cognitive architectures of embodied characters (45) (99). It was designed as a computational model of emotions. Ortony proposed several questions around emotions “*What causes us to experience emotions? What makes emotions vary in intensity? How are different emotions related to one another and to the language used to talk about them? What are the information processing mechanisms and structures that underlie the elicitation and intensification of emotions?. Despite an abundance of psychological research on emotions, many fundamental questions like these have yet to be answered*”. Ortony (88) argued that it might be enough to integrate only ten emotion categories, five positive and five negative, into the architecture of an agent when focusing on believability and consistency of an agent’s behaviour. When focusing on facial expressions of emotions this number can be further reduced to six “basic emotions” (41). Different approaches were proposed for modelling emotions (10) (19).

3.4.1 Modeling emotions

There are general models about the universality of emotions (34) (50). In many domains emotional models are used, such as, for example, in conversational agents (40). Various

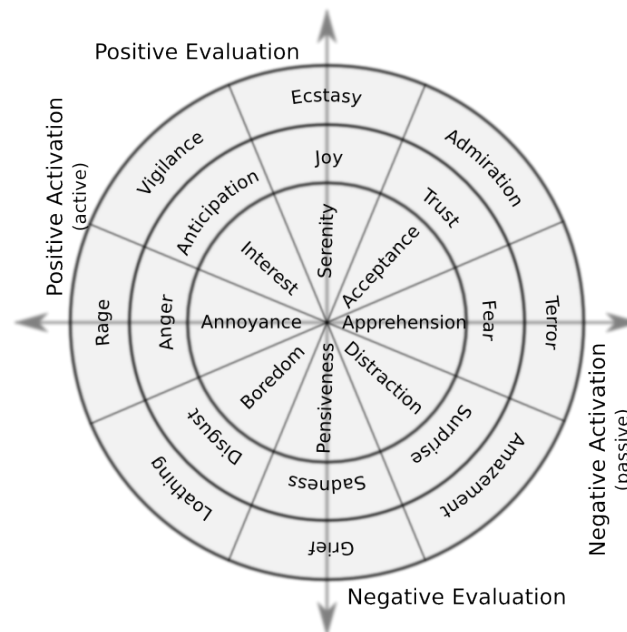


Figure 3.7: Simplified version of the Plutnik Wheel

models have been proposed and many of these models come up with different scales in which the different emotions can be located.

Basically emotions are categorized along two axis, activation (from passive to active) and evaluation (from negative to positive). Basic or archetypical emotions are situated in a circle around the centre point, which represents a neutral emotion-free state. The distance from the centre gives an indication of the emotional strength, in other words, the further away from the centre, the more intense the emotion as showed on Fig. 3.7

The two dimensions (arousal and valence) proposed by Breazal (19) are very often used in emotion modelling because they correspond to physical characteristics that can be recognized.

Fuzzy logic for modeling emotions

Fuzzy Logic allows to model complex emotional states and behaviours. A emotional state is fuzzy, i.e., at any point in time the system can be in various emotional states to certain degree. Fuzzy Logic has been already used in emotion simulation. Several works have been performed in the European Center for Soft Computing, for example to represent the emotional state in computer interfaces (127), or to explore the possibilities of modelling the state of stress of a person using as sensors a skin conductivity meter and accelerometers in (126).

Fuzzy Finite State Machines (FFSM) are suitable tools for modeling systems that evolve in time following an approximately repetitive pattern (118) (125). The input are basic emotional variables that are fuzzyfied. The FFSM acts as the memory of the internal emotional state as showed on Fig. 4.22, in which the system can be in various states at the same time, to a degree. The values of activation and evaluation are used to determine what emotional state the system is in, which is determined in a mapping.

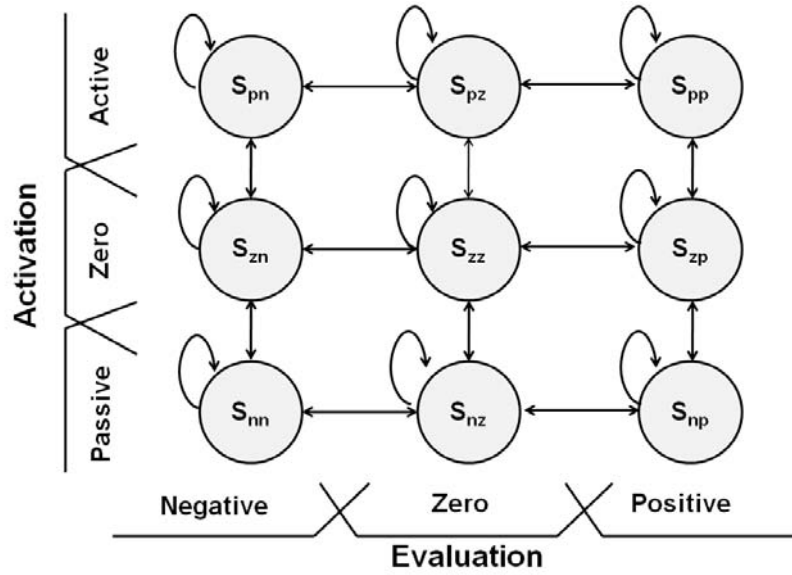


Figure 3.8: States of the FFSM

A FFSM is a tuple (U, S, f, Y, g, S_0) where:

U is the input vector $(u_1, u_2, \dots, u_{nu})$, where nu is the number of input variables.

S is the state vector $(s_1, s_2, \dots, s_{ns})$, where ns is the number of states.

f is a function to calculate the state vector at time step $t+1$, $S(t+1) = f(S(t), U(t))$.

Y is the output vector $(y_1, y_2, \dots, y_{ny})$, where ny is the number of output variables.

g is a function to calculate the output vector at time step t , $Y(t) = g(S(t), U(t))$.

S_0 is the initial state of the system.

3.4.2 Expressing emotions

The aim of expressing emotions in a multimodal output representation is to include emotional content in the combination of words, grammatical construction, voice and gestures while communicating. In face-to-face communication, the emotional state of the speaker is transmitted to the listener through verbal and the nonverbal communication. Facial expression, Speech processing and Language pragmatics model the output while expressing emotions.

- Faces

Faces convey information relevant to the expression and interpretation of emotions as they have a meaning of their own. Movements such as a smile, a nod or a shoulder shrug enrich semantics in communication and facilitate understanding. Ursula (59) proposed the following question, “*Why the same expression may not*

mean the same when shown on different faces or seen by different people?” . It is therefore important to not only focus on creating truthful semantic expressions but also to keep in mind that the face on which these expressions appear is not emotionless.

- Voice

Modifying the expressed voice is of importance in human interaction as it transmits added valuable information in a conversation. It is thus necessary to include affect in speech to differentiate emotional speech from normal speech (108) (109).

Synthetic speech nowadays is mostly intelligible. In some cases, it also sounds so natural that it is difficult to distinguish it from human speech. However, the flexible and appropriate rendering of expressivity in a synthetic speech is still on development.

The study of the stress, tune and rhythm of speech and how these features contribute to meaning is a challenging task. Prosody can have the effect of changing the meaning of a sentence by indicating a speaker’s attitude to what is being said.

- Language

While much attention has been paid to the effect of emotion on non-verbal behavior (54), less work has been done on the effects of emotion on the verbal behavior of embodied agents.

In language generation, Hovy asked (63) *“Why and how is it that we say the same thing differently to different people, or even to the same person in different circumstances?”* and concluded that we vary the content and form of our text in order to convey more information than is contained in the literal meanings of our words.

Hovy defined and develop a text generation program PAULINE that took into account the pragmatic aspects of communication. Pragmatics examines the distinction between the literal meaning of a sentence and the meaning intended by the speaker. In his thesis, Hovy (60) implemented a 3-valued (positive, negative, neutral) system of emotional shades in text generation. To characterize the pragmatics of its conversation used features with a number of possible values. In (48) described an integrated framework for modeling emotion in a virtual agent.

3.5 Language generation

Reiter and Dale (102) described natural Language Generation (NLG) as *“the developing techniques for automatically generating NL descriptions from non-linguistic information”*. In order to generate linguistic descriptions, the text generation process must be able to determine what information to communicate and how to organize this information to accomplish a communicative goal. This is to define why to communicate (motivation), define what to say (content definition) and define how to say it (content expression).

While communicating it is not enough to place together a set of sentences, but also to ensure that they are placed in a coherent and cohesive discourse. In a discourse the information is ordered in a set of ordered and related sentences displaying a pattern that the hearer can easily understand.

The NLG process can be decomposed as a pipeline of tasks. As described by Reiter and Dale (102) and Hovy (61) (60), while expressing in NL three key tasks are performed:

- Content determination

Content determination or content selection refers to the definition the desired output content and structure. Here it is defined what should be said in the final discourse.

The tasks related at this stage are those related with the identification of the appropriate content to express, to select relevant information for the discourse purpose. It is based in establishing criteria to identify information available and the relevancy in the content to express.

- Content planning

Content planning refers to the use of language resources required to express content such as lexical items and syntactic constructions. The tasks of text planning are based in arranging the messages from the defined content, as it is not enough to list sentences while generating language. At the end of this stage it is defined how should be said the discourse.

Content planning is in charge of organizing the information into a rhetorical coherent structure. This task is in charge of making grammatical decisions about how the propositions selected in content determination should be represented. The output of a sentence planner is a list of clause units containing a syntactic specification for each clause.

It must be agreed, that the following text is not satisfactory, *“The consumption in 2000 is high. The consumption in January is high. The consumption in February is high.”*. In order to understand it, to make sense of it avoiding unrelated pieces of text, it is required a structured relation between clauses, *“The consumption this year is quite high. The highest consumption was during the months of January and February.”*.

Different models to link successive sentences have been studied. The two principal methods for performing the text planning tasks involve schemas and rhetorical relations. Text Schemadata was developed by McKeown (82), where schema data are *“ representation of standard pattern of discourse structure which efficiently encodes the set of communicative techniques that speaker can use for a particular discourse purpose”*. The Rethorical Structure Theory defined by Mann and Thomson (80) is *“a descriptive theory of a major aspect of the organization of natural text”*.

- Content realization

Content realization makes reference to the tasks involved in the production of the final text string. Here the output text is instantiated following previous definition of what and how should be said.

Content realization is in charge of performing the linguistic surface generation. The system takes the individual sentence plan and creates human-readable sentence text by making the final decisions about words, their inflections and their order. The systemic grammars represent sentences as collections of functions and maintain rules for mapping these functions into explicit grammars forms. Widely used systems are Penman (79) later extended as KPML (12) based on systemic grammars and FUF/SURGE (86) (44) (43).

An implicit assumption of most NLG techniques is that the non-linguistic input information comes from knowledge bases with well-defined semantics. In practice, however, in most application domains where automatic textual descriptions are required, such knowledge bases do not exist. Data-to-text NLG is an extension to traditional NLG to allow to describe data linguistically, as natural language makes data more accessible to human users than traditional databases environments (100).

Several approaches have been reported in the literature. Kukich (70) summarized stock market data, Yu (133) reported a NLG system that produces textual descriptions of time series data from a gas turbine. Sripada reported a NLG system that produces textual weather forecasts and a system that described data recorded by a scuba dive computer (113) (114). Reiter with a broad work in the area defined Baby talk, that interprets large amount of clinical data to be presented effectively to the medical staff (93). HYSSOP, is a system that generates natural language summaries of insights resulting from a knowledge discovery process involving OLAP and data mining (104).

To resume all the process and operations performed while communication could be used the Hovy expression “*Pity the poor speaker. All this information, and more, must be packed into each clause!*”.

3.5.1 Gricean Maxims

A discourse contains a set of sentences or clauses that have to be coherent, using connected and comprehensible sentences.

In 1975, Paul Grice proposed a conversational maxims in “*Logic and Conversation*” (55) (56) to fix grounds in a conversation, however, there have had critics of these maxims because of being partial.

Grice proposed that a coherent discourse has to be informative, accurate, understandable and relevant and he sets four maxims. It has to be informative, presenting new and interesting information. It has to be accurate, containing accurate and reliable information. It has to be understandable, representing information that the user can understand. It has to be relevant, including information which is relevant to the current discourse goal and not redundant.

The four maxims described by Grice are:

- The maxim of quantity, where one tries to be as informative as one possibly can, and gives as much information as is needed, and no more.
- The maxim of quality, where one tries to be truthful, and does not give information that is false or that is not supported by evidence.
- The maxim of relation, where one tries to be relevant, and says things that are pertinent to the discussion.
- The maxim of manner, when one tries to be clear, brief and orderly as possible in what one says, and has to avoid obscurity and ambiguity.

3.5.2 Discourse Coherence and Cohesion

The study of discourse in linguistics is concerned with analyzing how phrase or clause level units of text are related to each other within a larger unit of text. Coherence is a matter of conceptual unity and cohesion is a matter of linguistic form.

- Coherence

Coherent text (62) can be defined as text in which the hearer knows how each part of the text relates to the whole. A text is cohesive when the hearer knows why it is said and the hearer can relate the semantics of each part to a single framework.

Coherence occurs when the hearer or reader is able to fit the different discourse elements into a mental representation, as opposed to a random sequence of sentences (66).

- Cohesion

Halliday (57) (78) described a text as a unit of language in use, not a grammatical unit, like a clause or a sentence, but a semantic one. He also described the clause complex as an enabler to functional organization of sentences. The relation between clauses can be interpreted in terms of the logical components of the linguistic systems.

Text derives from the fact of functioning as a unit with respect to its environment and has a texture. There are different resources for creating texture, for providing a cohesive relation between sentences.

Cohesion occurs where the interpretation of the elements in a discourse are dependent. Cohesion is mainly a semantic relation and is expressed as relations to information reported in clauses or sentences. As described by Halliday and Hassan the texture is provided by cohesive relations or ties that exist between sentences.

Referring expression, substitution, ellipsis and conjunction

As expressed by Halliday and Hassan (57), “*potential for cohesion lies in the systematic resources of reference, substitution, ellipsis, conjunction and lexical cohesion*”. For example, the text “*This year the efficiency is high*” has a referring expression where “*This year*” refers to a temporal referent which identifies a time period.

Referring expression generation involves identifying specific entities to the reader. A referring expression is an expression used to refer to another object or person known in the context. A variety of algorithms have been developed in the NLG community to generate different types of referring expressions (11) (17) (101). Referring expressions are broadly used, e.g. “*He (the advisor) suggested reducing the use of devices.*”.

Substitution is a relation implemented using a different word that maintain meaning while referring to something else, e.g. “*The efficiency was low in February this year (In 2013).*”. The substitution may function as a noun, as a verb, or as a clause.

Ellipsis is a substitution by zero, it occurs when something is omitted because has been mentioned previously and do not need to be repeated “*Your consumption in January was high, and (your consumption) low in February*”.

Quirk (96) described the semantic use of conjunctions. Halliday and Hasan (57) subdivided conjunctions into four categories, according to the relationship they express. They explored the function of conjunctions in great detail. For example discourse connectives include coordinating conjunctions, e.g. *and, or, ...*, subordinating conjunctions e.g. *because, when, since, ...* and discourse adverbials, e.g. *however, previously, nevertheless, ...*

They also gave a detailed review of additive, adversative, causal and temporal conjunctions. As a review, additive relation link units of semantic similarity, e.g. *and, or, likewise, furthermore, ...* Adversative conjunctions indicate a contrary result or opinion to the content mentioned previously, e.g. *but, however, on the contrary, ..*. Causal relations are used to introduce result, reason or purpose e.g., *so, thus, hence, therefore, consequently, ...* Temporal connectives express the time order of events, e.g. *then, previously, before that, ...* Temporal relation also includes the sense of conclusiveness, e.g., *finally, to sum up, in short, ...*

3.6 Question Answering Systems

Automated question answering systems (QA) have been a topic of research and development since the earliest AI applications. There are many references to natural language interfaces. A QA system is a type of information retrieval environment, based on natural language, where input queries requires understanding and deductive reasoning over a background knowledge (131) (23) (142). They can be classified as closed-domain or open-domain whether the domain knowledge is previously completely defined or not.

Zadeh said, that if he were asked, “*What is the most challenging problem in the realm*

of information science and technology?” his unequivocal answer would be conception and design of question-answering systems. The goal of a question answering system is to retrieve answers to questions rather than full documents or best-matching passages, as most information retrieval systems currently do.

QA system may be faced with the tasks of extracting the meaning for mapping questions to answers covering lexical, syntactic, semantic and pragmatic relationships between question string and answer string. An introduction involving interfaces to data bases can be found in (92). Also some fuzzy approaches have been developed (14) (72). In (14) it was proposed a comparison between traditional interfaces to data bases in order to obtain the desired results.

3.6.1 Perceptions in Question Answering Systems

Zadeh underlies three key issues in Question Answering systems. The first obstacle is that world knowledge is perception based. e.g., “*Most of your devices have a low efficiency*”, the second obstacle is the concept of relevance of the answer to the question proposed, and the third one is the deduction capabilities from perception based-information (139) (Qin et al.).

Zadeh underlined that Question Answering systems takes responsibility for providing answers, rather than a searchable list of links to potentially relevant documents “*There is a basic capability which existing search engines do not have: deduction capability – the capability to synthesize an answer to a query by drawing on bodies of information which reside in various parts of the knowledge base*”.

Zadeh proposed Precisiated Natural Language (PNL) for reasoning with problems expressed in natural language and computing with perceptions (138). PNL protoforms, as discussed by Zadeh, is one representation of natural language based on constraint-centered semantics, which is convenient for computing with words. A protoform is a concept that facilitates the reasoning with PNL expressions. The protoform language is the set of protoforms of the elements of a generalized constrained language (GCL). A prerequisite to mechanization of question-answering is mechanization of natural language understanding, and a prerequisite to mechanization of natural language understanding is precisiation of meaning of concepts and proposition drawn from a natural language.

In PNL, the meaning of a proposition p , can be represented as a generalized constraint on a variable. Schematically, this is represented as:

$$p: X \text{ is } R$$

where X is a proposition

$$X = \text{Consumption (January ())}$$

and R is a constraint relation that express the degree of truth for the distribution of X .

$$\mu_{\text{LOW CONSUMPTION}}(x) \begin{cases} 1 & \text{for } x < 100 \\ (200 - x) / 10 & \text{for } 100 < x \leq 200 \\ 0 & \text{for } x > 200 \end{cases}$$

Figure 3.9: Membership function example.

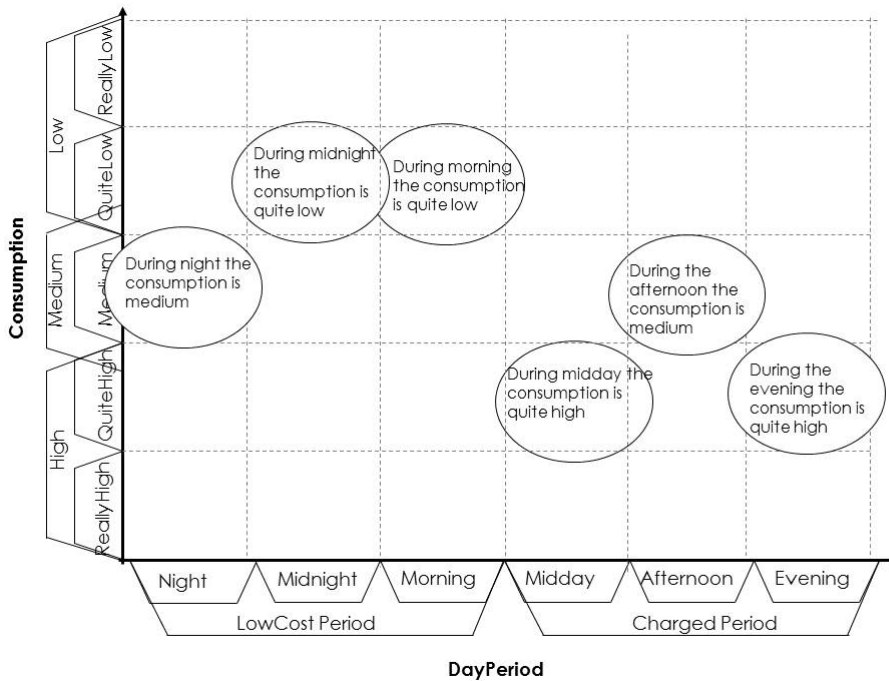


Figure 3.10: Fuzzy graph.

R is identified as a fuzzy set that represents its meaning. It is defined in the universe U which is the domain of the base proposition. For example, in the sentence “*The consumption is low*” the consumption value is treated as a linguistic variable *Consumption* and “*low*” is the value assigned to it, which may be represented by a fuzzy set, e.g. by the membership function showed on Fig. 3.9, defined in the domain $[0,1]$ of the value variable associated with the linguistic variable *Consumption*.

These linguistic statements make possible to represent imprecise information about the value of a variable. Represented as a fuzzy graph constraint, the constrained variable could be a complex proposition, e.g. “*During the low cost period the consumption is quite high*” and R is a fuzzy graph showed on Fig. 3.10. Here the fuzzy graph constraint is represented as P is $\mu(AS) = (\mu_A(u)\mu_S(y))$.

How is the consumption in January?
 q : *Consumption (January) is ?R*
Consumption in January is quite high.

p: Consumption (January) is quite high

During interpretation, query language expressions are modelled as perceptions to capture imprecision inherent to natural language. A query is precisiated and abstracted into perception protoforms that captures the semantics in formal structures. A perception will be treated as unitary element of computation. Some approaches can be used to precisiate linguistic expressions (18) e.g. “*Most of the months the consumption is high*”.

q: How is my consumption during the morning ?
p: (Consumption(MorningPeriod)) is ? R

3.6.2 Question Interpretation

To understand language a computational system must implement the models and processes able of decomposing a sentence into its basic constituents, understanding and coding them in a formal model. The formal model stores the semantic meaning of the lexical constituents, the emotional contents associated in the sentence and of the global grammatical structure that defines the semantic relation between each constituent. Also in question interpretation the system must analyze the question in the context of ongoing interactions through the general dialog.

Actual systems lack interpretation capabilities because exists several reasons for which a natural language interface to databases could fail. A basic, but main reason is that the user has to understand the domain of the query system that underlies usually over a restricted domain of knowledge. Some previous approaches can be find in PLANE (132), an English language question Answering system for a large relational database.

Here are reviewed some of the steps followed in the interpretation process.

- Tokenization

Tokenization is the task of splitting a string into identifiable linguistic units that constitute an independent piece of language data. The key step in this stage is to identify the word stems, clean empty tokens and prepare each token for the rest of the process.

- Syntactic Tagging

The syntactic process incorporates algorithms for detecting syntactic relations between clause elements.

Tagging is used to categorize words; a basic process for tagging could differentiate between nouns, verbs, adjectives, prepositions and adverbs. Unigram taggers are based on a process where for each token, is assigned the tag that is most likely for that particular token. For example, it will assign the tag “*adjective*” to any occurrence of the word “*Hourly*”, but while asking for “*Hourly consumption*” the tagging and meaning would be misunderstood.

“*What is my consumption on Monday hourly?*”, should return the consumption on Monday per hour, and “*What is my hourly consumption on Monday?*” should return the average hour consumption on Monday.

In the case of n-gram tagging, basically are considered not only the current token, but each token in a n-dimensional context. An n-gram tagger is a generalization of a unigram tagger whose context is the current word together with the part-of-speech tags of the n preceding tokens.

- Semantic Tagging

The semantic process incorporates algorithms for detecting semantic relations between clause elements (87).

Question semantics depends heavily on the type of question. Question taxonomy deals with the classification of question in a subset of classes, that narrow the search space of an answer associated. An specific review can be found in (91). “The Five Ws” is a simple and common classification of questions in English “*Who, What, When, Where, Why, and How*” (105). Taxonomies also considers the relation between the question and the answer to the question (53) “*Verification, Disjunctive, Concept completion, Definition, Comparison,..*”.

- Emotional tagging

The emotional tagging process incorporates algorithms for detecting emotional elements in the clause elements.

In Question Answering systems manage two directions of emotions, the side of the speaker and the side of the user who is consulting. This information extracted from the text, and referred as emotional meaning. Emotions can be extracted from text interpreting expressions with negative content and others with positive one, e.g., what a pity, ups, etc.

3.7 Virtual Assistants

Some research have been done in the field of virtual interactive agents. The advantages of this kind of interaction are well studied (97). Virtual Assistants simulate to think and interact like a human. Basically they should be able to hold a conversation understanding user requests and delivering related responses, even simulate that have a personality making the interaction more human-like. In a conversation with a virtual agent a user can express what he wants in the same terms and language that he would use in a shop or service centre, and get a human-like answer instantly. The fact is that most of the Virtual Assistants actually available are closer to a string pattern matching for question - answer that to a more intelligent behaviour.

Websites have become the principal service channel for an increasing number of users. Studies show that people prefer to visit websites when they need to gather information or make a purchasing decision. The online experience lacks the personal service that differentiates the web use and the personal attention that guides the visitor to the information or solution that they are looking for. A Virtual Assistant should be able to help more directly, focusing on user interaction and requests, instead of having to navigate through pages in complex website structures or search through frequently asked questions lists. Interacting with a personal agent, usually means to find a helpful person who is there

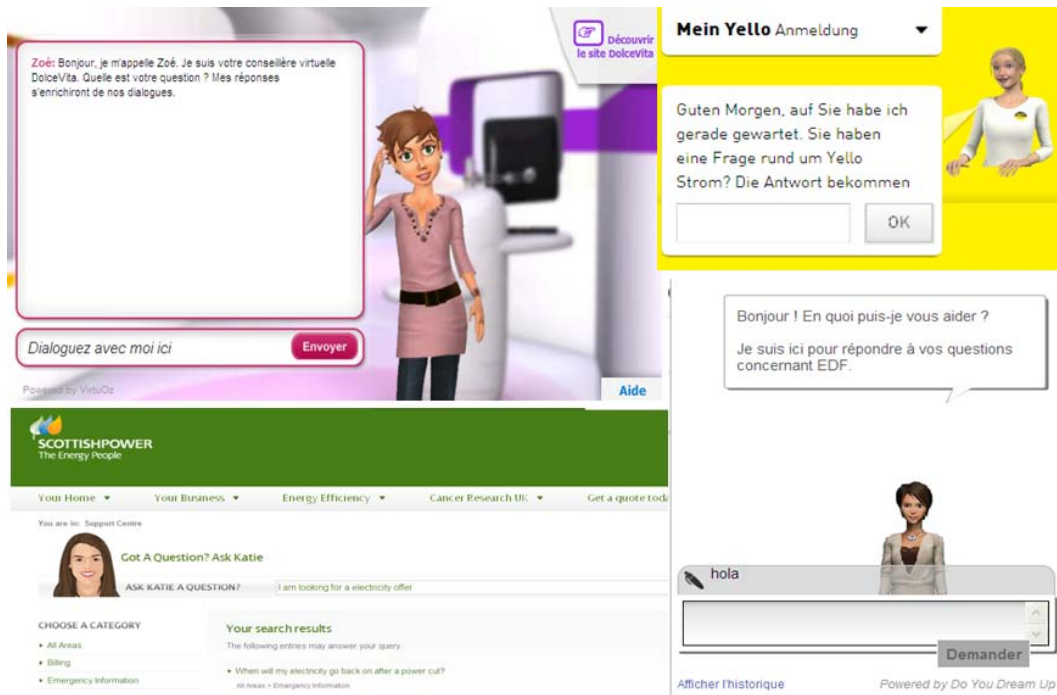


Figure 3.11: GDF, Scottish Power, YellowStrom and EDF Virtual assistants examples.

to make it easier for a customer to find what they are looking for, or to find the expert service to get something fixed.

Possibly A.L.I.C.E. (130) is the most famous generalist English speaking interactive agent. It comprises a pseudo natural language interpreter which emulates human conversation in order to make the user feel comfortable during information retrieval. The conversation is carried out based on an Artificial Intelligence Markup Language (AIML) knowledge-base and an AIML interpreter. The knowledge-base is made of pairs of patterns “*Question-Answer*” and templates, which can be linked together semantically and recursively.

Industry has identified that a new generation of Virtual Assistants can offer a personal touch. Organizations like GDF, EDF, Scottish Power or Yellow Strom have seen benefits that Virtual Assistants can bring. Some of them are showed on Fig. 3.11. Zoe from GDF SUEZ can help you find the best solution for your energy needs. Laura is the Virtual Advisor of EDF Bleu Ciel, she answers over 10,000 questions a day on various topics such as billing or opening contracts. Eve is a Virtual Assistant that helps people using the website of the German electricity supplier Yellow Strom. Katie is a chatbot from ScottishPower that helps to navigate through information in the web. Elin from E.ON helps users with frequently asked questions.

3.8 Architecture Design

The availability of a generic architecture is required when dealing with the project of designing a new computational system. The research and development in autonomous

agents has produced different architectures (15).

CPA7L has been explained in detail by Trivino (120). This architecture allows a robust design of autonomous agents. Here is presented a brief description of the main components of this architecture.

The distinction between Concepts, Procedures and Attitudes (CPA) has its origin in Cognitive Psychology and appears in the Theory of Elaboration introduced by Reigeluth (98). These researchers were investigating the contents of the learning process in the general context of Education. Some years later this classification was used as one of the theoretical bases of the Spanish educational system. In the book by Coll (31) definitions of the key concepts of the theory are provided:

- *Concepts* designate a set of similar objects, events, or symbols with common characteristics. Examples of concepts are: mammal, triangle, cloud, etc.
- *Procedures* designate a set of ordered actions aimed at a goal. Examples of procedures are: to subtract two numbers, to draw a map, to write a summarization, etc.
- *Attitudes* correspond with a prevailing tendency to act in a certain way and arise in front of determined situations, objects, events or people. Examples of attitudes are: to keep your clothes clean, to enjoy listening to classical music, etc.

For example, let us suppose that the objective of a class is to teach the students how to use a map. An analysis of what this objective involves can reveal the following elements: First, the students must learn the concept of map, the concept of scale, symbols for road, forest, etc. Next the students can learn the procedures to locate their position, to define a path, etc. The attitude to teach is that a map as a useful tool in certain situations. This attitude will help the students to use maps when the situation requires it, for example, when the planning of a holiday trip.

Figure 3.12 shows the seven levels of the architecture (arrows represent relations of use). The higher level components add new functionality to the system using the resources provided by the components situated below. The CPA7L layers are described as follows:

- *PhysicalLevel*

This level contains a description of the physical elements of the system, such as sensors, actuators, visual/physical appearance, communication devices, as well as other parts of hardware of the processing system.

- *SymbolicLevel*

For processing information about the physical world it is necessary to establish a bridge between it and the symbolic world (94). The Symbolic level includes a description of primitive concepts (communication channels with hardware) and primitive procedures (software primitive functions) which are used to access sensors.

- *ReactiveLevel*

Describes the mechanisms with which the system reacts autonomously in real time, using the basic concepts and procedures provided by the Symbolic level. As in the previous levels, the elements at this level are usually highly optimized and cannot be easily modified (e.g. by learning) in a practical application (20).

- *StructuralLevel*

The previous levels could be sufficient to describe a simple autonomous system. Normally additional software infrastructure is needed for a more complex semi-autonomous system, such as, an operating system, a communication system, a database management system, etc. These elements are organized at the Structural level and provide the support needed for the higher levels. It isolates higher level functionality from lower level functionality.

- *ConceptualLevel*

Contains a network of concepts corresponding to the world where the agent operates. This World Model not only contains concepts related with the perceptions in its environment, but also concepts related with the perception of the elements that are part of the itself, as well as abstract concepts related with the objectives.

- *ProceduralLevel*

The view of the architecture at this level contains the procedures associated with the network of concepts available at the conceptual level. Using these procedures it is able to create a possible plan of action that will lead the system from the current conceptual situation to a new situation with a higher degree of goal achievement. This level describes the high level operations that the agent can perform. In the procedural level is the definition of methods of a class like in Object Oriented Programming, and in the conceptual level, there are the attributes.

- *AttitudinalLevel*

This level describes the expected system behaviour in different circumstances. It includes descriptions of the willingness of the agent to achieve certain objectives, or the preferences of some actions over others. Note that these descriptions will be done by making reference to the concepts, procedures and basic attitudes introduced in the previous levels.

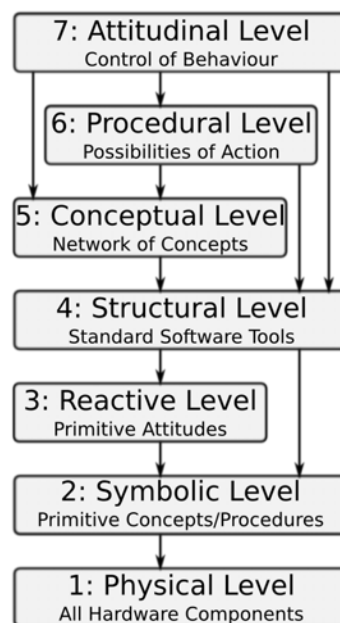


Figure 3.12: A view of the architecture structured in seven levels including the relations of use.

Chapter 4

Extension proposed to the applied technologies

4.1 Introduction

The abilities of computational systems to query and extract relevant knowledge from available data using natural language to communicate are limited. The industry demands a new generation of human computer interfaces able to provide users with enough expressiveness, flexible querying and intelligent searching.

For example, imagine a user calling to a telephone contact centre asking for information about his efficiency, e.g., a quite simple query as “*how is my consumption?*”.

This simple query requires quite a high volume of meaning interpretation and answer reasoning. To support the query there exists a large database with the registered hourly consumption and the expert knowledge of an *Energy advisor*.

The data base model contains data about consumption registered by time, date, customer and geographic area. A direct query is of any use, since there is a first step of interpretation to find out when consumption can be considered *high* or *low*. The advisor decides to use the average consumption of customers in the region fixing thresholds to be used for comparison to identify if the user consumption is high or not when compared to the average. He requires a data base analyst to introduce the query in the data base system using the necessary SQL statements.

SQL will not provide the level of access required in a simple query. Even when finding the result with different queries, e.g., the consumption of the last months for the customer and the average consumption for the customers of the region. The advisor would have to summarize the information obtained, in order to hide the irrelevant data and communicate a final response to the user.

In the research line of Computational Theory of Perceptions proposed by Zadeh, this thesis defines and develops a computational system to implement a *Personal Energy Advisor*. The thesis is developed upon previous research exploring the possibilities of obtaining from natural language queries, computational perceptions that define responses using the granular linguistic structure for modelling the knowledge. A *Personal Energy Advisor* should be able to generate, as requested by users, linguistic reports from the

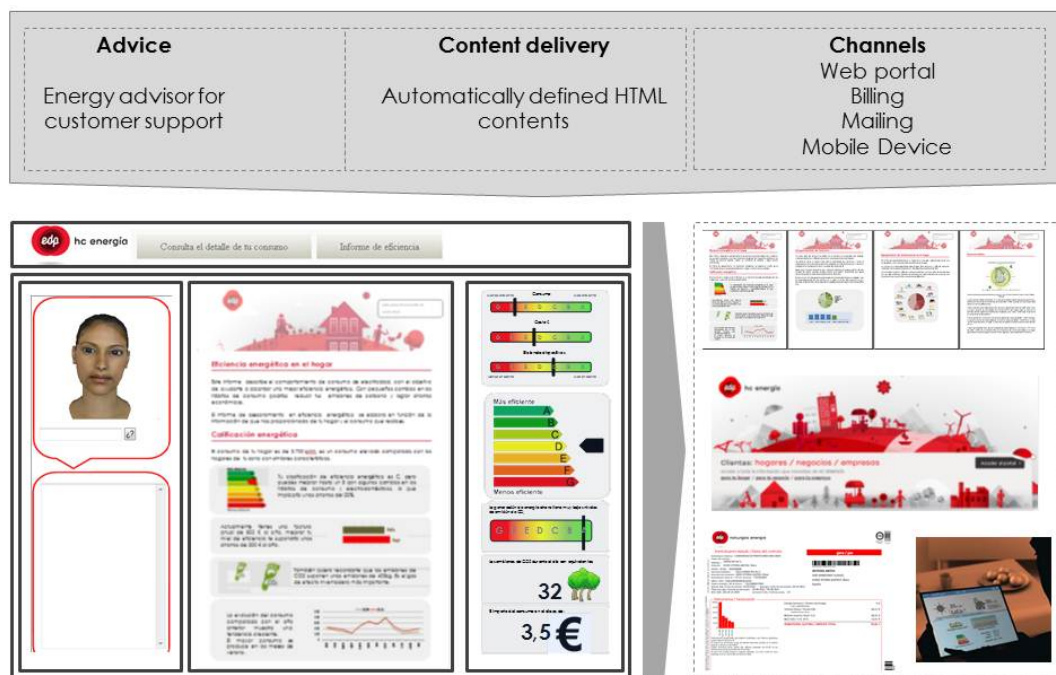


Figure 4.1: Information sample

available information in a database adding reasoning closer to user understanding and expert advisory services.

This chapter presents a review of all the extensions proposed to the paradigms described in previous chapter. These extensions placed together, will support the objectives of the *Personal Energy Advisor*, that should be able to extract advice and deliver contents using different potential channels to communicate with the users as showed on Fig. 4.1.

The solution proposed is based on previous studies on the fields of computational theory of perceptions, data processing, natural language processing, affective computing and autonomous virtual agent modelling.

In the section “*Dynamic navigation through perceptions*” is described the dynamic navigation through perceptions retrieved from a data base expressed in natural language. With this purpose a Granular Linguistic Model over an OLAP schema is defined, this model is referred as GLMP-OLAP. The typical operations in OLAP, namely drilling, rolling and slicing/dicing are applied to navigate by hierarchical granular structures of fuzzy perceptions. At this stage the dynamic retrieval of language using perceptions is performed over a defined knowledge domain over a data base. This proposal “*OLAP Navigation in the Granular Linguistic Model of a Phenomenon*” has been published in (85).

In the section “*Dynamic discourse generation based on perceptions*” the resulting perceptions acquired through navigating the knowledge domain are arranged in a structured discourse. Here are explored the possibilities to reorganize the structure of the GLMP to deploy a coherent and cohesive linguistic discourse, avoiding repetition and linking individual granular sentences in a set of connected ones. It is performed due to a process of (a) content selection to generate an initial GLMP from an input domain and (b) content planning for the reorganization of the resulting GLMP in order to define the discourse structure and (c) content realization to instantiate the GLMP. The reorganization is performed by pruning non relevant perceptions and by aggregating and linking related computational perceptions. The concept of relevancy is attached to each perception to identify the relevant information in a dynamically generated GLMP. This proposal “*Selection of the Best Suitable Sentences in Linguistic Descriptions of Data*” has been published in (83).

In the section “*Emotional perceptions*” the dynamic generation of perceptions include an emotional component. In this case the GLMP will be able to express emotions based on user behaviour increasing the expressivity of the language in communication. A new dimension with an emotional state associated to each perception, and an emotional process based on a Fuzzy Finite State Machine (FFSM) is used to compute emotions dynamically in the OLAP-GLMP. This proposal “*Generating advices with emotional content for promoting efficient consumption of energy*” has been accepted for publication in the International Journal of Uncertainty, Fuzziness and Knowledge-based Systems (84).

In the section “*Question Answer perceptions processing*” in order to increase and facilitate the user interaction, is proposed a question interpretation model based on perceptions over the knowledge domain. This extension allows dynamically interpreting user questions and dynamically computing the answers in a dialog with the user.

In the section “*Energy Advisor Architecture*” the different components for the *Personal Energy Advisor* are arranged to allow the system to manage the user request and provide response advising about the consumption efficiency.

All the elements described in this chapter placed together, define a proposed framework for a “*Personal Energy Advisor*” based on GLMP as a computational environment using natural language. The framework is based on the mechanization of queries representation via language constraints and the precisiation of the meaning of concepts for answer deduction in a granular environment. The different components will be placed together into an application framework able to implement the *Personal Energy Advisor* as showed on Fig. 4.2.

Two stages are defined, an off-line stage that describes the knowledge domain to interpret questions and generate the answers in natural language, and an on-line process that based on the user questions making use of the knowledge defined is in charge of generating dynamically the required responses requested by users.

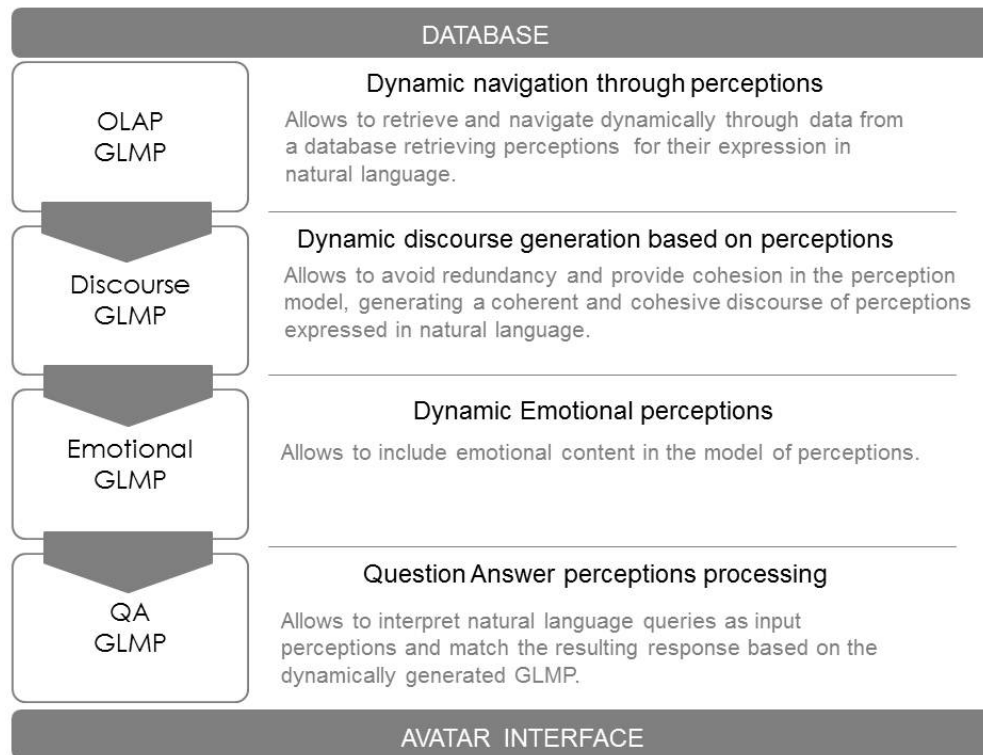


Figure 4.2: Modules based on GLMP for the definition for the Personal Energy Advisor

Note that the process follows a path from input query to a computed response based on a knowledge domain. The modules allow retrieving data from the database in a content structure of perceptions, arranging the perceptions in a structured discourse, including emotional content for a closer relation and integrating the process in a question answer dynamic process.

4.2 Dynamic navigation through perceptions

Data analysis frameworks, as Multidimensional Data models (MD) and On-Line Analytical Processing (OLAP), provides users with flexible ad-hoc analysis of information. They improve the limitation of conventional database models, whose structures are not well suited for queering large amounts of historical data. Usually, the final user of these technologies is a business analyst who interprets the resulting information expressed in tabular numerical reports. Nevertheless, for non-experts users, will be required a personalized natural language explanation of the information more than numeric reports.

Analysis performed by traditional OLAP systems is numeric oriented and natural meaning is missing, e.g., whether a 2.500kWh/year electricity consumption is high or low. Semantics defined by granular perceptions are imprecise but more understandable, as perceptions rather than raw numbers convey the meaning of the data. Traditional OLAP systems are based on numeric data-oriented computing, while perception systems are based on linguistic knowledge-oriented computing.

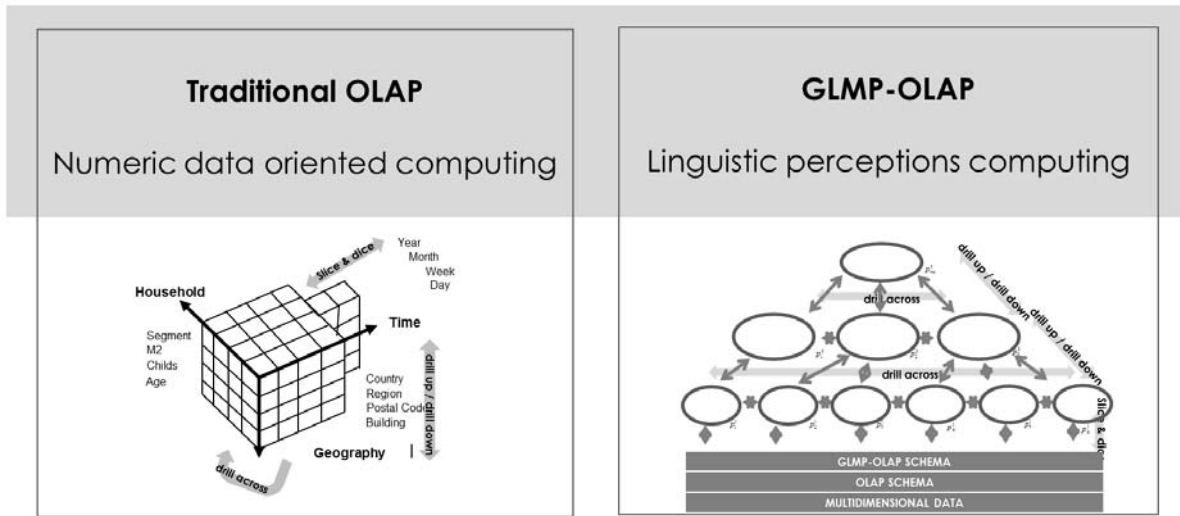


Figure 4.3: From OLAP to GLMP

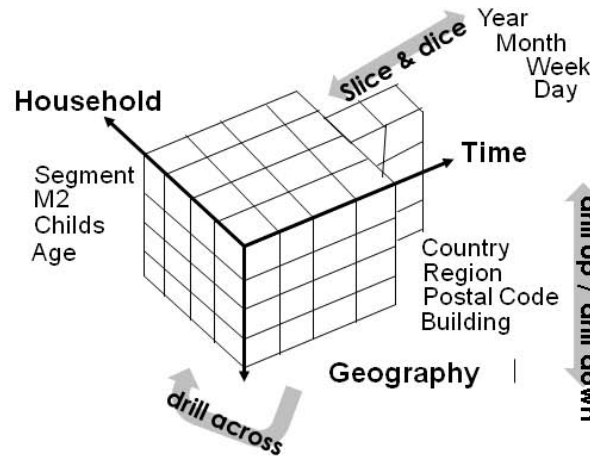
For dynamic navigation through perceptions a GLMP is built over an OLAP schema as showed on Fig. 4.3. In other words, from the point of view of the designer, the phenomenon is the *fact* and related *dimensions* together with the navigation possibilities described by the traditional OLAP schema, e.g., the dimensions and the measures associated with a query are translated to computational perceptions. Note that with this approach traditional OLAP 4.4 schemas are enhanced to deal with perceptions in a GLMP-OLAP schema as showed on Fig.4.5.

In GLMP applied over OLAP, 1PMs are applied to OLAP measures and dimensional attributes to obtain 1CPs. Now, the attribute *DayPeriod* could be a linguistic variable taking as values linguistic labels representing fuzzy sets, e.g., the designer defines a trapezoidal function to assign to each hour in the day the degree of membership to the linguistic label *Midday*. Then, over this set of 1CPs, the designer builds the remainder structure of a GLMP.

4.2.1 GLMP OLAP Schema

A GLMP-OLAP schema, $GLM-MD_q^n$ is defined as a tuple (D, M, F, V) as follows:

- D is a set of (n) dimensions and the dimensional attributes are CPs, e.g., the attribute *DayPeriod* of the dimension *Time* is a fuzzy linguistic variable that takes as values linguistic labels
 $Hour \in \{Morning, Midday, Afternoon, Night\}$.
- M is a set of CPs, e.g., the measure of
 $Consumption \in \{Low, Average, High\}$.
- F is a fuzzy function that maps a set of fuzzy constraints in a set of CPs.
 $F(GLMP-MD_q^n) : dom(D_1) \times \dots \times dom(D_n) \rightarrow V$.



An **OLAP schema** MD_q^n is a logical structure defined as a tuple (D, M, F, V) :

- D** is a finite set of dimensions $D=\{D_1, \dots, D_n\}$ with domains $dom(D) = \{dom(D_1) \dots dom(D_n)\}$. E.g. $D=\{Geography, Time, Date, Household\}$. Domains are composed by a set of dimensional attributes that provide a hierarchical organization of data from the most detail to the most general granularity, e.g. $Time=\{Hour, Day, Week, Month\}$, where attributes takes the values $Hour=\{1, 2, \dots, 24\}$
- M** is a set of measures of the facts $M=\{M_1, \dots, M_m\}$, e.g. $Time=\{Consumption[kWh], Cost[euros]\}$
- F** is the mapping function based on OLAP operations. $F_q(MD_q^n): dom(D_1) \times \dots \times dom(D_n) \rightarrow V$. It maps a set of dimensional attributes to the corresponding measures.
- V** is a set of measures $\{m_1, \dots, m_m\}$ obtained in a specific context $\{dom(D_1) \times \dots \times dom(D_n)\}$.

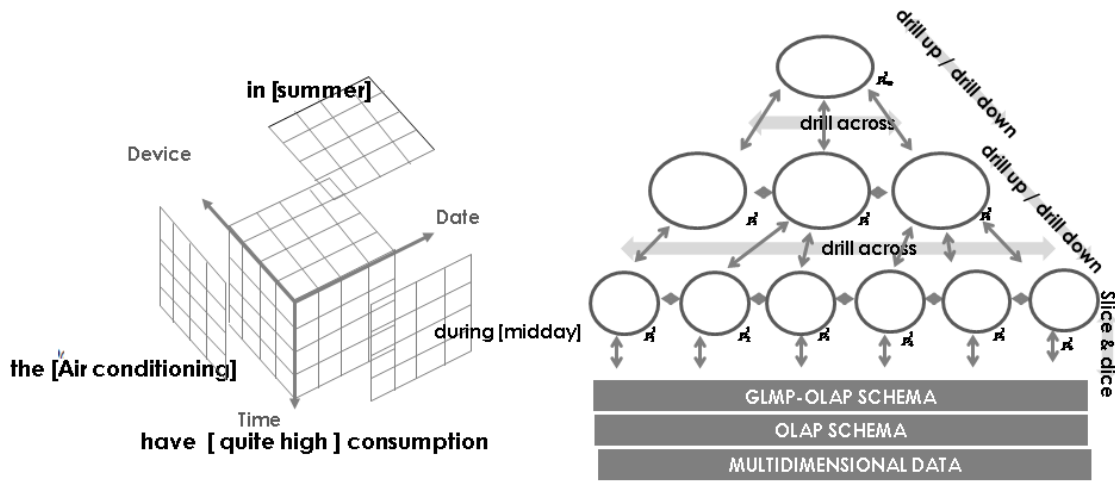
Figure 4.4: OLAP multidimensional schema definition, the data cube.

- V is a vector of CPs (m_1, \dots, m_m) in a specific context $(dom(D_1) \times \dots \times dom(D_n))$. According to Zadeh (136), a granule of information V , e.g., the answer to a user's question, is determined by applying a set of constraints $\{D_1 \times \dots \times D_n\}$.

The schema $GLMP-MD_q^n$ can be seen as a system designed over traditional OLAP meta-data using a new GLMP meta-data layer. In this system:

- The inputs are, namely, the query q , the OLAP meta-data, and the GLMP meta-data (defining the GLMP). Here, the query q is built using an extension of the basic OLAP operations, namely, *drilling*, *rolling* and *slicing/dicing*. They are used to navigate by the hierarchical granular information of fuzzy perceptions of a phenomenon.
- The output is a CP answering the question. It is the output of a 2PM, e.g., a set of NL propositions with associated degrees of validity (A_i, w_i) . Note that, a single specific answer could be given by choosing the proposition with highest degree of validity.
- The process is a traditional OLAP search followed by a CTP constraint propagation process (136).

While navigating through a GLMP-OLAP schema, natural language is returned for the queries instead of numeric reports. For example on Fig 4.6 it is showed graphically



A **GLMP-OLAP** schema, $GLMP-MD_q^n$ is defined as a tuple (D, M, F, V) :

- D** is a finite set of dimensions $D = \{D_1, \dots, D_n\}$ with potentially fuzzy attributes. Day Period is a fuzzy linguistic variable that takes values $\{Morning, Midday, Afternoon, Night\}$
- M** is a set of CPS, e.g., the measure of Efficiency $\in \{Low, Medium, High\}$
- F** is a fuzzy function that maps a set of fuzzy constraints in a set of CPS.
 $F_q(MD_q^n) : dom(D_1) \times \dots \times dom(D_n) \rightarrow V$.
 It maps a set of dimensional perceptions to other perceptions.
- V** is a set of CPS $\{m_1, \dots, m_m\}$ in a specific context $(dom(D_1) \times \dots \times dom(D_n))$.

Figure 4.5: GLMP OLAP Schema definition. The perceptions data cube.

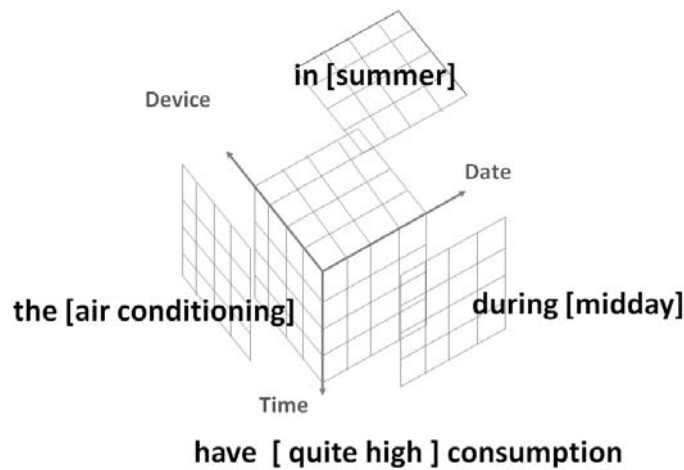


Figure 4.6: GLMP-OLAP data cube that allows to generate the sentence “The air conditioning, in summer, during midday, has quite high consumption”.

an example of a cube obtained as result of a search that produces the output: “The air conditioning, in summer, during midday, has quite high consumption”. Note that here, “midday” and “quite high” are fuzzy linguistic labels.

4.2.2 GLMP OLAP Application

In this section, a practical application is detailed to illustrate the functioning of a GLMP-OLAP scheme. The scope of this example is limited to demonstrate the concepts previously defined.

Here, the electricity consumption data for testing purposes is based on real data retrieved by hourly meter devices, and enriched with information provided by householder surveys to obtain device characteristics and behaviour in use. In this application, OLAP meta-data and GLMP meta-data are used to define a historical energy consumption multidimensional database for granular linguistic access, retrieval and analysis.

According to the definitions above, the designer described the GLMP schema over the OLAP schema. The designer defined 1PMs over the measures of the *fact* and over the attributes of dimensions. Then 2PMs were defined over subordinate perceptions.

The GLMP-MD_qⁿ was defined around the *Fact*: “Consumption of electricity” as a tuple (D, M, F, V) where:

- Dimensions: $D = \{Time, Device, Date, Household\}$.
- Measures $M = \{Consumption, Efficiency\}$ that take values:
 - $Consumption \in \{Low, Average, High\}$.
 - $Efficiency \in \{Highly\ inefficient, Inefficient, Moderately\ efficient, Highly\ efficient\}$.
- Mapping function F based on a combination of OLAP operations and GLMP aggregation functions. $F(MD_q^n) : dom(D_1) \times \dots \times dom(D_n) \rightarrow V$.
- V is a vector of measures $(Consumption, Efficiency)$ obtained in a specific context defined by the values: $(Time \times Device \times Date \times Household)$.

The set of dimensional attributes were defined as follows:

- *Time*: $\{Hour, DayPeriod, CostPeriod\}$
with values:
 - $Hour \in [0, 23] \in N$ (crisp).
 - $DayPeriod \in \{Morning, Midday, Afternoon, Night\}$
(fuzzy).
 - $CostPeriod \in \{Charged, LowCost\}$
(Fuzzy). Households pay less for their energy consumption during low cost period hours, so the higher the consumption in this period the better the behavior maintaining a rational global consumption.
- *Device* $\{Device, YearsOld, Quality\}$
with values:

- $Device \in \{AirConditioning, WashingMachine, \dots\}$ (crisp).
 - $YearsOld \in \{QuiteOld, AverageAge, QuiteNew\}$ (fuzzy).
 - $Efficiency \in \{High, Medium, Low\}$ (fuzzy). Obtained from the device technical specifications.
 - $Quality \in \{Low, Medium, High\}$ (fuzzy). Obtained using a set of fuzzy rules with antecedents, namely, age and device efficiency.
- Date $\{Year, Month, Week, Day\}$ with values:
 - $Year \in N$ (crisp).
 - $Month \in \{Jan, \dots, Dec\}$ (crisp).
 - $Week \in [1, 52] \in N$ (crisp).
 - $Day \in N$ (crisp).
 - Household $\{UserId, Area, Size, Type\}$ with values:
 - $UserId \in N$ (crisp). Household identifier.
 - $Area \in \{Small, Medium, Large\}$ (fuzzy). The area of either the flat or apartment.
 - $Size \in \{Low, Medium, High\}$ (fuzzy). The number of members in the family.
 - $Type \in \{Single, Couple, Family1, Family2\}$ (fuzzy). These categories are the result of an ongoing work of clustering aimed to identify different types of consumers. Note that the meaning of linguistic propositions, e.g., “You have low consumption”, must be suited to each type of user receiving the information depending on the $Type$ of household.

Figure 4.7 shows a diagram of an instantiation of the GLMP-OLAP schema. It is worth noting that the designer must suit linguistic labels and membership functions to each situation type. Also note that crisp values are particular cases of fuzzy values.

The following are three typical examples ($Q1$, $Q2$ and $Q3$) of OLAP style queries made on this GLMP-OLAP scheme.

The user started by logging in the system and selecting a specific $UserId$ in the dimension $Household$. This is a Slice/Dice OLAP operation that make the analysis performed over this specific $household$. ($Q1$) Then, he performed a new query slicing the cube to obtain the devices $YearsOld$ in the dimension $Device$. ($Q2$) Afterwards, by selecting $Year=2010$ in the dimension $Date$ (slice/dice) and then drilling-across and rolling-up, he navigated by the GLMP to obtain the measure *Low cost consumption* in 2010 when $Time = Low\ cost\ period$. ($Q3$) Finally, he rolled-up to obtain the $Efficiency$. $Efficiency$ is a 2CP that is explained using a set of subordinate perceptions, namely, *low*

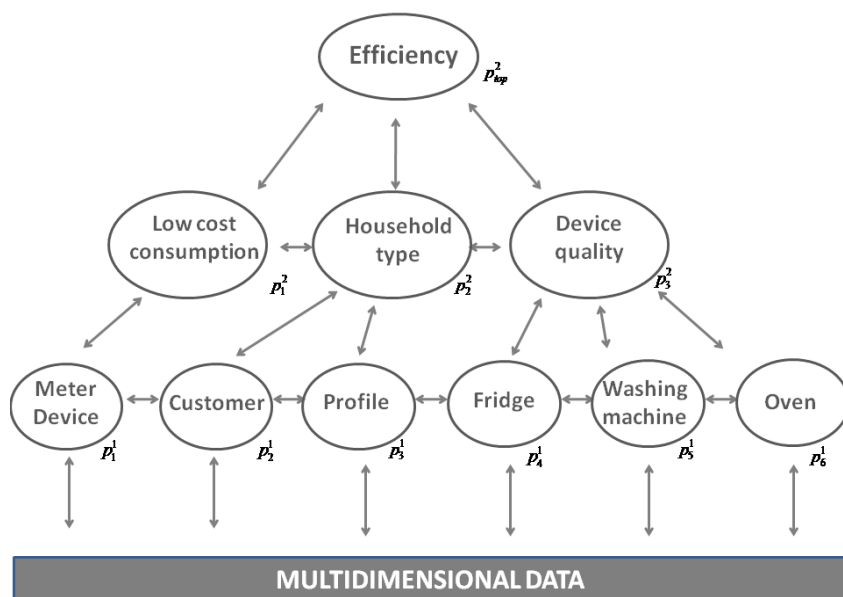


Figure 4.7: GLMP-OLAP scheme of the application prototype.

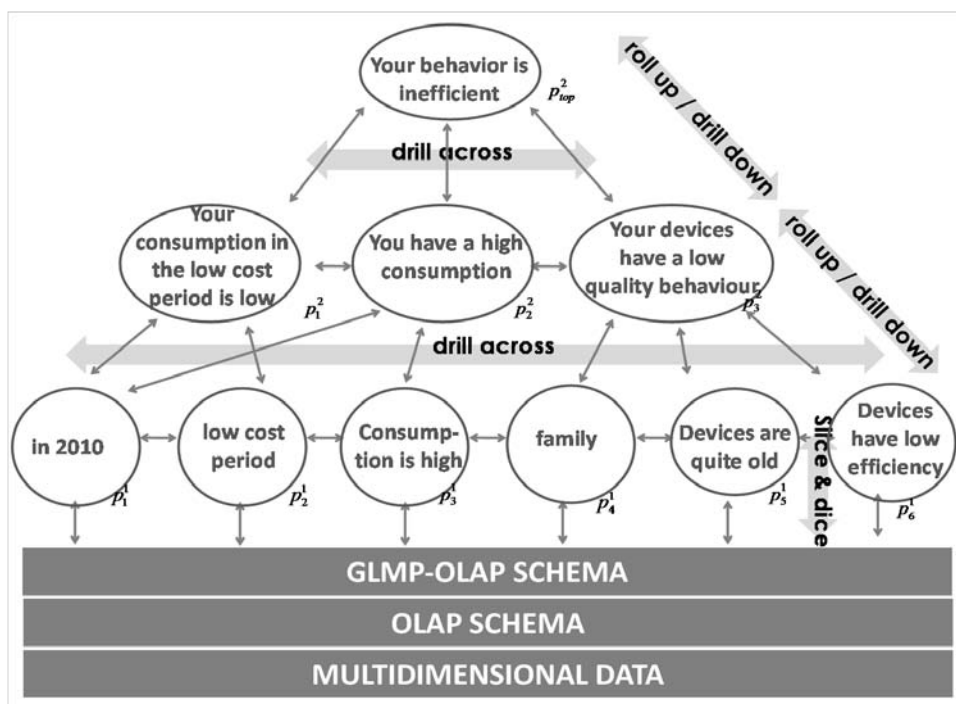


Figure 4.8: Graphical representation of the answers to queries in the application example.

cost consumption, Household type and Devices quality. Fig. 4.8 shows the whole structure of answers obtained in this application example.

Q1. Device YearsOld

For querying “Give me the devices age” a traditional OLAP slicing based on *Device* dimension is computed. The result is a table/view of the cube with the numeric value of the average age of devices in the household. Expressed as a traditional OLAP query function, it is defined by

$$F(\text{MD}_q^4)(\text{Household.UserID}, \text{Device.YearsOld}) = 9.$$

A 1CP is used to define the fuzzy dimensional attribute *Device.YearsOld*.

In agreement with the definition of 1CP, *Device.YearsOld* is a tuple (u, y, g, T) where:

- u is the input variable value defined by the traditional OLAP query result, e.g., 9 years old. Note that, it is the value stored in the database when the query is performed.
- y is the output CP with values $y = (A, W)$, e.g.,
 $y = (\text{“Your devices are quite old”}, 0.7),$
 $(\text{“Your devices have an average age”}, 0.3),$
 $(\text{“Your devices are quite new”}, 0.0).$
- g is built with a set of three trapezoidal membership functions.
- T is the template “Your devices (are quite old | have average age | are quite new)”.

Expressed as a GLMP-OLAP query function, it is defined and instantiated over the previous crisp OLAP result, e.g., $F(\text{MD}_{Q1}^4)(\text{Household.UserId}, \text{Device.YearsOld}) = \text{“Your devices are quite old”}$.

The resulting approach for Q1 is showed on Fig. 4.9.

Q2. Low Cost Consumption in 2010

For querying “Give me the consumption in 2010”, traditional OLAP aggregates consumption based on *Date* dimension context for *Year=2010*. Expressed as a traditional OLAP query function, it is defined by $F(\text{MD}_q^4)(\text{Consumption}, \text{Year} = 2010) = 3.534kWh$.

For querying “Give me the consumption in the low cost period”, GLMP-OLAP aggregates consumption based on the *Time* dimension fuzzy perception for the *Cost period*.

In agreement with the definition of 2PM, *Low cost consumption* is a tuple (U, y, g, T) where:

- U are the set of input variables, namely, the 1CP of the measure of *Consumption*, the crisp perception of the dimension *Date* (e.g., *Year = 2010*), and the CP of the dimensional attribute *CostPeriod* (e.g., *LowCost*).
- y is the output CP with values $y = (A, W)$, e.g., $y = \{(\text{“You have a high consumption in the low cost period”}, 0.0), (\text{“You have an average consumption in the low cost period”}, 0.35), (\text{“You have a low consumption in the low cost period”}, 0.65)\}$.
- g is calculated using the concept of fuzzy cardinality of *Consumption* in the membership function *LowCost*, e.g., counting the consumptions during the year 2010 belonging to the fuzzy set *LowCost*.

Practical application.

Q1- "Give me the devices age"

QUERY	Query: "Give me the devices age"
OLAP	OLAP: Slicing based on device dimension, the result is a table/view of the cube with a numeric value of the average devices age in the household. $F_q(MD_{Q1}^4)(Device\ Years\ Old)=9\ years$
GLMP OLAP	<p>GLMP-OLAP: A 1-PMP define the fuzzy dimensional attribute device.YearOld. It is a tuple (u, y, g, T) where:</p> <ul style="list-style-type: none"> u is the input variable value, the traditional OLAP query result (9 years old) y is the output perception with values $y=(A,W)$, e.g. $y= \{ ("Your\ devices\ are\ quite\ old", 0.7),$ $("Your\ devices\ have\ an\ average\ age", 0.3),$ $("Your\ devices\ have\ are\ quite\ new", 0.0)\}$. g is built with a set of three trapezoidal membership function T is the template "Your devices (are quite old have average age are quite new)" <p>Expressed as a GLMP-OLAP query function $F_q(GLMP-MD_{Q1}^4)(Device\ Years\ old)='Your\ devices\ are\ quite\ old'$.</p>

Figure 4.9: Q1. Device YearsOld

- T is a simple template: "You have a (high | average | low) consumption in the low cost period".

Expressed as a GLMP-OLAP query function, it is defined by $F(MD_{Q2}^4)(Consumption, UserId, Year = 2010, Time = LowCost) = "You\ have\ a\ high\ consumption\ in\ the\ low\ cost\ period"$.

The resulting approach for Q2 is showed on Fig. 4.10.

Q3. Efficiency

In this last example, the query was: "Give me the efficiency in 2010". Here, the information retrieved by the GLMP-OLAP system as second-order perceptions, namely, the *Low cost consumption*, the *Devices quality*, and the *Household consumption* were used to define the *Efficiency* second-order perception.

- *Low cost consumption* is calculated as described above.
- *Devices quality* is a general perception of quality of the set of devices in the household. It is obtained by aggregating the perceptions of *Device.Efficiency* and *Device.YearsOld*. It takes values $\{Low, Medium, High\}$.
- *Household consumption* is the typical consumption in the type of household. It is obtained by aggregating the measure of average *Consumption* and the dimensional attribute *Household.Type*.

QUERY	Query: "Give me the consumption in the low cost period in 2010"
OLAP	OLAP: Aggregate consumption for 2010 and the low cost period, the result is a table/view of the cube with a numeric value of the consumption. $F_q(\text{MD}_{Q_2^4})(\text{Consumption ; Year=2010; Period=LowCostPeriod})=3.534\text{kWh}$
GLMP OLAP	GLMP-OLAP: A 2-PMP define low cost consumption. It is a tuple (U, y, g, T) where: U is aset of CPPs input variables, $U=\{\text{consumption, year=2010, Period}\}$ y is the output perception with values $y=(A, W)$, e.g. $y= \{ ("You have a high consumption in the low cost period", 0.0),$ $ ("You have average consumption in the low cost period", 0.35),$ $ ("You have a low consumption in the low cost period", 0.65)\}$. g is built using the concept of fuzzy cardinality T is the template $"You have a (high average low)consumption in the low cost period"$ Expressed as a GLMP-OLAP query function $F_q(\text{GLMP-MD}_{Q_2^4})(\text{Consumption; Year = 2010; Period =LowCost})=$ "You have a high consumption in the low cost period".

Figure 4.10: Q2. Low Cost Consumption in 2010

Efficiency is a tuple (U, y, g, T) where:

- U is the set of input CPs $U = \{u_1, u_2, u_3\}$, $U = \{ \text{low cost consumption, Household consumption, Devices quality} \}$.
- y is the output CP with values $y = (A, W)$, e.g., $y = \{ ("Highly inefficient", 0.0), ("Inefficient", 0.68), ("Moderately efficient", 0.32), ("Highly efficient", 0.0) \}$.
- g : An aggregation function $W = g(u_1, u_2, u_3)$ was implemented using a set of Mamdani-type fuzzy rules as follows:

IF (u_1 is *High*) AND (u_2 is *Family2*) AND (u_3 is *High*) THEN *Highly efficient*
 IF (u_1 is *Medium*) AND (u_2 is *Family2*) AND (u_3 is *High*) THEN *Moderately efficient*
 IF (u_1 is *Low*) AND (u_2 is *Family2*) AND (u_3 is *High*) THEN *Inefficient*
 ...
 IF (u_1 is *Low*) AND (u_2 is *Single*) AND (u_3 is *Low*) THEN *highly inefficient*

Here, the perception of efficiency is defined based on the study of available data about consumption of families and implemented as expert knowledge.

- T is a simple template: "Your behavior is (highly inefficient | inefficient | moderately efficient | highly efficient)"

Expressed as a GLMP-OLAP query function, it is defined by $F(\text{MD}_{Q_3^4})(\text{Efficiency, UserId, Year} = 2010, \text{Time} = \text{LowCost}) = \text{"Your behavior is inefficient"}$.

QUERY	Query: "Give me the efficiency".
GLMP OLAP	<p>GLMP-OLAP: A 2-PMP define efficiency. By drilling-across we get the result. It is a tuple (U, y, g, T) where:</p> <p>U is the set of CPPs input variables, $U = \{ \text{Low cost consumption, Household type, Devices quality} \}$</p> <p>y is the output perception with values $y = (A, W)$, e.g. $y = \{ ("highly inefficient", 0.0),$ $("inefficient", 0.68),$ $("moderately efficient", 0.32),$ $("highly efficient", 0.0) \}$</p> <p>g is an aggregation function using a set on mamdani-type fuzzy rules <i>IF (u₁ is High) AND (u₂ is Family₂) AND (u₃ is High) THEN Highly efficient</i> <i>IF (u₁ is Medium) AND (u₂ is Family₂) AND (u₃ is High) THEN Moderately efficient</i> <i>IF (u₁ is Low) AND (u₂ is Family₂) AND (u₃ is High) THEN Inefficient</i> $\vdots \vdots$ <i>IF (u₁ is Low) AND (u₂ is Single) AND (u₃ is Low) THEN Highly inefficient</i></p> <p>T is the template <i>"Your behavior is (highly inefficient inefficient moderately efficient highly efficient)"</i></p> <p>Expressed as a GLMP-OLAP query function $F_q(\text{GLMP-MD}_{\alpha_3^4})(\text{Efficiency}; \text{Year} = 2010) = \text{"Your behavior is inefficient"}$.</p>

Figure 4.11: Q3. Efficiency

Using this GLMP, we can generate a more complete explicative sentence, e.g., *"Your behavior is inefficient, due to the high consumption, the quite old devices and the low consumption at the low charge period"*.

The resulting approach for Q3 is showed on Fig. 4.11.

4.3 Dynamic discourse generation based on perceptions

An assumption that can be found in most NLG techniques is that the non-linguistic input representation comes from knowledge bases with a deep semantic definition. However, in most application domains where natural language communication is required, such knowledge bases do not exist. The GLMP contains a deep semantic structure linked to computational models and linguistic expressions due to the perception mapping definition.

In this section the process to reorganize the structure of an initial GLMP to deploy a coherent and cohesive linguistic discourse is explored. The reorganization identify relevant perceptions, avoids redundancy and links individual granular sentences in a set of connected ones.

For text generation from GLMP, the CP and PM definitions are extended with the concept of relevancy. This concept allows to identify the relevant information in dynamically generated GLMP from OLAP.

The process is defined by the steps of (a) content selection to generate an initial

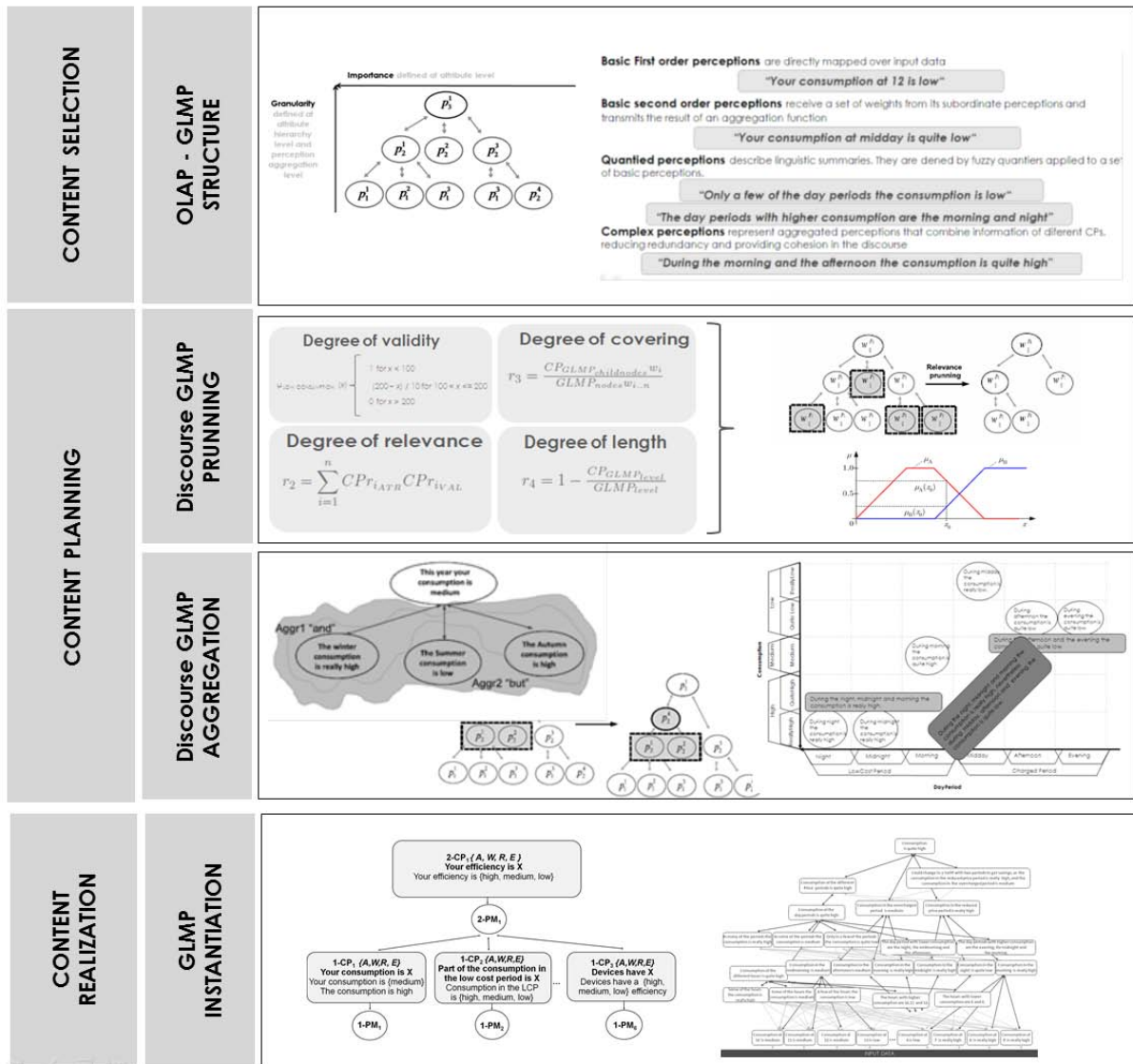


Figure 4.12: Process followed for language generation through perceptions in the GLMP.

GLMP from an input domain, (b) content planning for the reorganization of the resulting GLMP in order to define the discourse structure and (c) content realization to instantiate the GLMP. The overall process followed is showed on Fig. 4.12 and described in more detailed in this section. The reorganization is performed by pruning non relevant or redundant perceptions and by aggregating and linking related computational perceptions.

The rest of this section is organized as follows:

- A short interpretation of the Gricean Maxims for the model proposed.
- A GLMP extension to include the computation of relevance.
- The process for discourse generation following the steps of content selection, content planning and content realization.

- The application of the model proposed.

4.3.1 Gricean Maxims

The criteria for performing the task of text generation follows the Gricean Maxims (56) (55) described in previous chapter. In our model, the way of applying these maxims is proposed as follows:

- Maxim of Quality

It requires to analyze the candidate report from the viewpoint of its truth in the context current conditions. After instantiating the GLMP with input data, for each CP, each possible sentence a_i has associated a degree of validity w_i .

- Maxim of Quantity

It requires evaluate if the quantity of statements used in the report are the right for expressing the relevant ideas or if there is redundancy in the text.

- Maxim of Relation Relevance

The relevance is very subjective, since it depends of the background of final user. The designer will assign to each perception sentence a degree of relevance in function of the application context.

- Maxim of Manner

The candidate report must be evaluated considering if it uses the adequate vocabulary, if the order of the ideas is appropriate or if the used expressions are the right ones. The designer must choose linguistic expressions and emotional representation.

Here are used a set of heuristics derived from an initial GLMP to decide what types of simple perceptions constitute a relevant information, as well as the ways in which simple sentences are combined into most complex ones. The aim for this reorganization is to avoid redundancy and provide cohesion. It is done by pruning non relevant perceptions instances and by aggregating and linking related perceptions instances.

4.3.2 Computational Perceptions with relevance

The concept of relevancy is attached to each perception to identify the relevant information in dynamically generated GLMP.

Here a *CP* is the computational model of a unit of information acquired by the designer about the phenomenon to be modelled. In general, *CPs* correspond with specific parts of the phenomenon at certain degrees of granularity. A *CP* is a tuple (A, W, R) described as follows:

A is a vector (a_1, a_2, \dots, a_n) of linguistic expressions (sentences in NL) that represents the linguistic domain of the CP. Each a_i describes the value of the CP in

each situation with specific degree of granularity. These sentences can be either simple, e.g., $a_i = \text{“The consumption in the morning is quite high”}$ or more complex, e.g., $a_i = \text{“Some days you could save a lot of money”}$. During the preliminary off-line stage, these values are assigned by the designer extracting the most suitable sentences from a linguistic corpus of the application domain.

W is a vector (w_1, w_2, \dots, w_n) of validity degrees $w_i \in [0, 1]$ assigned to each a_i in the specific context. The value of validity depends on the application, e.g., it is a function of the suitability of each sentence to describe specific input data. During the on-line stage, these validity values are assigned (and updated) in function of the phenomenon current state. Typically, A is a strong fuzzy partition of the domain of existence of CP and therefore $\sum w_i = 1$.

R is a vector (r_1, r_2, \dots, r_n) of relevancy degrees $r_i \in [0, 1]$ assigned to each a_i in the specific context. The concept of relevancy depends on the application. It is used as complementary data to select the most suitable sentences. During the off-line stage, these values are assigned by the designer of the computational system in function of the specific requirements in the application context.

For example, a computational perception that models the consumption during the morning $CP_1 = (A_1, W_1, R_1)$ could be instantiated as:

$$a_1 = \text{“The consumption in the morning is high”}, w_1 = 0.7, r_1 = 0.8$$

$$a_2 = \text{“The consumption in the morning is medium”}, w_2 = 0.3, r_2 = 0.3$$

$$a_3 = \text{“The consumption in the morning is low”}, w_3 = 0, r_3 = 0.8$$

Here, the first sentence has currently the highest validity = 0.7 and, according with the designer criteria, it has high degree of relevance = 0.8 for the addressed user. Therefore is a good candidate sentence to be included in the final report.

4.3.3 Perception Mapping with relevance

Perception mapping (PM) is called to a function that performs the constraint propagation process, e.g., the function that converts the input perceptions in output perception (136).

PM s are used to create and aggregate CP s. A PM is a tuple (U, y, g, T) where:

U is a vector of input CP s,

$$U = (u_1, u_2, \dots, u_n), \text{ with}$$

$$u_i = (A_{u_i}, W_{u_i}, R_{u_i}). \text{ They are called first order perception mappings.}$$

y is the output CP , $y = (A_y, W_y, R_y)$.

$g()$ is an aggregation function employed to calculate the degrees of validity $W_y = (w_1, w_2, \dots, w_{n_y})$.

It is a fuzzy aggregation of the input vectors $W_y = g(W_{u_1}, W_{u_2}, \dots, W_{u_n})$, where W_{u_i} are the degrees of validity of U . In Fuzzy Logic many different types of aggregation functions have been developed. For example g could be implemented using a set of fuzzy rules. In the case of $1PM$ s, g is built using a set of membership functions as follows:

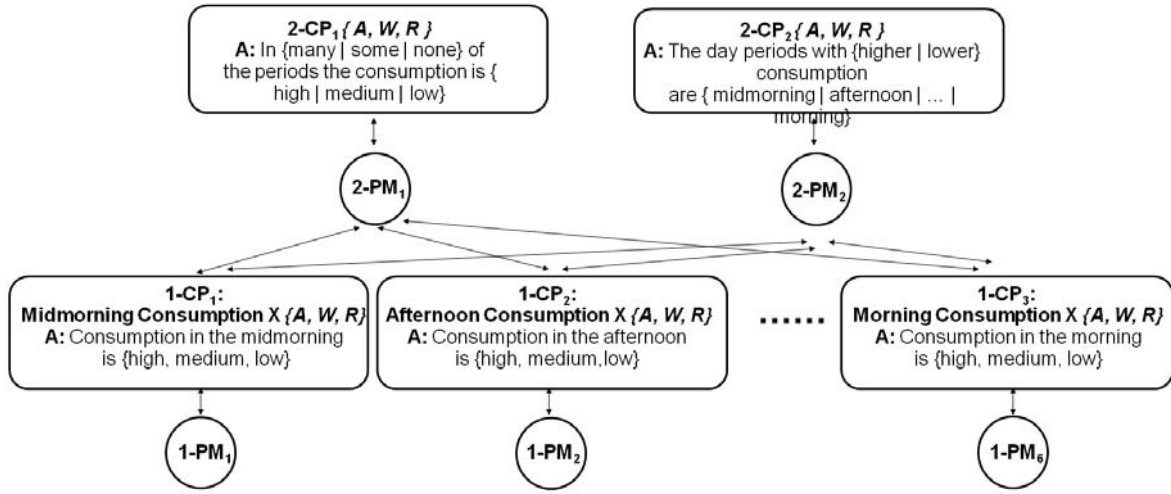


Figure 4.13: Example of Granular Linguistic Model of a Phenomenon.

$$W_y = (\mu_{a_1}(z), \mu_{a_2}(z), \dots, \mu_{a_{n_y}}(z)) = (w_1, w_2, \dots, w_{n_y})$$

where W_y is the vector of degrees of validity assigned to each a_y , and z is the input data.

T is a text generation algorithm that allows generating the sentences in A_y . In simple cases, T is a linguistic template, e.g., “*The efficiency in this building is {high | medium | low}*”.

It is worth remarking that A_y and R_y are defined by the designer during the off-line stage.

4.3.4 Content Selection

Content selection is the task of choosing the right information to communicate. Content selection for CP extraction involves the computation and deduction of information from an input domain and the assessment of relevance of CPs to define initial GLMP corpora as showed on Fig. 4.14. After content selection the GLMP structure is deployed in an initial corpus using a hierarchical layout. It provides a first approach to what information has to be deployed as chunks of coordinated content in the ongoing discourse.

During the selection process information is computed and spread upwards. Computing constraints for new CP go from lower level perceptions till reaching a top order CP. With this purpose an aggregation function spread subordinate perceptions weighs through aggregation along a hierarchy function W_y , or by fuzzy (IF-THEN) rules. In the GLMP are distinguished the use of different kind of perceptions: basic perceptions, quantified perceptions and complex CPs.

- Basic Perceptions

Basic perceptions provide information of a single feature, e.g. “You have a high consumption in 2008”. Basic perceptions can be 1CP and 2CP. First order perceptions are directly mapped over input data. To extend perceptions expressiveness are

developed granulation mechanisms which group individuals into larger meaningful units that can express accumulative properties of the resulting collections.

- Basic second order perceptions

Basic second order perceptions receive a set of weights from its subordinate perceptions and transmit the result of an aggregation function. The aggregation function is implemented through hierarchy by average mean, or by using a set of fuzzy rules. This set of fuzzy rules allows calculating the weight associated with sentences having as antecedents the weights of the subordinate perceptions. Due to this structure can describe the meaning of sentences, e.g., “Your behavior is inefficient”, could be explained by combining other perceptions as “Consumption is high”, “Devices are quite old”, “Consumption in the low cost period is low”.

- Quantified perceptions

Quantified perceptions describe linguistic summaries. They are defined by adding fuzzy quantifiers to a set of basic perceptions. In this work are used two quantified perception structures:

- “*Q people in the data base are S*”. Where S is the summarizer and Q is the quantity in agreement, e.g. “*Most of the devices have a low efficiency*”.
- “*Q S are X*”. Where Q is the entity summarized, S is the max/min summarizer and X are the detail perceptions, e.g. “*The devices with the higher consumption are the fridge, the bulbs and the TV*”.

Initial GLMP layout

The first GLMP layout is instantiated in a tree structure that provides integration between discourse level and clause level descriptions. Only those CPs with enough degree of truth are defined in the GLMP. The layout is defined by two different dimensional axes as showed in Fig. 4.14. Both combined generate a granular ordered structure. The vertical hierarchy is defined using the attribute hierarchy levels e.g. *Date = (Year, Month, Day)* and the perception level, e.g., *summarizer, max/min, second order, first order*. Vertical axis defines the degree of covering. Importance of CPs establishes an order in the discourse from left to right defining the horizontal axis layout. Importance is computed using the degree of relevance.

The hierarchical structure in the GLMP allows inferring formal relations between sentences providing cohesion as it is spread from bottom perceptions till top-level perceptions.

Sentence relevance is determined based on the layout of the CPs in the tree. Behind this approach is that information that is important is usually placed at a leading position within a text.

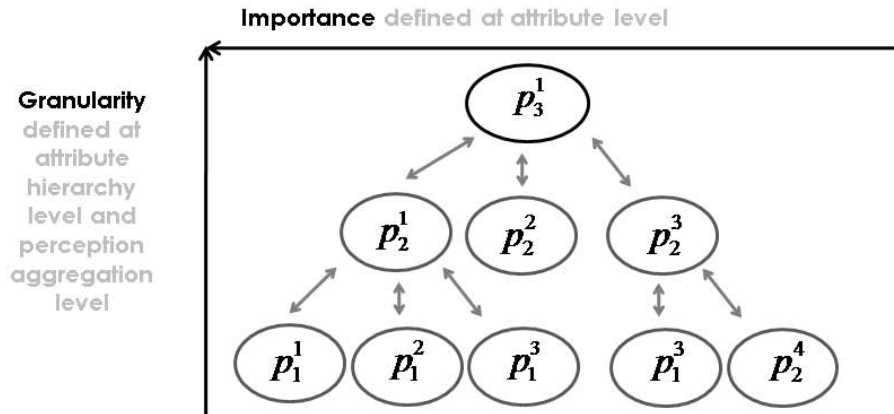


Figure 4.14: GLMP Layout defined by dimensional constraints on perceptions.

4.3.5 Content Planning

In order to generate linguistic descriptions, the GLMP must be able to determine what information to communicate and how to organize this information to accomplish a communicative goal. The aim for organization is to avoid redundancy and provide cohesion.

With this purpose the reorganization of the GLMP is performed in the planning stage. A set of heuristics are developed derived from the initial corpus which decide in what order perceptions need to be listed, what types of simple perceptions constitute a relevant linguistic description, as well as the ways in which simple sentences are combined into most complex ones. It is done by pruning non relevant perceptions and by aggregating and linking related perceptions. For the text generation purpose are needed complex perceptions that must be able to determine which connectives best links its embedded sentences e.g. but, because, etc. Information is granulated extracting data from input perceptions with different levels of degree of aggregation as showed on Fig. 4.15.

Pruning

Pruning defines, in the instantiation, what information to communicate in order to provide relevant linguistic descriptions. While pruning, a suitable weight for each sentence is computed for afterwards excluding irrelevant ones.

The problem of selecting the most suitable sentences is faced by calculating a weight of suitability S_i for each expression with three components, namely s_1 degree of validity, s_2 degree of relevancy and s_3 degree of covering.

The degree of validity, $s_1 = w_i$ determines the degree to which a sentence is valid as a basic criteria. It is performed in a first step during content planning, maintaining in the GLMP only those sentences with the highest degree of validity, e.g., the degree of validity specified by the membership function is higher than a threshold value.

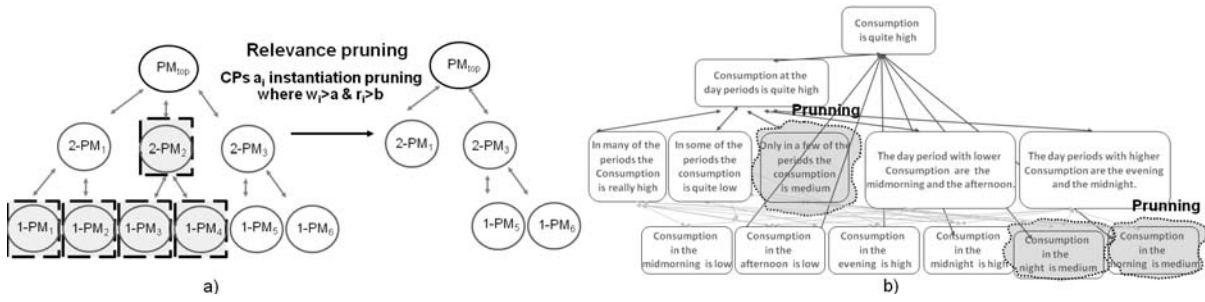


Figure 4.16: Suitability Pruning Layout and Example of Sentence Instantiation.

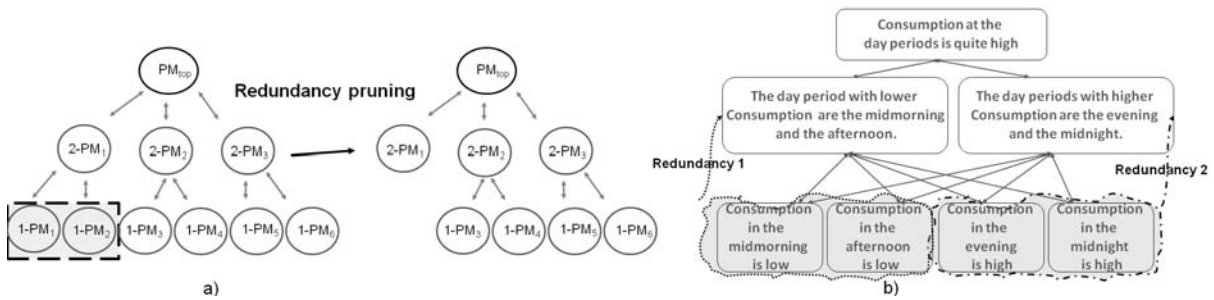


Figure 4.17: Redundancy Pruning Layout and Example of Sentence Instantiation.

highest validity are showed, e.g., “Only in a few of the periods the consumption is medium” is pruned because of its low suitability.

A second pruning is performed following a top-down path, when parent perceptions are explained by the children ones, and therefore additional meaning is not deployed but redundancy.

Conceptually, in Fig. 4.17.a is showed a generic GLMP structure and its redundancy pruning. Independently on Fig. 4.17.b it is showed an example of a GLMP Linguistic expression redundancy pruning, for simplicity only the sentences for each CP with the highest validity are showed, e.g., “Consumption in the morning is low” is pruned because it is already expressed in its parent perception.

Aggregating and linking the granular linguistic model

Aggregation provides, in the instantiation, cohesion to linguistic expressions. It merges linguistic expressions of different CPs if they can be expressed in a complex sentence. While aggregating, relations between perceptions sentences are computed by similarity, and if applies, clustered together using a connective. The choice of the connective depends on the relations that hold between perceptions and their semantics, e.g., *additive* “and” or *adversative* “but” conjunctions.

Complex perceptions represent linked perceptions that combine information of different types of CP with the aim of generating more complex linguistic expressions reducing redundancy and providing cohesion in the discourse, e.g. “The periods with higher consumption are the morning and the afternoon, both are in the overcharged period”.

The clustering of perceptions is performed in two steps, first looking for participants

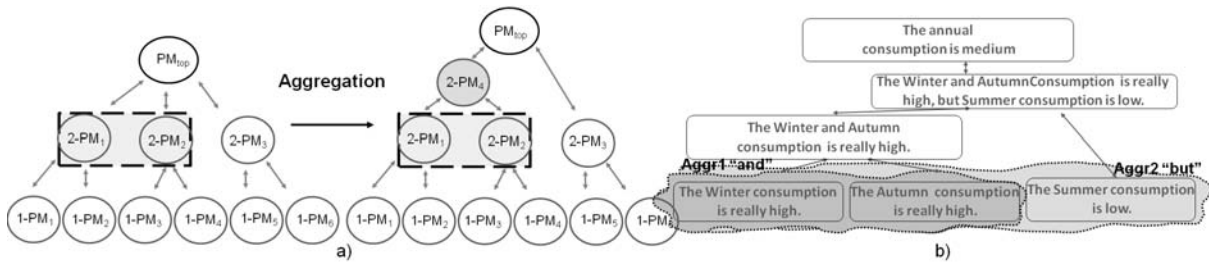


Figure 4.18: GLMP Aggregation Layout and example of Sentence Instantiation.

and later looking for shared structures using the distance function over the perceptions for an specific tree level. The first cluster is selected using the first perception of the tree level. In each iteration the algorithm decides whether a new perception should be inserted in an existing cluster or should originate a new one. Simple conjunction denoting opposite behavior are performed in a last step based on dissimilarity of the share structure.

To control the expressiveness it is used the degree of length of the aggregated perceptions. For each individual complex perception is calculated the cardinality of participants $\text{card}(P)$ (density) and the cardinality of the shared structure $\text{card}(SS)$ (dimensionality).

Second order perceptions, can also be merged based on their semantic relationship in order to represent not only a meaning but the explanation of this behaviour, e.g. “Your efficiency is low, due to the high overall consumption, the low consumption in the low cost period and the households with similar characteristics”, “Changing your fridge with efficiency C to one with efficiency A could generate you savings around 20 Euros per year.”.

Additive aggregation is computed by the similarity of perceptions sentences and applied as shared participants, e.g. “*The Winter and Autumn consumption is really high.*”. Adversative aggregation is computed by dissimilarity of perceptions sentences and applied as shared structures, e.g. “*The Winter consumption is really high but the Summer consumption is low.*”. Note that causal aggregation is already defined by IF-THEN rules in second order perceptions, e.g., “*The efficiency is low due to the high consumption and low efficiency devices*”.

Conceptually, in Fig. 4.18.a is showed a generic GLMP structure and the merging. Independently in Fig. 4.18.b it is showed an example of a GLMP Linguistic expression merging, for simplicity only the sentences for each CP with the highest validity are showed.

The similarity measure (SM) between sentences is calculated for simplicity over trapezoidal shaped fuzzy sets. A trapezoidal shaped fuzzy set can be characterized by a vector of four values. Over the similarity measure are defined the fuzzy sets that defines the connectives, e.g. connectives “and” and “but”.

Aggregations are computed in order, first additive and later adversative at each GLMP

layer. The algorithm decides for each CP, if a new parent CP should be created aggregating children or should remain. Note that parent perceptions are completely explain by children ones, that because of redundancy are pruned.

CPs are related to other CPs performing cohesion. Directly related CP are treated in aggregation, but in some situations, there is just an allusion from one CP to another. A reference relation is used to avoid repetition, using a referring expression rather than the nominal. While producing a referring expression, the perceptions are transformed in order to link the expression rather than describe the object. Reference relations involve the usage of pronouns to refer to an entity mentioned in the preceding text. Distance between the sentence containing the reference and the referred object are computed into a paragraph unit to reduce distance and provide understandable text.

References are used for the introduction of an object e.g. “The fridge had a high consumption in 2011. It is quite old”, or by a subsequent reference that refers to an entity already introduced in the discourse e.g. “In 2011 your consumption was high. This year you also had low efficiency behaviour.”. CPs are substituted by definite or indefinite determiners or pronouns. Also context references required to be translated by referring expressions, e.g. “Today consumption was high” instead of “On the 4th of January the consumption was high”.

4.3.6 Text Realization

For each vector of linguistic expressions in a CP, individual expressions are selected or not in an output text based on its validity and relevance. Only those linguistic expressions with the highest degree of validity will remain in the resulting text, e.g., the degree of validity provided by the membership function and relevance defined by the relevancy weight is higher than a threshold value.

Text realization is based on each perception template composed dynamically based on a fact and the context of it.

4.3.7 Generation Process Application

In previous section was proposed the navigation through the information stored in a GLMP schema. Using a subset of the previous GLMP schema, it is defined the historical energy consumption database. The dimensions associated with a query over the consumption measure are translated to computational perceptions. The definition of fuzzy Consumption measure has values

$$Consumption = \{Really\ High, Quite\ high, Medium, Quite\ Low, Really\ Low\}.$$

The dimensional fuzzy attributes for the Time dimension are defined as:

$$\begin{aligned} Time &= \{Price\ Period, Day\ Period, Hour\} \text{ with values:} \\ - PricePeriod &= \{ChargedPeriod, ReducedPeriod\} \end{aligned}$$

- *Day Period* = {*Morning*, ..., *Night*}
- *Hour* = [1, 24] ∈ *N*

A *Summarizer* = {*Only a few*, *Some*, *A lot*, *Most of*} and a maximizer and minimizer quantifiers *Max/Min* = {*the highest*, *the lowest*} are defined and applied to fuzzy attributes. A second order perception based on if-then rules is generated for tariff advice to get savings, based on consumption in the overcharged and reduced price period, e.g. “*IF Consumption in ChargedPeriod is low AND Consumption in the ReducedPeriod is high THEN TwoPeriods Tariff*”.

The communicative objective is to identify relevant information as response to the query “*How is the consumption this year*”. Input data is retrieved from a query on *Time* dimension. The result is a table with the numeric value of the average consumption for each hour as it is the lower time hierarchy attribute, e.g. *as tuples (Hour-Consumption in wh)* “*H22 - 1.320, H23 - 1.796, H24 - 586,1 - 217, H2 - 131,3 - 97, H4 - 45, H5 - 65, H6 - 161, H7 - 129, H8 - 945, H9 - 1.243, H10 - 116, H11 - 145, H12 - 132, H13 - 87, H14 - 161, H15 - 129, H16 - 171, H17 - 131, H18 - 253, H19 - 275, H20 - 876, H21 - 934*”.

During GLMP definition, first order perceptions are computed from these input data for each lower level attribute values, e.g. “*Your consumption at 23 is quite high*”. Second order perceptions are spread upwards for each attribute value along the *Time* hierarchy till defining the overall GLMP structure.

The hierarchical layout is defined using the dimensional axis to generate the granular ordered structure, using the *Time* hierarchy. The resulting GLMP is showed on Fig. 4.19.

For sentences selection and pruning, it is performed the computation of suitability S_i , namely (1) the degree of validity, (2) the degree of relevancy and (3) the degree of covering. Here is followed the simple criteria of using

$$S_i = \frac{1}{n} \sum_{i=1}^n s_i$$

Pruning represents the process of identification of relevant information among all the CPs instantiated in the GLMP. As defined in the model, perceptions with a suitability degree lower than 0,65 are pruned. Results are displayed on Fig. 4.20.

After pruning, remaining CPs are aggregated if their meaning can be expressed in an individual complex perception. The choice of connectives is based on relations that holds between perceptions and their semantics. First step is looking for similar perceptions, expressed by the “and” connective. Second aggregation is performed looking for opposed behaviour expressed by a “but” connective.

The output text generated by the instantiation of the reorganized GLMP is as follows:

“Your consumption is quite high. You could change to a tariff with two periods to get savings, as your consumption in the reduced price period is really high and the consumption in the overcharged price period is medium. In many of the day periods the

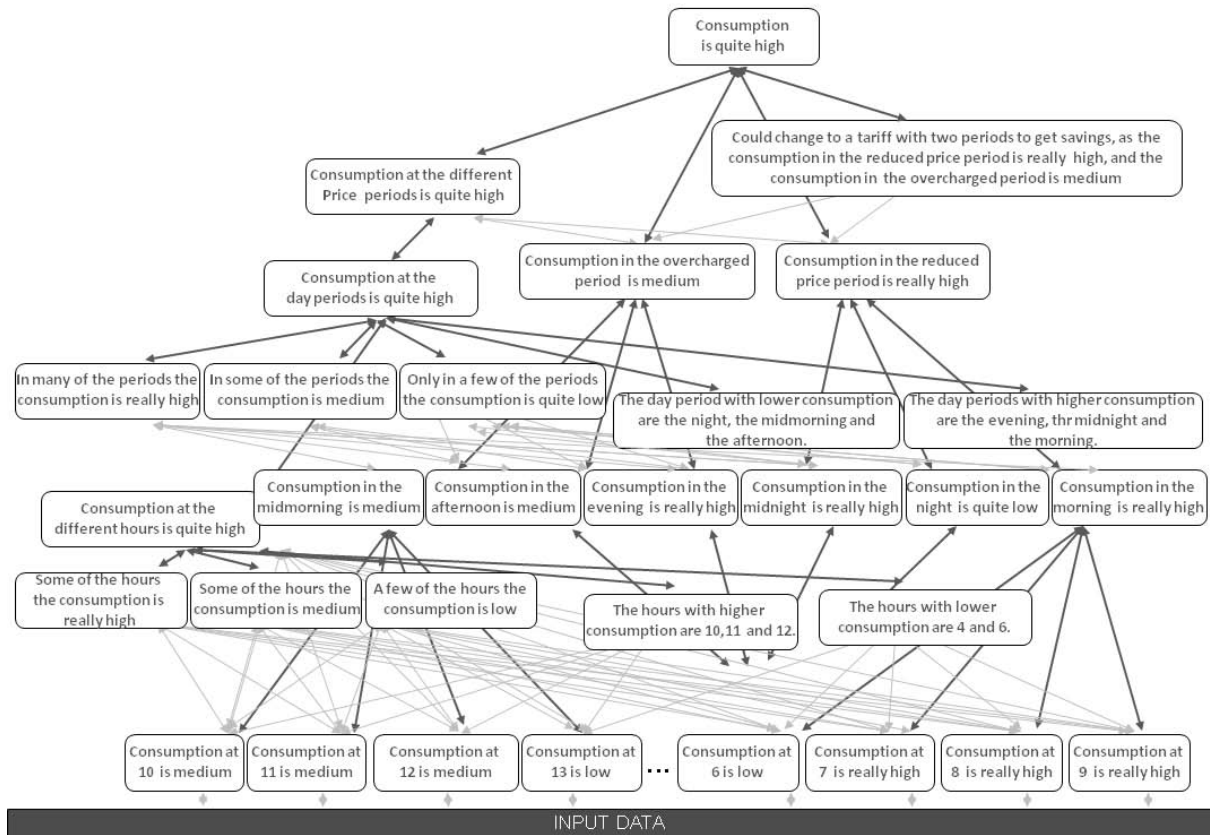


Figure 4.19: Linguistic Expressions Instantiation

consumption is really high, but only in a few, the consumption is quite low. The day periods with higher consumption are the evening and the midnight, whether day periods with lower consumption are the midmorning and the afternoon”.

4.4 Emotional perceptions

In order to increase the effectiveness and acceptance of a *Personal Energy Advisor*, interface would need to show a closer human-like behaviour including emotions.

Here it is extended the computational capabilities of the GLMP to include an emotional engine that computes emotional states associated to each perception based on user behaviour.

The proposed approach to include perceptions in a GLMP includes a new dimension with an emotional state associated to each perception, and an emotional process to apply emotions dynamically to each perception in the OLAP-GLMP.

	Degree of Validity	Degree of Relevancy	Degree of Covering	Suitability
Your consumption is quite high	1,000	0,625	1,000	0,850
You could change to a tariff with two periods	0,960	1,000	0,762	0,897
The consumption at the different price periods is quite high	0,973	0,625	0,952	0,826
The consumption in the reduced price period is really high	0,960	0,750	0,881	0,844
The consumption in the overcharged price period is medium	0,987	0,300	0,881	0,670
In many of the periods the consumption is really high	1,000	0,750	0,643	0,757
In some of the periods the consumption is medium	1,000	0,300	0,643	0,577
Only in a few of the periods the consumption is quite low	1,000	0,750	0,643	0,757
The day periods with higher consumption are the evening and midnight	0,900	0,750	0,571	0,709
The day period with lower consumption are the night, the midmorning and the	0,900	0,750	0,571	0,709
The consumption at the different day periods is quite high	0,973	0,625	0,524	0,654
The consumption in the midnight is really high.	0,917	0,750	0,095	0,521
The consumption in the night is quite low	0,962	0,625	0,095	0,480
The consumption in the morning is really high	1,000	0,750	0,095	0,538
The consumption in the midmorning is medium	0,962	0,300	0,095	0,350
The consumption in the afternoon is quite low	1,000	0,625	0,095	0,488
The consumption in the evening is really high	1,000	0,750	0,095	0,538
.....				
The consumption at the different hours is quite high	1,000	0,625	0,286	0,564
The consumption at 00 is quite high	1,000	0,625	0,024	0,460
The consumption at 1 is medium	0,671	0,300	0,024	0,324
The consumption at 2 is quite low	1,000	0,625	0,024	0,460
The consumption at 3 is quite low	1,000	0,625	0,024	0,460
The consumption at 4 is really low	0,851	0,750	0,024	0,480
.....				

Figure 4.20: Suitability assignment for perception instances.

4.4.1 Emotional-GLMP

Here is proposed a computational model to include emotional behaviour in the GLMP, based on the close relationship between perceptions and emotions. The model enriches raw semantics, compute an emotional behaviour and express emotions. It is called Emotional-GLMP. It will enrich the communication process increasing the impact and expressiveness of the linguistic messages generated by the GLMP.

An emotional-GLMP is able to compute emotions based on fuzzy states. Here a perception has associated a structure which satisfies a given set of semantic and emotional constraints.

The GLMP model is designed to have different emotional states depending on the communication actors, namely the speaker, the hearer and the communicative goals. The communicative goals are defined by the designer, e.g., following a behaviour of consumption considered efficient.

Each emotional state is applied to individual perceptions; when the efficiency of the hearer has a good behaviour and there is a fluent relationship between speaker and hearer the emotional state of the system improves and is closer, or when the efficiency behaviour is under performance and the hearer is casual the emotional state and confidence deteriorates. The designer defines for each perception an emotional constraint.

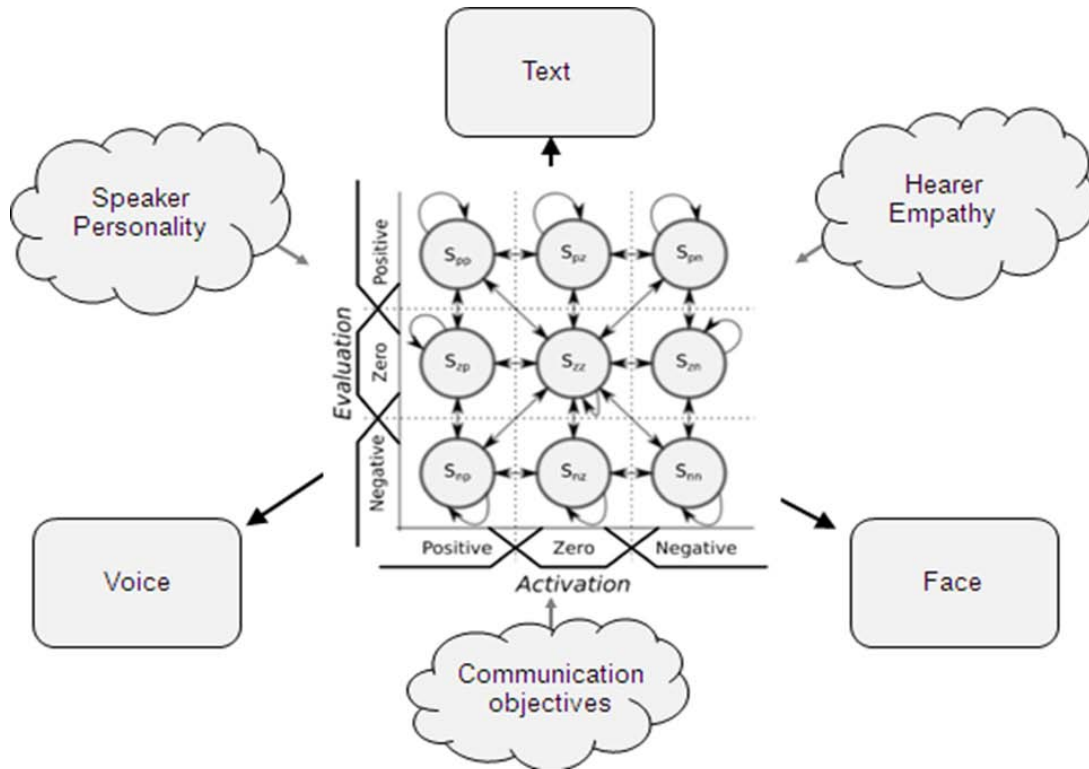


Figure 4.21: Emotions inputs and outputs

4.4.2 Expressing emotions

The emotional state is transmitted to the listener through verbal “*Why and how is it that we say the same thing differently to different people, or even to the same person in different circumstances?*” (61) and the nonverbal “*Why the Same Expression May Not Mean the Same When Shown on Different Faces or Seen by Different People?*” (59) communication.

The aim of multimodal output representation is to describe the combination of words, grammatical construction and gestures while communicating.

In face-to-face communication, the emotional state of the speaker is transmitted to the listener through verbal and the nonverbal communication, including face expression, voice intonation and the language generated.

In fact a perception is defined as a set of instantiations that reflect the personality of the transition matrix. Also perceptions affect the voice and the face expression to increase communication purposes.

4.4.3 FFSM for modelling emotions

Modeling the temporal structure of emotional state and translate this state in language generation is a difficult challenge. The approach used here considered emotions catego-

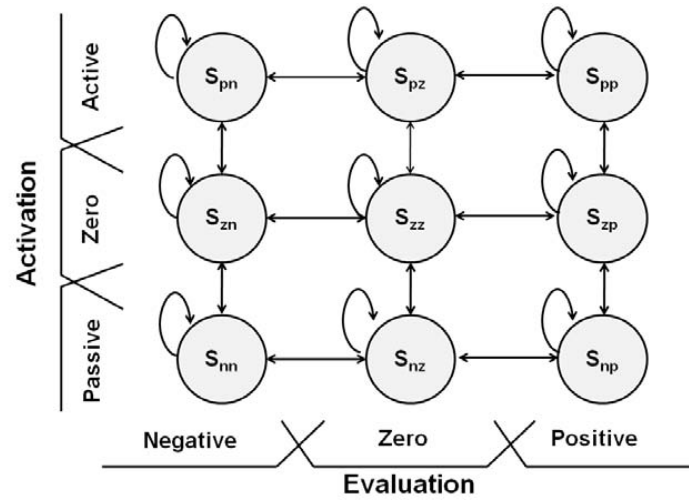


Figure 4.22: States of the FFSM

rized along two axes, activation (from passive to active) and evaluation (from negative to positive).

Fuzzy Logic allows to model complex emotional states and behaviours. A emotional state is fuzzy, e.g., at any point in time the system can be in various emotional states to certain degree. Fuzzy Logic has been already used in emotion simulation(42).

Emotional states assigned to each perception are computed by a FFSM is a tuple (U, S, f, Y, g, S_0) where:

U is the input to the FFSM based on two linguistic variables *evaluation* u_a and *activation* u_e , e.g., the vector (u_a, u_e) . Both variables are fuzzyfied with linguistic labels (negative, zero, positive). Activation is computed based on the number of times the user has been in touched in the last 12 months, evaluation is the value assigned to the a_i with the highest degree of truth.

S the state diagram of the FFSM used in the prototype is given in Fig.4.23. A simplified version of states of the Plutknik Wheel with nine states is used. The model allows only to change the state to “neighbour” states.

f is a function to calculate the state vector at time step $t+1$, $S(t+1) = f(S(t), U(t))$. The transition function is implemented using a set of fuzzy rules combining the actual state and the input.

Y is the output vector (e_i, a_i) . Each state is defined based on the possible values of inputs u_a and u_e . The output vector (y_a, y_e) has the values of the activity and evaluation variables for the FFSM “memory” and input.

g is the function used to calculate the degree of emotional truth for each a_i of each perception, namely W_s . It is described in more detail in the next section.

S_0 is the initial state of the system.

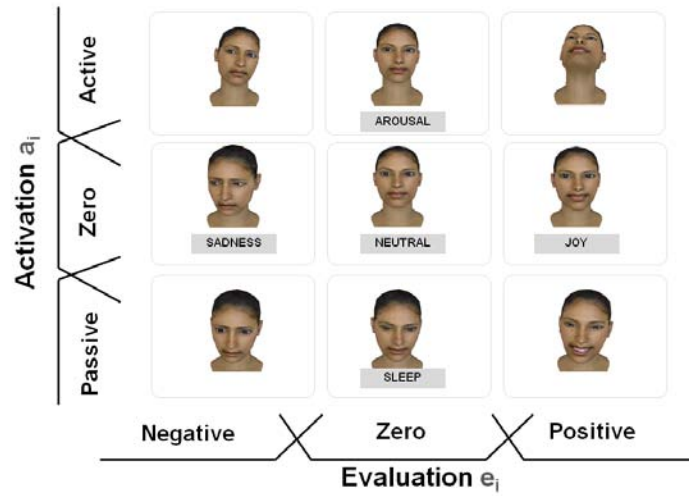


Figure 4.23: Example of computing emotions (Character by Hapteck).

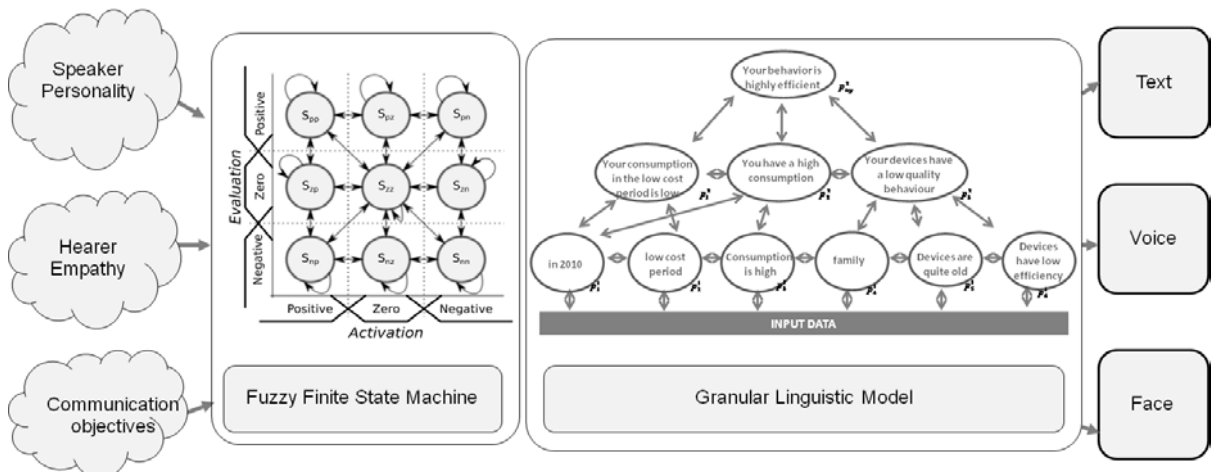


Figure 4.24: GLMP Functional modules for computing emotions

4.4.4 Modeling and Computing Emotions

Content selection for CP extraction involves the computation and deduction of information from an input domain. Each of the perceptions defined have assigned an emotional weight. The emotional weight is defined from an input vector of variables namely the proximity between speaker and hearer and the behaviour related to the communication goal. These inputs, are modelled translating them into basic emotional variables (activation and evaluation) as showed on Fig. 4.25. The collection of input constraints are represented in the output emotional state weight W_s resulting from the FFSM as showed on Fig. 4.24.

An initial state is computed in the overall GLMP and for each perception it will transitate without high impact, moving through immediate proximal states for each CP value. The transition Function ($S_{t+1} = f(S_t, U_t)$) is used to calculate the next state. This model of FFSM, does allow staying in the same state or move forward. This function

produces the degree of validity for the state vector S_{t+1} , combining the previous state and the input vector U_t using a set of if-then rules.

4.4.5 Computational Perception with emotional states

For the inclusion of emotions, the definition of a CP and PM are extended in order to include the emotional state associated to each perception in the GLMP and for the overall GLMP.

A *CP* is the computational model of a unit of information acquired by the designer about the phenomenon to be modelled. In general, *CPs* corresponds with specific parts of the phenomenon at certain degrees of granularity. A *CP* is a tuple (A, W, R, S, W_s) where:

A is a vector (a_1, a_2, \dots, a_n) of linguistic expressions (sentences in NL) that represents the linguistic domain of the CP. Each a_i describes the value of the CP in each situation with specific degree of granularity. These sentences can be either simple, e.g., $a_i = \text{"The consumption in the morning is quite high"}$ or more complex, e.g., $a_i = \text{"Some days you could save a lot of money"}$.

W is a vector (w_1, w_2, \dots, w_n) of validity degrees $w_i \in [0, 1]$ assigned to each a_i in the specific context. The concept of validity depends on the application, e.g., it is a function of the truthfulness of each sentence for specific input data.

R is a vector (r_1, r_2, \dots, r_n) of relevancy degrees $r_i \in [0, 1]$ assigned to each a_i in the specific context. The concept of relevancy depends on the application. During the initialization stage, these values are assigned by the designer of the computational system in function of specific requirements in the application context.

S is a vector (s_1, s_2, \dots, s_n) of emotional states $s_i \in [0, 1]$ assigned to each a_i in the specific context. Each s_i , has the activation and evaluation values assigned as a tuple (a_i, e_i) . It is used for calling the FFSM. The concept of states depends on the application. During the initialization stage, these values are assigned by the designer of the computational system in function of specific requirements in the application context, e.g. in an application for the evaluation of power consumption, when the consumption is low the evaluation is good, and when the consumption is high, the evaluation is low. This evaluation degree enriches text generation process. For each activation and evaluation values, it is assigned a linguistic expression (a_i) as showed in Fig. 4.25.

W_s is a vector $(ws_1, ws_2, \dots, ws_n)$ of emotional degrees $ws_i \in [0, 1]$ assigned to each a_i as defined by output of the FFSM.

It is increased the expressivity of *CP* by including the vector of emotional values for activation and evaluation S in its definition, and a vector of validity assigned for each specific emotional state. It is used as complementary data to increase the expressivity of each sentence. Note that a perception has an emotional space defined as showed in Fig. 4.25.

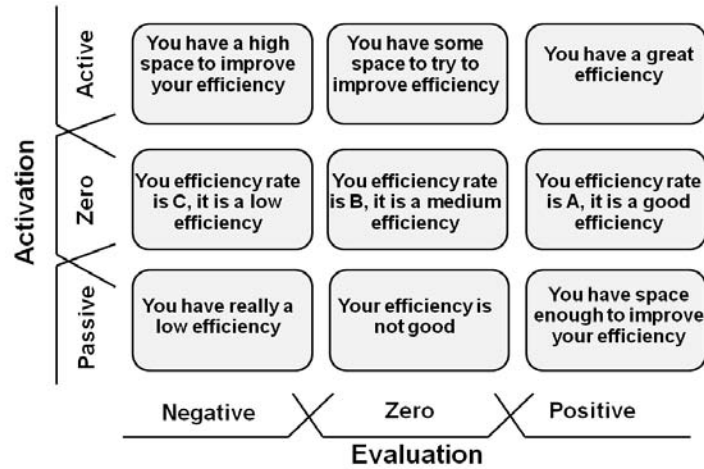


Figure 4.25: Example of computational perception emotional states

	a_i	w_i	r_i	e_i	a_i	w_s
You have a high space to improve your efficiency		0,7	1,0	0,0	0,0	0,7
You efficiency rate is C, it is a low efficiency		0,7	1,0	0,3	0,0	0,3
You have really a low efficiency		0,7	1,0	0,7	0,0	0,0
You have some space to try to improve efficiency		0,3	0,5	0,0	0,3	0,7
You efficiency rate is B, it is a medium efficiency		0,3	0,5	0,3	0,3	0,3
Your efficiency is not good		0,3	0,5	0,7	0,3	0,0
You have a great efficiency		0,0	1,0	0,0	0,7	0,7
You efficiency rate is A, it is a good efficiency		0,0	1,0	0,3	0,7	0,3
You have space enough to improve your efficiency		0,0	1,0	0,7	0,7	0,0

Figure 4.26: Example of a computational perception that models the consumption efficiency

For example, a computational perception that models the consumption efficiency $CP_1 = (A_1, W_1, R_1, S_1, Ws_1)$, could be instantiated as showed in Fig. 4.26

4.4.6 Perception Mapping

PM s are used to create and aggregate CP s. There are many types of PM s and this paper explores some of them. A PM is a tuple (U, y, g, s, T) where:

U is a vector of input CP s, $U = (u_1, u_2, \dots, u_n)$, where $u_i = (A_i, W_i, R_i, S_i, Ws_i)$. In the special case of first order Perception Mappings ($1PM$ s), these are the inputs to the GLMP and they are values $z \in R$ being provided either by a sensor or obtained from a database.

y is the output CP , $y = (A_y, W_y, R_y, S_y, Ws_y)$.

g is an aggregation function employed to calculate the vector of fuzzy degrees of validity assigned to each element in y , $W_y = (w_1, w_2, \dots, w_{n_y})$, as a fuzzy aggregation of input vectors, $W_y = g(W_{u_1}, W_{u_2}, \dots, W_{u_n})$, where W_{u_i} are the degrees of validity of

the input perceptions. In Fuzzy Logic many different types of aggregation functions have been developed. For example g could be implemented using a set of fuzzy rules.

s is a function employed to call the FFSM with the input vector (e_i, a_i) assigned to each a_i . The FFSM calculate the vector of fuzzy degrees of validity assigned to each element in y , $W_{s_y} = (ws_1, ws_2, \dots, ws_{n_y})$,

T is a text generation algorithm that allows generating the sentences in A_y . In simple cases, T is a linguistic template, e.g., “*The consumption of electricity is {high | medium | low}*”. Note that there is la linguistic template assigned to each emotional state.

4.4.7 Emotional-GLMP Application

In a previous section was detailed the navigation through the information stored in an GLMP schema. Using a subset of the previous GLMP schema, it is defined the historical energy consumption database. Also linguistic description expressiveness was extended by selection of the most suitable sentence in a GLMP generating a coherent and cohesive linguistic discourse, pruning non relevant perceptions and by aggregating related ones avoiding redundancy.

The dimensions associated with a query over the consumption measure are translated to computational perceptions. The definition of fuzzy Consumption measure has values $Consumption \in \{High, Medium, Low\}$, $LowCostPeriodConsumption = \{High, Medium, Low\}$, $DeviceEfficiency = \{High, Medium, Low\}$.

The communicative objective is to identify relevant information as response to the query “*How is my efficiency?*”. It is a second order perception, computed from a query on the three measures listed above.

Efficiency is a tuple (U, y, g, T) where:

U is the set of input CPs $U = \{u_1, u_2, u_3\}$, $U = \{ Consumption, LowCostPeriodConsumption, DeviceEfficiency \}$.

y is the output CP with values $y = (A, W, R, S)$, e.g., $y = \{ (“Highly inefficient”, 0.0), (“Inefficient”, 0.68), (“Moderately efficient”, 0.32), (“Highly efficient”, 0.0) \}$.

g An aggregation function is implemented $W = g(u_1, u_2, u_3)$ using a set of Mamdani-type fuzzy rules as follows:

IF $(u_1$ is *High*) AND $(u_2$ is *Low*) AND $(u_3$ is *High*) THEN *Inefficient*

IF $(u_1$ is *Medium*) AND $(u_2$ is *High*) AND $(u_3$ is *High*) THEN *Moderately efficient*

IF $(u_1$ is *Medium*) AND $(u_2$ is *High*) AND $(u_3$ is *High*) THEN *Moderately efficient*

...

IF $(u_1$ is *Low*) AND $(u_2$ is *High*) AND $(u_3$ is *Low*) THEN *highly inefficient*

Here, the perception of efficiency is defined based on the study of available data about consumption of families and implemented as expert knowledge.

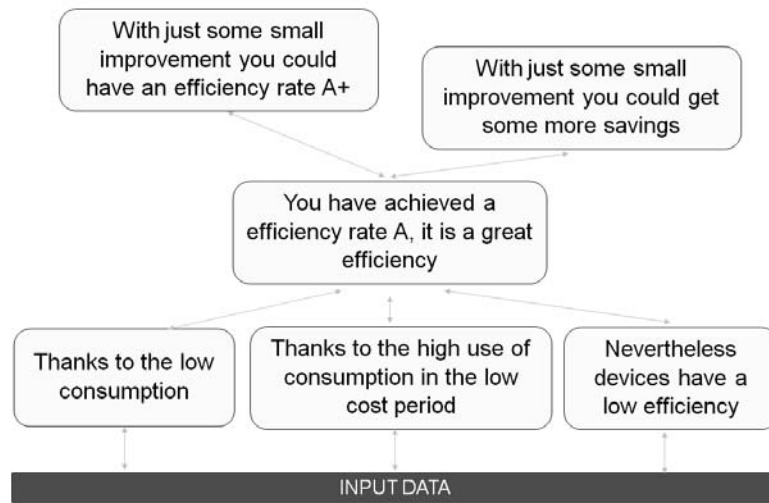


Figure 4.27: Example of GLMP for a high efficiency rate.

s is the input vector for the the FFSM that returns the weights associated for each emotional state u_i .

T is a simple template: “Your behavior is (highly inefficient | inefficient | moderately efficient | highly efficient)”. Note that there is a template assigned to each emotional state. High efficiency is correlated with a efficiency rate A, and the lowest efficiency is correlated to a efficiency rate C.

During GLMP definition, first order perceptions are computed from these input data for each lower level attribute values, e.g. “*Your consumption is low*”. Second order perceptions are spread upwards for each attribute value to compute the second order perception and two more explicative perceptions obtained from the efficiency result, evaluating the possible efficiency to obtain and a overall description of the savings.

Emotional perceptions are constrained and instantiated in a set of linguistic expressions from a set of alternatives that express the same semantic content with different emotional values. Facial expression makes behaviour more understandable and improves communication.

In order to increase expressiveness the efficiency GLMP is explained in more detail with perceptions related to the improvements that could be achieved. Fig. 4.27 shows the structure of answers obtained in this application example for a user with efficiency A. Using this GLMP, a more complete explicative sentence can be generated. Note that the evaluation value associated with each perception instance, enrich the natural language generated, knowing if the impact of the elements is positive or negative and therefore increasing expressivity.

For a medium efficiency rate, the linguistic description would be:

“Your efficiency rate remains in a rate B. You could have an efficiency rate A+ and get savings. Your efficiency rate is B because although your consumption is medium, you have a low consumption in the low cost period and a low efficiency in your devices.”

For a low efficiency rate, the linguistic description would be:

“Your consumption was quite bad, but now it is even worse. Your efficiency rate has decreased to a rate C. It is really a low efficiency rate. You have a high space to improve your efficiency and get high savings. To improve the efficiency you have to reduce your consumption as it is really high compared with households of similar characteristics. Also you have to move consumption to the night period as it is cheaper. Also you should change some of inefficient devices.”

For a high efficiency rate, the linguistic description would be:

“You have achieved a efficiency rate A, it is a great efficiency. With just some small improvement you could have an efficiency rate A+ and get some more savings. Your efficiency rate is A thanks to the low consumption and the high use of consumption in the low cost period. To improve the efficiency you could improve some of the devices. ”

Discussion of results

With the proposal of this section and the previous ones have been shown the basic components of a computational system for linguistic description of complex phenomena. These components as express in (84) perform functions in the whole system that can be listed as follows:

1. Monitoring real phenomena (Data acquisition).
2. Include a semantic model of the input data (Granular Linguistic Model).
3. Handling the imprecision of input data (Instatiation).
4. Lexico-grammar model of phenomena (Report template).
5. Natural Language generation (Report generation).
6. Include a model of the temporal evolution of the avatar’s emotions (Granular Linguistic Model).
7. Include emotional content in texts (Report template).
8. Include a graphical representation of the agent’s emotional gestures (Report template).

Implementing this list of functions requires a multidisciplinary approach where each function can be focused and solved from several perspectives.

In table 4.1, is showed a brief comparison of several research works just by considering if

the work deals with each specific function. Note that the list is not exhaustive and it is not compared the quality of each approach but mentioned some works of other researchers working with similar goals.

Table 4.1: Comparison with some previous research works.

Author	ref.	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)
Fleischman2002	(49)	-	-	-	✓	✓	-	✓	✓
Barbat2003	(9)	-	-	-	✓	-	-	✓	✓
Mahamood2011	(77)	✓	✓	-	✓	✓	-	✓	-
Rita2011	(24)	✓	✓	✓	✓	-	-	-	-
Arguelles2013	(7)	✓	✓	✓	✓	✓	-	-	-
this proposal	-	✓	✓	✓	✓	✓	✓	✓	✓

4.5 Question Answering perceptions processing

In this section it is described a framework for a QA based on GLMP as a deduction process based on natural language computation as showed in Fig. 4.28. The framework is based on the mechanization of query representation via language constraints and the precisation of the meaning of concepts for answer deduction in a granular environment.

Usually while queering and answering are express concepts imprecise in nature, e.g., when not a direct yes/no answer is requested, e.g., “*Is high my consumption? Yes*” but instead a more general description with deductive reasoning is required, e.g., “*Is high my consumption? Your consumption at 12h is quite high, around 30Wh.*”. The solution proposed includes capabilities based on perceptions for question interpretation and answer generation in a structured discourse.

In a question answering system the model moves from a crisp treatment:

*What is my consumption at 12h?
Your consumption is 30kWh.*

to a fuzzy approach based on linguistic computing.

*Is high my consumption?
Your consumption at 12h is quite high, around 30Wh.*

Question interpretation capabilities include a range of query expressions linked to perceptions, including linguistic quantifiers and fuzzy expressions. All this process is supported by a OLAP-GLMP definition that describe the knowledge context, semantic perceptions definition and deduction capabilities. The queries are limited to the knowledge domain.

Response generation determines what information is relevant to communicate for the query and how to organize this information. It allows navigating through information.

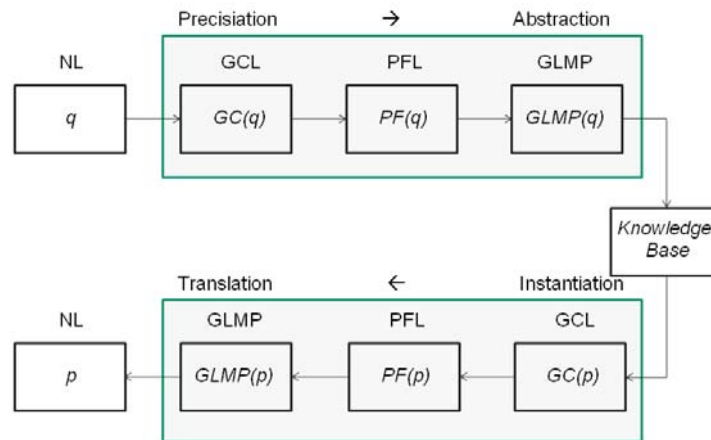


Figure 4.28: GLMP QA deduction process

The computation is performed over a GLMP that describes and enrich knowledge from input data, extending deduction capabilities by perception reasoning. Input data is understood as a set of data, e.g., *data that could be retrieved from a database*.

Question Interpretation based on perceptions

During interpretation, query language expressions are modelled as perceptions to capture imprecision inherent to natural language. The goal is to represent the meaning of propositions expressed in natural language as a constraint perception.

Queries are precisiated and abstracted into perceptions that captures the semantics in a formal structure. A perception will be treated as unitary element of computation. Perceptions are modelled in the GLMP definition, and are used to calculate the precisiated implications of propositions. The queries are restricted to use a subset of natural language, which is limited in terms of vocabulary and syntax, related to expressions defined in the GLMP.

q : *How is my consumption during the morning ?*
 p : *(Consumption(MorningPeriod)) is ? R*

In the presence of an input query, it is composed the meaning of query propositions from the meaning of its constituents e.g. *facts, time, date,..* and query type e.g. *what, when,.. or if it is a yes/no question*.

Perception mappings are modeled in the GLMP definition, and are used to calculate the precisiated implications of propositions and for the description of an input domain.

The process followed to parse the input query is quite basic. Identify semantic type of constituents and key facts, determine additional constraints on the answer, identifying relations, e.g., *year(2008)*, *period(morning)*,..., and identifying the question type, e.g., *when looks for a date or time, etc.*

The process followed to interpret a question is decomposed in different steps:

- Tokenization. Tokens are identified and extracted from the input string.
- Syntactic Tagging. Here is performed an n-gram tagger to identify and tag each independent word or pairs.
- Semantic Tagging. Synonyms and pattern matching for each string.
- Emotional Tagging. Identify the user expression to extract an emotional level.
- Perceptions. Linguistic expressions are precisiated in perceptions, e.g. *Most of the months the consumption is high.*

To analyze a sentence, it is decomposed and each element is tokenized and tagged. Each token is classified as a member in the syntactic structure and a semantic decomposition of the sentence is performed. Linguistic labels are identified as fuzzy elements and transformed into a perception linguistic definition. Also an emotional state is assigned, interpreting emotional claims in the user query expression.

Language expressions are allowed to refer not to the information itself but about the navigation over the granular information structure inherent in the GLMP, e.g., *command expressions like, more detailed information.* These commands allow extending the description of the information stored in a GLMP through the hierarchical granular information.

4.5.1 Answer Generation

Answer generation involves the computation and representation of information from an input domain and the assessment of relevance of the information to define based on the question the resulting response.

The computation and representation of information from an input domain is defined in a GLMP. An input domain is understood as raw level and crisp data. In a previous section was proposed an OLAP-GLMP approach, to dynamically build GLMP based on input data retrieved from a database.

The assessment of relevance is computed as a matter of degree, as a fuzzy concept to provide significance to the GLMP perceptions. It is required a method for computing the degree of relevance based on the meaning of the question to extract only those CPs relevant enough. A relevance function $R(q/p)$ is defined, where q is the question or topic,

and p is each perception in the GLMP, and R is the degree to which p is relevant to q .

For each query it is computed the relevance over each perception in the GLMP $R(p/q)$. It is computed based on the distance for each constituent of the query in the dimensional space. Distance is computed for the dimensional attribute values, e.g., $\{Midnight, Morning, \dots\}$, the attributes at the different aggregation levels, e.g., $Time = \{Price\ Period, Day\ Period, Hour\}$, and the measure values, e.g., $\{really\ low, quite\ low, medium, \dots\}$.

4.5.2 QA Application

In this section, a practical application is used to illustrate the details of the proposed approach. In a previous section was proposed the navigation through the information stored in an GLMP schema and the generation of a coherent and cohesive linguistic discourse. Using a subset of the previous GLMP schema, it is defined the historical energy consumption database. The dimensions associated with a query over the consumption measure are translated to computational perceptions.

The definition of fuzzy Consumption measure has values $Consumption \in \{Really\ High, Quite\ high, \dots, Quite\ Low, Really\ Low\}$.

The dimensional fuzzy attributes for the Time dimension are defined as $Time = \{Price\ Period, Day\ Period, Hour\}$ with values:

- $Price\ Period = \{Charged\ Period, Reduced\ Period\}$
- $Day\ Period = \{Morning, \dots, Night\}$
- $Hour = [0, 23] \in N$

A *Summarizer* = $\{Only\ a\ few, Some, A\ lot, Most\ of\}$ and a maximizer and minimizer quantifiers $Max/Min = \{the\ higher, the\ lower\}$ are defined and applied to fuzzy attributes.

The queries for testing purposes are:

- a) “Which ones are the hours with higher consumption?”.
- b) “In any of the day periods, do I have a low consumption?”.
- c) “How is the consumption this year?”.
- d) “Could I get more detail”.

Query interpretation is performed by items. Note that question c) is a command question:

- a) “hour - higher consumption”
- b) “some - day period - low consumption”

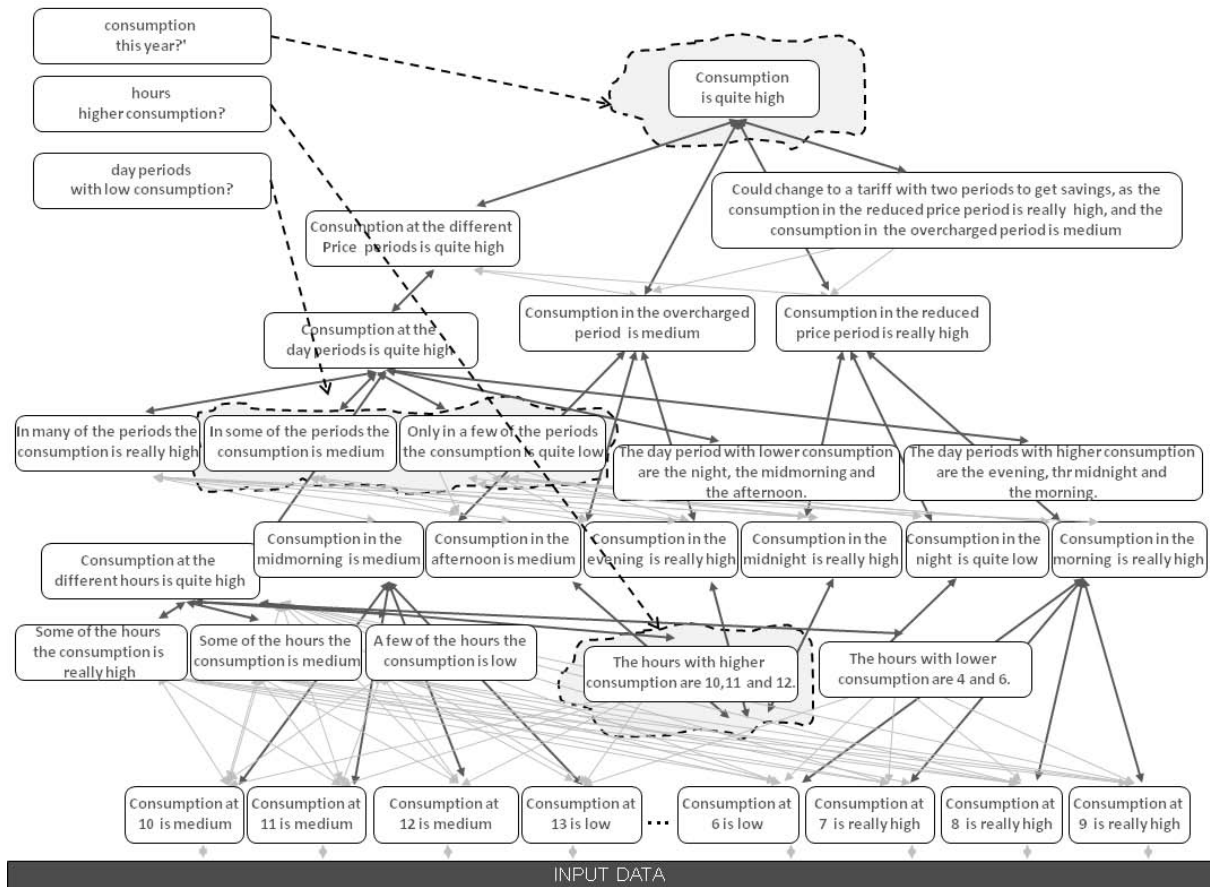


Figure 4.29: Linguistic Expressions Instantiation

c) “consumption - this year”

Translated to perceptions, where a_i is:

- “consumption.higher(hour) is ? ”
- “consumption.low((day periods)) is ? some”
- “consumption (thisyear) is ?”.

Input data is retrieved from a query on *Time* dimension. The initial result is a table with the numeric value of the average consumption for each hour as it is the lower time hierarchy attribute, e.g. as tuples (*Hour-Consumption in wh*) “H22 - 1.320, H23 - 1.796, H24 - 586,1 - 217, H2 - 131,3 - 97, H4 - 45, H5 - 65, H6 - 161, H7 - 129, H8 - 945, H9 - 1.243, H10 - 116, H11 - 145, H12 - 132, H13 - 87, H14 - 161, H15 - 129, H16 - 171, H17 - 131, H18 - 253, H19 - 275, H20 - 876, H21 - 934 ”.

During GLMP definition, first order perceptions are computed from these input data for each lower level attribute values, e.g., “Your consumption at 23 is quite high”. Second order perceptions are spread upwards for each attribute value along the *Time* hierarchy till defining the overall GLMP structure.

The hierarchical layout is defined using the dimensional axis to generate the granular ordered structure, using the Time hierarchy. The resulting GLMP is showed on Fig. 4.29.

For each query it is computed the relevance over each perception in the GLMP, namely R and $R(p/q)$. Pruning represents the process of identification of relevant information among all the CPs instantiated in the GLMP. As defined in the model, perceptions with lower relevance degree are pruned. Results are displayed on Fig. 4.29.

After pruning, remaining CPs are aggregated if their meaning can be expressed in an individual complex perception. The choice of connectives is based on relations that hold between perceptions and their semantics. First step is looking for similar perceptions, expressed by the “and” connective. Second aggregation is performed looking for opposed behaviour expressed by a “but” connective.

The output text generated by the instantiation of the reorganized GLMP is as follows:

- a) *“The hours with higher consumption are 10,11 and 12. ”.*
- b) *“Yes, in some of the periods the consumption is quite low. Consumption is low in the midmorning and afternoon.”*
- c) *“Your consumption is quite high.”.*
- d) *“Your consumption is quite high. Your consumption in the reduced price period is really high. In a few of the day periods the consumption is quite low. The day periods with higher consumption are the evening and the midnight, whether day periods with lower consumption are the midmorning and the afternoon.”.*

4.6 Energy advisor architecture

Within the Energy Advisor architecture, the different components are arranged to allow the system to manage user requests and provide responses about the energy consumption and to support the ability to generate personalized advisory reports. The objective of the virtual energy advisor defined is to describe and advice the customers about their efficiency.

4.6.1 Off-line information and On-line processes

The process followed to generate the natural language advice, is based on the definition of two stages as showed on Fig. 4.30. The off-line information that describes the knowledge domain to interpret questions and generate the answers in natural language, and the on-line process that based on the user questions making use of the knowledge defined is in charge of generating dynamically the required responses requested by users.

- Off-line Knowledge Domain

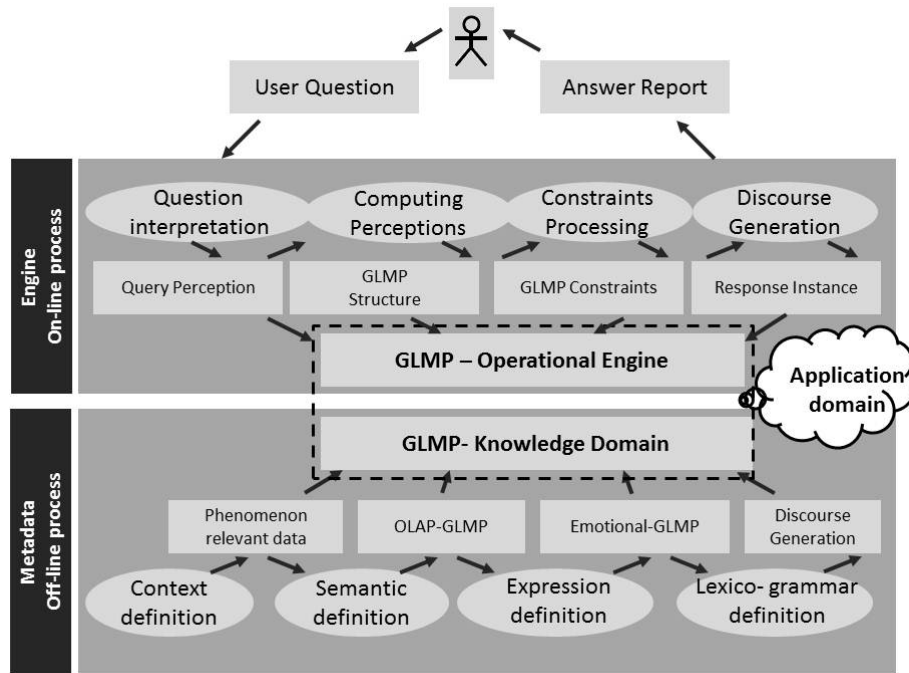


Figure 4.30: Functional architecture for the autonomous personal advisor.

The off-line Knowledge Domain contains the information required to allow the on-line process to execute. It models the personal Energy Advisory Knowledge, and is defined by a designer. In this module is defined the required information to operate, namely the context definition, the GLMP-OLAP definition, emotional contents and Natural language. In the definition of the off-line knowledge, will be set the capabilities to access and understand the data, interpret user questions and generate answer reports using natural language.

- On-line application engine process

The online process is in charge on dynamically, under the user requests get the right responses. The overall process is decomposed in different steps to interpret queries from users, access the physical data, compose perceptions, identify relevant information, and instantiate in natural language easily understandable for the user. The process follows sequential stages, with interrelated functionality that refines due to constraints the information generated from previous layers.

4.6.2 Concepts, Procedures and Attitudes

An analysis of what involves the Personal Energy Advisor in terms of Concepts, Procedures and Attitudes reveals the following elements:

- Concepts

The Energy Advisor must learn the concepts around the environment and knowledge base, e.g., energy consumption, emotional state or communication channels. Concepts are defined in an off-line stage.

- Procedures

The Energy Advisor must learn the procedures to use the concepts defined in order to provide response, namely, locate the information requested by the user, access the database, extract a subset of the information, translate it into natural language and gain empathy with emotional behavior. Procedures are defined in an off-line stage.

- Attitudes

The Energy Advisor manages the procedures and concepts to identify the most relevant information based on user requests and compose the optimal messages with higher impact. Attitudes use concepts and procedures to interpret and retrieve responses. Attitudes are executed in an on-line stage.

Concepts and procedures support data access with independence from the query language and the underlying data structure in the database adding semantic and lexicogrammatical knowledge in the translation from crisp data into fuzzy linguistic perceptions e.g., a query over the consumption at “9h” in the morning will result not in a crisp value “10Wh”, but in a linguistic expression “*Your consumption at 9 is quite low*”.

Attitudes will provide the ability and goals to generate the overall GLMP structure from low level perceptions, spreading upwards to top order perception. Lower level perceptions are at the bottom of the GLMP and while summarizing there is a movement to higher order ones. Linguistic description over the complete phenomena in the GLMP requires extracting a summary of the available information where certain suitable elements are remarked while other irrelevant aspects are pruned. To add meaning to the linguistic expressions and make the message more understandable by the user emotions are associated with perceptions. It is performed by the definition of a matrix of states and the transition between one state to another. Instantiation of the GLMP generates different verbal and non verbal behaviour associated.

The different levels of the CPA7L architecture are showed in Fig. 4.31 and described as follows.

4.6.3 Physical Level

The virtual agent gets inputs from the environment. These inputs are the customer hourly consumption measures. The electricity consumption data is based on real data retrieved by hourly meter devices. To get them, the current consumption meters at the customers houses that only takes accumulative measures, are replaced by smart-meters that store and send measures hour by hour. In this way a customer consumption profile can be built.

Additionally, there is the need to communicate the resulting reports to the customer. The main channels are based on mobile devices where the virtual agent response reports could be accessed. A possible channel is the printed bill because it reaches all the customers. Based on this model, a number of different channels of communication may be used.

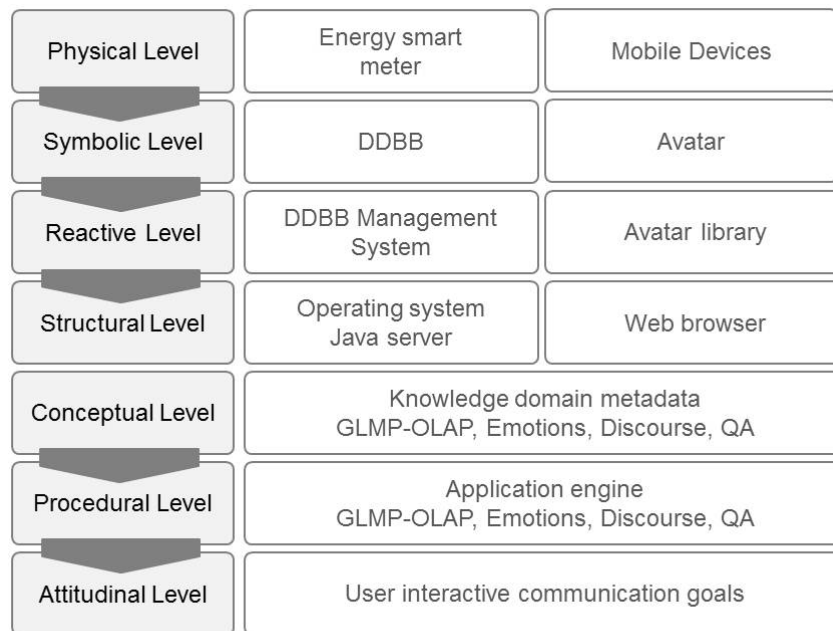


Figure 4.31: CPA7L architecture elements for the definition of the Personal Energy Advisor.

4.6.4 Symbolic Level

The hourly consumption measures are accessible in a relational database. A relational database has a model defined by tables and fields to store values assigned to the consumption.

Some basic routines are needed to select the appropriate emotional expression in the virtual agent for a visual representation. These routines only show the corresponding image at the defined channels, but the selection mechanism is implemented in the upper levels.

The virtual agent need a visual representation that makes it easier for the customer to create a mental representation of it as showed in Fig. 4.32.

Animated characters are usually the front end of an NLP system. Speaking avatars in a website empower the online communication.

4.6.5 Reactive Level

At the reactive level, low level routines to execute, retrieve and process queries over the database model are needed.

The same way, it should have a mechanism to publish information to the customer through different channels. The common way to export the information, can be text files with the generated advices. The resulting text, can be included to the billing process in order to be printed on each customer or publish them into the web using the avatar



Figure 4.32: Visual representation of the five emotional states of the virtual agent: *Disappointed*, *Concerned*, *Neutral*, *Happy*, *Excited*.

interface. Then routines to write and handle text files are needed. More complex or sophisticated methods needed, they belong to this level. In fact, these routines are just calls to low level routines or services already implemented on the Operating System, a Web Server, or a DBMS.

4.6.6 Structural Level

At the structural level is defined the software environment where the virtual agent is embedded. It should be a software, with enough computational power to process the data obtained from each customer and support for previous layers. This server should have access to the database, to deliver its results and to get the basic information of the customers in order to know who is each customer.

The software runs over a Java Server and is delivered in a web browser, all services that must be implemented require no special application or hardware.

4.6.7 Conceptual Level

GLMP concepts are used to define historical energy consumption by using the GLMP-OLAP schema. The emotional state is defined from an input vector of variables namely the proximity between speaker and hearer and the behaviour related to the energy consumption and efficiency.

With the addition of the emotional component, new concepts must be included in the virtual agent. These concepts allow it to simulate the emotional state, and process the data taking into account that state. Also, the conceptual elements of the FFSSM will be described in this level and the functions are described in the next level.

Lexical and grammatical definitions associated to each perceptions, are defined at the conceptual level for their instantiation of the semantic and emotional meaning in natural language expressions.

4.6.8 Procedural Level

Related to the concepts that the virtual agent must handle are the procedures to manage them. At the procedural level are defined the processes required to build the virtual agent messages based on the attitudes.

The procedures defined are those required to retrieve data, aggregate the CPs, compute emotions, and build a GLMP discourse with the Perception Mappings that allows the transformation and analysis of the customer consumption data. Also those procedures related to the transition through emotions for communication are defined in here.

For example to identify relevant information as response to the user query “*How is the consumption this year?*”. Input data is retrieved from the database on *Time* dimension. The result is a table with the numeric value of the average consumption for each hour as it is the lower time hierarchy attribute, e.g. *as tuples (Hour-Consumption in wh)*.

4.6.9 Attitudinal Level

In the final level, the attitudes of the virtual agent are defined. Attitudes are focused on the ability to interact with the users using the procedures defined to manage the underlying concepts.

Attitudes define the objectives and motivations that it has, basically a reactive behavior looking for the right answer related to increase the efficiency. This may include a list of attitudes, e.g., achieve energetic efficiency, promote the reduction of the consumption peaks, promote an increase of the consumption on low cost hours, and keep the customer informed of his consumption habits.

The Attitudes, concerning the emotions, are to provide the report and advices about the energy consumption, in a natural way, gaining empathy with the customer and taking into account the objective of increase the consumption efficiency during the weekends.

Chapter 5

Energy Advisor Development Framework

5.1 Introduction

The software to implement the Energy Advisor has been developed in order to support the proposed paradigms described in the previous chapter, e.g., dynamic navigation through perceptions, dynamic discourse generation, emotional perceptions and question-answering processing.

The software developed is an application framework used to define the Personal Energy Advisor modelling the knowledge domain around energy efficiency. The knowledge domain contains the information required to process advice about energy efficiency.

The framework allows reducing the gaps between design and implementation facilitating development. It simplifies the development of the knowledge domain for the Energy Advisor due to a visual and declarative environment, providing independence of technical development for the designer and an execution environment for the user.

The environment structures the knowledge available based on concepts, procedures and attitudes. Concepts are the knowledge information designed by the designer and defined in the framework, procedures are the steps followed to process the information and attitudes direct the execution of the procedures based on the available knowledge and the user requests.

The framework is based on two interfaces that allow to develop and to execute a natural language tool for queering energy consumption knowledge, (1) the developer interface allows defining the knowledge domain required by the Personal Energy Advisor to provide service, (2) the user interface provides the infrastructure for the execution of the interface of the Personal Energy Advisor providing access to users to perform requests and get personalized responses focused in advice.

The functional architecture is based on off-line and on-line functionalities.

- For the off-line functionality are described the contents of the knowledge domain

implemented using metadata and the interfaces that support the definition of the contents based on perceptions, namely the semantic meaning, computational requirements and lexico grammatical elements.

- For the on-line functionality, the application engine is described detailing the process followed for once given a user request, how input data is retrieved from the database and how information is generated for instantiation in natural language.

The main benefit of the framework is that it allows designers and programmers to minimize the time-to-market, focusing on functional requirements and their design rather than dealing with programming technical details. For the definition of the Personal Energy Advisor a metadata-driven approach for knowledge definition is followed. The development of the avatar is based on iterative approaches delivering to the customer prototypes that are intended to help understanding and acceptance of the system. This allows an easier implementation to reach the customer acceptance for the different prototypes.

In this chapter the objectives and the functional architecture of the software are reviewed, and the interfaces and processes supported by the framework for development and execution are described.

5.2 Objectives

For companies the customer plays a key role in the business model. One of the most common techniques that have been developed to include customers in the business model is Customer Relationship Management (CRM) (21) (5). CRM uses technology to organize, automate and synchronize customer service activities.

Industry has identified that Virtual Assistants can offer a personal touch to increase service levels, but the fact is that most of the Virtual Agents actually available are closer to a string pattern matching for questions and answers than to a more intelligent behaviour able to deal with numeric information and computational reasoning capabilities.

Looking for a more intelligent and effective interaction with the customers to achieve an improvement in customer relationship management, the objective is to design an autonomous Personal Energy Advisor able to interact with each customer and make personal suggestions about their behaviour as an expert would do.

With this approach users would move from a monthly period tariff bill to a personalized advice report generated by the agent as showed on Fig. 5.1. The Personal Energy Advisor would generate useful information extracted from the huge amount of consumption data collected at hourly periods, providing an interactive description of the personalized consumption based on requests, identifying relevant information and reasoning recommendations to improve efficiency and get savings for customers.

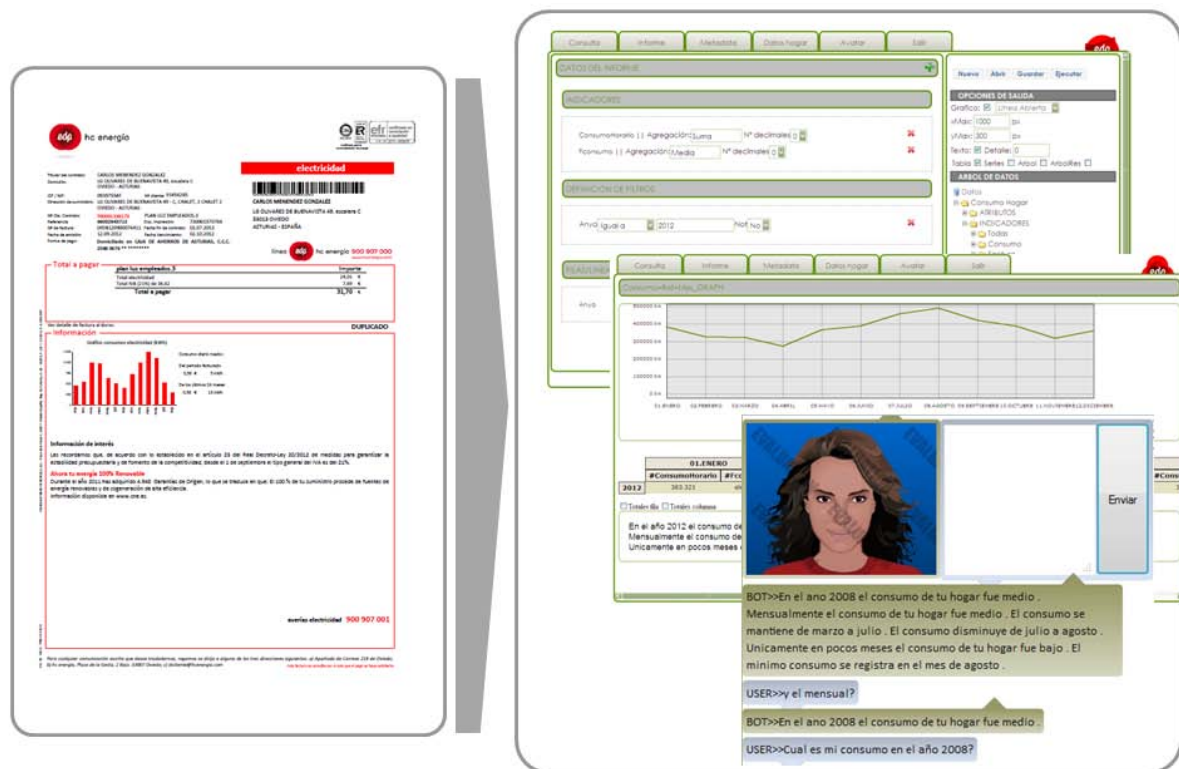


Figure 5.1: Objective of improvement in the communication. From the bill to a closer customer interaction insights.

The objective of the software framework is to allow designers to minimize the time-to-market, dedicating time to meet software functional requirements rather than dealing with the more standard low-level technical details and programming knowledge. As a drawback for developers can be underlined that is required to invest time in learning the framework.

- The integrated framework accelerates the development reducing the required software lifecycle efforts, namely design, development, deployment time and ongoing maintenance.
- The integrated framework minimizes the gap between analysis and the resulting application, as it allows developing applications without technical programming,

5.3 Functional Components

The functional definition of the application framework is represented on Fig. 5.2. The process brings the opportunity to move from traditional reporting tools to a natural lan-

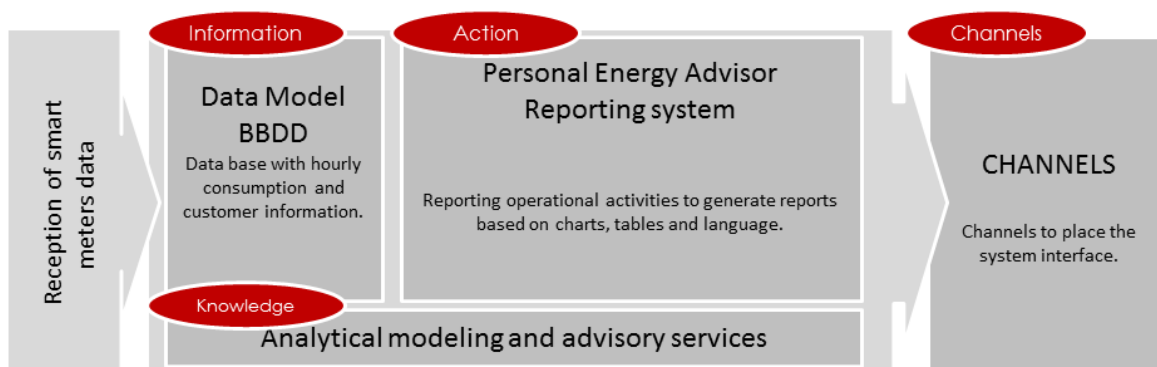


Figure 5.2: Functional definition of the application framework.

guage communication as showed on Fig. 5.3.

The functional components are organized in modules with the principal functions and areas of activity. Functional components set the initial required components to manage and deliver the personal advisor.

Each module is part of the global process, where using information stored and modelled in a database from hourly measures of energy at households and based on the advisor system knowledge, is reported and delivered to customers advice through different channels.

- Data Model (Information)

The data model structures the data that should be available in the database with the corresponding structure. Hourly consumption information is enriched with user data to define the overall model.

- Analytical modelling and advisory services. Knowledge definition (Knowledge)

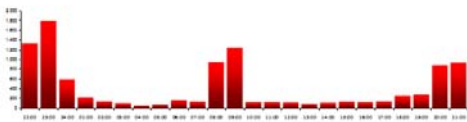
Here the tasks to identify what information is available are developed and how and what information has to be defined in order to be delivered to the customer are set. The information of the energy advisor is identified, translated and modelled into lower level language to be included in the energy advisor. The information and the abilities for computation that would be delivered in the system are defined in this module.

- Personal Energy Advisor Reporting System (Action)

In this module the elements required to read the data from the database, transform the data in useful information expressed in natural language are defined. It can be done due to the knowledge domain defined and data available.

The output of this module is the advice to communicate to the customer. The advice is structured in a report, understood as a set of texts and graphs that describe user

TARIFF	PRICE PERIOD	DAY PERIOD	HOUR	CONSUMPTION
ONE PERIOD TARIFF	P2 REDUCED PRICE PERIOD	MIDNIGHT	22	1.320
			23	1.796
			24	586
		NIGHT	1	217
			2	131
			3	97
			4	45
			5	65
			6	161
			7	129
	MORNING	8	945	
		9	1.243	
		10	116	
	P1 OVERCHARGED PERIOD	MIDMORNIGN	11	145
			12	132
			13	87
		AFTERNOON	14	161
			15	129
			16	171
	EVENING	17	131	
		18	253	
19		275		
20		876		
		21	934	



Your daily consumption is quite high.

To get savings you could change to a tariff with two periods, as most of your consumption is in the reduced price period.

In many of the day periods the consumption is really high, but only in a few the consumption remains low.

The day periods with higher consumption are the evening and the midnight, whether day period with lower consumption is the afternoon

Figure 5.3: Information is transformed from numeric tables and charts, to a natural language description.

behaviour and advice about efficiency to get savings. Contents in the report are focused in maintaining customer attention and therefore in providing information relevant and not repetitive.

- Channels (Channels)

This model takes into account the desired channels to deliver the information. Users can access the contents generated by the Personal Energy Advisor through different channels. Channels to use could be the billing system, a web avatar, personalized printed reports or others. The web is the primary channel to be used in the framework due to the avatar interface, but it could be extended to others, e.g., a telephone customer attention centre to interact directly with the users.

5.4 Functional Architecture

The application framework was developed covering the different conceptual levels of the CPA7L architecture. The framework separates concepts, procedures and attitudes.

The framework works due to the separation definition of on-line and off-line functionalities. The application framework implements these functionalities to define the knowledge domain and to execute user requests. The application framework interfaces available for designer and users are showed on Fig. 5.4.

- The off-line functionality is managed through the interfaces that allow defining the different levels of knowledge used to interpret and generate language texts based

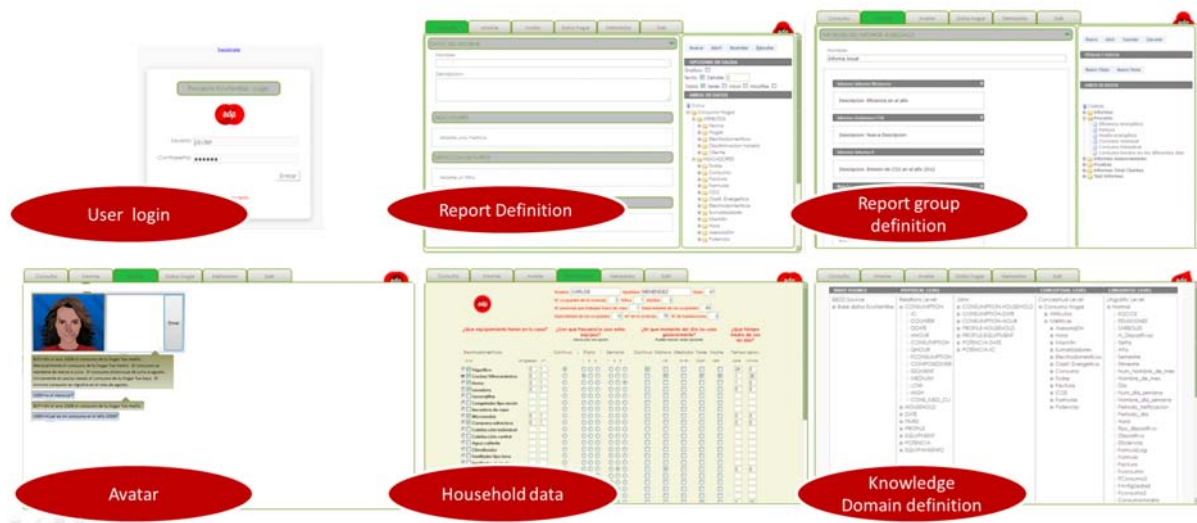


Figure 5.4: Application framework interfaces environments available for designer and users.

on perceptions.

The interface for the designer allows defining the domain knowledge. The designer has access to the metadata definition, where knowledge is defined. A developer interface allows defining the parameters required to include functionality into the framework, basically the data available and linguistic knowledge. For application definition is followed a metadata-driven approach for software designing and development of the final software product.

The designer defined the knowledge domain through the interface showed on Fig. 5.6.

- The on-line functionality is managed by the processes that interpret questions and generate answers in natural language or deliver predefined advice reports as requested by users, e.g., *“how is my consumption? Your consumption is high.”*

The interface for the user allows performing requests to the Personal Energy Advisor. Based on the on-line stage processes and the existing knowledge domain, the system dynamically manages requests and generates the resulting reports in natural language.

The user has access to the interfaces required to retrieve information, the consulting report interface and the avatar interface as showed on Fig. 5.5.

5.5 Designer Interface. Knowledge domain definition

In the application framework, the definition of the information knowledge domain follows a metadata driven approach to facilitate the design and implementation. Developer interface support the designer in the description of the metadata patterns for the application’s knowledge domain independently from technical details.

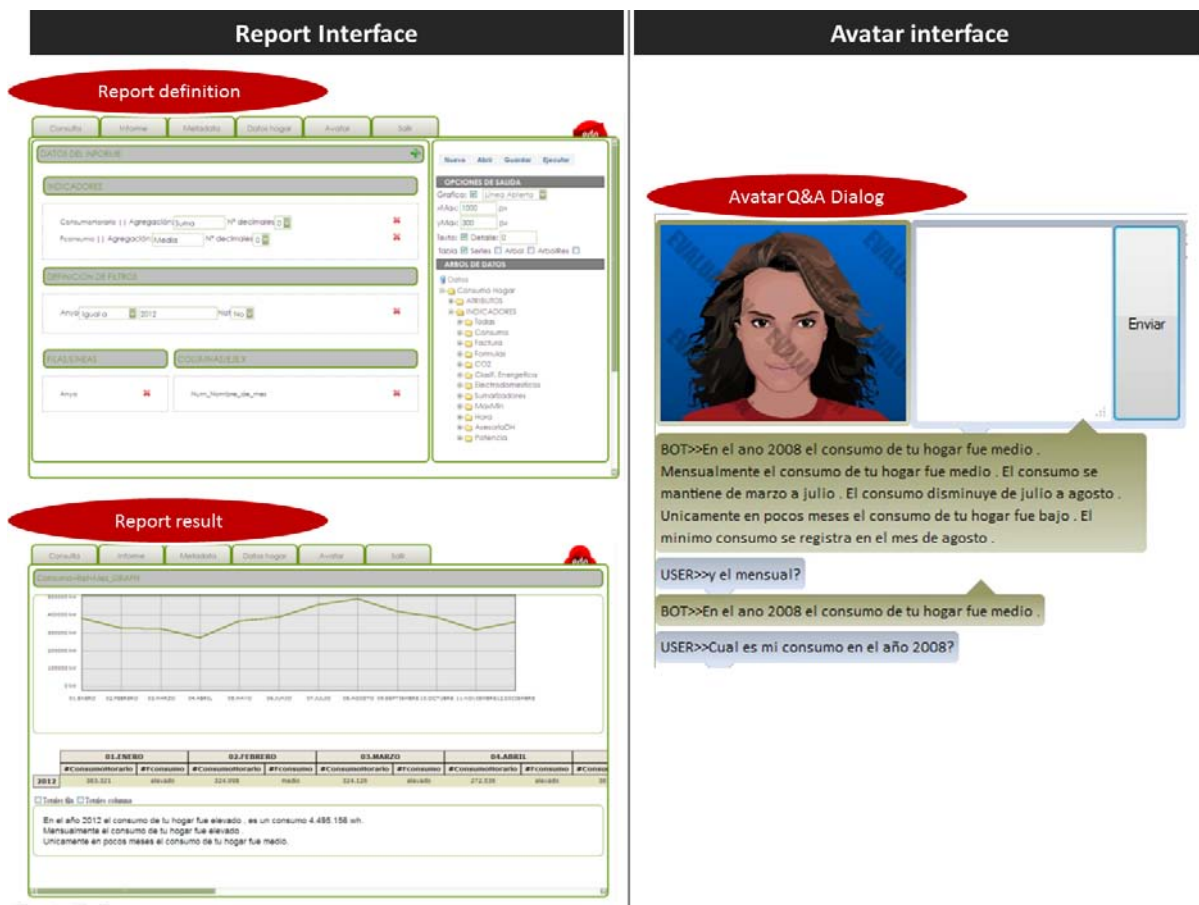


Figure 5.5: Advisory Report and Avatar interfaces for users to request information from the Personal Energy Advisor.

The term “*metadata*” was coined in 1968 by Philip Bagley, in “*Extension of programming language concepts*” (8), it refers to “*data about data*”. Metadata, in the application framework, allows designing a structured information domain with different levels of conceptual design. Metadata, as data, is stored in a database and managed by the application framework. The use of the metadata as a semantic model is a fundamental characteristic in the application framework. If the use of software frameworks is defined by patterns, then metadata is the language used to describe those patterns. The metadata stores the link between source data bases and the definition of concepts managed by the knowledge domain. The metadata describes perceptions based on patterns that describe input data, computations, emotional content and lexical and grammatical issues associated for their instantiation in natural language.

Describing and modelling the patterns of perceptions mappings that the application must employ, more than coding them helps to promote understanding of what features provides the personal energy advisor. The interface defines also a safer, more controlled environment in which the knowledge domain can be dimensionate easily. Also there is an easier visibility of the knowledge defined, facilitating to meet requirements limiting the gap between design and development.

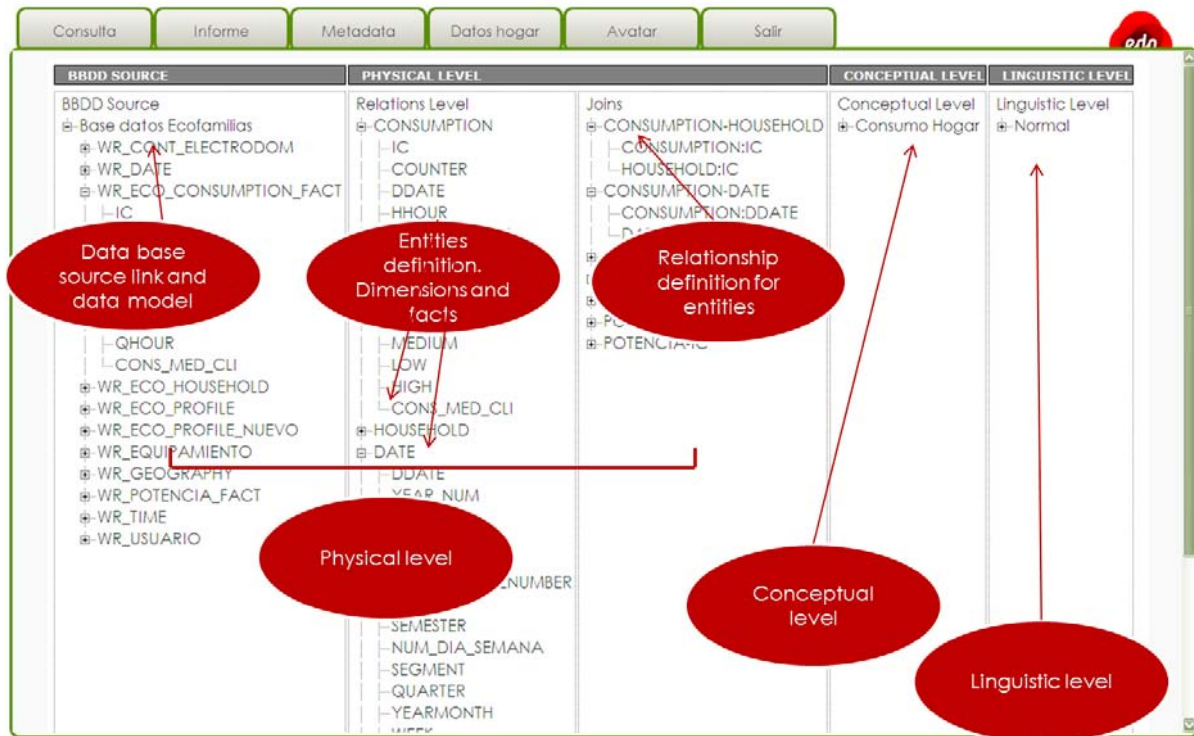


Figure 5.6: Application metadata for the physical layer definition.

5.5.1 Metadata Definition of the knowledge domain

The interface for the metadata definition of the knowledge domain is divided in three layers, namely physical, conceptual and linguistic. Each layer increments the level of abstraction going from a physical database definition to the parameters for language generation using perceptions. It is captured on Fig. 5.6 the view of the overall metadata definition interface.

- Physical level

The Physical layer of the application allows to define the physical data from data sources. Information physically stored in the database is mapped into the framework. The Physical layer defines the objects and relationships available for writing physical queries to retrieve initial data.

The physical layer with the entities and their relationships definition is showed on Fig. 5.6.

Elements from the relational database columns/attributes and tables/entities are selected and imported in a logical data model required for later physical access with structured query language (SQL). Information mapped from the database is modelled due to the relationship definition. Joins between entities are defined in order to allow the application framework to access the data independently from the

query language.

Input data defined in the physical layer follows a star schema that defines relationship between the entities of the multidimensional data model. The star schema defines relations between tables, where the fact tables are placed conceptually in the centre of a star schema and dimensions tables are placed around referencing the fact using the joins. The fact hold the measurable, quantitative data about a phenomenon, and dimensions are used as a set of descriptive hierarchical attributes to contextualize related fact data.

- Fact tables

Fact tables store numeric values as measures of a specific event and foreign keys to dimensional data with descriptive information. Fact tables are designed with a low level detail, so usually a large number of records are stored in fact tables over time. Measures are for example consumption or CO2 emissions.

- Dimension tables

A dimension is a structural set of attributes all of which are of a similar type in the user's perception of the data, e.g., months, quarters and years make up a time dimension.

Dimension tables usually have a relatively small number of records compared to fact tables, but each record may have a large number of attributes to describe the fact data. Dimensions usually define a wide variety of characteristics to describe the facts in a hierarchical structure.

Dimension's attributes are organized following a parent-child relationships, where a parent member represents the consolidation of the members which are its children. The result is a hierarchy, and the parent/child relationships are hierarchical relationships. Dimensions allow moving around different levels of granularity whether facts store events at an atomic level.

- Conceptual level

The conceptual level includes several definitions to model concepts required for perceptions generation, they are the perception mappings. Perceptions are implemented using the metadata configuration interface as showed on Fig. 5.8.

Note that the natural language template associated with perceptions instantiation is defined in the linguistic layer described later.

Concepts or semantic data elements that compose perceptions are defined from the entities, attributes and measures previously defined in the physical data model or over other concepts previously defined at the concept layer. Physical data is transformed into closer understandable references for the user independent from the physical structure.

Note that analysis performed by traditional OLAP systems is numeric oriented and natural meaning is missing, e.g., whether a 2.500kWh/year electricity consumption

BASIC	"Consumption in January is high"
COMPOSED	"Your efficiency is low due to the high consumption"
RECOMENDATION	"Your consumption at night is quite high, changing to a two price periods tariff you could get savings"
MAXIMUM / MINIMUM	"The maximum consumption was in January" "The minimum consumption was in May"
GENERAL	"At the different months the consumption is high"
SUMARIZATION	"Consumption in the different months is high"
SERIE INCREASE DECREASE MAINTAIN	"The consumption increase from September to October" "The consumption decrease from May to June" "The consumption maintains from June to July"
VARIABILITY	"Consumption is change highly from January to February"
GROUPING BASIC PERCEPTIONS	"In January and February consumption was high" "Consumption was high and efficiency low"

Figure 5.7: Discourse Perceptions defined in the application framework.

is high or low. Semantics defined by perceptions are more understandable, as perceptions rather than raw numbers convey the meaning of the data. Here the solution moves from traditional OLAP systems based on numeric data-oriented computing, to a computational perception system based on linguistic knowledge-oriented computing.

Concepts and semantic elements will be available for the user while consulting the information structured in hierarchies, e.g., using a structured set of measures and attributes with labels closer to concepts the user manage more than technical names in the database structure.

An overall description of available perceptions is showed on Fig. 5.7. There is the possibility to define different types of perceptions.

- Crisp Perceptions

They are considered perception generated by perceptions mappings with a crisp aggregation function, e.g. *summary or average*. This perceptions use directly obtained crisp values from the database and therefore there is no linguistic variable assigned.

This kind of perceptions allows retrieving numeric data from the knowledge domain with a linguistic description around the crisp values, e.g., "*Your consumption this year is 2.000.000 Wh*".

- Crisp Composed Perceptions

They are considered perception generated by perceptions mappings with a crisp composed aggregation function, e.g., *Consumption in kWh is equal to the consumption in Wh divided by 1.000*. They allow performing traditional numeric computations to enrich the numeric results obtained from the database. This kind of perceptions allows retrieving numeric data and performing additional computations with a linguistic description around the crisp values, e.g., *“Your consumption this year is 2.000 kWh”*.

- Basic Perceptions

They are traditional described perceptions. An aggregation function is used to define linguistic labels through fuzzy trapezoidal functions. They are used as they reflect quite accurately the linguistic variables and are easily defined. This kind of perceptions allows retrieving linguistic expressions related to crisp values, e.g., *“Your consumption this year is quite high.”*

- Complex perceptions

Complex perceptions are defined using a set of Mamdani-type fuzzy rules $W = g(u_1, u_2)$ as follows :

IF (u_1 is *High*) AND (u_2 is *Low*) THEN *Inefficient*
 IF (u_1 is *Medium*) AND (u_2 is *High*) THEN *Moderately efficient*
 IF (u_1 is *Medium*) AND (u_2 is *High*) THEN *Moderately efficient*
 ...
 IF (u_1 is *Low*) AND (u_2 is *High*) THEN *highly inefficient*

This kind of perceptions allows retrieving linguistic expressions related to other perceptions, e.g., *“Your behavior is inefficient due to the high consumption.”*

- Recommendation perceptions

These perceptions are used as a set of Mamdani-type fuzzy rules as complex perceptions, but instead of generating a new complex perception generates a recommendation for the user.

This kind of perceptions allows retrieving recommendation linguistic expressions related to other perceptions, e.g., *“You could reduce the consumption to get savings.”*

Others perceptions are composed dynamically while generating a discourse. These are the maximum and minimum perceptions, the aggregated perceptions and the variation of perceptions in time. They are used to enrich the knowledge extracted from initial data and language communication.

This kind of perceptions allows retrieving additional summarization linguistic expressions related to other perceptions, e.g., *“Your highest consumption is in the morning.”*

- Linguistic level

The linguistic level defines the lexical and grammatical items to facilitate the instantiation of linguistic sentences from the underlying perceptions. The linguistic

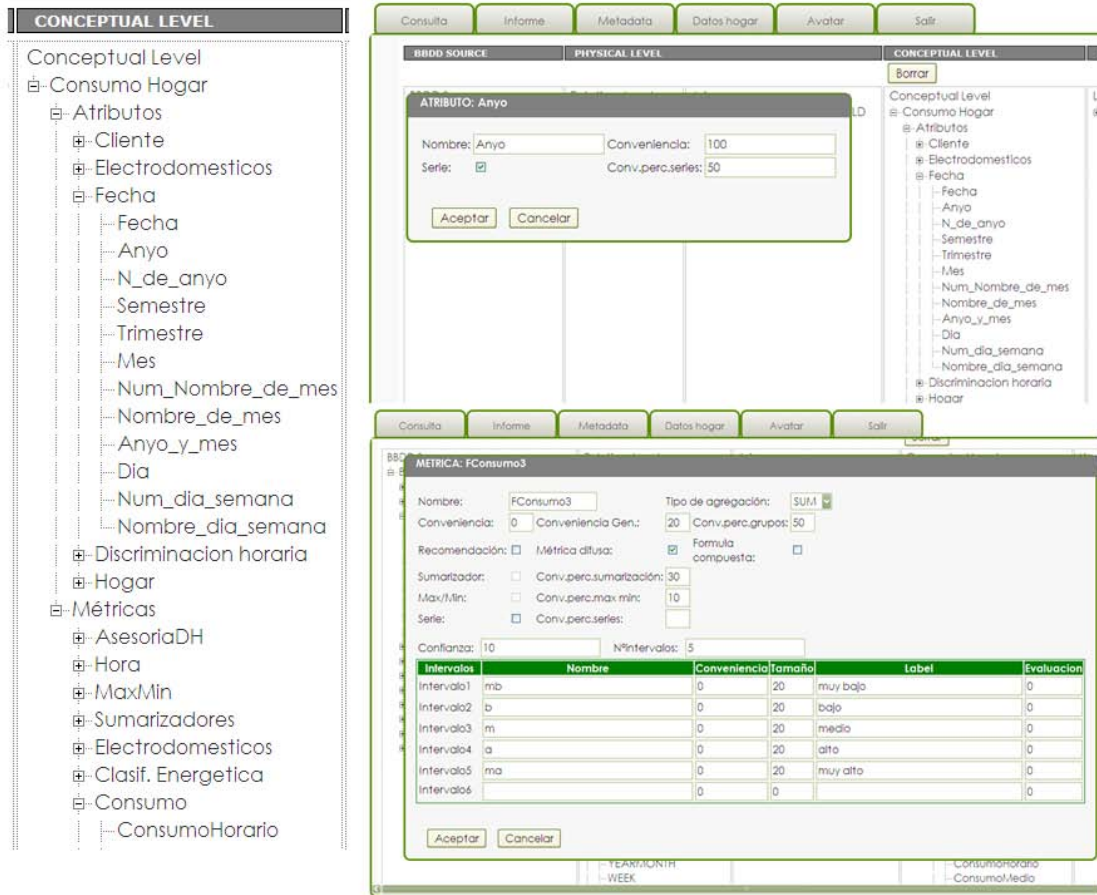


Figure 5.8: Application metadata for the conceptual layer definition.

definition at the perception level includes the verbal forms required and the linguistic labels used to define the context around for instantiation. Emotional linguistic expressions associated to language and contents are defined in here.

Linguistic definition interface is showed on Fig. 5.9.

5.6 User interface to query the knowledge domain

The interaction of the user with the Personal Energy Advisor is performed due to the interfaces for queering the knowledge domain. The application engine is called from the user interface while executing a report or while chatting with the Avatar. While performing queries in the application framework, the user navigates through the knowledge base, extracting in each query a particular subset of data.

The application engine is in charge of giving response to input queries. The engine process the input query identifying the concepts defined in terms of perceptions attributes and perceptions measures following the objective of getting insight over the consumption behaviour.

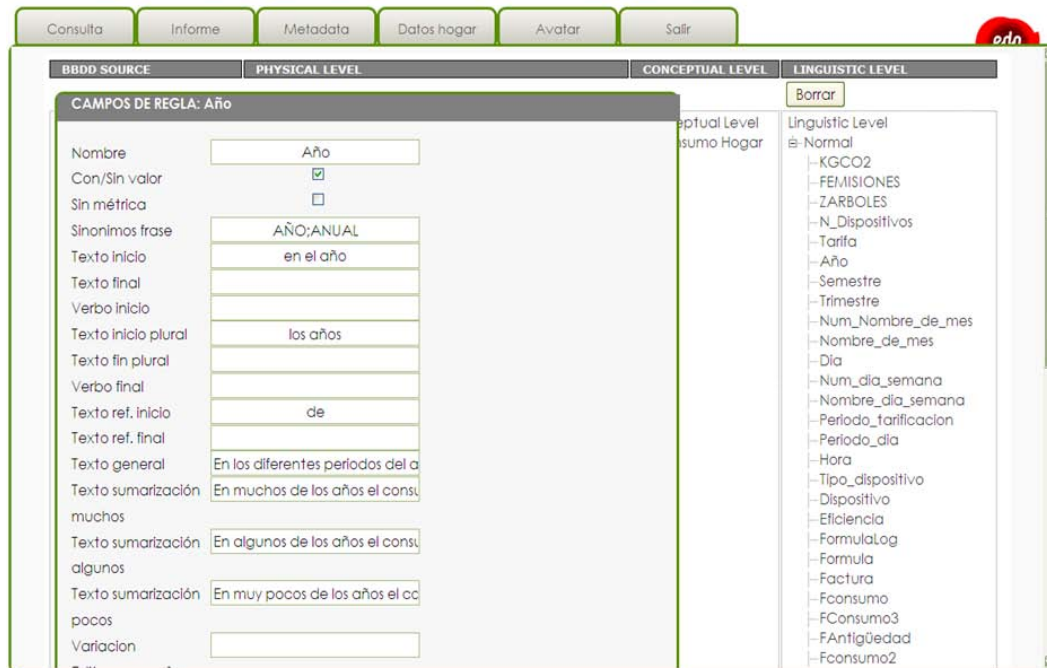


Figure 5.9: Application metadata for the Language layer.

Each query of the user is computed by the interpretation of the request and the computational capabilities processed over the knowledge domain to generate the related responses. From a user request, the process follows the required steps to generate the right answer using the knowledge base and to deliver an output response.

- User request

The interface for the user allows two different ways of retrieving information, a reporting interface and an avatar interface. These environments are showed on Fig. 5.5.

- The reporting interface allows users to define reports by selecting the underlying information dragging and dropping attributes and measures into the report, e.g., *Consumption measure and January filter*. Once executed retrieve from the knowledge domain requested data and process a natural language personalized report as response.
- The Avatar interface allows users to interact using natural language. Here the user can query the system using natural language in order to get the requested responses, e.g., *Tell me my consumption in January.*

- Process

Based on user requests the process generates an output advice report or the response of the avatar. The process follows the theoretical grounds applying the extensions proposed in previous chapters. Based on user request the input data required to



Figure 5.10: Sample report generated by the application framework with contents expressed by charts, tables and natural language that answers user request about monthly consumption.

generate existing first order perceptions is retrieved from the data base. Perceptions are structured in a GLMP with a layout following a granular structure, able to provide knowledge in a coherent discourse. In a second step information is constrained by pruning and avoiding redundancy and aggregating the data providing cohesion to the final discourse used in the report.

- Output response

An output advisory report or answer from the avatar is the information returned for the information requested by the user. To interpret the reports users do not require to have advance analytical skills, so responses generated must be easily understandable by users, maximizing the readability, e.g., for non-experts user, is easier to interpret a personalized natural language report more than numerical reports. Reports contents are organized in topics from general to particular information, e.g., initially describing the efficiency, and afterwards explaining each element that defines efficiency like consumption or power demand.

5.6.1 Report Definition interface

By defining reports, the user navigates through the knowledge base, extracting in each query a subset of data. The main screen for reports definition is showed on Fig. 5.11. The Report definition interface allows users to create their own reports in a controlled, user-friendly environment.

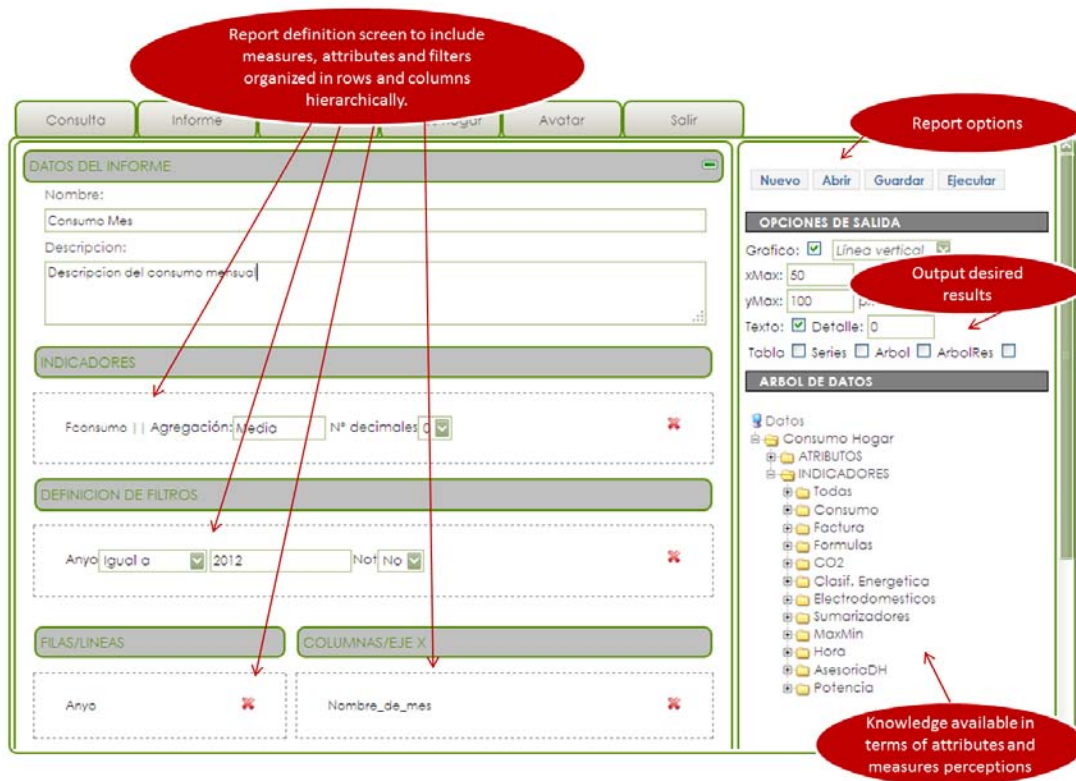


Figure 5.11: Interface for report definition.

A report represents a request for a specific set of formatted data from the knowledge domain. They allow users to gather insight through data analysis. While defining reports it is followed an analytical process where specific selections of attributes, measures and filters defined in the conceptual domain are arranged in order to restrict the set of consumption data retrieved. Examples of selections include efficiency per year or a more detailed evolution per month.

Report definition is performed by dragging and dropping attributes and measures from a data tree to the report layout. Report layout is composed of measures, attributes and filters on measures and attributes. The different components of a report define what information is requested and how it is organized. Once a report is defined, it will contain a layout of attributes, measures and filters that determine what information is required to generate the report.

In the interface the knowledge is available for selection at the left bottom of the screen based on the definitions of the metadata. It is organized in terms of dimensional attributes and groups of measures.

Filtering restricts the amount of data displayed on the report, by controlling the subset of data to be displayed from the data retrieved from the database. Measures define what information is being requested, and dimensional attributes constrains the context

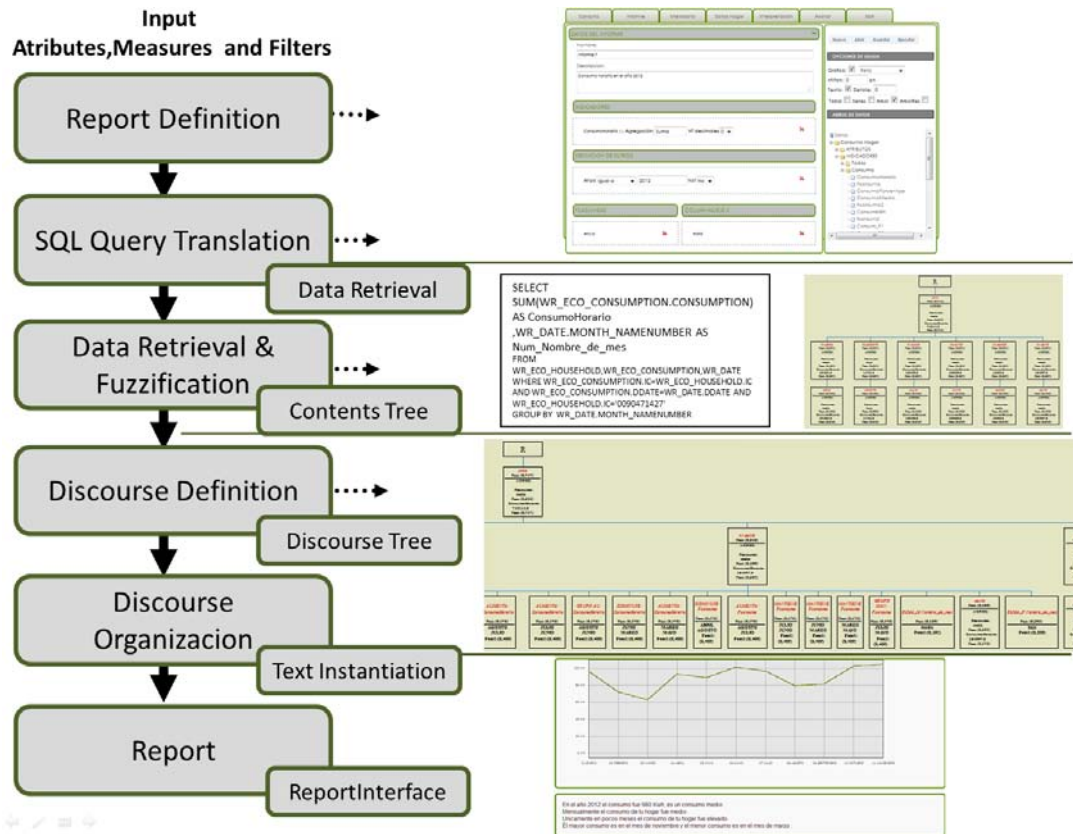


Figure 5.12: Report processing stages, from its definition to the results.

required.

Drilling down or up allows the user to navigate through the different levels of dimension hierarchies from the most aggregated to the most detailed. The drilling paths are defined by the attributes organization in hierarchies within dimensions. For example, when viewing consumption per year, a drill-down operation in the time dimension would then display quarterly consumption and a further drill-down the consumption per month.

Drilling across the user navigates through the attributes of different dimensions changing the context of the measure analyzed. The drilling across paths is defined by attributes of different dimensions. For example, rotating a cube rows and columns of an output table response are changed. When analyzing the consumption per year, a drill across operation would replace the year for the location of the household.

In a report the user defines the output required for the information requested. Output is defined in terms of tables, natural language and charts as showed on Fig. 5.10. Combining the different elements, tables, charts and language is incorporated a higher level of description of data.

- Table

A table represents tabular numeric structured data. It is used to display data in a structured format showing multiple rows and columns of data that cross in cells.

It is the most basic way to organize the data with a predefined structure.

Based on what data the user is requesting, each table may include its own set of fields. Attributes are represented in rows and columns and measures are represented in each cell.

- Charts

Charts are visual representations of the output results. Charts often makes it easier to understand than the data in a table because users can visually extract patterns and trends illustrated in the chart that are otherwise difficult to see.

Charts can usually be read more quickly than the raw data, so are often used to facilitate understanding of data and the relationships between attributes and measures.

Certain types of charts are more useful for presenting a given data set than others, for example trends are usually showed with lines in a line chart, and while comparing a small set of attributes are used slices in a pie chart.

- Language

Language incorporates more information that the data represented in a table or chart, as it express inner thoughts and emotions, making sense of complex thoughts using perceptions.

While using language it is generated a structured discourse with the response where relevant information is incorporated and irrelevant one is excluded.

- Avatar

A face-to-face Avatar incorporates information by using face signals or gestures and due to voice intonation. It also promotes empathy for a closer relation with the user.

Reports can be stored in the metadata to be able to define predefined reports and retrieve them. Also a set of individual reports can be organized in a report set, that allows including different reports in a global definition.

5.6.2 Report Execution

Report execution is initiated from the report definition interface by the execution option. The execution option launches the report execution based on the information associated to it. Execution initiates the process required to extract the data, generates an initial granular structure in a content basic tree and extends and organizes initial contents in a discourse. In the discourse tree, composed measures are calculated on-the-fly with the data available and instantiated in a report output dynamically.

The steps followed are showed on Fig. 5.12 and described as follows.:

- SQL Query translation

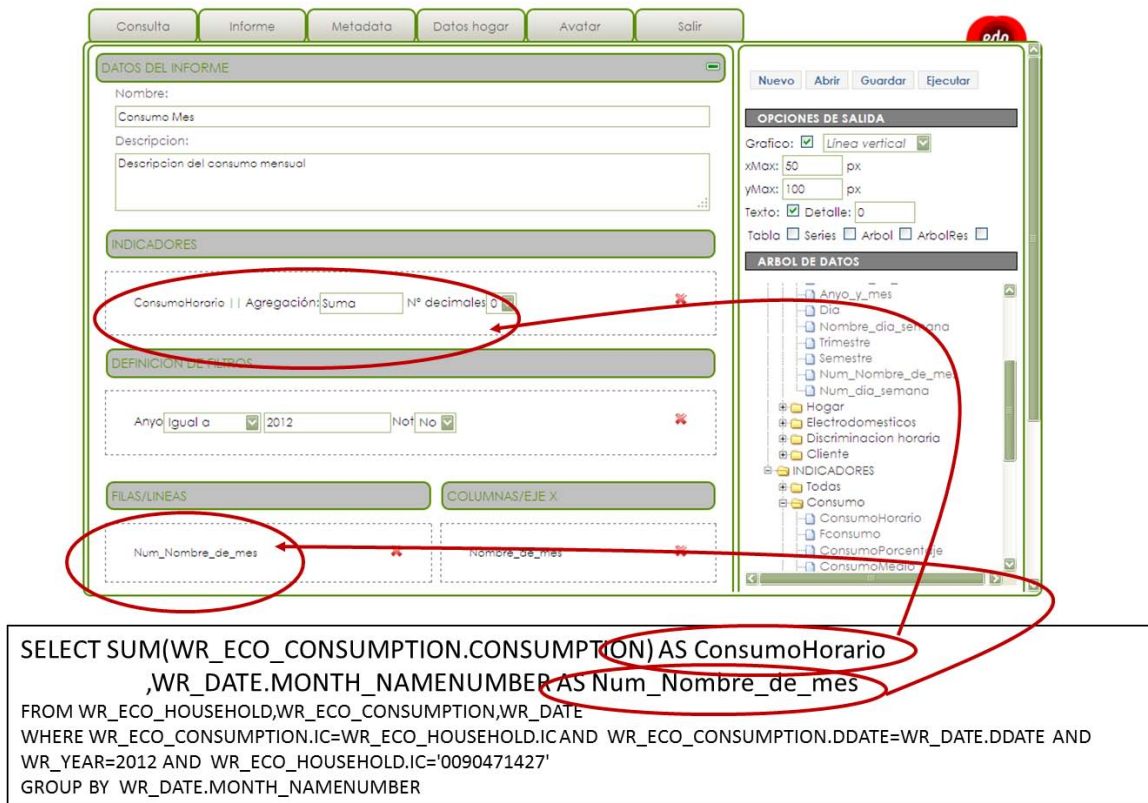


Figure 5.13: Report definition and the execution generating a SQL to access the database.

The first step is to prepare the access to the data base to retrieve the input data as showed on Fig. 5.13. Each attribute, measure and filters associated defined in the report are translated into a SQL statement for execution over the database to obtain input raw data.

While executing a report information is consolidated. Consolidation involves computing all of the data relationships for one or more dimensions and the related fact data, for example, adding up all hourly consumption to get the consumption per month. While such relationships are normally summations, any type of computational operation might be defined for each perception mapping input value.

A single measure perception is defined by the intersection defined by selecting one member from each dimension and a fact measure. For example, with the dimensions time and hour if attributes selected are date and day period, and the fact measure is consumption, then Consumption at 2nd January 2012 at 5:00h specify a precise intersection along all dimensions that uniquely identifies a single data cell, which contains the value of the consumption at 2nd January 2012 at 5:00h.

- Data retrieval and fuzzyfication

Numeric data is retrieved from the data base and transformed due to computa-

tional requirements, e.g., translating numeric values into linguistic statements for fuzzyfication.

Information extracted is represented in the form of a granular linguistic model. Initial data is structured in perceptions that define each node of the tree. In this tree, concepts are represented language independently, and each node is augmented with semantic, grammatical and pragmatic information based on the information defined in the perceptions mappings of the knowledge domain.

While generating the content tree, information about the relevance and evaluation information is added as described in previous chapter and showed on Fig. 5.14.

Each perception has associated a fuzzy value, a linguistic expression and a set of weights:

- Weight: Is the fuzzy weight assigned to the principal linguistic value assigned. Determines the degree to which a sentence is true as a basic criteria.
- Attribute Convenience: Is the convenience or relevancy degree of importance of the attribute or measure associated to the perception.
- Value Convenience: Is the convenience or relevancy degree of importance of the value associated to the attribute or measure.
- Level degree: Is the level of the perception in the tree layout.
- Covering degree: Is the coverage used to assign more importance to a summary than to individual perceptions and by identifying summaries with a low covering, avoiding local behaviour without representativeness.
- Perception degree: is the overall suitability of the perception. The degree of relevancy determines the importance of the sentence defined in the semantic context of the requested report.
- Evaluation: Each perception has an evaluation degree, in first order perceptions it is assigned by the evaluation value defined in the knowledge domain, in second order perceptions it is computed by an aggregation function.

- Discourse Definition

In the discourse definition step a tree is built based on input data retrieved from the database and the logical semantic definition as showed on Fig. 5.14. This first tree of perceptions is called *content tree*, as it reflects initial basic perceptions. The tree is organized in levels as defined by the hierarchy dimensions, from the most aggregated information at the top to the details at the bottom.

- Discourse Organization

In the discourse organization step the discourse tree is built by organizing the content tree and adding related second order perceptions. Note that the discourse organization in layers will be used to order the resulting text layout by instantiation as showed on Fig. 5.15.

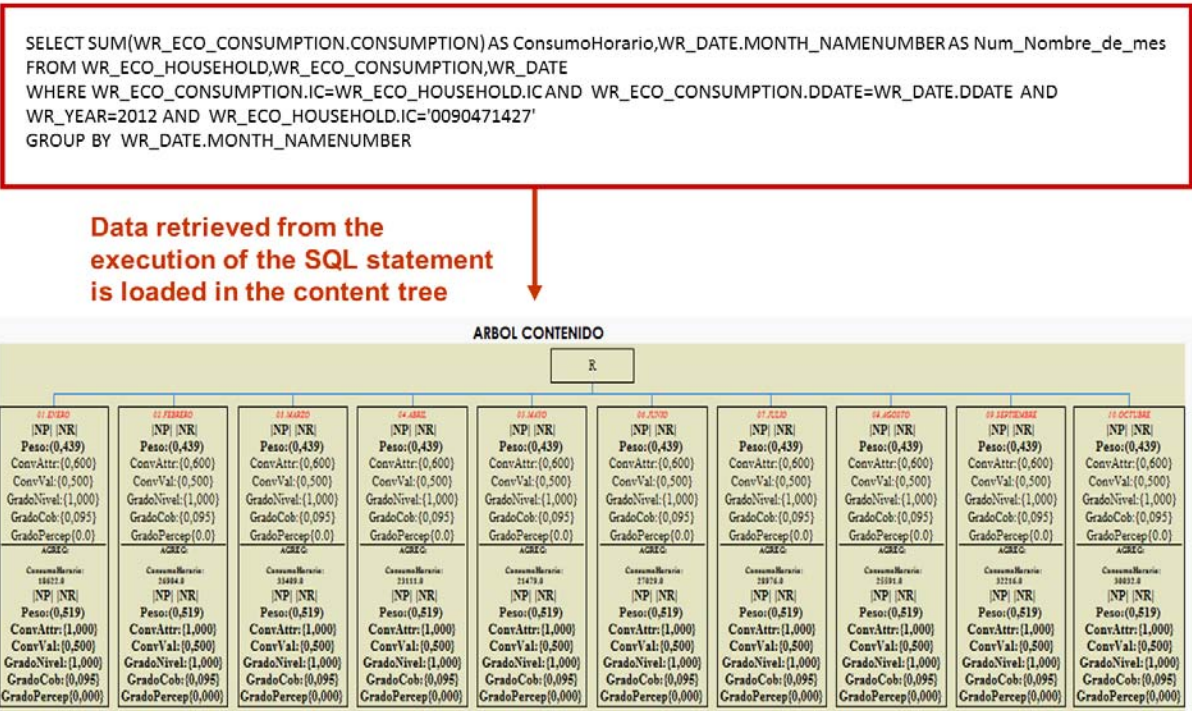


Figure 5.14: SQL for data retrieval and report tree generation.

The layout is instantiated in a tree structure that provides integration between discourse level (the overall tree) and clause level descriptions (perceptions of the tree). It generates a granular ordered structure. The hierarchical structure in the GLM allows inferring formal relations between sentences providing cohesion as it is spread from bottom perceptions till top-level perceptions.

In the discourse tree new perceptions are extracted from existing perceptions defined in the *content tree*. Each perception is added to the discourse tree as showed on Fig. 5.16. Second order perceptions as described earlier include the computation of aggregated perceptions like max and min perceptions, summarization perceptions and series description perceptions in order to enrich communication, e.g., *The maximum consumption is in January.* determines that between all the months the maximum consumption is in the month January.

A second step in perceptions generation includes the computation of aggregation. Aggregation provides, in the instantiation, cohesion to linguistic expressions. It merges linguistic expressions of different CPs if they can be expressed in a complex sentence. While aggregating, relations between perceptions sentences are computed by similarity, and if applies, clustered together using a connective. The choice of the connective depends on the relations that hold between perceptions and their semantics, e.g. *“The Winter and Autumn consumption is really high.”*.

Relevance will be used while pruning, in the instantiation of the discourse tree, as it defines what information to communicate in order to provide relevant linguis-

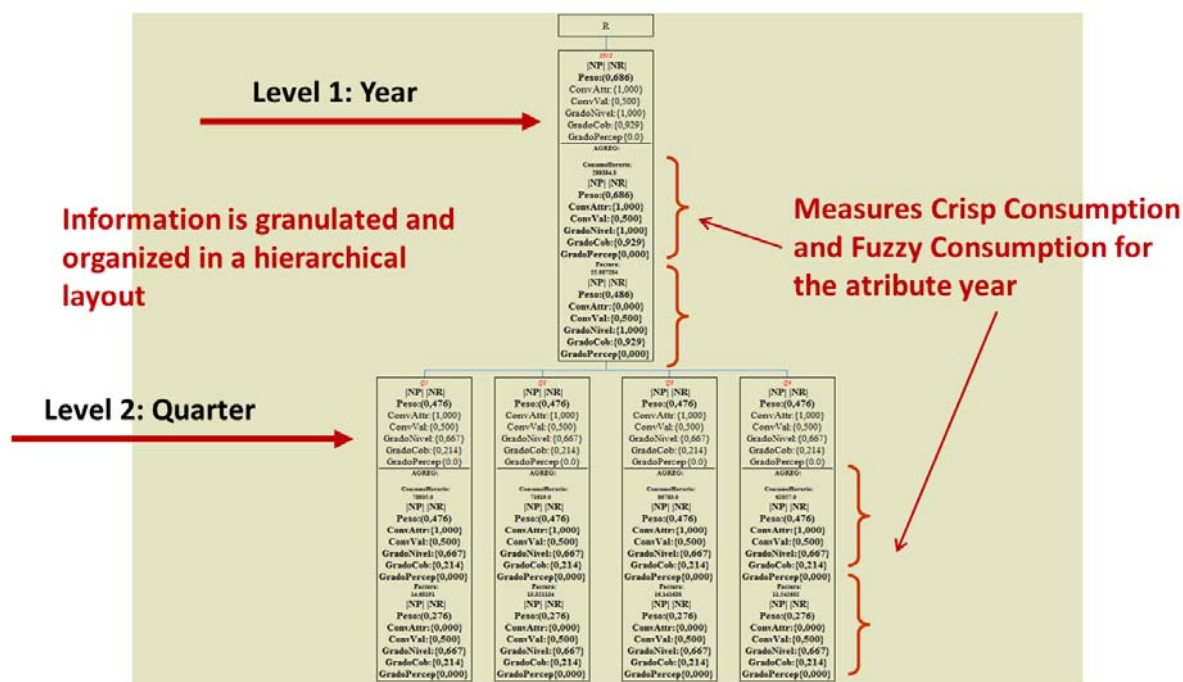


Figure 5.15: Discourse organization layout.

tic descriptions. While pruning a relevancy weight for each sentence is analyzed excluding irrelevant ones as showed on Fig. 5.17. A second pruning is performed following a top-down path, when parent perceptions are explained by the children ones, and therefore additional meaning is not deployed but redundancy e.g., As showed on Fig 5.17 the sentence “In January the consumption was high” is pruned because it is already expressed in a parent perception. Reference relations are used to avoid repetition or redundancy. Each node of the tree is marked as redundant is the semantic contents are already expressed in a parent or brother perception.

Aggregation provides, in the instantiation, cohesion to linguistic expressions. It merges linguistic expressions of different perceptions if they can be expressed in a complex sentence. While aggregating, relations between perceptions sentences are computed by similarity, and if applies, clustered together using a connective. The choice of the connective depends on the relations that hold between perceptions and their semantics, e.g., *additive* “and” or *adversative* “but” conjunctions.

Each emotional state is applied to individual perceptions; when the efficiency of the hearer has a good behaviour emotional state of the perception improves, or when the efficiency behaviour is under performance the emotional state deteriorates. Based on the evaluation assigned to each perception, the instantiation process will compute the related emotional state to generate the final linguistic expression.

- Response generation

Last step is the instantiation of perceptions in natural language, it is performed

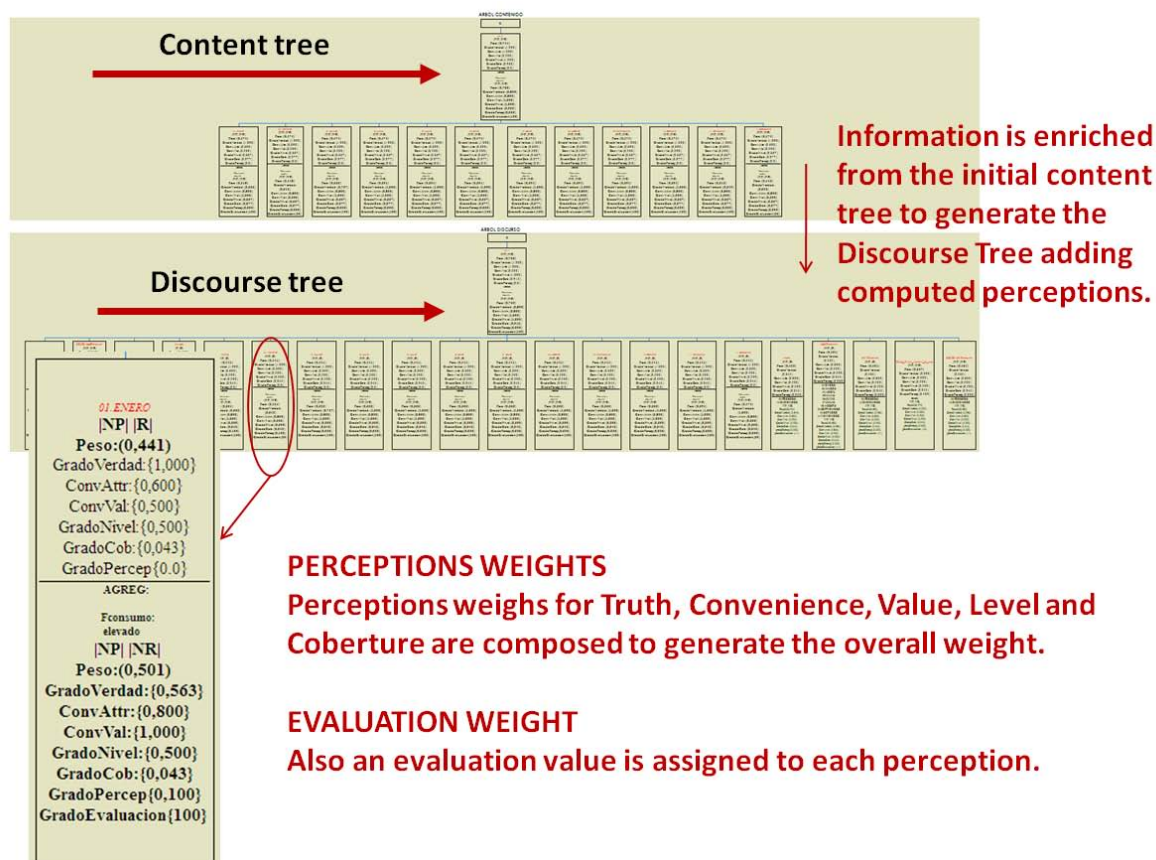


Figure 5.16: Discourse enrichment with complex perceptions.

through the realization process as showed on Fig. 5.18. The realization or instantiation process will generate the resulting text of each perception active in the discourse tree.

The report result will be a personalized information for the user, where each perception is computed and personalized for each user as showed on Fig. 5.19.

Instantiation of each perception generates a text sentence based on measures (verb) and a context (attributes) associated. The measure is instantiated based on the verb, the formal tense and person. Context attributes are instantiated based on the singular or plural form.

The generation process is able to generate the constituent words in the target language, arrange perceptions grammatically and make the necessary changes to the stems of the words.

5.6.3 Avatar interface

The Avatar interface implements a computer animated character and a messenger chat dialog. The chat messages show the ongoing dialog with the history of messages of the user and the Avatar. The character facilitates a closer relation in a face to face communication, and allows developing a human-like online assistant.

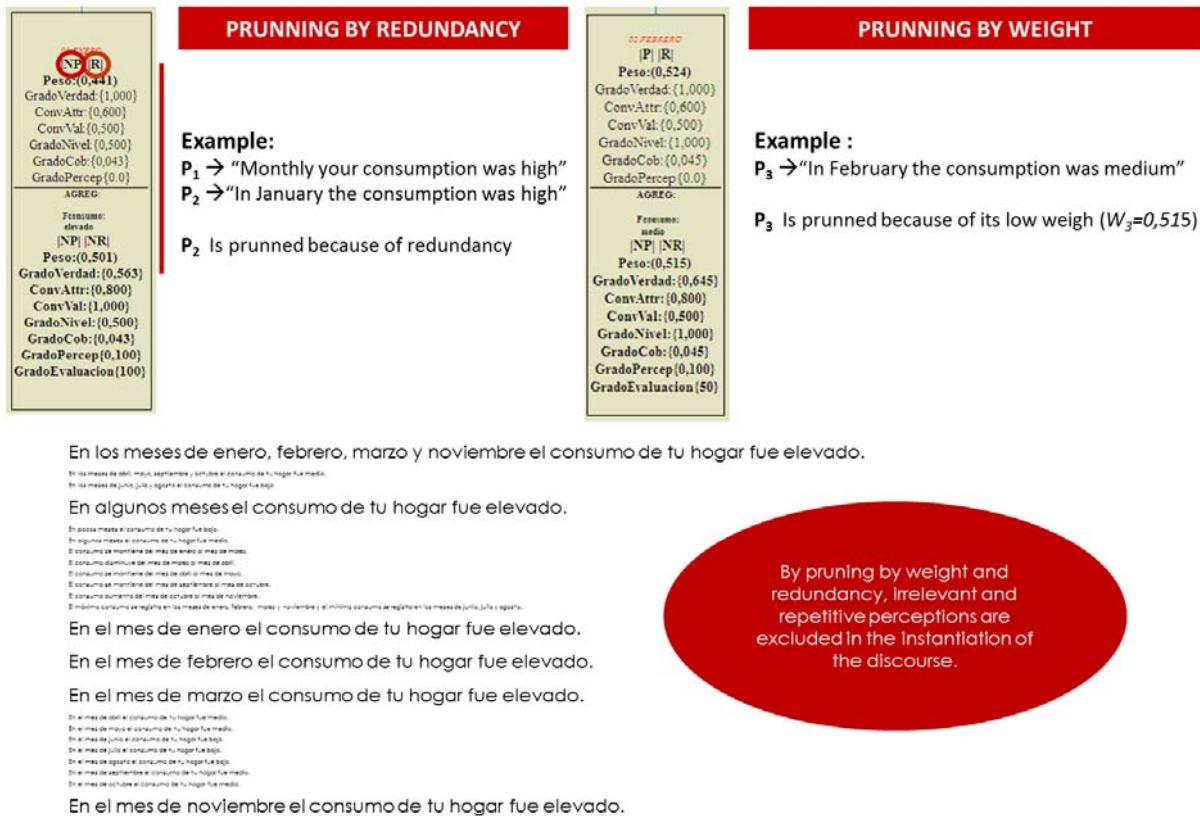


Figure 5.17: Discourse pruning over perceptions.

While performing queries in natural language to the avatar, the user navigates through the knowledge base, extracting in each query a subset of data expressed in natural language. It is possible due to the definition of semantic enrichment to the attributes and measures defined in the knowledge base.

The process followed by the avatar to get responses is based on the reporting capabilities described earlier in the report definition and execution section.

The avatar interface, to interpret the user question, incorporates the ability for natural language interpretation. A query is translated into a set of attributes, measures and filters as in a report definition. Basically in this process, user queries are translated in a report definition for its execution.

- Tokenization

To perform the tasks of natural language understanding the input question is tokenized to decompose the constituents of the input text.

- Tagging

Each of these tokens is tagged, using the knowledge domain, in order to identify the semantic meaning. Each tag identifies the relation with the attributes, measures

MAX el consumo es Wh Los consumos son el máximo consumo se registra el mínimo consumo se registra Sinónimos TextDescrip	1SEM En el los semestres En los diferentes semestres Sinónimos TextDescrip	MIN el consumo es Wh Los consumos son el máximo consumo se registra el mínimo consumo se registra Sinónimos TextDescrip	2SEM En el los semestres En los diferentes semestres Sinónimos TextDescrip	1SEM En el los semestres En los diferentes semestres Sinónimos TextDescrip	651920.0 el consumo es Wh Los consumos son el máximo consumo se registra el mínimo consumo se registra Sinónimos TextDescrip	2SEM En el los semestres En los diferentes semestres Sinónimos TextDescrip	427401.0 el consumo es Wh Los consumos son el máximo consumo se registra el mínimo consumo se registra Sinónimos TextDescrip
---	--	---	--	--	--	--	--

El máximo consumo se registra en el primer semestre y el mínimo consumo se registra en el segundo semestre .
 En el primer semestre el consumo es 651.920 wh.
 En el segundo semestre el consumo es 427.401 wh

Figure 5.18: Discourse tree instantiation in natural language.

and filters identified in the knowledge domain. A semantic enrichment is processed at this stage, using semantic synonyms of expressions related to each element and previously defined in the knowledge domain.

- Semantics

Once the tokens are tagged and the semantic meaning is assigned, input text is transformed to a perception for evaluation in the reporting process described in previous section.

- Emotions

Emotions are extracted from the input interpreting tokens with negative content and others with positive one expressed by the user.

- Perceptions

Perceptions are the overall definition of language in a sentence, and apply constraints to input linguistic labels.

- Execution

While speaking with the Personal Energy Advisor, the multidimensional capabilities implemented are used by the avatar to generate responses based on user questions in a dialog based environment remembering the conversation maintained in the past, e.g. *“What is my annual consumption ? ... And monthly?”*.

Drilling down or up allows the user to navigate through the different levels of dimension hierarchies from the most aggregated to the most detailed.

Drilling across the user navigates through the attributes of different dimensions changing the context of the measure analyzed.

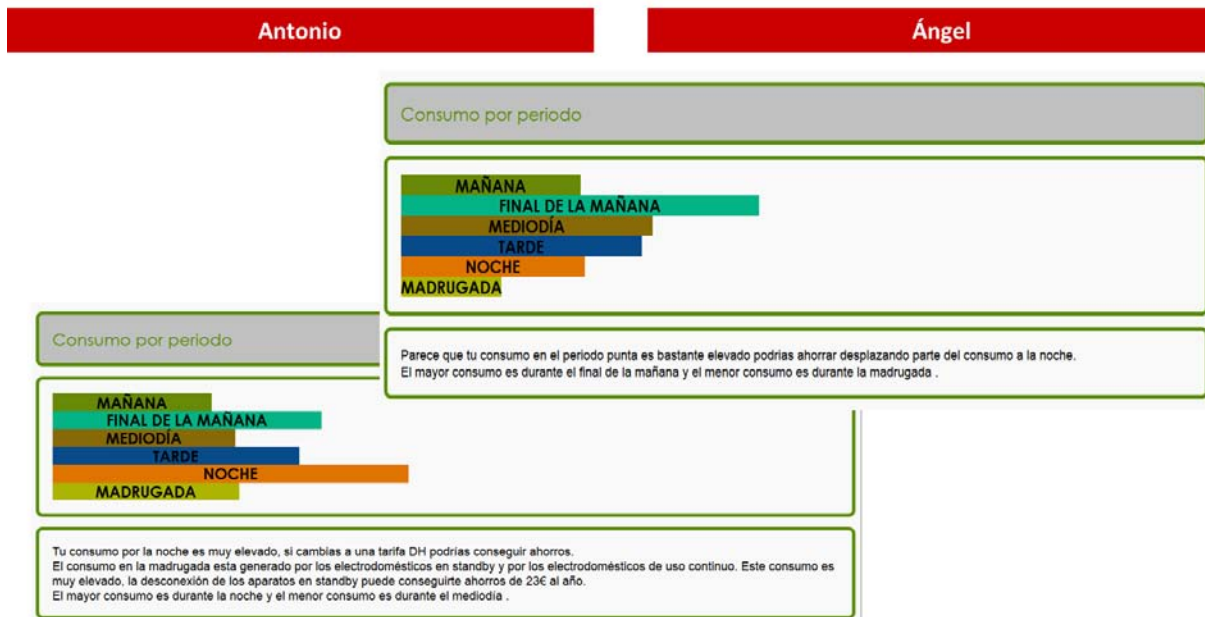


Figure 5.19: Report results example. Consumption evolution per day period for different users.

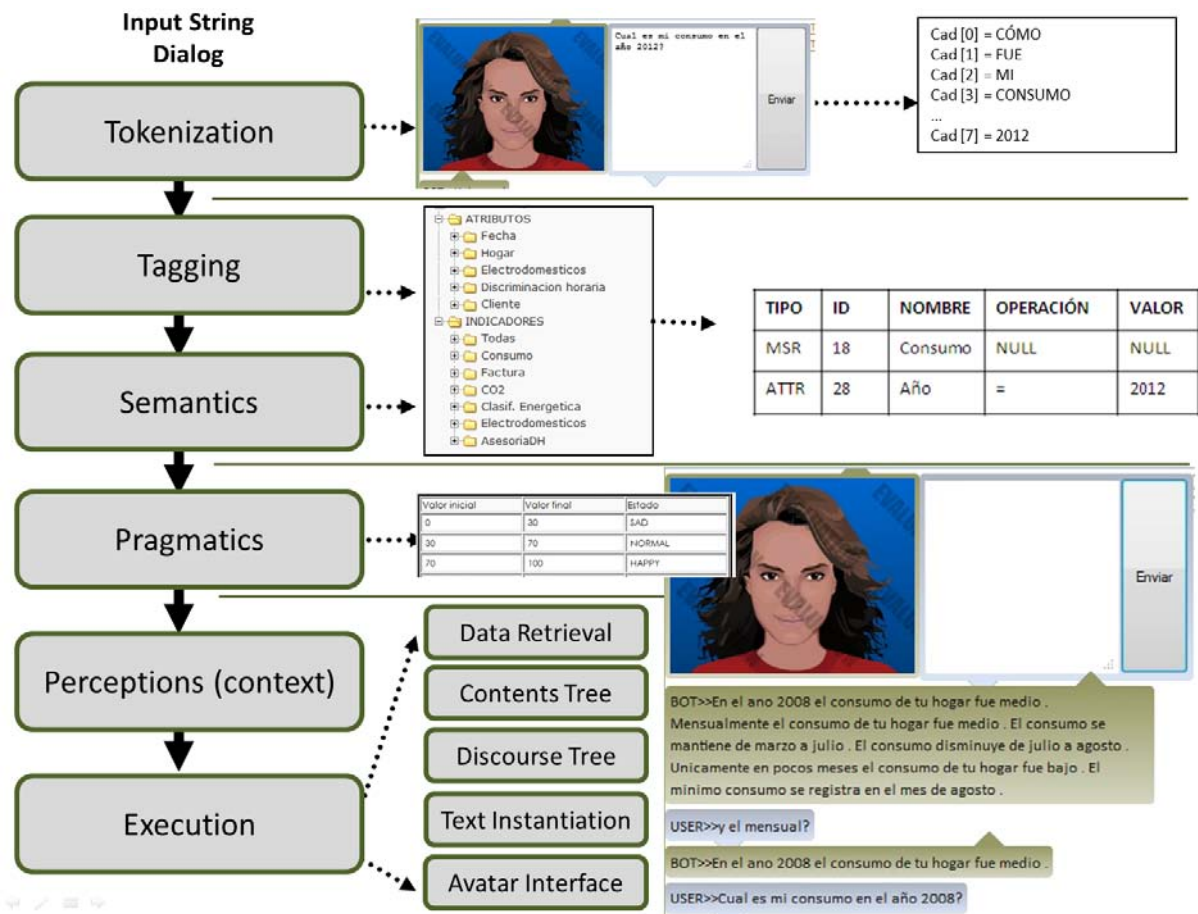


Figure 5.20: QA Process, from the user natural language request, to the answer.

Chapter 6

Energy Advisor Development Methodology

6.1 Introduction

Formal methods in software engineering are an increasingly important application area for intelligent systems. Research environments often produce prototypes that demonstrate the feasibility of a method and leave efficient and more complete implementation to industrial exploitation.

Software projects, in order to meet requirements from customers in time, are managed using a methodology to formalize the engineering process. Software Engineering deals with the application of a systematic, disciplined, quantifiable approach to the design, development, operation, and maintenance of software (2).

In Natural Language Generation (NLG), research results from software engineering and software technology are limited. While generating language for communication, human being performs a set of complex task. These tasks have to be designed and automated in software systems, to extract relevant information from the environment and describe it in Natural Language.

This chapter describes a proposed methodology developed upon previous research to build natural language interactive systems. The methodology allows developing computational systems able to generate linguistic descriptions of input data with emotional content using perceptions. In the methodology are defined the steps and deliverables that should be followed in an industrial implementation based on the framework defined in previous chapter. Three research areas are linked together: (1) Natural language generation (NLG) processes to generate understandable and useful information to users, (2) Knowledge discovery in databases (KDD) that describe the process of discovering useful knowledge from a collection of data obtained from the environment and (3) software methodologies for application design, development and maintenance.

The rest of the chapter is organized in different sections. Initially there is an introduction of methodologies and processes required for natural language generation, and later

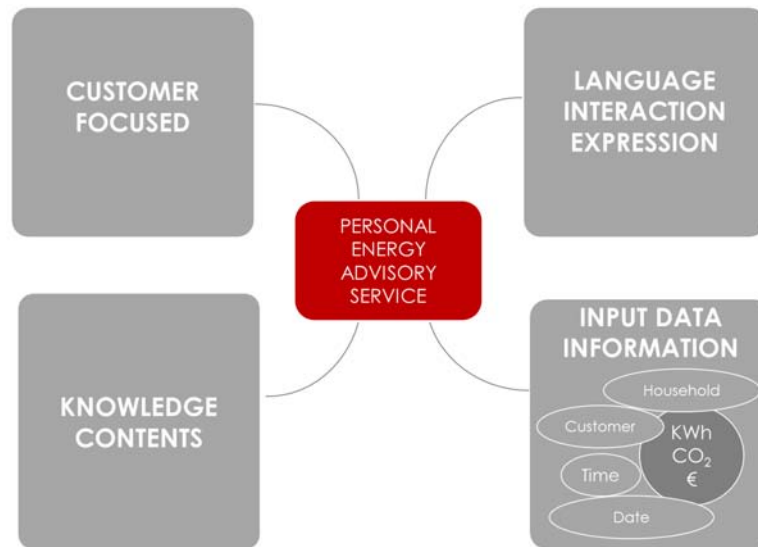


Figure 6.1: Four key principles for the definition of the Personal Energy Advisor.

a review of the general characteristics and processes involved in existing methodologies for knowledge discovery in data bases and software development.

Later, based on previous existing methodologies, is described a methodology for the development of systems with the objective of generating linguistic descriptions of data using the development framework proposed.

Also initial steps in the application of the methodology for the development of the Personal Energy Advisor are presented. In the next two chapters are described in detail the steps and applied stages of the methodology, the analytical modelling and the software development activities.

6.1.1 Context Definition for the application

The definition of contents for the energy advisor is based on key four principles described on Fig. 6.1. Here is defined at high level the context of development in order to meet the final requirements for the user.

- Customer

Customer is the focus point of the application, as the advisory services are, effectively a service for the customer. All information used and defined in the application has a direct result in the customer, the receptor of the information.

- Input data information

The information is the base data required and available that should be stored and accessible. Information available from different databases is integrated in a common data model design that specify the grounds of the application knowledge.

- Language Interaction

The language is used in the process of interpreting questions and providing responses, and should meet customer expectations in order to provide a dialog interaction easily understandable. Language should be significant in order to awake customer attention on those key aspects where actions can be taken to improve the efficiency.

- Knowledge Contents

Knowledge contents are expressed in a report as the tangible asset to deliver to the customer. The report structures the advice in a deliverable for the customer. Report contents are based on charts, statistical tables, and language.

The description of an specific structure of a report is defined in order to meet customer expectations covering the phenomenon. The report objective is to be similar as much as possible to the one that an expert should deliver to the customer, so the application has to be intelligent enough to simulate this behaviour. The report is conceptually defined as a list of sub reports going from the most general concepts to a more detailed description in a well defined structure.

6.2 Methodologies in Natural Language Generation

Computational systems are able to store huge amounts of data about the phenomenons in our environment and in some way incorporate the ability to extract relevant knowledge from these data and express it in understandable language showing human-like behaviour.

In the use of natural language specific communicative goals are followed and a set of complex tasks to go from some available knowledge to a sequence of meaningful sentences are performed. These tasks have to be automated in systems designed to extract relevant information from data to describe complex phenomena in natural language.

There are little references in the area of software engineering and natural language generation covering the description of a phenomenon. The NLG process has been decomposed as a pipeline of tasks. As described by Reiter and Dale (102) and Hovy (61) (60).

In Natural Language Generation, research results and applications proposed, have not given too much detail about the methodology to follow. Even when there are factors that add complexity to the task of NLG systems beyond conventional software systems (102) (75).

6.3 Software Development and Knowledge Discovery Methodologies

A methodology is a tool to manage projects, specifically to manage that they are delivered on time and on budget, meeting customer's requirements, maximizing resources and

minimizing risks. Software projects require the reduction of time to market and manage pressures for improved quality. In many cases the failure on a project is the result of either not using a methodology, not using the right one or not using it correctly (26).

6.3.1 Software Development Methodologies and Unified Process

Software development methodologies like Waterfall, RUP and Agile have all become key tools for software developers and project managers (38) (16).

During the 1960s, “*code and fix*” was the method employed by software developers. Due to the difficult nature of “*code and fix*” approach, Winston Royce in 1970 proposed the waterfall methodology to deal with the increasing complexity of aerospace software. The waterfall approach emphasizes a structured progression between defined phases. Each phase consists of a definite set of activities and deliverables that must be accomplished before the following phase can begin. These methods for system development are structured in standard, well-defined processes, e.g., requirements/design/build paradigm.

Methodology trends enhance the intrinsic limitations of waterfall models, considering more agility to adapt to change, reducing time to deliver and minimizing risks. They are not focus on a long development cycle but rather on short iterations and rely on close customer involvement (52) (13) using successive refinements from abstract specifications into concrete ones, allowing software to evolve, not be produced in one huge effort (65) (69).

As described by Booch, Rumbaugh and Jacobson in “*The Unified Software Development Process*” (65), the Unified Process (UP) is a use-case-driven, architecture-centric, iterative and incremental development process framework.

Here is presented a review of the main UP characteristics and the UP process in terms of phases and workflows.

UP Main characteristics

Up main characteristics are that it is iterative and incremental, use case driven and architecture-centric.

- a) Iterative and incremental because an iterative approach allows an increasing understanding of the software through successive refinements, and to incrementally grow an effective solution through multiple iterations.

An iterative approach addresses risk along stages in the lifecycle through executable releases that enable continuous end user involvement and feedback. Development team stays focused on producing results and ensure that the project stays on schedule.

- b) Use case driven because use cases drive iterations, bind workflows together and synchronize the content of different models. Use cases drive the development activities

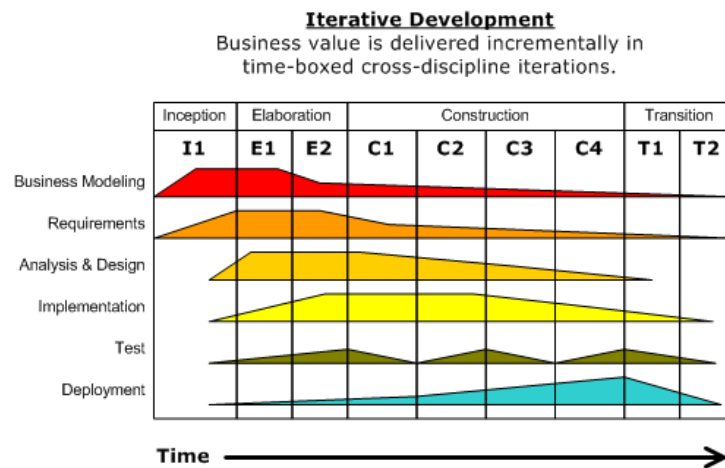


Figure 6.2: Steps in UP

for the creation and validation of the system. They also drive the definition of test cases and procedures, planning of iterations and the creation of user documentation.

Whether use case specifies function, architecture specifies form.

- c) Architecture centric because uses a systematic approach to define an architecture using new and existing components. These are assembled in a well-defined architecture.

The UP Process

The UP process can be described by two dimensions, or along two axis as showed on Fig. 6.2. The first dimension represents time and shows the lifecycle aspects of the process. It is expressed in terms of phases, iterations, and milestones. The second dimension represents the static aspect of the process, how it is described in terms of process components, activities and workflows. The graph shows how the emphasis varies over time. For example, in early iterations, spend more time on requirements, and in later iterations, is spent more time on implementation.

The phases in UP process are Inception, Elaboration, Construction and Transition. Each phase can be further decomposed into iterations. An iteration is a complete development loop resulting in a release of a subset of the final product, which grows incrementally from iteration to iteration to become the final system.

- a) Inception: Justify the project and define the business case and its scope. Based on requirements initial use cases are developed, identifying the most important functionalities of the system and risks associated. Functionality is described at high level identifying external actors. In these first steps, an initial skeleton of a draft architecture is also created and the first prototypes are delivered. The project plan is defined in a first attempt, including phases and iterations.

- b) **Elaboration:** The use case requirements are detailed and extended in this phase covering almost the functionality, and the system architecture is validated and designed.

During this phase, the known risks factors are addressed. The implementation of a basic core system is just intended to demonstrate that the project is feasible mitigating most technical and architecture risks. It will be improved and completed later on, in the following iterations but at this stage a first prototype is delivered to users. At the end of this phase there should be an accurate plan (costs and schedule) for the development of the project.

- c) **Construction:** Remaining components and application features are developed and added to the system that is built by adding functionality in small and short-time iterations. At the end of this phase is built the first complete software product. Whether the software engineering effort has taken place in previous steps, construction could be understood as a manufacturing process.

The architecture of the system is stable in this phase, although some minor improvements could be applied.

Each iteration of this process will produce an internal release of the system that will be deployed and tested within the development environment. At the end of this phase, all the use cases and user requirements should be covered.

- d) **Transition:** It is the transition of the product to the users. Here it is decided if the software can be operational without risk, generating a beta release. A deployment and post-deployment support plan is defined.

The users are trained and interact with the system. Feedback provided is used to include some improvements in the final release or introduced into the next version of the system in new iterations.

There are five workflows that cut across the four phases: Requirements, Analysis, Design, Implementation, and Test. Each workflow is defined by a set of activities that various project workers perform. UP is based on Models based on UML as standard language for modelling techniques. Models are used as vehicles for visualizing, specifying, constructing, and documenting the software development process.

- a) **Requirements.** The primary activities of the requirements are aimed at building the use case model, which captures the functional specification to reach agreement on the system capabilities.

The use case model consists of use cases, actors and use case diagrams. Use cases are first identified and then explained following a common template.

- b) **Analysis.** The primary activities of the analysis workflow are aimed at building the analysis model. It helps to refine and structure the functional requirements captured within the use case model and lend themselves to design and implementation.

The analysis model is a conceptual representation of the main system. A general view of the system can be represented by using package and class diagrams.

Some diagrams of classes can be used for representing a limited part of the system. Packages can represent conceptually complete subsystems.

The use cases are usually a sequence of actions where different objects interact. To represent the execution of use cases, interaction diagrams can be used. This representation will be done by using collaboration diagrams to identify requirements and responsibilities among objects. The collaboration diagrams show interactions between objects by using links and messages in these links.

- c) Design. The primary activities of the design workflow are aimed at building the design model, which describes the physical realization of the use cases from the use case model, and from the contents of the analysis model.

The design model serves as an abstraction of the implementation model. Attributes, relationships and methods defined in a class of the design model have a direct relationship with the implementation. The design model should be maintained during the complete lifecycle of the project due to its close relationship with the implementation model.

During the design work two different models are developed: design model and deployment model. The design model consists of a set of subsystem and classes diagrams, interaction diagrams and a textual description of the implementation requirements. The subsystem and class diagrams represent in higher detail parts of the system. The design model must also contain sequence diagrams, which represent the execution of use cases by showing detailed interactions between objects and subsystems, ordered in the actual timeline.

The deployment model describes the physical distribution of the system over the different computational nodes (hardware devices), e.g., the model shows the software-to-hardware correspondence in the system.

- d) Implementation. The primary activities of the implementation workflow are aimed at building the implementation model, which describes how the elements of the design model are packaged into software components, such as source code files.

The implementation model is composed of component diagrams. The component diagrams consists of subsystems with their dependencies and interfaces, and components with their dependencies.

- e) Testing. The primary activities of the test workflow are aimed at building the test model, which describes how integration and system tests will exercise executable components from the implementation model.

The test model contains test cases that are often derived directly from use cases.

6.3.2 Knowledge discovery Methodologies

The goal of a knowledge discovery process is to extract knowledge from data in the context of large databases. To guide the process there are defined standards with a detailed set of sequential steps organized in a methodology.

The term knowledge discovery in databases (KDD) was coined in 1989 to refer to the broad process of acquiring knowledge from data, and to emphasize the high-level application of particular data mining methods (46). There are considered five stages Selection, Pre-processing, Transformation, Data Mining and Interpretation/Evaluation. (71) (103).

The methodologies more commonly used are SEMMA and CRISP.

- a) SEMMA offers an easy to understand process, allowing an organized and adequate development and maintenance of data mining projects. The acronym SEMMA, defined by the SAS Institute considers a cycle with 5 stages for Sample, Explore, Modify, Model and Assess, helping to present solutions to business problems focusing on data mining business goals.
- b) CRISP-DM stands for Cross-Industry Standard Process for Data Mining. Its purpose is increasing knowledge extracted from the data, organizing the process around a way that the customer can use it. It consists on a cycle that comprises six stages: Business understanding, Data understanding, Data preparation, Modeling, Evaluation and Deployment.

CRISP has a broader view of the process, and is focused on customer understanding. It guides a set of proposed steps to go from user requirements to a final implementation.

- 1) Business understanding phase is focused on understanding the project objectives and requirements from a business perspective, then converting this knowledge into a DM problem. A preliminary plan is designed to achieve the objectives looking for a solution.
- 2) Data understanding phase starts with an initial data collection and proceeds with activities in order to be familiar with the data, to identify data quality problems, to discover initial insights to form hypotheses.
- 3) Data preparation phase covers all activities to construct the final dataset from the initial raw data.
- 4) Modelling phase covers the selection of modelling techniques to be applied.
- 5) Evaluation phase cover the steps to review that the models achieves the business objectives.
- 6) Deployment phase deals with the activation of the model in an industrial environment.

6.4 Personal Energy Advisor methodology

The contribution proposed here consists on defining a methodology that covers the lifecycle of a software product for generating the Personal Energy Advisor using the Granular Linguistic Model in the application development framework described in previous chapter.

The aim of the Personal Energy Advisor is to substitute an expert analyst, interpreting user queries and producing linguistic answer reports similar in quality and structure.

This kind of systems share the same core architecture that is sensible for reuse due to a design based on components. The architecture considers a huge amount of data stored in a data base, and the computational ability to interpret the data and express it in understandable language.

The process followed to develop the systems starts with an initial definition of the contents to generate in a report with the overview of the information contents and language in use, independently from the underlying architecture focusing on a business perspective. The definition of the report, identifying the suitable terminology and topics of relevance, is performed with the help of experts and users, and using articles related to the domain. Terminology is highly influenced by the users to which the application outputs are directed, e.g., the type of user and his/her experience.

The ground data used to generate reports is stored in databases. Databases constitute a model for information storage and recovery widely extended. The key achievements at this stage are to identify and validate the data required, and extract a detailed definition of key aspects and rules that define relevant knowledge from the data using the required analytical models.

Initial report, previously created, is analyzed from a technical perspective based on the granular linguistic architecture. This implies that report changes could be required at this stage, due to the limitations of the data available or the language expressions implementable with perceptions.

The report is divided in topics grouped around specific issues sharing common data. Each of this topics or paragraphs will be design and implemented using a granular linguistic model. Each of the sentences of the paragraphs is implemented using perceptions.

For each perception at the granular linguistic model, will be defined the input data, the computational requirements and relevance associated with each result, and linguistic descriptions including emotional contents.

6.4.1 Work Team

The number of members of the work group will depend on the size of the project. The typical working group will consist of a team of four people: Manager, Designer, Data Analyst and Developer.

In each phase and/or activity, each member will be responsible of one or more working roles.

- Manager is the person of the team who is in charge of managing communication with the customer and ensures that the project meets customer expectations. The manager is the main responsible of the correct development of the project. As responsible of the project, his/her goal is to ensure that the project will progress towards its final target. He/she has the initiative and assures the feasibility of the project.

- Designer is a person with global knowledge and skills, usually an experienced role with business and technical details. The designer is in charge of the project. He/she has a broad knowledge in the application and experience in the functions developed by the data analyst and the developer.
- Data Analyst is a person that owns a great knowledge on data analysis and the business issue. The data analyst is in charge of the data model design.
- Developer is a person with knowledge in the application framework and the technology that underlies. He will develop the implementation model, and will be responsible for the implementations and testing tasks. It is not a usual language developer, but a developer for the application framework.

6.4.2 Phases in the Methodology for the Personal Energy Advisor

To accomplish the development of the Personal Energy Advisor over the application framework, the process is divided in phases and workflows as showed on Fig. 6.4. Phases are detailed covering each specific steps. Each phase is divided in iterations, where initial and therefore general requirements are detailed and developed iteratively. The overview of the activities for the development of the Personal Energy Advisor is showed on Fig. 6.3.

The development process starts with the business modeling describing the motivation, the state of the art and the scope of the project.

a) Inception.

The process starts with a business model describing the motivation, the state of the art and the scope of the project.

Initial requirements are defined with a report defined by an expert in the subject. A report is structured in terms of topics or paragraphs of output text. Also data available is identified and described providing an overall view of information.

The report, each paragraph and sentences are modelled using use cases that describe the content structure. Independent from the underlying architecture. Each use case will identify the final users and the input data required.

A general use case will describe the overall report, and subuse cases will decompose the report into sub problems regarding the content scope and the input data, rules required for computational capabilities and the text associated. Some new minor use cases could be included later.

Note that the content and structure will be defined, first independently and later focus on technical details matching from the initial report a layout of perceptions. Data sources available are identified and described.

The first prototype, as a proof of concept, is generated covering some relevant use case even when the system is not yet integrated in the overall architecture.

At the end of this phase agreement with the customer is achieved. Having a minor example of the final result for a small scope of the overall project risks are mitigated.

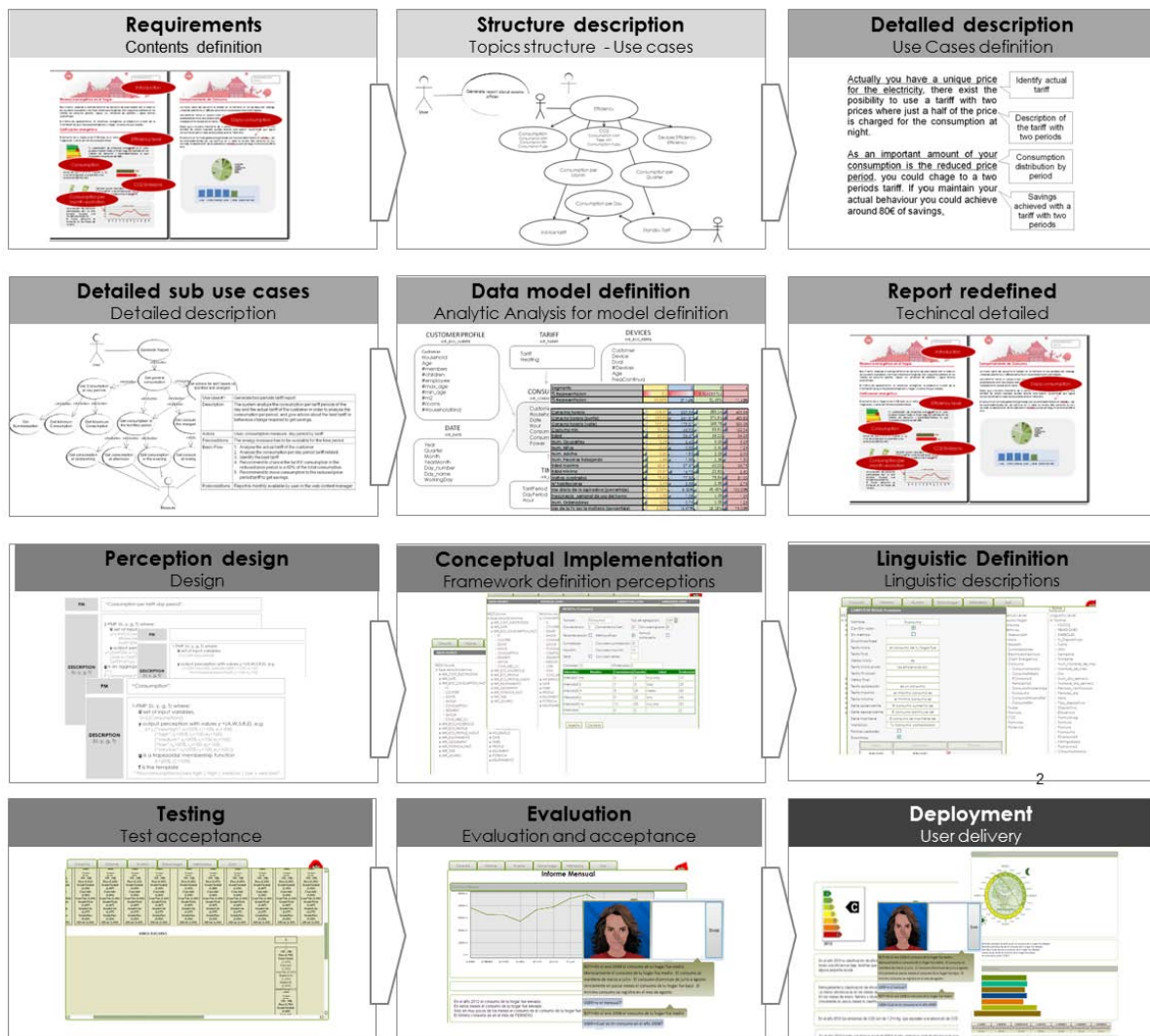


Figure 6.3: Overview of the activities to develop with the Personal Energy Advisor Framework.

b) Elaboration

During the elaboration phase, most use cases are detailed in order to design an initial system with the main functionality. Following iterations could include limited extensions for some minor use cases.

The general report is described with a detail description of use cases. Data is identified and modelled in order to optimize execution using precomputed values.

This prototype will have the architecture integrated in the overall system of the customer. Non functional requirements could be tested in this phase, e.g., time of execution involved an response time requested.

The planning of the project will be closed with a detailed description in terms of time required and resources needed.

c) Construction

During the construction phase, the components are developed, and the system is

built. It is performed in iterations producing releases for a subset of the global domain that can be deployed with significance in the environment and tested. The testing approach evolves from unit testing to integrated system testing.

The test procedure has to be planned covering all the use case scenarios implemented so far, checking that all the user requirements are covered. This activity covers the integrity and system tests, since the unit testing has to be developed in the implementation stage, at the same time as the system is being coded. These defects are noted down to be solved in the previous activities (design and implementation) of the next iteration.

d) Transition

During the transition phase, the system is delivered to the user. A beta release is delivered to receive feedback and acceptance. Some minor changes could be applied for the final release in order to fix specific issues basically to gain fluency in the language generated. New improvements identified would be included for next iterations.

In addition to the general plan design, each iteration is also planned in detail before its start. It is preferred that the iterations are relatively short in time.

6.5 Methodology Application Personal Energy Advisor

In this section the practical application of the methodology is described to illustrate the details of the approach. Only one iteration is described in the early stages of the process, to show the functionality required and developed. It is described in sequential steps developing a use case. In the next two chapters, the main stages of the methodology the analytical and the software development activities are described in more detailed, “*Efficiency Analytical Modeling*” and “*Energy Advisor Development*”.

6.5.1 Business Modeling

Motivation

The company EDP launched the project “Ecofamilias”, with the objective of providing useful information about consumption patterns in households.

According to studies about energy consumptions habits, 90% of customers are willing to consume less energy and 76% like to do more for the environment, but do not know what to do.

The ability to interpret and identify customers needs, and being able to communicate appropriately will provide differentiated services to households. Users to use the reports are not required to have advance analytical skills, so reports generated must be easily understandable by users and readability has to be maximized while determining content, structure and linguistic expressions

State of the art

Energy is consumed through a variety of devices and consumers hardly know how do they use the energy, how can they improve energy efficiency or how could they contribute to the energy system.

Hourly meter devices placed in households increase the amount of consumption data actually available.

By means of natural language processing, a description of the current situation of the household can be generated on easy terms to understand using the data available.

Scope

The scope of the application is limited to advice about day periods prices, as energy at night can have a reduced price and can be used by users to get savings and to improve the energy efficiency.

The ability to transform this data in easily interpretable natural language, extracting remarkable information and hiding the irrelevant one, will be a key in the process of identifying paths to improve efficiency.

The basic functionality of this system is based on the computational architecture of the GLM. The raw data is obtained from consumption measures hourly retrieved by meter devices. Information generated will be published in a content database accessible in a web for user access.

Reports should be translated into action by the customers. Customer reaction will be measured in terms of past and actual behaviour to track improvement and usability for the user.

The aim of the application is to generate a personalized energy advisor able to report about energy efficiency to households as requested by users. The answer provided to users in the response is a report with a linguistic summary similar to the one that an expert advisor would produce.

6.5.2 Requirements

To specify the requirements are defined a report with the contents to deliver. A report is a document with the information considered by experts in energy efficiency. It is composed of charts texts and detailed information in a table as required. The main tasks are developed by the customer with the support of the manager and the designer. It is developed using different iterations where the report is composed and delivered to the team project including many corrections.

The report contains a detailed example of advice to a customer or a set of customers around the different topics of interest and a detailed description around each topic as if

it were going to be delivered to the customer.

The final report of this stage is showed on Fig. 6.5. The report contains energy efficiency advice describing from top to down the characteristics of consumption of a household going from the highest level information to a more detailed explanation to facilitate the understanding if the customer. It follows a structure that begins with an introduction, a set of body paragraphs and conclusions. The first paragraph is an overview generated with the overall efficiency of the customer and a description of this efficiency in comparison with the households in the same geographic region. Efficiency is the main topic, that is decomposed in the following paragraphs in a annual terms in a description of the consumption, CO2 emissions, annual costs in and the average efficiency of the devices.

A more detailed description is added in a second area with the consumption per month that would allow to identify the evolution of consumption in time, and the comparison of this year with previous year.

The third area describes the detailed consumption per hour, and the main issues related with this consumption, e.g., underlying that consumption at night is mainly produced by stand-by devices.

It is required to detail the channels in use and the frequency of delivering to the users. The report will be generated in a web environment for each user on a monthly basis in a batch process.

The initial requirement is structured in a set of use cases, identifying actors as showed on Fig. 7.4. Use cases are used to represent the related contents and actors that are in charge of generating input data or to whom is delivered the resulting information. Use cases will include while moving forward in the methodology the detailed process associated to the generation of each content. The report is limited on generating personalized reports about efficiency achievable by optimizing the Consumption, CO2 emissions and devices efficiency, also are specified the consumption per month and the consumption at the different day periods.

6.5.3 Efficiency Analytical Modelling and Energy Advisor Development

In the next two chapters the detailed steps and the application of the methodology are described for the analytical modelling and the software development activities.

The Efficiency Analytical Modelling phase describes the analytical modelling phase that was carried out for the definition of the Personal Energy Advisor knowledge domain. The information defined at this stage refines the definitions of initial requirements and will be the input for the design and development of the application framework.

The main activities in the analytical process are:

- Analytic Requirements

Evaluate the input report in terms of data, linguistic expressions and computational requirements.

- Data Understanding

Identify data sources available and select input fields.

- Data Preparation

Evaluate the data available and transform the data to generate additional required information fields.

- Data Model Definition

Define the multidimensional data model with the information required integrated from the different data sources.

- Analytic Modelling

Model analytic computations, understand and identify relevant information.

- Report refinement

Redefine the initial report from requirements in terms of linguistic expressions and contents due to technical constraints and analytic capabilities.

- Evaluation

Evaluate the redefined report to gain feedback and acceptance for the implementation phase.

- Data Model Building

Build the data model and load the data.

The Energy Advisor Development phase describes the implementation of the Personal Energy Advisor. It is performed due to an analysis review, a design definition, the implementation and the test. This phase can be developed due to the analysis definition carried out in the previous phase for an implementable report. Along this phase the project moves to an implemented solution.

The main activities followed in the development process are :

- Analysis

Analyze contents as use cases defined in previous phase for design, e.g., input data, computations, lexico grammatical components and emotions.

- Design

Design the knowledge domain for implementation with a specific design for each use case, e.g., physical, conceptual and linguistic.

- Implementation

Implementation of the use cases designed in the framework, e.g., physical interface, conceptual interface and linguistic interface.

- Test

Test plan definition and validation of the solution.

	Inception		Elaboration					Construction				Transition			Production		
	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	T1	T2	T3	T4	T5
Business Modeling																	
<i>Motivation</i>																	
<i>State of the art</i>																	
<i>Scope</i>																	
Requirements																	
<i>Analyze the problem</i>																	
<i>Understand Stakeholders needs</i>																	
<i>Define the system</i>																	
<i>General Architecture</i>																	
<i>Manage scope of the system</i>																	
<i>Report definition independent from the architecture</i>																	
<i>Identify Data Sources</i>																	
<i>Channel Integration requirements</i>																	
<i>Refine the System Definition</i>																	
<i>Report definition contents</i>																	
<i>Identify risks</i>																	
<i>Manage Changing Requirements</i>																	
Analytical modelling																	
<i>Analytic Requirements</i>																	
<i>Report evaluation</i>																	
<i>Match data required to report</i>																	
<i>Report refinement</i>																	
<i>Data identification and understanding</i>																	
<i>Identify the data sources</i>																	
<i>Select Data</i>																	
<i>Describe data required</i>																	
<i>Data model definition</i>																	
<i>Data Preparation</i>																	
<i>Data evaluation</i>																	
<i>Data cleaning</i>																	
<i>Data Transformation</i>																	
<i>Data Model Definition</i>																	
<i>Data model design</i>																	
<i>Analytic Modeling</i>																	
<i>Information Discovery</i>																	
<i>Design the analytic model</i>																	
<i>Build the analytic model</i>																	
<i>Assess Analytic Model</i>																	
<i>Redefine Data required</i>																	
<i>Report refinement</i>																	
<i>Report redefinition</i>																	
Evaluation																	
<i>Evaluate results</i>																	
<i>Review process</i>																	
Build Data Model																	
<i>Data model building</i>																	
<i>Extraction of data to feed data model</i>																	
Build the system																	
<i>Analysis</i>																	
<i>Define architecture</i>																	
<i>Analyze Behaviour</i>																	
<i>Design Components</i>																	
<i>Design data source access</i>																	
<i>Design components</i>																	
<i>Implementation</i>																	
<i>Implementation model</i>																	
<i>Implement components</i>																	
<i>Integrate the system</i>																	
<i>Test</i>																	
<i>Define test plan</i>																	
<i>Verify test approach</i>																	
<i>Test and Evaluate</i>																	
Deployment																	
<i>Prepare the environment</i>																	
<i>Deployment plan definition</i>																	
<i>Support environment for iterations</i>																	
Support																	
<i>Change and deliver configuration items</i>																	
<i>Manage Releases</i>																	
<i>Manage Change requests</i>																	

Figure 6.4: Planning

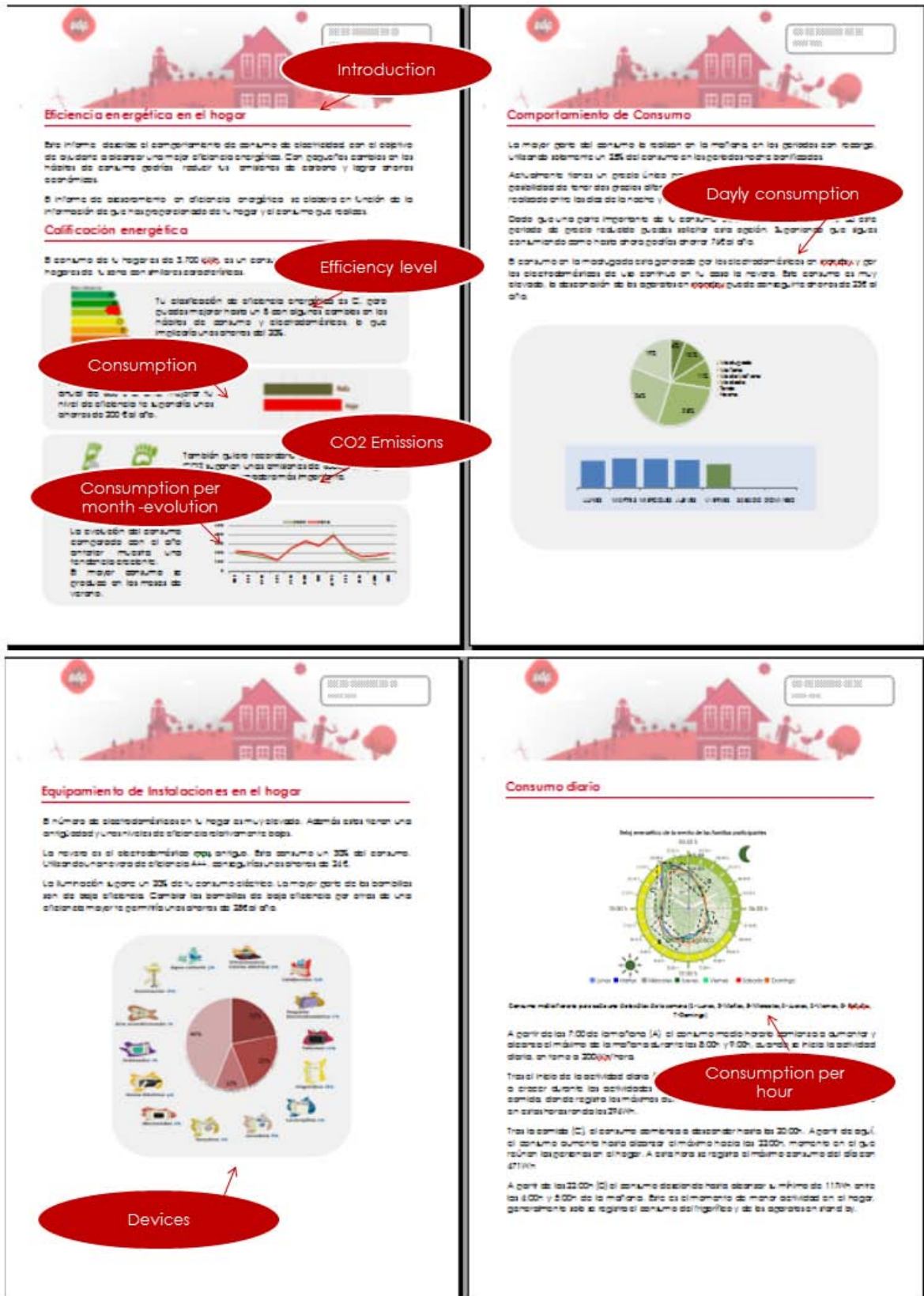


Figure 6.5: Initial report definition.

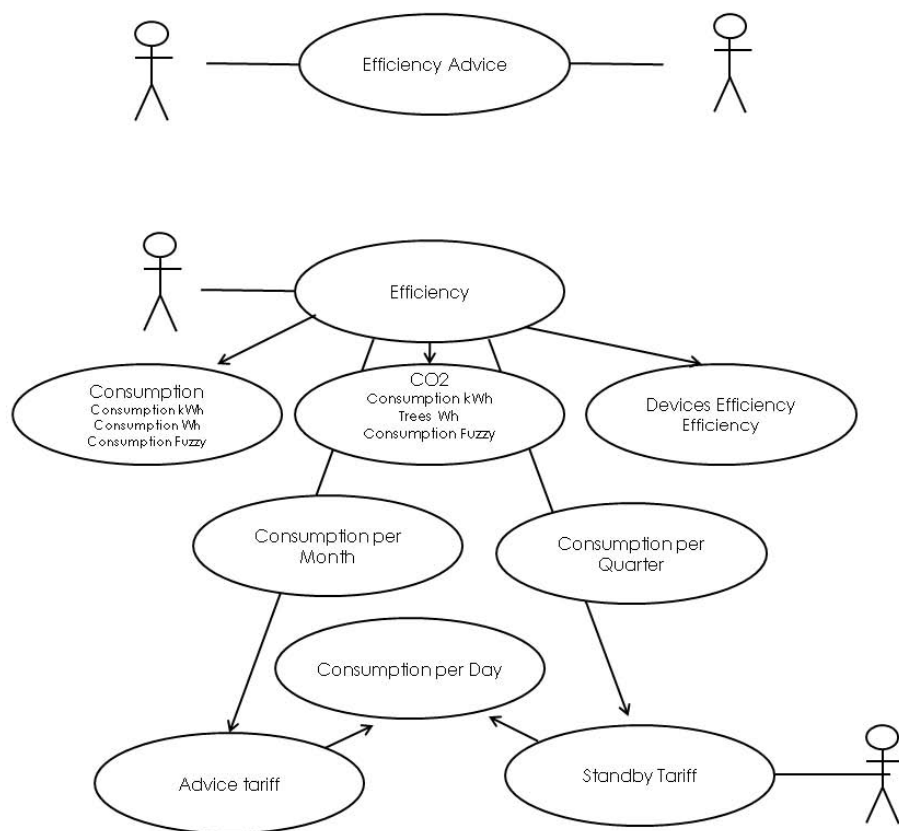


Figure 6.6: Use case for efficiency report

Chapter 7

Energy Advisor Efficiency Analytical Modelling

7.1 Introduction

This chapter describes the analytical modelling phase that was carried out for the definition of the Personal Energy Advisor knowledge domain. The information defined at this stage refines the definitions of initial requirements and will be the input for the design and development of the application framework in the following phase *Energy Advisor Development*.

Making use of available input data and expert knowledge, during this phase are covered the different analytical approaches taken to identify advice for the customers.

During the analytic modelling activity is performed the task of understanding the customer behaviour in order to be able to communicate to the customer an efficient way of use of energy. Only once understood the customer behaviour would be possible to give the requested advice, give exact details about the consumption to users, identify key patterns and model recommendations.

This phase will reveal the gaps between initial requirements defined in the initial report independent of the technical details and an implementable report due to technical details. Along this phase the project moves closer to how has to be modelled the knowledge domain.

7.2 Activities

The activities for the analytical modelling phase to define the Personal Energy Advisor are described with a review of the methodology introduced in the previous chapter. The main activities in the process are showed on Fig. 7.1.

- Analytic Requirements

Evaluate the input report in terms of data, linguistic expressions and computational requirements.

- Data Understanding

Identify available data sources and select input fields.

- Data Preparation

Evaluate available data and transform the data to generate additional required information fields.

- Data Model Definition

Define the multidimensional data model with the information required integrated from the different data sources.

- Analytic Modelling

Model analytic computations, understand and identify relevant information.

- Report refinement

Redefine the initial report from requirements in terms of linguistic expressions and contents due to technical constraints and analytic capabilities.

- Evaluation

Evaluate the redefined report to get feedback and acceptance for the implementation phase.

- Data Model Building

Build the data model, extract and load the data.

Input requirements are based on the report contents initially defined by experts. Initial contents will be constrained and redefined in order to be able to define the final energy advisory system. With this purpose use cases defined in requirements and extracted from the initial report, will be detailed. Some information at this analytical stage could be rejected from previous definition based on restrictions due to existing data in source databases or could be added based on new useful information obtained from data not identified earlier.

The main functional stages for Analytic Modelling are underlined in red on Fig. 7.2. During this phase, the analytic capabilities to report are identified, validated and described. Based on the report definition the information and analytic models required are identified and built.

The outputs of this phase will be the input for next steps to build the application.

- Redefined report contents

Once the problem is analyzed and understood and the scope is delimited, the output of this phase is an implementable redefined report constrained by technical details. At the end of this stage the key aspects are identified and the report definition is detailed based on the information available.

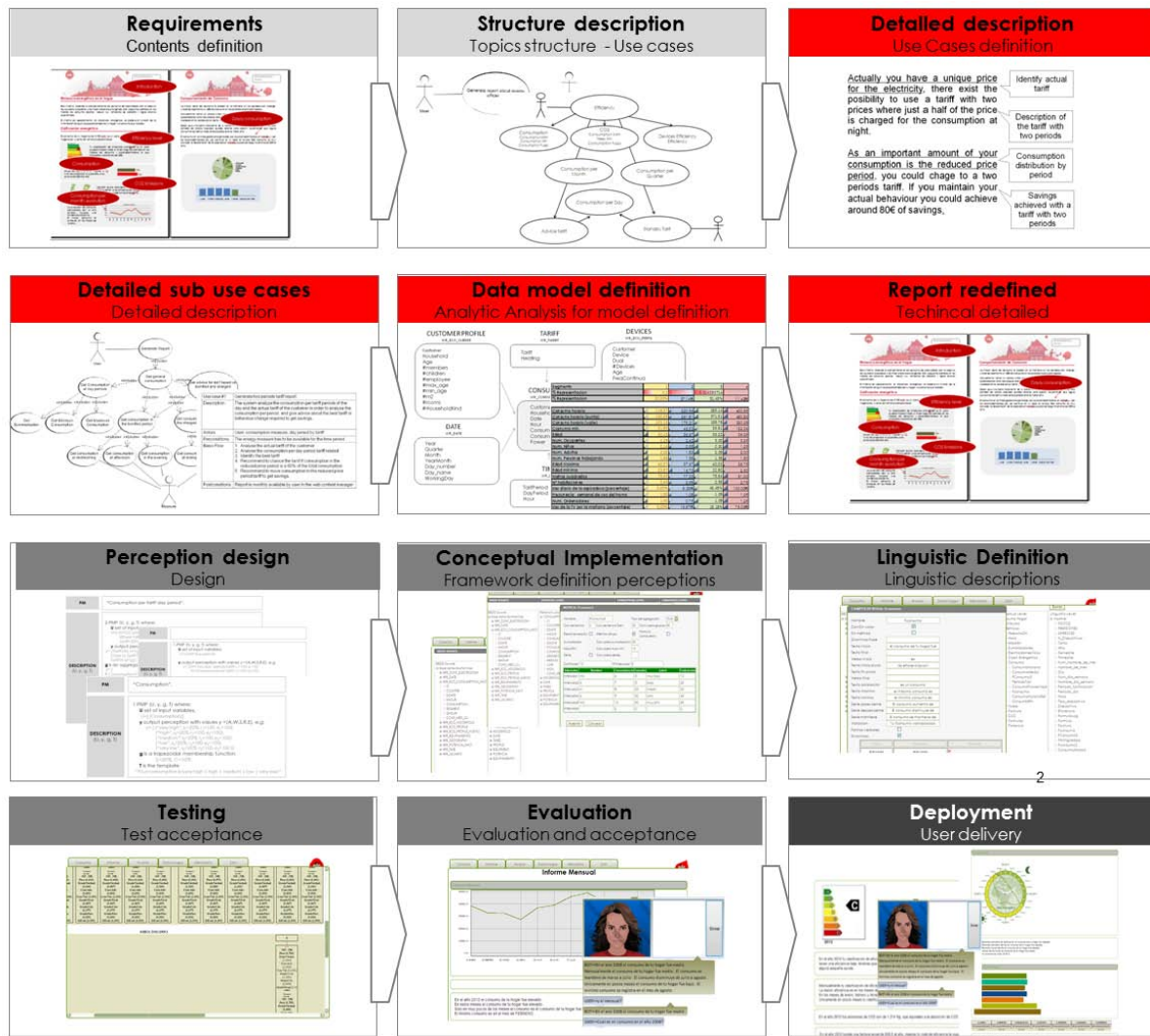


Figure 7.2: Overall methodology with the analytic stages in red, namely detailed description of use cases, detailed description of subuse cases, data model definition and report redefinition.

The process follows a path to go from existing information sources to the identification of key issues and most relevant aspects that can be obtained from the data, e.g., the analysis of when a two price periods tariff has to be recommended to users and the percentage of users to whom this recommendation applies. A large number of potentially useful statistical measures will be defined and computed on the data due to data transformation.

Information to be used in the report should follow two premises, the coverage and the robustness.

- By coverage is searched that the knowledge results are relevant enough for a representative subset of customers. For example if consumption in the standby period is a key element to use in advice, the generation of a recommendation to save in this kind of consumption has to be applied if there is a relative high number of customers for who is useful to apply this recommendation, and if is feasible to perform

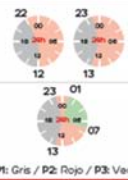
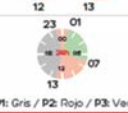
TARIFFS PER PERIOD											
TARIFA	POTENCIA	TÉRMINO DE POTENCIA (€/kW año)		TÉRMINO DE ENERGÍA (€/kWh)			DISCRIMINACIÓN HORARIA		% REACTIVA sobre (TFP+TFE)	IE	
		Sin DH	Con DH	Sin DH	Con DH			Invierno			Verano
					P1	P2	P3				
TUR	≤10 kW	21,893189		0.138658	0.167658	0.05719	-		Aplicable en función cos φ	4,864% de [(TFP+TFE+CM)1,05113]	
TUR SUPERVALLE	≤10 kW	21,893189		-	0.167558	0.069460	0.049695				

Figure 7.3: Day periods for tariff prices.

the improvement by the customers, e.g., their consumption in the standby period is quite high.

- By robustness is searched that the knowledge results remain in time. For example if consumption in the standby period is a key element to use in advice, the generation of a recommendation to save in this kind of consumption has to be applied if the behaviour of this kind of consumption remains in time, e.g., their consumption in the standby period remains in the last three months .

Understand the consumption behaviour

Energy is consumed through several devices and is influenced by internal and external factors. It is a complex task to translate data consumption into useful information for the users, e.g., an explanation about how do users use the energy or how could they to get savings.

Due to the complexity of behaviour explanation, it is required to understand how users consumption changes over time, find if exist users with similar characteristics or understand the internal and external context impacts in consumption.

The main problems with consumption data is not just the lack of detail but also the lack of context around the information and the difficulties for users to interpret and make sense of the information. Comparison of behaviour in time, and comparison with similar households behaviours should be a way of positioning each individual patterns in a context that could be used as a reference for the users.

Having the information of two households where, for one the consumption is 2.000 kWh per year and for the other 3.500 kWh, a basic analysis could make us fail in the conclusion supposing that the one with higher consumption is less efficient, if the one with higher consumption is a household of a family of five members, and the other one is a household of a single.

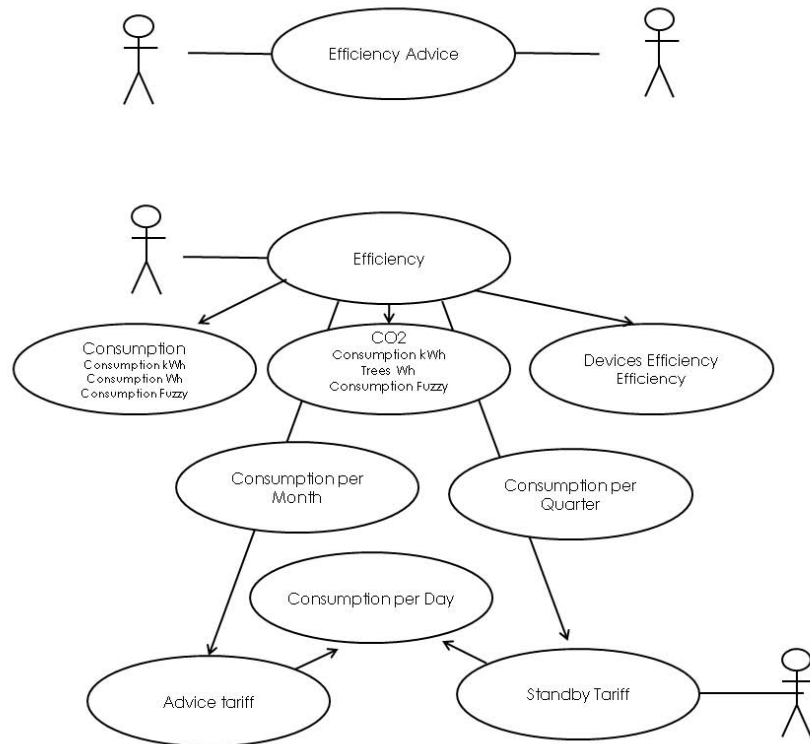


Figure 7.4: Use case for efficiency report

Process Review

The principal objective is to give advice to customers and to deliver it efficiently. Information is presented and structured from top to down, where aggregated information at the beginning gives a general overview of the actual situation, and once decomposed into more detailed elements the user can perform or act to modify behaviour. This allows getting a general view, a resume of the actual situation, and a more detailed decomposition in items where to take decisions.

The initial requirement defined in the report and in a set of use cases identifying actors and topics as showed on Fig. 7.4. Information in the report as expressed in use cases starts with a general efficiency description, and it is decomposed in lower level information, namely consumption, CO2 emissions and costs associated. After the initial review a detail analysis per month is reported. The last topic is the detailed consumption per hour. Organization in topics and initial decomposition of the report is showed on Fig. 7.8.

To complement the understanding of the customer, there are analyzed the “*charts contents*”, as it is required to deliver charts around the text to complete the information using a visual representation. For example an “Energy Watch” is used to represent for

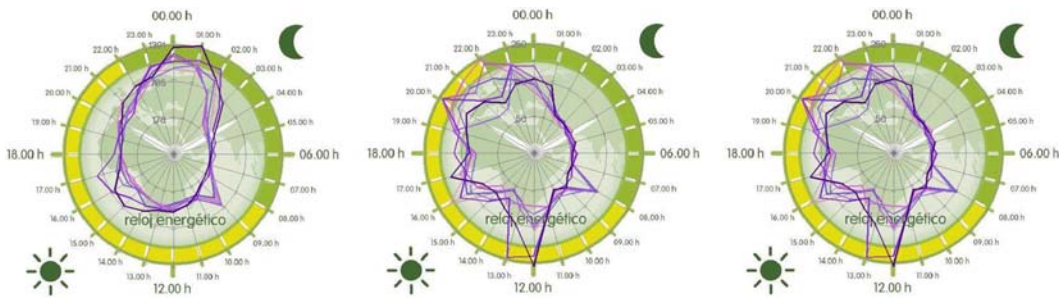


Figure 7.5: Energy Watch (Registered by EDP)

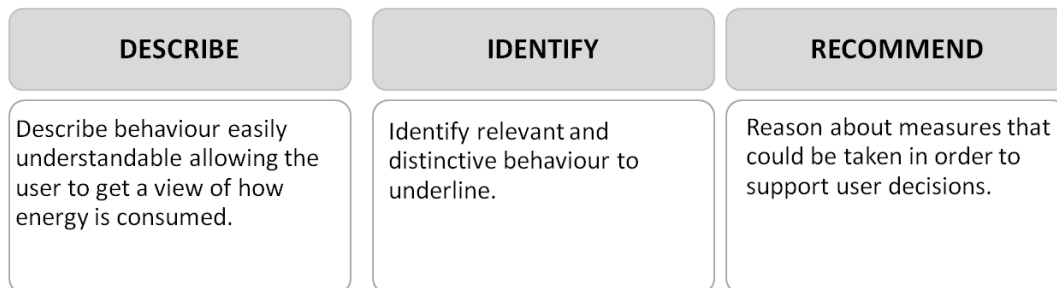


Figure 7.6: From data to detailed information to support users decisions.

the customer the energy consumption per hour as showed on Fig. 7.5.

Each sentence is analyzed in terms of output text generated, input data required and computations associated and the kind of perception to generate the output text. Sentences can be clasified in three main types as described on Fig. 7.6.

- Describe

Describe behaviour easily understandable using close natural language, e.g., the description of the average consumption per hour along the day.

A fuzzy set approach is applied to find natural language descriptions of the measures dynamically. Linguistic terms will be generated by fuzzifying input numerical measures. This will allow facilitating the customer understanding.

Membership functions have to be defined for the perceptions in order to compute the fuzzy values or linguistic labels. With this purpose each crisp measure is translated by a membership function into a defined fuzzy attribute with a set of linguistic labels. In the application framework trapezoidal fuzzy sets are used as showed on Fig. 7.7. It is noted that the membership function should be relative to the measure contextualized by the dimensions and should have the capabilities to be aggregated, e.g. *“Consumption per month vs Consumption per year”*.

- Identify

Identify relevant information to be described using the data available. This allows giving a more detailed advice focusing in key events where the user can react and take actions to improve efficiency and get savings.

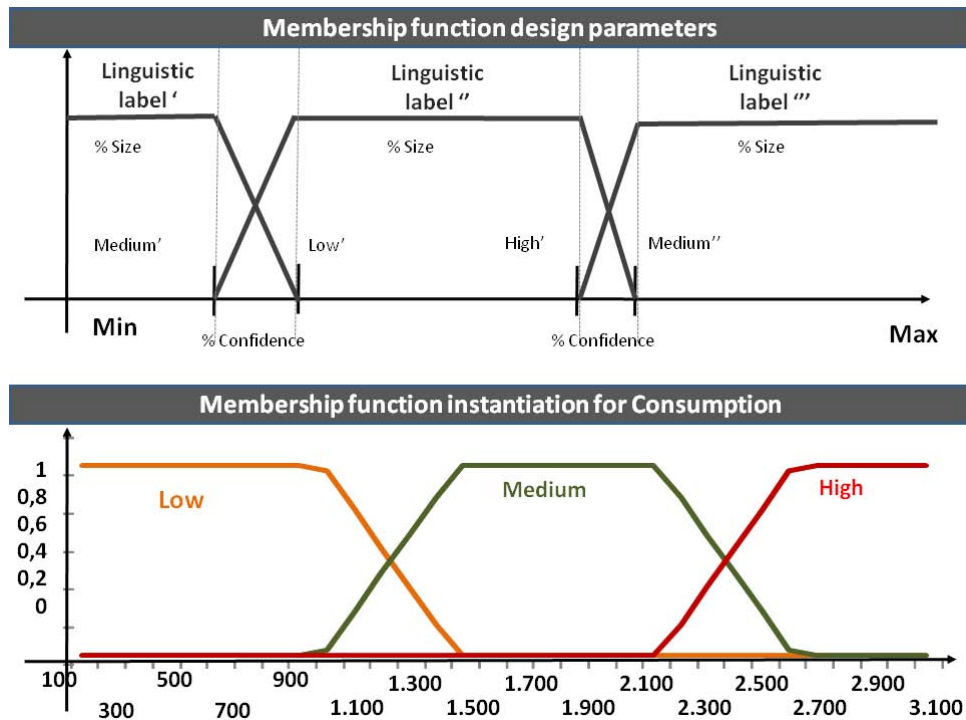


Figure 7.7: Membership definition of fuzzy values

For example identify the peaks of consumption when differ from the rest of the users. Key events are modelled by assigning relevance to contents to describe in each specific situation, e.g., to speak about *efficiency* is relevant but it is more relevant to speak about *very low efficiency* behaviour than to speak about *medium efficiency* behavior.

- Recommend

Here the objective is not only to identify key events, but to reason about measures that the user could take into account in order to guide and support decisions. For example depending on the period of the day when peaks are identified, could be reasoned a recommendation to reduce the peak, e.g., if it is at midday recommend how to reduce consumption while cooking.

Also recommendations about specific behaviours have to be identified, for example if the consumption in the low cost period is high and the customer does not use a two period tariff that has a reduced price period for the low cost period. Then the potential savings for this customer just changing the tariff could suppose important savings.

Recommendations are computed and modelled using a set of fuzzy if-then rules.

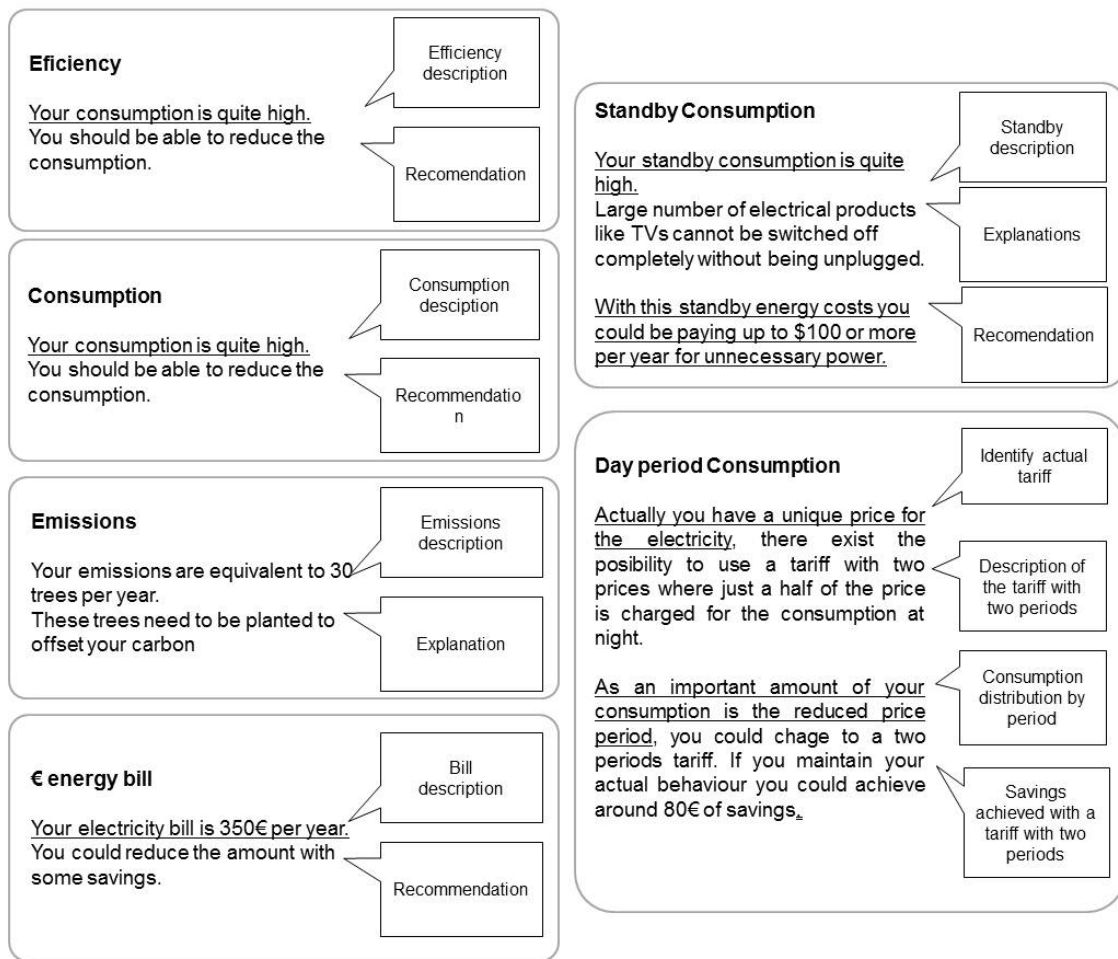


Figure 7.8: Report decomposition into topics and association of analytical requirements.

7.3 Analytic Requirements

At this stage requirements have been defined at high level around the elements to report and the information that should be delivered to the customers. Initial scope is limited on generating personalized reports about efficiency to help users to optimize their consumption. The scope of requirements is delimited due to the analytic results focusing in a system implementation. The review of the initial requirements is performed in order to match and constrained the desired results. Detailed use cases are produced for next steps.

Initial report is decomposed in topics. A topic is a set of sentences around a common content. For each topic, each sentence is identified and associated to a perception to describe its components. For each perception is identified what information is described, what and when is considered relevant the information and how the information is extracted from raw data. Also, the language to be used in each of the topics is identified and the emotional variations from initial report.

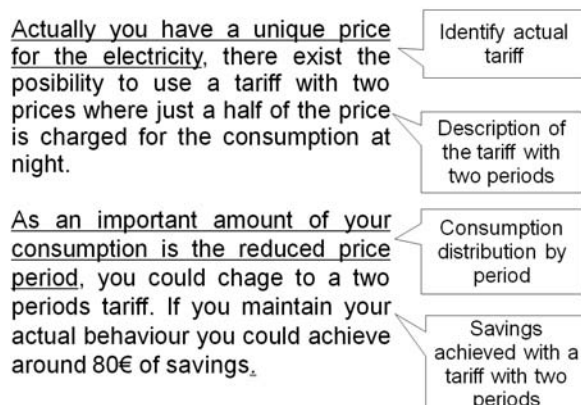


Figure 7.9: Text Report topic for day period consumption.

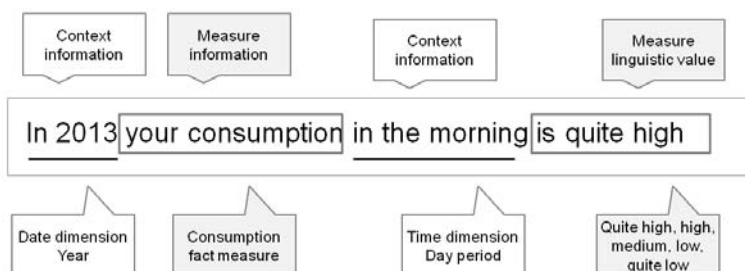


Figure 7.10: Example of a report sentence decomposition

For the initial use case the corpus of the text covering the requirement is extracted with the help of experts and existing documents. An initial set of topics is showed on Fig. 7.9. This text is an initial version independent from the data available and architecture.

Each sentence in the report is assigned to a specific type, description, identification and recommendation as showed on Fig. 7.6. Sentences in the report are decomposed identifying data elements required as showed on Fig. 7.10. While report refinement each sentence in the report is mapped to a perception. Each perception is composed of an event, e.g., consumptions or CO₂ emissions, and the context, e.g., in this year, on January. The initial report decomposed in topics is showed on Fig. 7.8.

The use case is refined and a detailed description is provided in detailed use cases showed on Fig.7.11. The process is followed for each topic identified in the initial report, resulting in a set of use cases.

Use case #1	Generate two periods tariff report
Description	The system analyze the consumption per tariff periods of the day and the actual tariff of the customer in order to analyse the consumption per period, and give advice about the best tariff or behaviour change required to get savings.
Actors	User, consumption measure, day period by tariff
Preconditions	The energy measure has to be available for the time period.
Basic Flow	<ol style="list-style-type: none"> 1. Analyse the actual tariff of the customer 2. Analyse the consumption per day period tariff related 3. Identify the best tariff 4. Recommend to change the tariff if consumption in the reduced price period is a 60% of the total consumption. 5. Recommend to move consumption to the reduced price period tariff to get savings.
Postconditions	Report is monthly available by user in the web content manager

Figure 7.11: Detailed description of the use case for day period consumption.

7.3.1 Report data matching

In report data matching activity, information contained in the report is related to measures (events) and attributes (context) adding the technical perspective in the analysis.

Measures describe *how* or *how much* an activity performed. Measures identified for efficiency reporting are:

- Consumption (Direct)
- CO2 Emissions (Computed)
- Expenses (Computed)
- Devices Efficiency (Direct)
- Efficiency (Computed based on previous perceptions)

The description of the measures answer the related question tags *when*, *why*, *who*, *where* or *what*. It is the related information to contextualize the measures.

- Time
- Date
- Household
- Devices

Efficiency is defined based on the consumption of the customer, the use of consumption in the low cost period, the device efficiency and the power demand.



Figure 7.12: Hourly consumption measures.

7.4 Data Identification and Understanding

Identify the data sources

Based on use cases the data sources available are identified and the related data sources are described.

- Consumption data

The initial data available for consumption per hour is provided as a text file, structured with columns names in the first row. Each row refers to one measurement of consumption, taken at a specific time for a specific client.

The detailed fields for CONSUMPTION are:

- A Column AIMP which states the consumption, usually in Watts.
- A client identifier CUSTOMER. By default it is assumed that there is a column with the id of the consumption user.
- A DATE and HOUR related data, which together uniquely specify the measurement of consumption for each date and time. This column will be spitted in two different ones, in order to manage separately date and time behavior.

- Household data

Information about the household geographic location and technical related data is considered. Related information includes the address and the tariff of the customer. The information is delivered in a text file HOUSEHOLD.

- Sociodemographic characteristics

Information about sociodemographic characteristics was collected in a specific survey for the project and delivered in a text file. This information is showed on Fig. 7.13.

- Devices

Information about the devices is also collected with a detailed description for each device using a survey. This information is showed on Fig. 7.13.

Nombre Apellidos Edad

Nº ocupantes de la vivienda Niños Adultos

Nº personas que trabajan fuera de casa Edad máxima de los ocupantes

Edad mínima de los ocupantes m² de la vivienda Nº de habitaciones

¿Que equipamiento tienes en tu casa?

¿Con que frecuencia usas estos equipos?
Marca solo una opción

¿En que momento del día los usas generalmente?
Puedes marcar varias opciones

¿Que tiempo medio de uso les das?

Electrodomésticos	Sí/no	Antigüedad	Nº	Continuo			Diaria			Semanal			Continuo		Mañana	Mediodía	Tarde	Noche	Tiempo aprox.	
				1	2	3	1	2	3	-12h	12-18h	18-20h	+20h	Horas	Minutos					
<input type="checkbox"/> Frigorífico	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Cocina/Vitrocerámica	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Horno	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Lavadora	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Lavavajillas	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Congelador tipo arcón	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Secadora de ropa	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Microondas	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Campana extractora	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Calefacción individual	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Calefacción central	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Agua caliente	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Climatizador	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Ventilador tipo torre	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/> Ventilador de lecho	<input type="checkbox"/>	<input type="text"/>	<input type="text"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Figure 7.13: Sociodemographic data related to the user. Print screen of the application framework.

Temperature was also provided, with a detailed description of the temperature for the geographic area. Basically all the information initially required is available and delivered in text files. Text files are loaded in a data base in order to manage the data and link the information to the application framework. Any transformation applied to the data is described later.

7.5 Data Preparation

7.5.1 Data Evaluation

Here data is reviewed in terms of quality. While evaluating data is ensured that data is received is correct and useful. Some routines are used to validate the data and in data cleansing are implemented some facilities to correct any issue if possible. Evaluation of the data is a crucial step in order to provide useful information. Errors in data should be the key issue in order to get confidence of the user in the interpretation of data.

7.5.2 Data Cleaning

In data cleaning transformations are applied to the data in order to clean them as required. The issues identified and the solutions adopted are described in detailed.

- Duplicate entries: The raw data file contains, for some measurements, two entries with different consumption values. In such a case the entry appearing in a row with the higher number is used.

- Missing entries: Apart from the fact that the measurements start and end at different points in time for the different clients, there are cases where intermediate measurement values are missing. Such missing values are detected and reported, but not corrected.
- Streaks of zeros: There are cases where several consecutive measurements report a consumption of 0 Watts. As this is unlikely to be correct, such streaks of no consumption are detected and reported if at least 4 consecutive measurement values are 0. They are replaced by missing values.
- Strangely large entries: In data a strange pattern of wrong consumption values were detected: the sum of two consecutive values exceeds 10000 Watts, even though the surrounding values are considerably lower. As this points to faulty measurement, such pairs are detected, reported and removed (replaced by missing values).

7.5.3 Data Transformation

Data is transformed due to analytic requirements. New required information is included in order to deliver the information requested in the use cases.

The result of this transformation step includes the following fields.

- A new table is created to describe the DATE field in the consumption table with a hierarchical structure. This table contains information in independent fields for the Year, the Quarters, the Month, the Week and finally the date. The date field is also present in the consumption table. The hierarchical structure is showed on Fig. 7.14.
- A new table is created to describe the TIME field in the consumption table with a hierarchical structure. This table contains information in independent fields for the Day Period, the Tariff Period and the Hour. The hierarchical structure is showed on Fig. 7.14.

Note that some of the transformations will be implemented in the data model and some others will be defined in the application framework and will be calculated on-the-fly while executing the reports as defined in the knowledge domain, e.g. the CO2 emissions as a transformation of consumption in kWh into CO2 Emissions kg= $\text{Consumption kWh} * 30 \text{ kg/kWh}$.

7.5.4 Design the data model

In the design the data model structure and the data to be used as input for the model are defined. The data model designed is the result of several iterations during the analytic process.

The final data model will include the results of the analytic modelling activity in later iterations. For example if considering the information of CO2 emissions it could be transformed into more meaningful information for the user like the amount of trees that

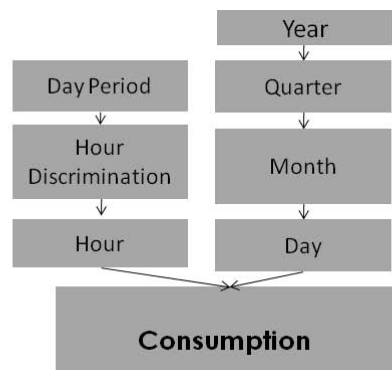


Figure 7.14: Dimensional data model definition.

are required to clean the CO₂ emissions. A kg of CO₂ emissions are equal to the amount of CO₂ that can clean a tree in a year. Also more complex results of the analytical model could be included later, for example the segmentation results of the clustering data model is included in the final design, each customer will be assigned to the corresponding segment.

The elements of the data model are described in terms of dimensions and facts. Due to the hierarchical organization of dimensions, details of measures extracted can vary according to the selected level chosen. The data model is defined as showed on Fig. 7.14. The fact table contains the consumption numeric value and a field to relate consumption data to the dimensions. To contextualize these measures dimensions are defined.

The final efficiency data model is represented in a multidimensional structure using the fact in the centre of the picture, and the dimensions around as showed on Fig. 7.15.

The entities, attributes and measures are described as follows:

Date dimension

- Year. The year of the measurement.
- Quarter. The quarter of the measurement.
- Month. The month of the measurement.
- Week. The week in the year of the measurement.
- Day. The day in the month of the measurement.

Time dimension

- Tariff period. It is the tariff period available Day and Night.
- Day period. Is the corresponding day period, e.g. morning, afternoon,..
- Hour. The hour in the day of the measurement as an integer.

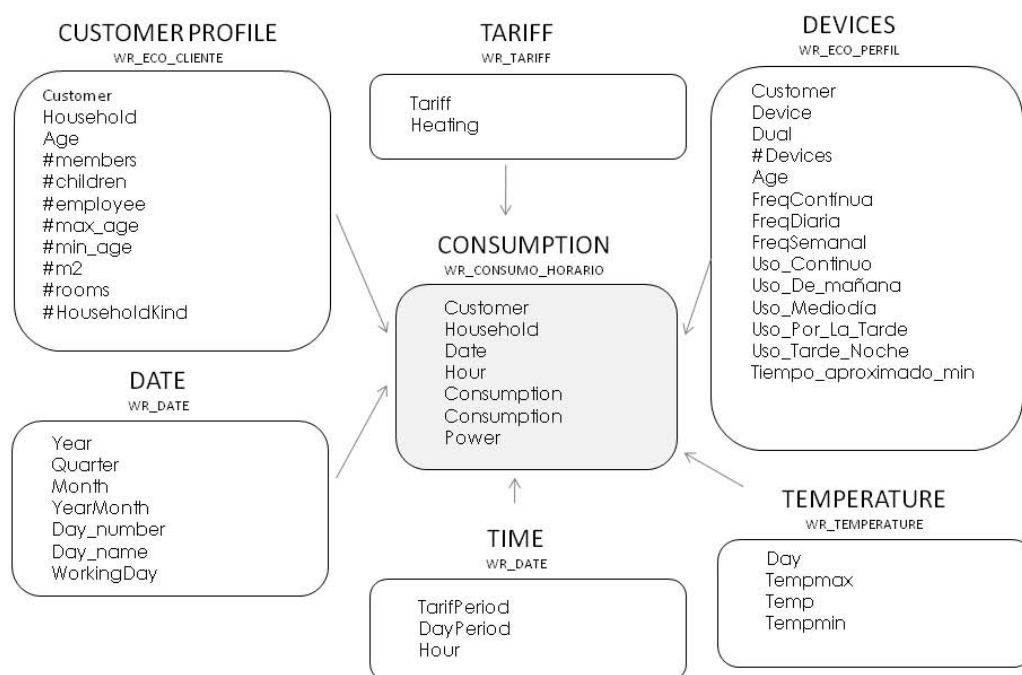


Figure 7.15: Modeling stages - Multidimensional data model.

Temperature

- Tempmin. The minimum temperature for the day of the measurement.
- Temp. The average temperature for the day of the measurement.
- Tempmax. The maximum temperature for the day of the measurement.

Customer Profile

- Id. The id of the customer is used as the key of the table to avoid repetition for each individual user.
- Name. The name of the user.
- #Members. Number of members at the household.
- #Childrens. Number of childrens at the household.
- #Employee. Number of persons working out at the household.
- MaxAge. The maximum age of the persons at the household.
- MinAge. The minimum age of the persons at the household.
- m2. Square meters of the household.

- #Rooms. Number of rooms of the household.
- HouseholdType. Type of household, e.g., detached, semidetached, apartment,...

Tariff

- Tariff. Actual contracted tariff at the household.
- Heating. If electricity is used for heating at household.

Devices

- Device Id. The id of the device used as the key of the table to avoid repetition for each individual device.
- Name. The name of the device.
- Use percentage. The consumption percentage assigned to the device. Note that this value is computed based on the input of the user about the device use specified.

7.6 Analytic modelling

7.6.1 Information Discovery

With the initial data model, the discovery phase is used to iteratively better understand how energy is consumed and how users behave. In later iterations could be included more information in the data model. The objective is to learn how to maximize the advice delivered to each customer. Definitions obtained from this stage will enrich the previous ones defined by experts or will support them. Knowledge discovered is verified numerically and contrasted over the global database.

During information discovery it is defined a picture of the information available. It is used as a first step while moving from data to information and should be periodically performed in order to maintain the application advice up to date. Later reviews, after application deployment, will ensure that the information remains valid, or identify new issues relevant enough to be included in later versions.

At this stage it is followed a process of question answering, questions are formulated by the experts and analyst, and answers are obtained navigating through the data. While navigating and looking for answers to the questions formulated, dimensions context change to analyze specific measures behaviour and information is aggregated or detailed. With this purpose visual analytics are used to support numeric results and increase understanding.

For example if using a two price periods tariff can benefit those customers with high consumption at night, how many customer could benefit from this advice? what are the average savings for those customers? Does the consumption per day period remain along

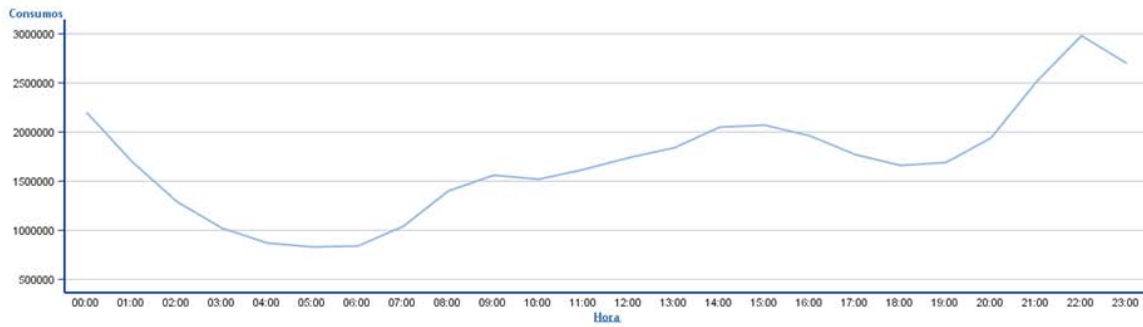


Figure 7.16: Hourly consumption load curve.

the year?. Or for example, it is understood that advice to the user about the consumption in the midnight should be useful because is mainly consumption of standby devices and therefore useless. This rational recommendation has some additional considerations before to be applied, due to the fact that some of the users work at night, and therefore the day period related to the standby consumption was not at midnight but during the morning.

Customer Behaviour

The results of the discovery phase increase the knowledge. To reflect the process followed, is presented a description of analysis used to conclude some of the key issues required to understand customers behaviour.

The first step was to identify the hourly load curve for the average of the customers as showed on Fig. 7.16. Basically users present a load curve that can be easily explained. The load curve reflects the customer behaviour along the different hours of the day.

- During the midnight (00:00 - 04:00) consumption is the lowest of the day. Usually in this day period only standby consumption and the fridge are consuming energy.
- During the morning (04:00 - 08:00) household start activity and the consumption increase.
- In the midmorning (08:00 - 12:00) the household activity continues, while cleaning the house, cooking and leisure activities.
- In the midday (12:00 - 16:00) consumption increase due to the cooking activities and the increase of number of people at home.
- In the afternoon (16:00 - 20:00) the consumption start to decrease due to the lower activity.
- In the night (20:00 - 00:00) consumption increase, this is the day period when most of the people remains at home and day light is lower.

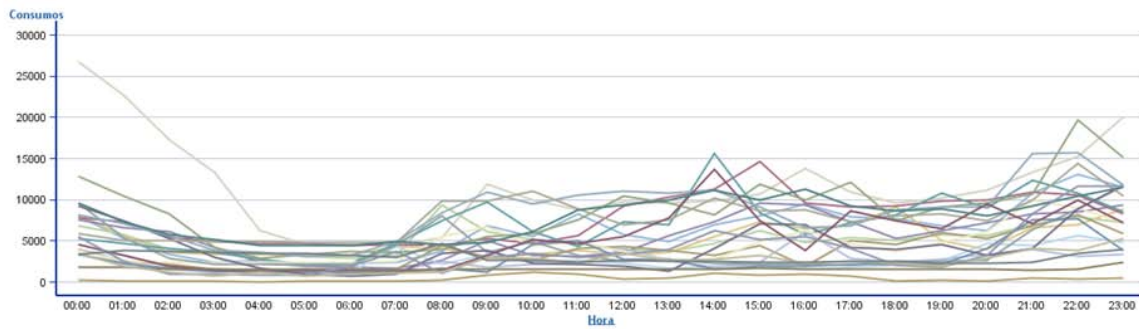


Figure 7.17: Hourly consumption load curve for a set of customers.

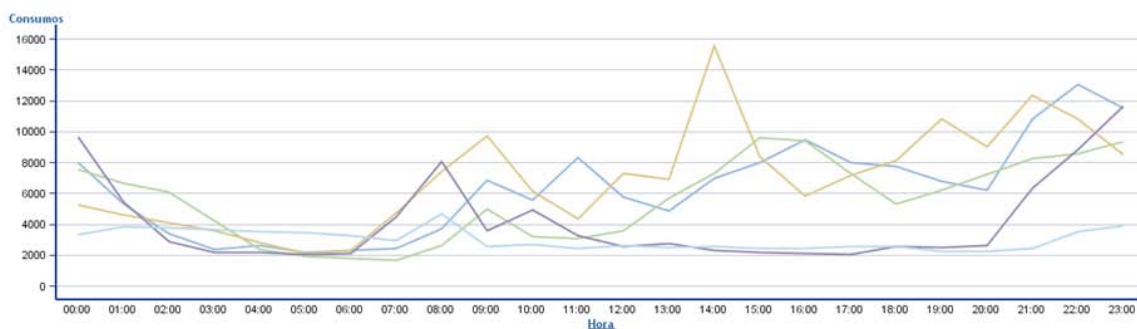


Figure 7.18: Hourly consumption load curve for a subset of customers.

The average load curve changes for each individual customer, each geography, each household demographics or each devices equipment. So when looking the hourly load curve for each individual customer the picture shows high dispersion as showed on Fig. 7.17. When this curve is analyzed per customer no common pattern is easily interpretable. In a set of customers can be reviewed different behaviours where no common patterns seems to appear and distance to the average load curve differs from one customer to the other.

If a smaller set of customer is analyzed can review different behaviours as showed on Fig.7.18. Consumption seems to have some similarities between some of customers. Note that the consumption load curve of the light blue customer is similar to the customer in purple, although his consumption is quite lower. Also the customer in green has a similar pattern of consumption to the one in blue, the total amount of consumption along the day is also quite similar.

- The customer in purple has a high peak in the morning at 8 and consumption decrease from there till 10:00 probably because cleaning activity at home. The consumption remains at the lowest level along the rest of day till 20:00, probably when the user comes back to home from work, from here till 23:00 are registered the highest levels of consumption.
- The customer is yellow has the peak in the morning at 9:00, from here consumption decrease. At 12:00 there is an increase of consumption due to work at home. The

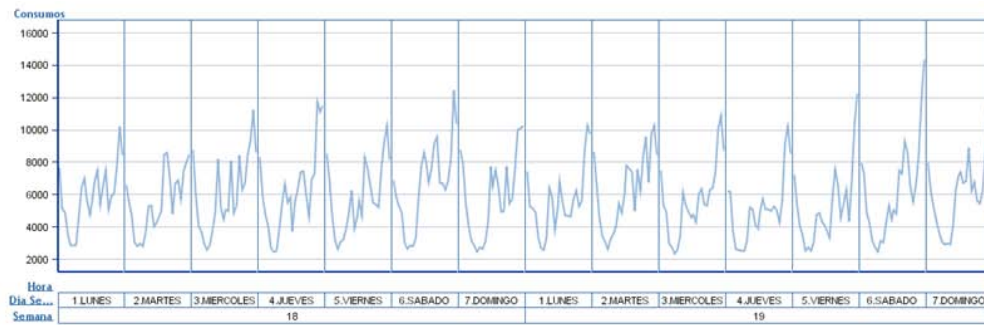


Figure 7.19: Hourly consumption load curve by day.

highest peak of the customer is at 14:00 at lunch time, probably some other activities are performed in this period. In the afternoon there are two peaks at 19:00 and 21:00, the consumption decrease till standby consumption.

- The customer in blue has a more plained curve. The peak in the morning is at 9:00, and there is another one at 11:00. From here the consumption decreases till 13:00. In the afternoon there is another peak at 16:00 and consumption reduces at 20:00. From here it is registered the highest consumption till 22:00, then decrease rapidly.
- The customer in green has a small peak in the morning at 9:00. Consumption increase again at midday for eating and remains a little higher along the rest of the dat. There is a second peak in the afternoon at 13:00, probably for cooking. From here it is registered the highest consumption till 22:00, then decrease rapidly.
- The customer in light blue has a really low consumption. The consumption peak in the morning is at 8:00, and it remains close to standby consumption till 22:00 at night. No activity appears to occur.

When consumption is visualized per customer along the weeks, a similar behaviour spreads along the different days for the same customer as showed on Fig. 7.19. The behaviour of each individual customer seems to remain similar along the different days, and so do for a different set of customers as showed on Fig. 7.20.

7.6.2 Customer behavior context analysis

Domestic energy consumption depends on several factors like the location, socio-demographic characteristics and technology used as heating systems or the efficiency of appliances.

Customers maintain a similar consumption in time, but there are differentiated patterns between different customers, it seems to be due to differences in external and internal factors. In order to understand the behaviour is required to analyze the individual impact of the factors.

The impact of internal and external factors is showed on Fig. 7.21.

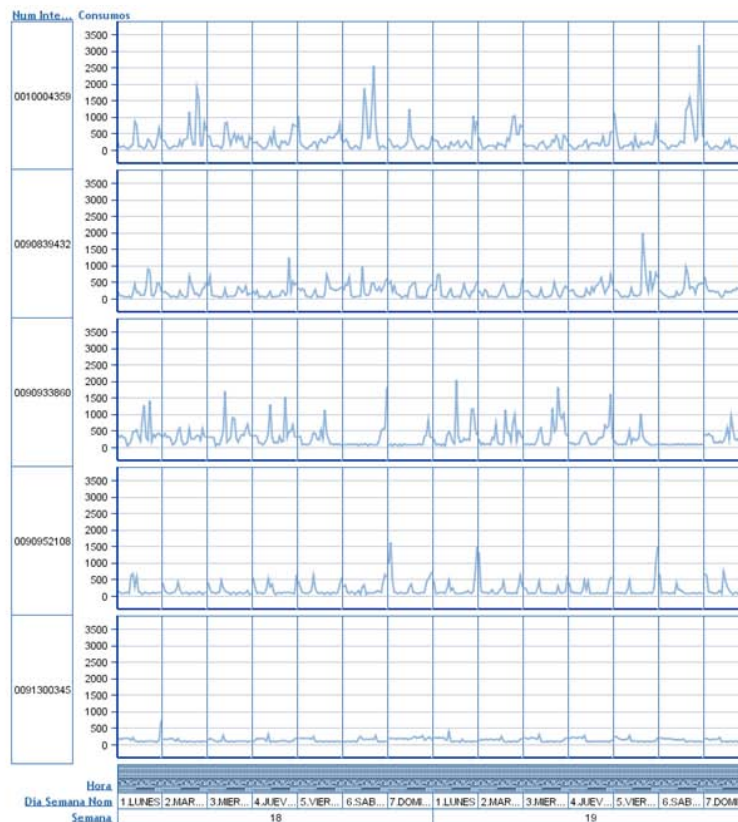


Figure 7.20: Hourly consumption load curve by day and customer in detail.

- Seasonality

The temperature is the variable with highest impact in consumption behaviour. Energy consumption is influenced due to temperature and the number of hours with day light.

Depending on the season there are two key periods, in summer from June to September with the use of air conditioning and the winter from November to April with the use of heating. There are intermediate months where temperature is medium, and the influence is not so crucial. The variation between summer and winter for household with electric heating is around twice the consumption in summer. For a household with air conditioning the increase of consumption in summer is similar to the consumption in winter.

- Working patterns

The difference of energy use is mainly underlined between working days and bank holidays. The effect of holidays is quite important as consumption reduces while people stay out of home. There are also different behaviours because of the working hours in the household.

- Age

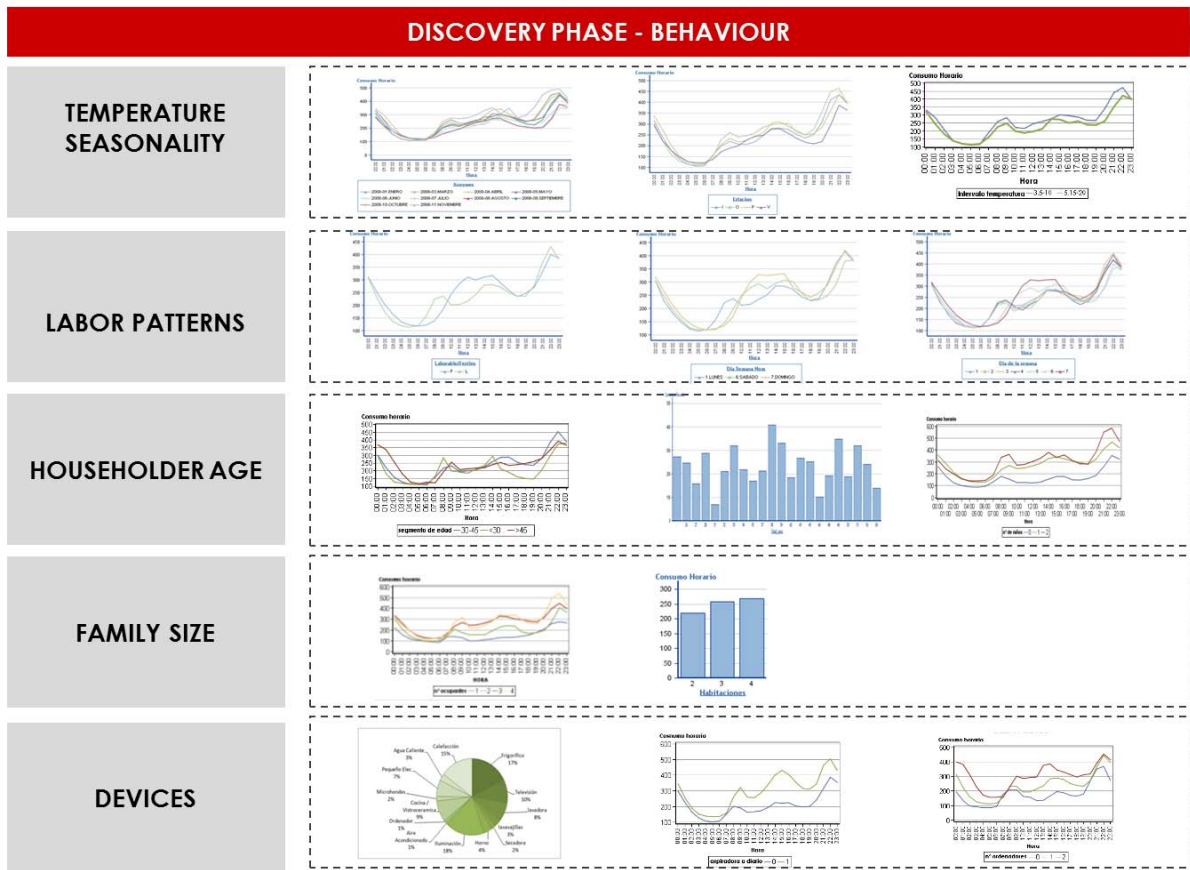


Figure 7.21: Impact on behavior of external factors.

The age of the older member of the family and the age of the younger are key factors in behaviour, e.g. retired people have smaller consumption, and families with babies increase consumption.

- Members at household

The number of members at the household and the size of the household are also patterns that define the final consumption. Peaks are higher due to the higher activity at home.

- Devices

The number of devices at each household will be deterministic in the final consumption. The kind of devices, their efficiency and the way of usage has a great impact on the final consumption. The consumption of the air conditioning or heating supposes more electricity than any other device. The fridge placed in most of the households is the third device with higher consumption, the efficiency of the fridge is quite relevant to get savings.

7.6.3 Customer behavior segmentation

Customers demand energy but efficiency cannot be measured by a specific defined formula. The approach used here is the use of references, so just by comparison between

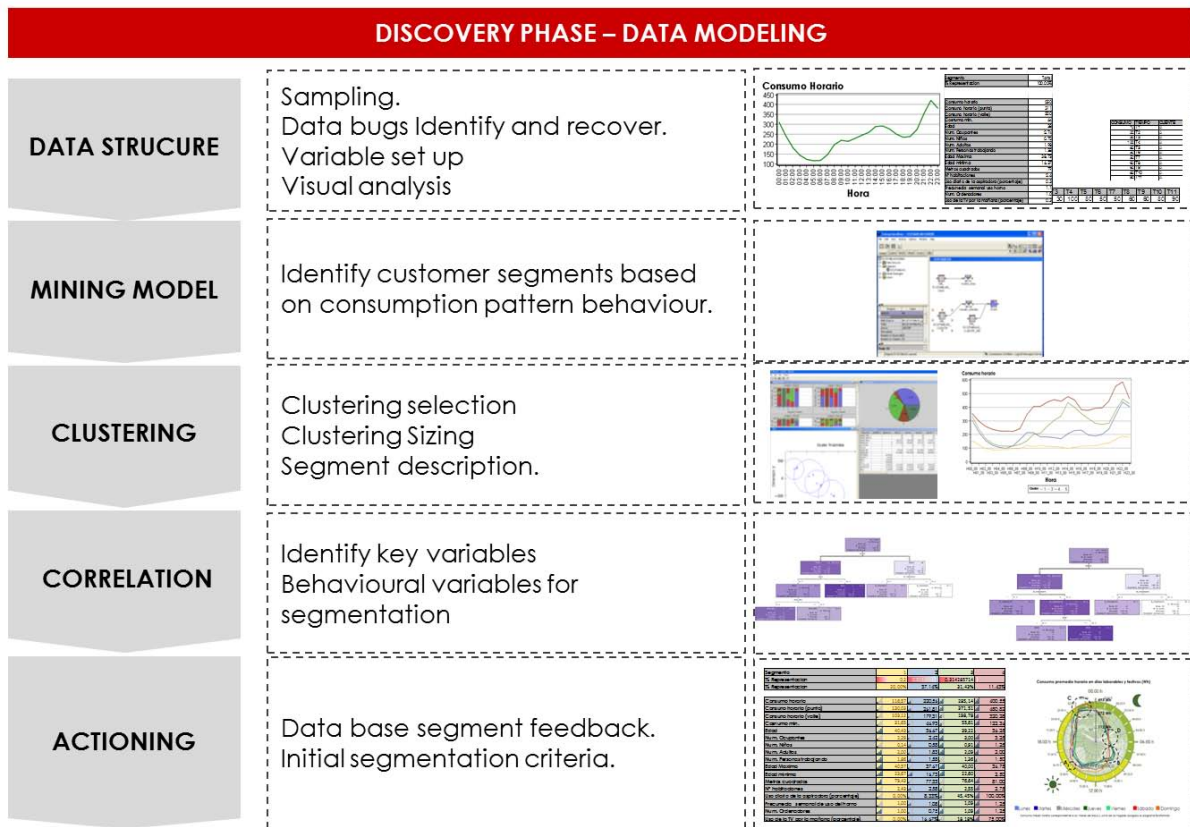


Figure 7.22: Modeling stages.

customers energy demand efficiency can be reported. For example having two customers one with a consumption of 1.000 kWh per year and another with a consumption of 3.000 kWh, which one is the most efficient?. At a first sight someone could conclude that the second customer with a consumption of 3.000 kWh is less efficient. But what if it is found that the second customer has a family of five members and uses electricity for heating and the first one is a single with gas heating?.

This kind of behaviour concludes that it is necessary to divide the users in groups of similar characteristics. Here a customer advice is translated in a qualitative measure related to the group of household characteristics. It allows giving a personalized advice to customers having a reference with customers with similar characteristics.

Analytic Modelling Clustering

The modelling technique used to segment customers is clustering. Clustering analysis is a generic term for a large number of techniques that have the common aim of determining whether a multivariate data set contains distinct groups or clusters of observations and, if so, find which of the observations belong to the same cluster. The process followed is showed on Fig. 7.22. It is not the purpose here to give a detailed description of the clustering computation models that could be used. To parameterize the clustering model has been used SAS Enterprise Miner. For data transformation has been used the SAS

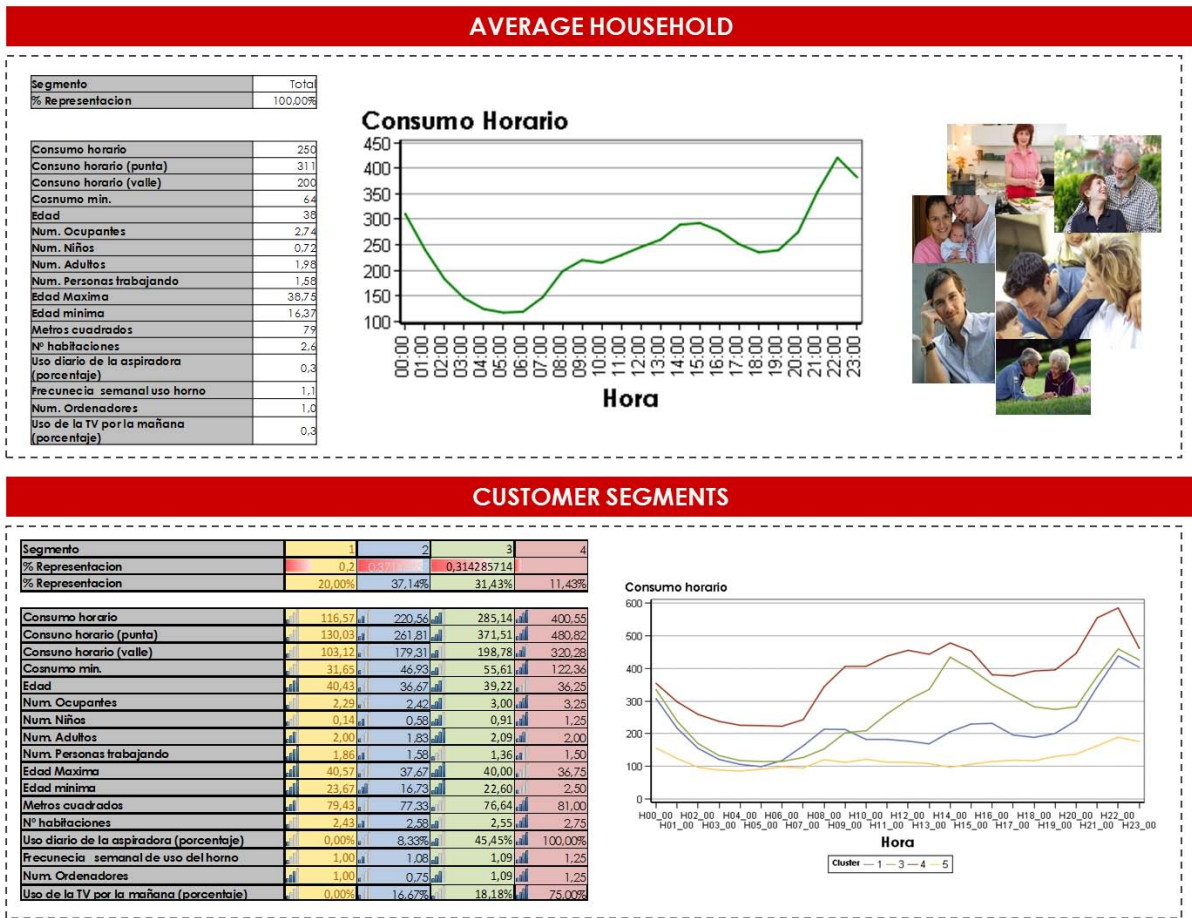


Figure 7.23: Customer segmentation.

Enterprise Guide.

The objective of clustering is to find similar users in behaviour in order to use the average consumption of the cluster as a reference for each individual customer in the segment. Results will be that some of the users will be over the average and some others above. The users above will try to reduce their consumption and therefore the segments characteristic will change over time. One of the expected behaviours is to reduce the total consumption. Once the non efficient consumers of the segment change their habits, the effect will imply that the average behaviour of the overall segment will improve and therefore along the time each segment will be more exigent in terms of comparison with the customers inside.

The results of the model are showed on Fig. 7.23. There were defined four clusters with clear distinctive characteristics.

Note that the data has been transformed in order to meet the clustering objective: Find customers with similar patterns of behaviour. Each record in the initial database is transposed and aggregated to generate a load curve for the last 12 months. The results have been proved in terms of covertures and precision and final clustering was a deliver-

able by itself with the description of each cluster.

7.7 Report Refinement

Using the results obtained from the previous discovery phase and available data sources the report is redefined. The report is understood as a set of topics, where each topic represents a specific view of the data. Each topic has a significant amount of related information in order to clarify customer understanding and report structure.

The report is organized in order to provide information going from a general overview of the user behaviour to a more detailed description, going from general knowledge to particular one.

Also experts initial knowledge could be extended due to discovered behaviour or rejected due to unproven results. For example one important point to be used as advice to the user is the recommendation to move consumption from the charged price periods to the bonified periods. This fact could reflect that this advice should be rejected if it only occurs in a small subset of users and therefore to work around this kind of advice is useless.

Finally expert knowledge has to be modelled in terms of formal mathematical and logical definitions. The analytic process is the only one that can deliver this final knowledge that will be used as the input for the final advisory system. For example the recommendation to change from a one price period tariff to a two price periods tariff has to be modelled in terms of fixed values. If consumption in the low cost period is a 60% of the total consumption and it remains during the most of the year then can be applied to the user to send a recommendation.

Also the degree used in linguistic terms has to be modelled in precise terms, as Zadeh argued "*fuzzy logic is a precise logic of imprecision and approximate reasoning*". Degrees of recommendation has to be measured when delivered to the customer as it could be highly recommended to change the tariff, or it could be recommended in a lighter expression, e.g., it seems that you could change to a tariff with two periods due to your actual consumption.

Due to all this stages the final report is defined as follows in different sections. The sections are structured like a discourse, going from the general overall terms to concrete individual items.

The report contains the following sections:

- Overall efficiency

Here it is defined the overall efficiency based on the energy label. It is rated in terms of consumption compared with similar households and the distribution of consumption per period of the day and tariff.

- Consumption description

Here it is used the comparison with related segments, e.g., to measure if consumption is high or low.

- CO2 Emissions

This information is translated into more understandable values to the final users, as the amount of Kg emitted are meaningless to the users. The equivalent kilometers and the equivalent trees required to capture the CO2 emissions are used to complete the information.

- Expenses

The amount of money expend in energy consumption is usually more easily understandable that the amount of KWh of consumption of the users. Also it is a factor that can include more control over behaviour of consumption patterns.

- Evolution of consumption per month

The evolution of consumption per month show the patterns followed and help to identify specific measures.

- Hourly Consumption

Hourly consumption gives a reference about the use of energy along the day. This could help to take decisions about their use and make corrections.

The hourly consumption and the tariff are used to advice about savings reducing the electricity consumption during day.

The generation of linguistic sentences, translating numeric data into fuzzy linguistic terms, will be used in the report while describing the data. Emotional behaviour is included by defining the instantiation linguistic sentences associated to each perception value.

- Identify linguistic variables to apply to the report and associated language and emotional content.
- Generated and identify computations required for recommendations.

7.8 Evaluation

The evaluation activity is used to describe the results, get feedback from the customer and get acceptance. Note that evaluation has been performed at different stages along the analytic phase, but this is a more formal step.

Evaluation is performed in a final step with the customer in order to get the acceptance of the report in terms of usability and understandability.

Not only customers but user feedback is key at this stage as usually while developing it is lost the point of view of a final and standard user.

7.9 Build the data model

Once the final redefined report is accepted, the data model defined can be built. Information to load the data is described due to ETL (Extraction, Transformation and Load Data).

The only information that will be used in a report is the available in the data model, it is true that the information will be enriched with other knowledge like semantic, lexical and grammatical values, but the primary base of the knowledge resides in the database.

At this stage of the project data was loaded from a text a file and it is not required a detailed definition of extraction transformation and load of data.

Chapter 8

Energy Advisor Development

8.1 Introduction

This chapter describes the implementation of the Personal Energy Advisor. It is performed due to the analysis review, the design definition, the implementation and the test. This phase is developed due to the analysis definition carried out in the previous phase. Along this phase the project moves to an implemented solution.

Note that this phase is developed over the application framework and the result obtained is the implementation of the Personal Energy Advisor accessible due to the user interfaces.

Use cases from initial analysis are prepared for implementation in the design phase. The main functional stages are underlined in red on Fig. 8.1.

8.2 Activities

The activities for the implementation of the Personal Energy Advisor are defined and described with a review of the methodology introduced in the previous chapter. The main activities followed in the process and showed on Fig. 8.2 are :

- Analysis
Analyze contents as use cases defined in previous phase for design, e.g., input data, computations, lexico grammatical components and emotions.
- Design
Design the knowledge domain for implementation with a specific design for each use case, e.g., physical, conceptual and linguistic.
- Implementation
Implementation of the use cases designed in the framework, e.g., physical interface, conceptual interface and linguistic interface.

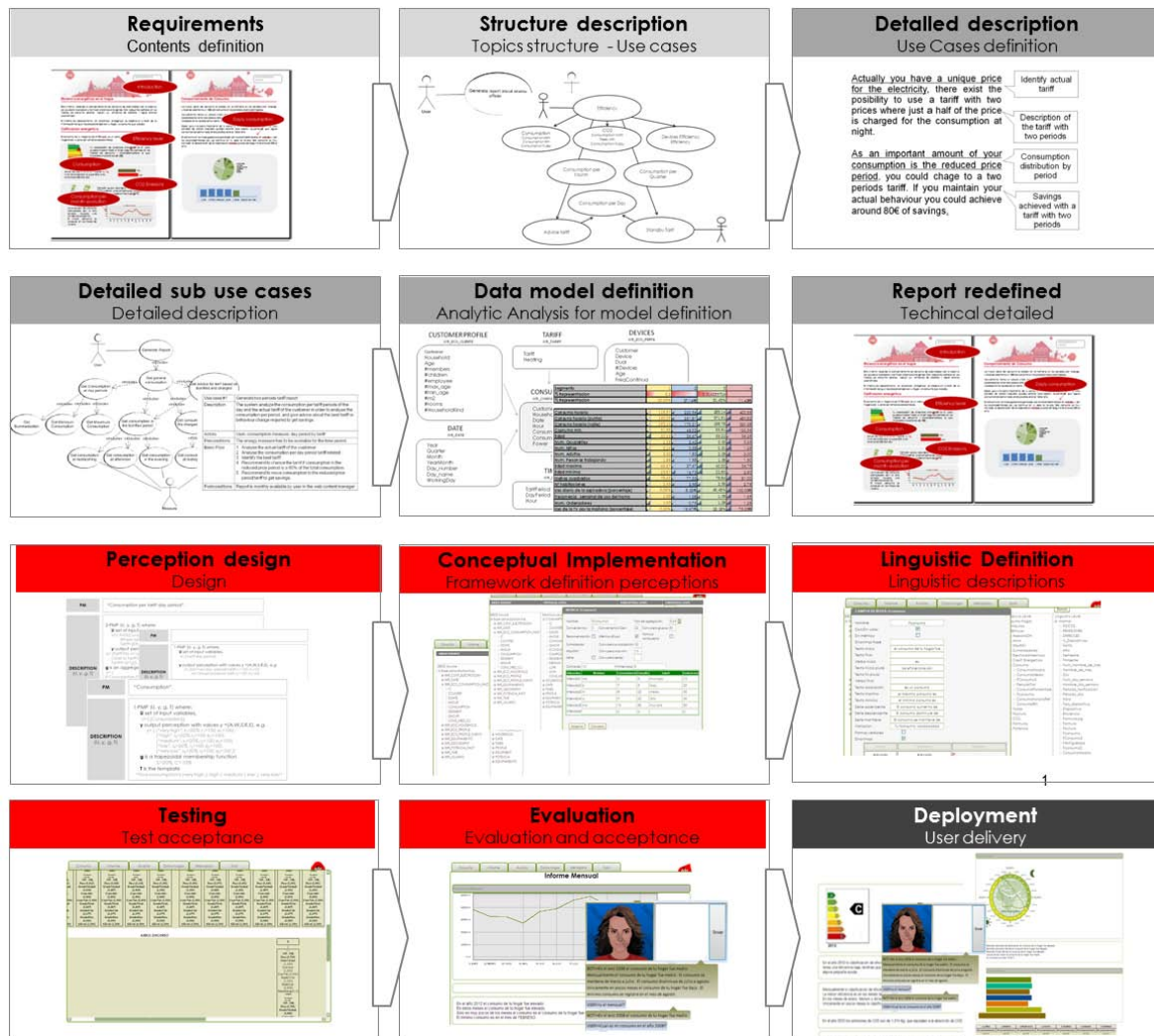


Figure 8.1: Overall methodology with the development stages in red, namely perceptions design, conceptual implementation, linguistic definition, testing and evaluation.

- Test

Test plan definition and validation of the solution.

The outputs of this phase are a closed design for approval, and the tested and accepted implementation.

8.3 Analysis

Here the behaviour of the Personal Energy Advisor is reviewed. In fact the analysis has been defined in previous phase with the analytic modelling and the report refinement. Here analysis is used to review the detailed use cases in the global architecture. Their description will be detailed in the design phase for implementation.

	Inception			Elaboration					Construction					Transition			Production				
	I1	I2	I3	I4	I5	I6	I7	I8	I9	I10	I11	I12	I13	I14	I15	I16	I17	I18	I19	I20	I21
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10	T11	T12	T13	T14	T15	T16	T17	T18	T19	T20	T21
Build the system																					
Analysis																					
Define architecture																					
Analyze Behaviour																					
Design Components																					
Design data source access																					
Design components																					
Implementation																					
Implementation model																					
Implement components																					
Integrate the system																					
Test																					
Define test plan																					
Verify test approach																					
Test and Evaluate																					
Deployment																					
Prepare the environment																					
Deployment plan definition																					
Support environment for iterations																					
Support																					
Change and deliver configuration items																					
Manage Releases																					
Manage Change requests																					

Figure 8.2: Methodology.

Analysis starts with a review of the defined use cases by the designer with the analyst in order to get understanding. Note that the report has been used to define the contents, not only for a generic and individual report, but to define the knowledge domain. Use cases and the architecture is showed on Fig. 8.3.

The architecture is reviewed in terms of channels to be used in the communication with the customer and requirements for integration. The behavior is analyzed in terms of implementation and the integration with other systems. In the solution proposed the application framework is used as the direct channel to be used by the customers. The application framework as described previous has two user interfaces, the report definition and the Avatar.

8.3.1 Data Model Review

The input data for the model has been defined in previous phase as a set of measures and dimensional attributes. They have been built in the database as defined in the data model.

- Measures $M = \{Consumption, Savings\}$ that take values:
 - $Consumption \in \{Low, Average, High\}$.
 - $Savings \in \{More\ than\ 100\ euros, Around\ 80\ euros, around\ 50\ euros, around\ 30\ euros\}$.
- Customer $\{Id, Tariff\}$ with values:

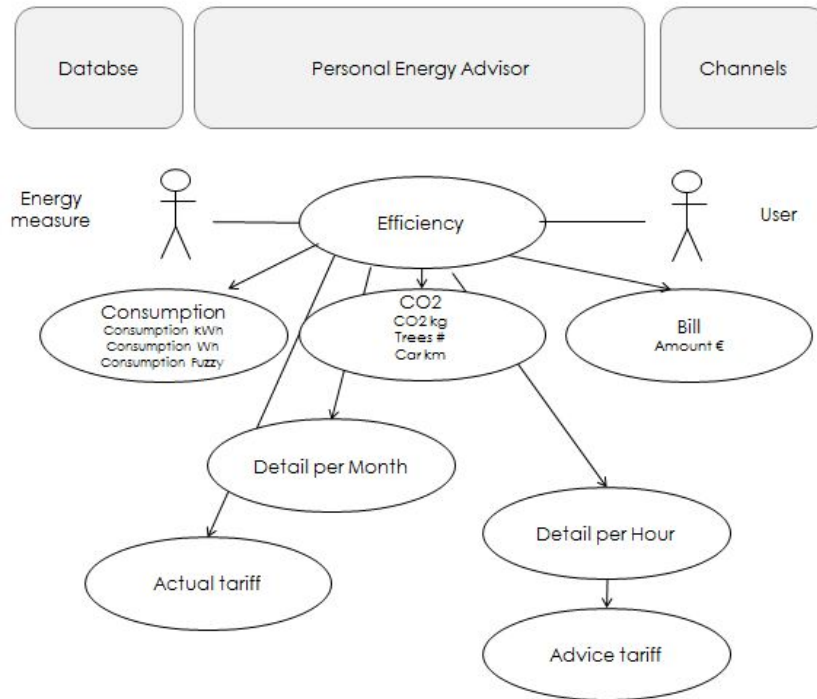


Figure 8.3: Use cases review.

- $Tariff \in \{Oneperiodtariff, Twoperiodstariff\}$ (crisp).
- $Time: \{Hour, DayPeriod, CostPeriod\}$ with values:
 - $Hour \in [1, 24] \in N$ (crisp).
 - $CostPeriod \in \{Charged, LowCost\}$ (Fuzzy). Households pay less for their energy consumption during low cost period hours, so the higher the consumption in this period the better the behavior maintaining a rational global consumption.
 - $DayPeriod \in \{Morning, Midday, Evening, Night, Midnight\}$ (Fuzzy). Households use energy along the different periods of the day based on the mains activities at household.
- $Date: \{Year, Month\}$ with values:
 - $Year \in N$ (crisp).
 - $Month \in \{Jan, \dots, Dec\}$ (crisp).

8.3.2 Use Cases Review

The perceptions are detailed for each use case. A perception is defined using input data, output expressions and computation required. Here the designer links for each use case of described perceptions the elements of input data , computations required, lexico grammatical components and emotional contents. Here are detailed some examples for tariff

PM	"Consumption".
DESCRIPTION (U, y, g, T)	<p>1-PMP (U, y, g, T) where:</p> <p>U set of input variables, U={ (Consumption)}</p> <p>y output perception with values $y=(A,W,S,R,E)$, e.g. $y=\{$ ["very high", $s_1=20\%$, $r_1=100$, $e_1=100$], ["high", $s_2=20\%$, $r_2=100$, $e_2=100$], ["medium", $s_3=20\%$, $r_3=100$, $e_3=100$], ["low", $s_4=20\%$, $r_4=100$, $e_4=100$], ["very low", $s_5=20\%$, $r_5=100$, $e_5=100$]} g is a trapezoidal membership function $S_1=20\%$, $C=10\%$</p> <p>T is the template "Your consumption is (very high high medium low very low)"</p>

Figure 8.4: Analysis definition for Consumption.

and consumption per day period:

Tariff is a tuple (U, y, g, T) where:

U the actual tariff, $U = \{ 2,0A, 2.0DHA \}$.

y is the output CP with values $y = (A, W, R, S)$, e.g., $y = \{ (\text{"unique price"}, 0.0), (\text{"two periods price"}, 0.68) \}$.

g It is directly translated by a mapping function

T is a simple template: "Actually you have a (| unique price | two periods price) for the electricity.)"

Consumption by Tariff period is a tuple (U, y, g, T) where:

U consumption in the overcharged period $U = \{ \text{Consumption overcharged period} \}$.

y is the output CP with values $y = (A, W, R, S)$, e.g., $y = \{ (\text{"Really high"}, 0.0), (\text{"High"}, 0.68), (\text{"Medium"}, 0.68), (\text{"Low"}, 0.68), (\text{"Really Low"}, 0.68) \}$.

g It is implemented by a membership function

T is a simple template: "Actually you have a (| unique price | two periods price) for the electricity.)"

Efficiency is a tuple (U, y, g, T) where:

U is the set of input CPs $U = \{u_1, u_2, u_3\}$, $U = \{ \text{Consumption}, \text{LowCostPeriodConsumption}, \text{DeviceEfficiency} \}$.

y is the output CP with values $y = (A, W, R, S)$, e.g., $y = \{ (\text{"Highly inefficient"}, 0.0), (\text{"Inefficient"}, 0.68), (\text{"Moderately efficient"}, 0.32), (\text{"Highly efficient"}, 0.0) \}$.

PM	Query: "Efficiency".
DESCRIPTION (U, y, g, T)	<p>2-PMP (U, y, g, T) where:</p> <p>U set of input variables, U={ Consumption , Low cost consumption Consumption.P2}</p> <p>y output perception with values $y=(A,W,S,R,E)$, e.g $y=\{["A", 0.0],$ $["B", 0.68],$ $["C", 0.32]\}$</p> <p>g is an aggregation function using a set on mamdani-type fuzzy rules IF [Consumption is Low] AND [Consumo.P2 is High] THEN Efficient IF [(Consumption is Medium) OR (Consumption is Medium)] THEN Moderately efficient IF [Consumption is High] AND [Consumo.P2 is Low] THEN Inefficient</p> <p>T is the template "Your behavior is (highly inefficient inefficient moderately efficient highly efficient) Emotional content "It is really a poor efficiency!" "Great efficiency! Well done!"</p>

Figure 8.5: Analysis definition for Efficiency.

g We implemented the aggregation function $W = g(u_1, u_2, u_3)$ using a set of Mamdani-type fuzzy rules as follows:

IF (u_1 is *High*) AND (u_2 is *Low*) AND (u_3 is *High*) THEN *Inefficient*

IF (u_1 is *Medium*) AND (u_2 is *High*) AND (u_3 is *High*) THEN *Moderately efficient*

IF (u_1 is *Medium*) AND (u_2 is *High*) AND (u_3 is *High*) THEN *Moderately efficient*

...

IF (u_1 is *Low*) AND (u_2 is *High*) AND (u_3 is *Low*) THEN *highly inefficient*

Here, the perception of efficiency is defined based on the study of available data about consumption of families and implemented as expert knowledge.

s is computed by the fuzzy state machine.

T is a simple template: "Your behavior is (highly inefficient | inefficient | moderately efficient | highly efficient)". Note that there is a template assigned to each emotional state. High efficiency is correlated with a efficiency rate A, and the lowest efficiency is correlated to a efficiency rate C.

An example of analysis definition for Consumption and Efficiency perceptions is showed on Figures 8.4 and 8.5. These examples will be described later in the design and implementation phases.

8.4 Design

The design activity of the system is performed iteratively. The knowledge base is designed and next developed following the generic sequence of prototypes defined in the process. The prototypes are intended to help the development of the system in the initial stages, since the first prototypes will be very basic and the following prototypes will have

increased complexity.

The designed components imply a view of use cases based on a physical, conceptual and linguistic definition. The designer moves closer to the implementation. According to the definitions of use cases, the designer associate to perceptions the input data required.

During design the database access is defined in technical terms for later implementation. Also the report structure is defined, showing the content following the order and output required.

The conceptual level includes several definitions to model concepts required for perceptions generation. Concepts or semantic data elements that compose perceptions are defined from the entities, attributes and measures previously defined in the physical data model or over other concepts previously defined at the concept layer. Physical data is transformed into closer understandable references for the user independent from the physical structure.

The linguistic level defines the lexical and grammatical items to facilitate the instantiation of linguistic sentences from the underlying perceptions. The linguistic definition at the perception level includes the verbal forms required and the linguistic labels used to define the context around for instantiation. Emotional linguistic expressions associated to language and contents are defined in here.

An example of design definition for Consumption and Efficiency perceptions is showed on Figures 8.6 and 8.7.

8.5 Implementation

The implementation of the system is performed once designed, e.g., implementing prototypes of increasing complexity, iteratively and sequentially.

Implementation is performed over the application framework. Data defined in the database model is mapped into the application framework as a data source. Later is linked in the physical definition to concepts and relations between entities mapped. In the conceptual level the perceptions elements are defined, directly related to input data or other perceptions.

An example of implementation definition for Consumption and Efficiency perceptions is showed on Figures 8.8 and 8.9.

8.6 Test

Although individual tests have been performed for each development, here a more global testing is performed, e.g., system testing and acceptance testing.

PM	"Consumption".
Physical Entities and attributes or measures	WR_CONSUMPTION.Consumption
Conceptual Computations and linguistic fuzzy definitions	<p>Convenience:20 Type: <i>Fuzzy Measure</i> Trapezoidal membership function Confidence=10% Values</p> <p style="padding-left: 20px;">Label: "very low"; Convenience:100; Size:30; Evaluation: 100 Label: "low"; Convenience:50; Size:10; Evaluation: 80 Label: "medium"; Convenience:0; Size:20; Evaluation: 60 Label: "high"; Convenience:50; Size:10; Evaluation: 20 Label: "very high"; Convenience: 100; Size:30; Evaluation: 0</p>
Linguistic Lexical and gramatical expression	<p>Output sample: "Your consumption is (very high high medium low very low)"</p> <p>Verb: to be Perception context: "Your consumption is" Include linguistic values</p>

Figure 8.6: Design definition for Consumption.

Tests performed are functional and non functional. Functional from the point of view of results obtained to check the capability and ability to deliver the intended results. Non functional from the point of view of performance working conditions.

The acceptance test is the last test that is performed before the software is delivered to the users. It is carried out to ensure that the software that has been developed meets all customer requirements. Should be used two types of acceptance testing, one that is carried out by the members of the development team and it is known as internal acceptance testing or Alpha testing, and the other that is carried out by the customer and it is known as external acceptance testing or Beta testing.

Detailed steps in perceptions generation for the content and discourse trees for testing purposes are showed on Fig. 8.10.

8.7 Deployment

The goal of the Deployment Phase is to place the solution into a production environment with access for users. It is the transitioning of the project to operation and support. After deployment should be performed a project review and a customer satisfaction survey. Stabilizing activities may continue during this period.

To prepare the deployment, the physical infrastructure, system software and application software are installed and configured in the production environment, e.g., operating system, database platform, network configuration. Moving from the development to the

PM	"Efficiency".
Physical Entities and attributes or measures	Consumption , Tariff Period
Conceptual Computations and linguistic fuzzy definitions	<p>Convenience: 100 Input Perceptions: Consumption, Consumption.P2 Type: Recommendation Values</p> <p style="padding-left: 20px;">Label: "A"; Convenience: 100; Evaluation: 100 Label: "B"; Convenience: 100; Evaluation: 80 Label: "C"; Convenience: 100; Evaluation: 60</p>
Linguistic Lexical and grammatical expression	<p>Output sample: "Your behavior is (highly inefficient inefficient moderately efficient highly efficient))</p> <p>Verb: to be Perception context: "Your efficiency is" Include linguistic values Emotional content: "It is really a poor efficiency!" "Great efficiency! Well done!"</p>

Figure 8.7: Design definition for Efficiency.

production environment requires a migration of the metadata and the multidimensional database. It is basically a copy of the databases already defined.

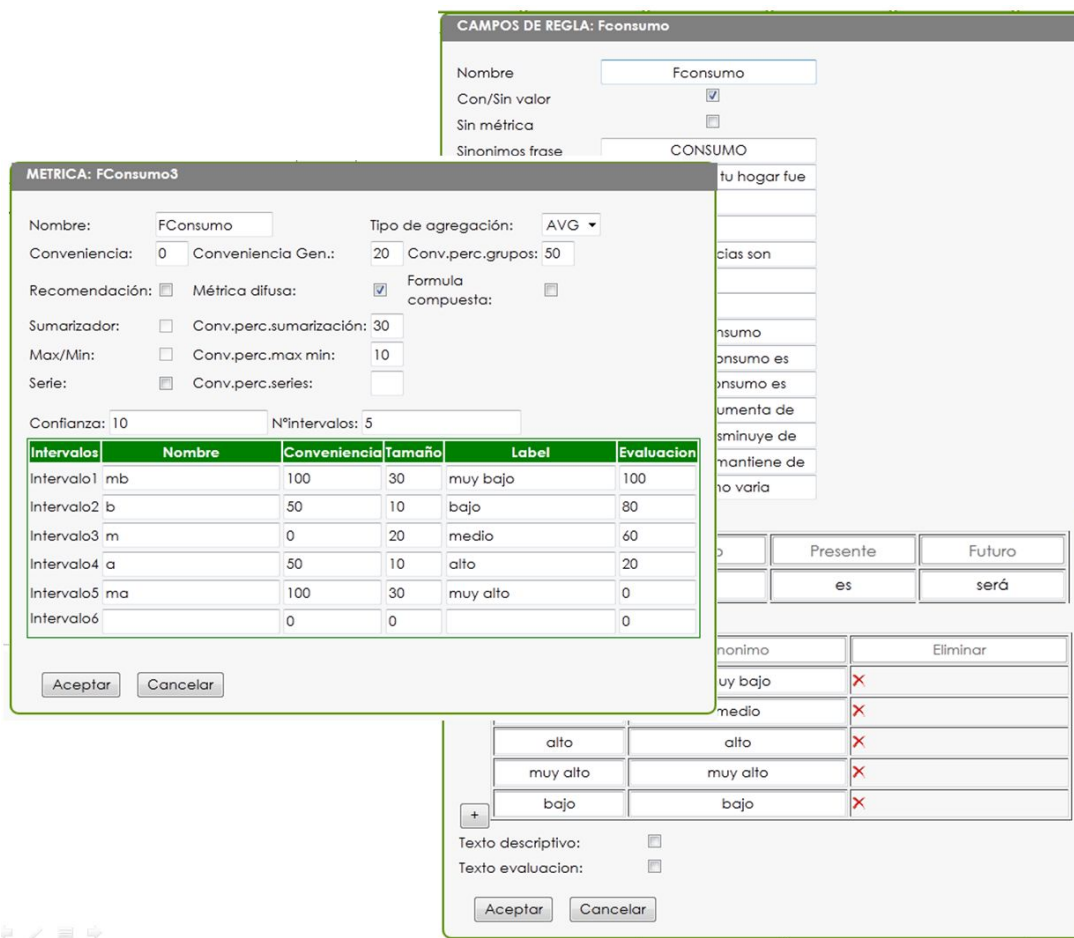


Figure 8.8: Implementation for Consumption.

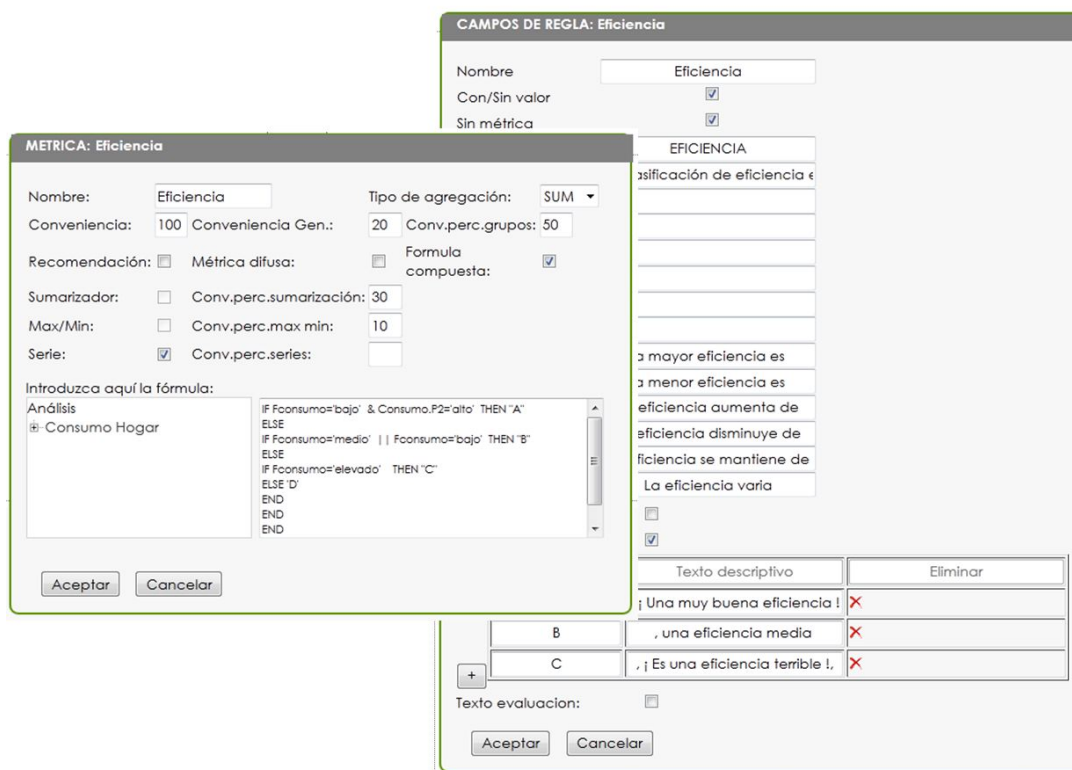


Figure 8.9: Implementation for Efficiency.

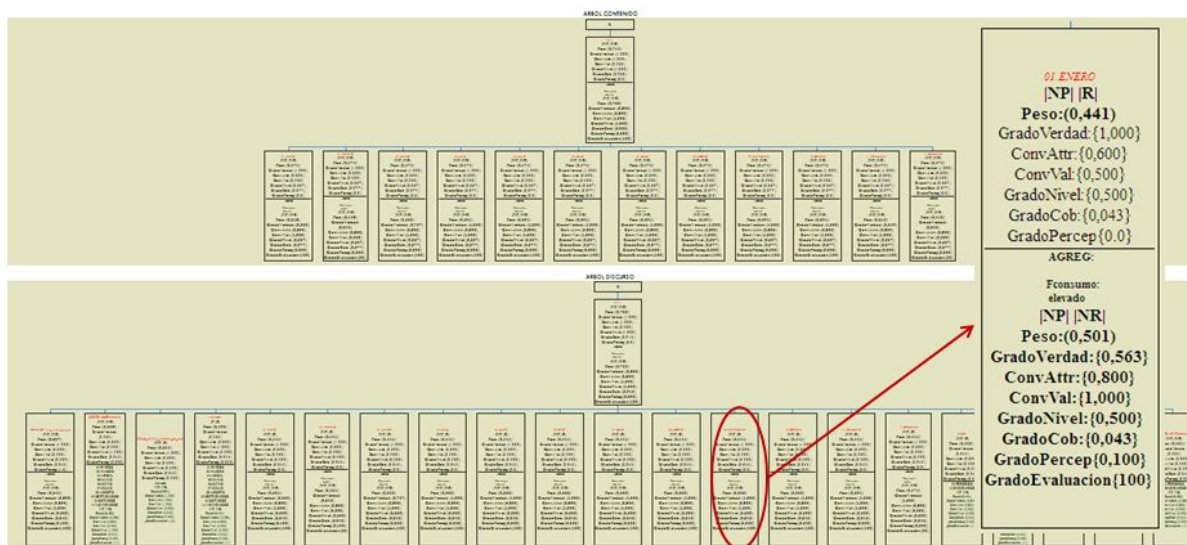


Figure 8.10: Detailed steps in perceptions generation for the content and discourse tree are facilitated in the framework for testing purposes.

Chapter 9

Functionality of the Personal Energy Advisor

9.1 Introduction

This chapter describes the functionality of the Personal Energy Advisor from the user perspective. The purpose is to review the functionality covered and the communication abilities. It is performed using different scenarios underlying specific abilities. The developed Personal Energy Advisor is tested for both the report and the avatar interfaces over the application framework.

The interaction of the user with the Personal Energy Advisor is performed due to input queries defined in the user interfaces:

- The reporting interface allows users to define reports by selecting the underlying information dragging and dropping attributes and measures into the report, e.g., *selecting the measure of consumption and selecting the filter month to January*. The report once executed retrieves from the knowledge domain requested data and processes a natural language personalized report as response.
- The Avatar interface allows users to interact using natural language. Here the user can query the system using natural language in order to get the requested responses, e.g., *“Tell me my consumption in January.”*

The application engine is called from the user interface while executing a report or while chatting with the Avatar. The application engine is in charge of giving response to input queries. The engine processes the input query identifying the concepts defined in terms attributes and measures following the objective of getting insight over the consumption behaviour.

While performing queries in the application framework, the user navigates through the knowledge base, extracting in each query a particular subset of data. Each query of the user is computed by the interpretation of the request and the computational capabilities processed over the knowledge domain to generate the related responses.

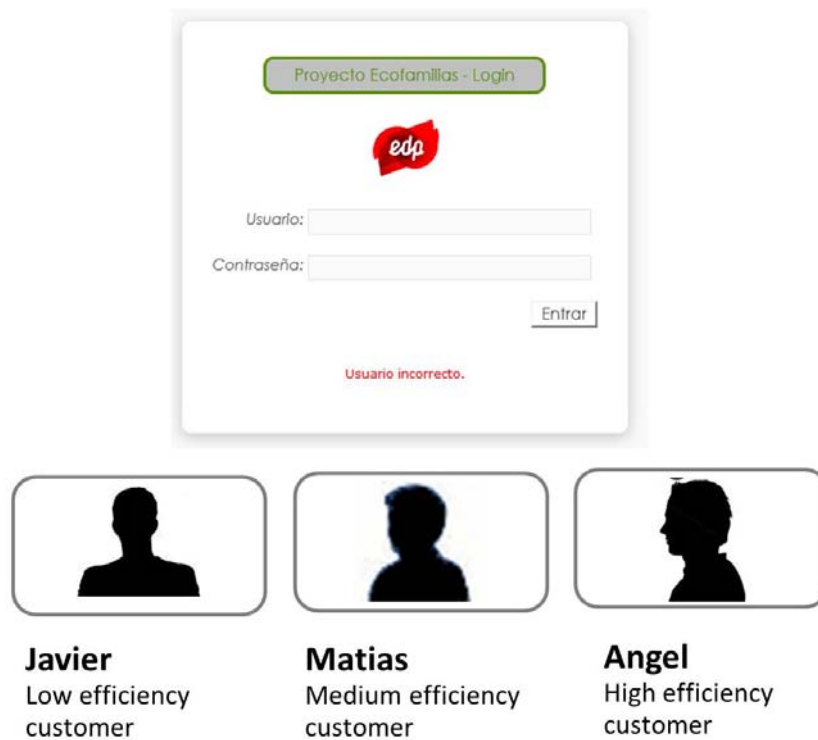


Figure 9.1: Access to the Personal Energy Advisor and users profiles.

9.2 Users

Not all customers have the same questions, the same behaviours or the same needs. Therefore, the Personal Energy Advisory system deals with certain questions and give only certain prescription answers personalized to users.

To access the Personal Energy Advisor the user logs into the framework with a user id and password to consult his consumption information and behaviour in the use of energy.

For testing purpose a demonstration of use for three customers with different efficiency profiles, namely Matias, Javier and Angel is included. As showed on Fig. 9.1, Matias is a medium efficiency customer, Javier is a low efficiency customer and Angel is a high efficiency customer.

9.3 Channels

The web is the communication channel used in the framework to access the report and the avatar interface. Note that these interfaces could be extended to others, e.g., the avatar could be used in a telephone customer attention centre to interact directly with the users, and the reports could be used to be delivered to the customers in the bill.

9.4 Avatar functionality

Looking for a more intelligent and effective interaction with the customers to achieve an improvement in relationship management, the Avatar interface is able to interact with each customer and make personal suggestions about their behaviour as an expert would do.

The avatar interface allows users to interact just using natural language. This interface processes user questions expressed in natural language and generates natural language answer reports. Users have a closer and easier relation while interacting and getting answers for their questions and concerns.

The Avatar implements a computer animated character and a messenger chat dialog. The chat messages show the ongoing dialog with the history of messages of the user and the advisor.

The avatar knowledge is limited to the concepts defined in the knowledge domain. Based on the dimensional attributes and measures, any combination can be retrieved as text.

The rest of this section is used to describe specific functionality, namely conversation about general topics, advice on the use of energy, the communication of emotions, the use of extended contents and the dialog navigation through information.

9.4.1 Conversation about general topics

The Personal Energy Advisor can maintain an initial conversation based on general concepts. This allows transmitting a closer relation with the customer and a human like behaviour, e.g., while welcoming the user shows a friendly and close style.

General knowledge is included to provide abilities for a natural language conversation. It allows maintaining a conversation related to general concepts using a string pattern matching solution.

On Fig. 9.2 is showed a dialog making use of this kind of conversation. It is a first level of knowledge in the communication with the user.

The dialog is performed as follows:

- *User: Hello.*
- *Avatar: I am your personal energy advisor. How are you Matias?*
- *User: Fine, and you?*
- *Avatar: I am looking forward in helping you to be energy efficient.*
- *User: How can you help me?*
- *Avatar: I have access to your consumption information and I can identify patters of behaviour. Ask me more details.*

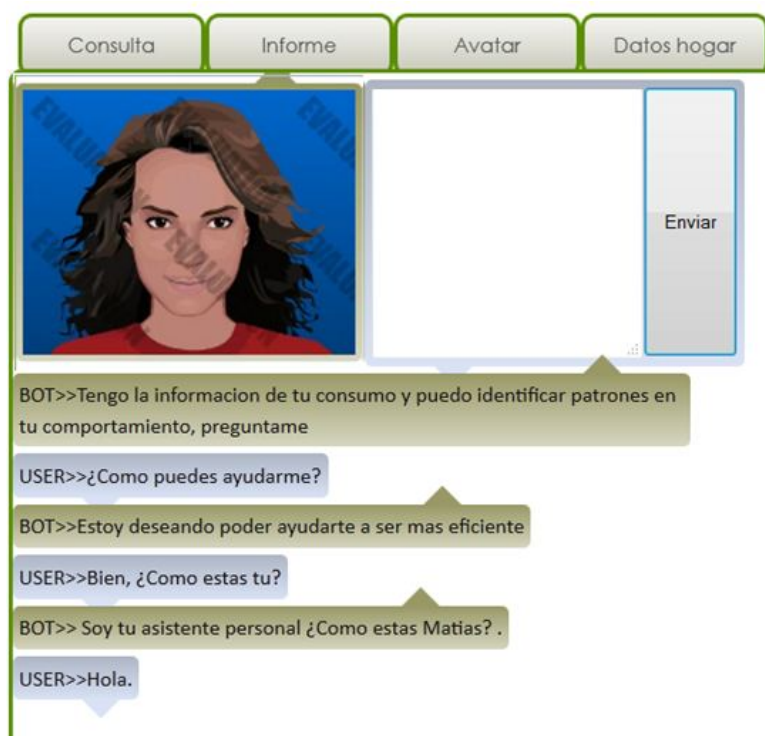


Figure 9.2: Personal Energy Advisor scenario for General knowledge.

9.4.2 Advising on energy use

Queries about efficiency and consumption behaviour require a more complex reasoning than a question-answer string pattern matching. Here we move from (1) a generic string matching approach to a personalized analysis to generate advice and from (2) a traditional numeric oriented analytical approach to a natural language communication.

The Personal Energy Advisor is able to decide what the questions issues are and deliver to the user an answer with relevant contents. Note that not yes no answers are provided, but a more complex reasoned responses are used to provide and an associated explanation.

Once the user is identified and logged into the system, information and natural language expressions can be personalized for each user. Question interpretation, information extraction, computational requirements and instantiation in natural language is performed due to the model proposed.

This kind of queries is showed on Fig. 9.4, here is showed an example to retrieve data about CO2 Emissions and later for consumption.

The dialog is performed as follows:



Figure 9.3: Personal Energy Advisor scenario for natural language and extended answers.

- User: *What are my CO₂ emissions?*
- Avatar: *In 2012 your CO₂ emissions are 1214kg. In 2013 your CO₂ emissions are 84kg. The lower part of your emissions are in 2013.*
- User: *What is my consumption?*
- Avatar: *In the different periods of the year your household consumption was high. The lowest consumption is in 2013.*

9.4.3 Communicating Emotions

The Personal Energy Advisor can express emotional contents. Emotions allow getting closer to the customer and improving understanding. Emotions are used to clarify the message and get a direct impact to the users.

This kind of answers is showed on Fig. 9.3, here is showed an example to communicate a high efficiency rate to the user underlying the positive behaviour.

The dialog is performed as follows:

- User: *What is my monthly efficiency?*
- Avatar: *Your monthly energy efficiency rating is A. It is a very good efficiency! The consumption of your household was low. The lower efficiency is in the months of January and December. Only in a few months your energy efficiency rating is B, an average efficiency.*

The figure displays two screenshots of a chat interface for a Personal Energy Advisor. Both screenshots feature a top navigation bar with buttons for 'Consulta', 'Informe', 'Avatar', and 'Datos hogar'. An avatar of a woman is visible on the left of each chat window, and an 'Enviar' button is on the right.

Left Screenshot:

Bot: >>En los diferentes periodos del ano el consumo de tu hogar fue elevado El mínimo consumo es en el ano 2013 . En algunos de los anos el consumo es el consumo de tu hogar fue medio .

User: >>cual es mi consumo?

Bot: >>En el ano 2012 tus emisiones de CO2 son de 1 214 Kg . En el ano 2013 tus emisiones de CO2 son de 87 Kg . La menor parte de tus emisiones de CO2 son en el ano 2013 .

User: >>Cuales son mis emisiones este año?

Right Screenshot:

Bot: >>Mensualmente el consumo de tu hogar fue elevado El mínimo consumo es en los meses de enero , febrero, marzo, noviembre y diciembre . Unicamente algunos meses el consumo de tu hogar fue medio .

	#Fconsumo
01.ENERO	medio
02.FEBRERO	medio
03.MARZO	medio
04.ABRIL	elevado
05.MAYO	elevado
06.JUNIO	elevado
07.JULIO	elevado
08.AGOSTO	elevado
09.SEPTIEMBRE	elevado
10.OCTUBRE	elevado
11.NOVIEMBRE	medio
12.DICIEMBRE	medio

Figure 9.4: Personal Energy Advisor scenario for natural language and extended answers.

9.4.4 Expressing extended contents

The Personal Energy Advisor can work with extended contents. Information delivered by the avatar is not restricted to text, usually charts or tabular data can increase and facilitate understanding.

Extended contents allow including in the answer more information than language text, including tables and charts.

This kind of queries is showed on Fig. 9.4. Here the same information is retrieved with enriched contents in a table.

9.4.5 Dialogue through information

Information can be accessed and retrieved in a natural way through an interactive dialog. Dialog capabilities are supported by the database multidimensional structure, that allows to navigate through the information while chatting with the Personal Energy Advisor.

This kind of dialogue is showed on Fig. 9.5, here the user asks about the consumption in 2012 and later in 2013. Afterwards move to the consumption detailed per month and get specific values for January, and later the hourly details.

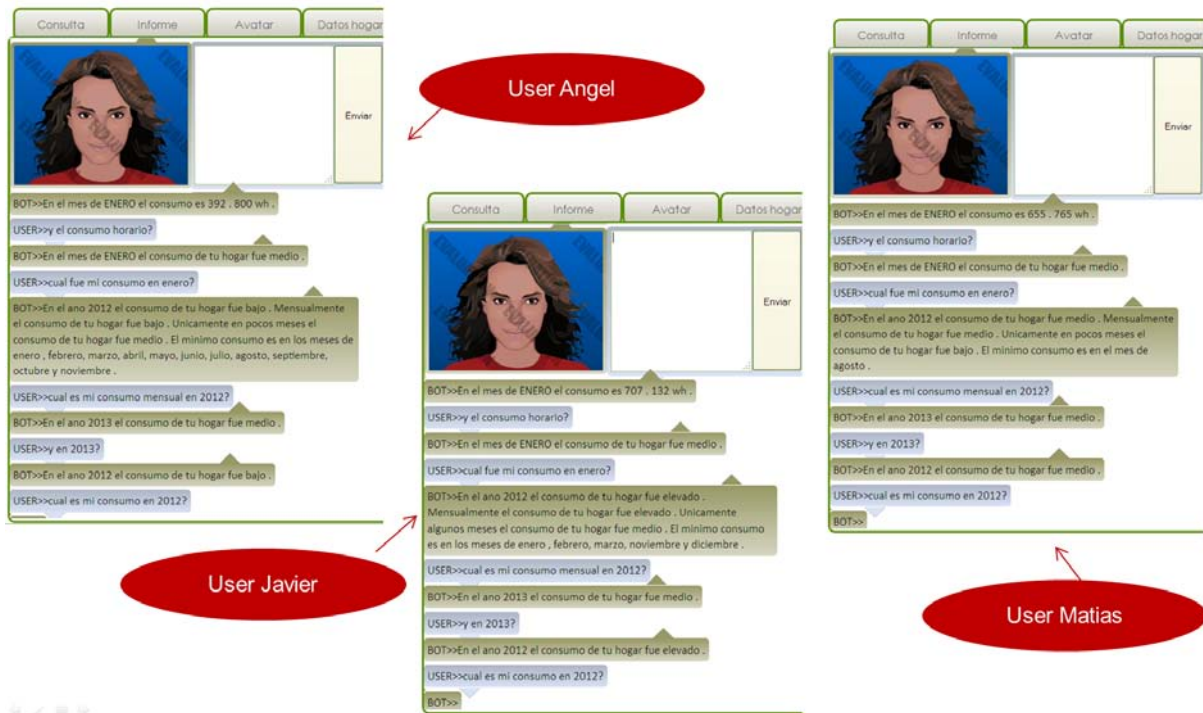


Figure 9.5: Personal Energy Advisor scenario of interaction for Angel, Matias and Javier.

The dialog is performed as follows:

- User: *What is my consumption in 2012?*
- Avatar: *In the year 2012 your consumption was low.*
- User: *and in 2013?*
- Avatar: *In the year 2013 your consumption was medium.*
- User: *what is my monthly consumption in 2012?*
- Avatar: *In 2012 the consumption of your household was low. Monthly consumption of your household was low. Only in a few months your consumption was medium. The lowest consumption is in the months of January, February, March and April.*
- User: *What is my consumption in January?*
- Avatar: *In January your consumption was medium.*
- User: *and the hourly consumption?*
- Avatar: *The hourly consumption was 392 wh.*

9.5 Report functionality

9.5.1 Report

The report definition interface is used to query contents using the report definition layout. Each report represent a specific content, defined by a set of measures and attributes arranged in rows or columns, and a set of filters applied over measures or attributes.



Figure 9.6: Personal Energy Advisor report interface.

Reports can be defined by the user due to dragging and dropping attributes and measures defined in the knowledge domain into the report layout. Reports constitute a model for query information related to energy efficiency.

A report sample is showed on Fig. 9.6. This report is used to describe efficiency and get recommendations. It is defined as follows:

- Efficiency is the measure selected.
- A filter on the year attribute is defined for 2012.
- The attribute year is placed in rows and no attribute for columns is defined.

In the report definition the user specifies the output required for the information requested. Output is defined in terms of tables, natural language or charts. Combining the different elements, tables, charts and language is incorporated a higher level of description of data.

In the report sample showed on Fig. 9.6, the output is requested as a line chart, a table and the associated text. Results of the report execution is showed on the same figure.

9.5.2 Report Group

Reports contents can be arranged in a more complex report set by the report group definition screen. A report group is defined by a set of contents. Contents are defined by predefined reports. This structure allows placing advice from different perspectives in a unique structure to compose a more complex report. Users can define reports and report groups as desired or access predefined ones in order to analyze their consumption behaviour.

To facilitate access to reports or reports groups, they can be saved on a folder structure to be retrieved by their selection for execution.

A report group is showed on Fig. 9.7. It is defined by including a set of individual reports previously defined.

Reports included in the sample report group are:

- Efficiency
- CO2 Emissions
- Expenses information

9.5.3 Predefined Reports

Predefined reports can be defined by the designer or the user to be used and consulted frequently. For this purpose has been defined an Overall efficiency report with detailed information.

With this approach users could move from a monthly period tariff bill to a personalized advice report generated by the Personal Energy Advisor.

The contents of the report go from high level information to a lower level description. The contents are structured as follows: (a) Annual details about efficiency, consumption, CO2 emissions and billing information, (b) Monthly consumption and efficiency for a more detail analysis and (c) Hourly consumption details to extract advice to get savings.

A report for each user is showed on Figures 9.8, 9.9 and 9.10. It underlyies the personalization of the information defined and the utility of contents.

Report options

Selected reports for the report group

Available predefined reports

2012

En el año 2012 tu clasificación de eficiencia energética es B debido a que y el consumo de tu hogar fue medio . Tienes una eficiencia relativamente buena, podrias intentar mejorar la eficiencia para alcanzar un nivel A. Esto te supondria ahorros de en torno al 20%..

Mensualmente tu clasificación de eficiencia energética es B debido a que y el consumo de tu hogar fue medio . La menor eficiencia es en los meses de enero , febrero, marzo, abril, mayo, junio, julio, septiembre, octubre, noviembre y diciembre . Unicamente en pocos meses tu clasificación de eficiencia energética es A.

Figure 9.7: Personal Energy Advisor report group interface.

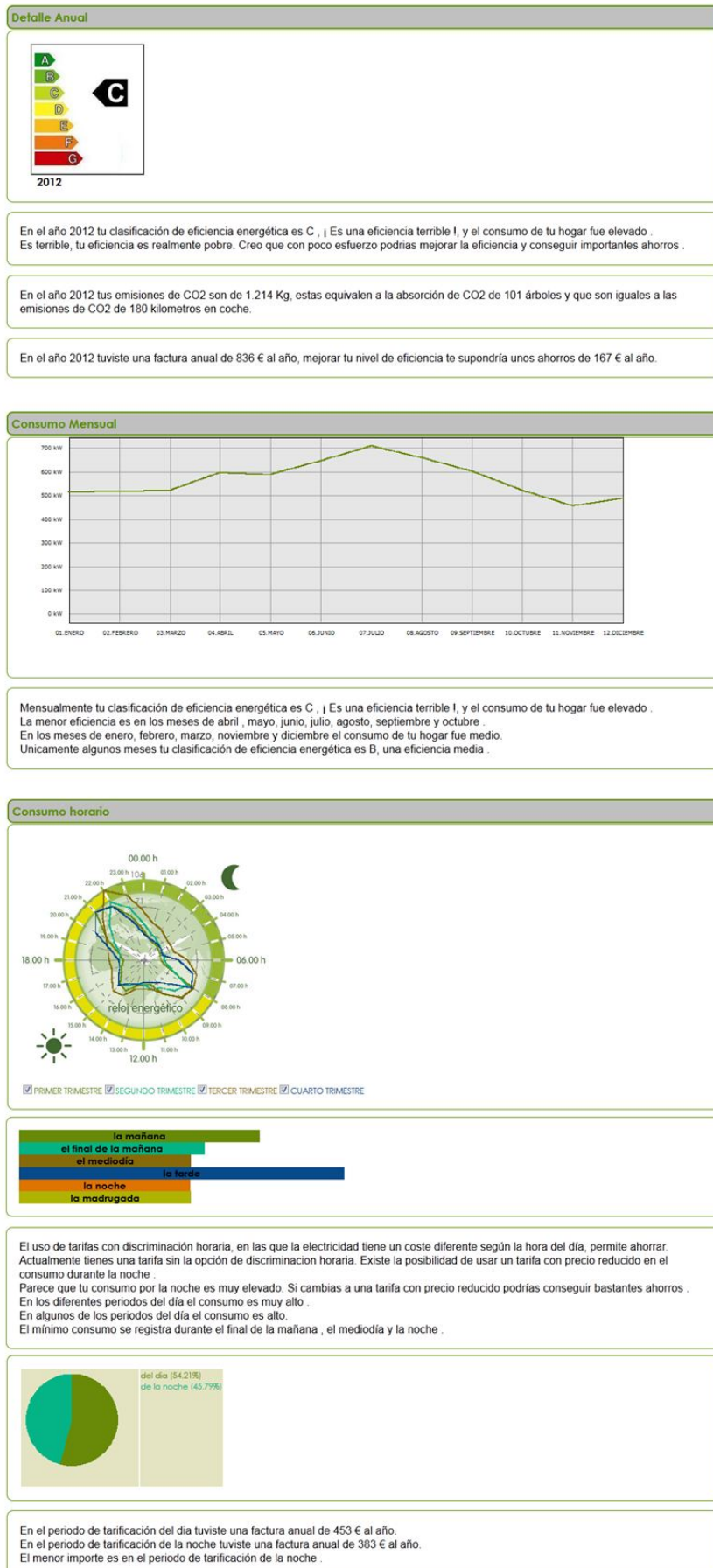


Figure 9.8: Efficiency report definition for Javier.

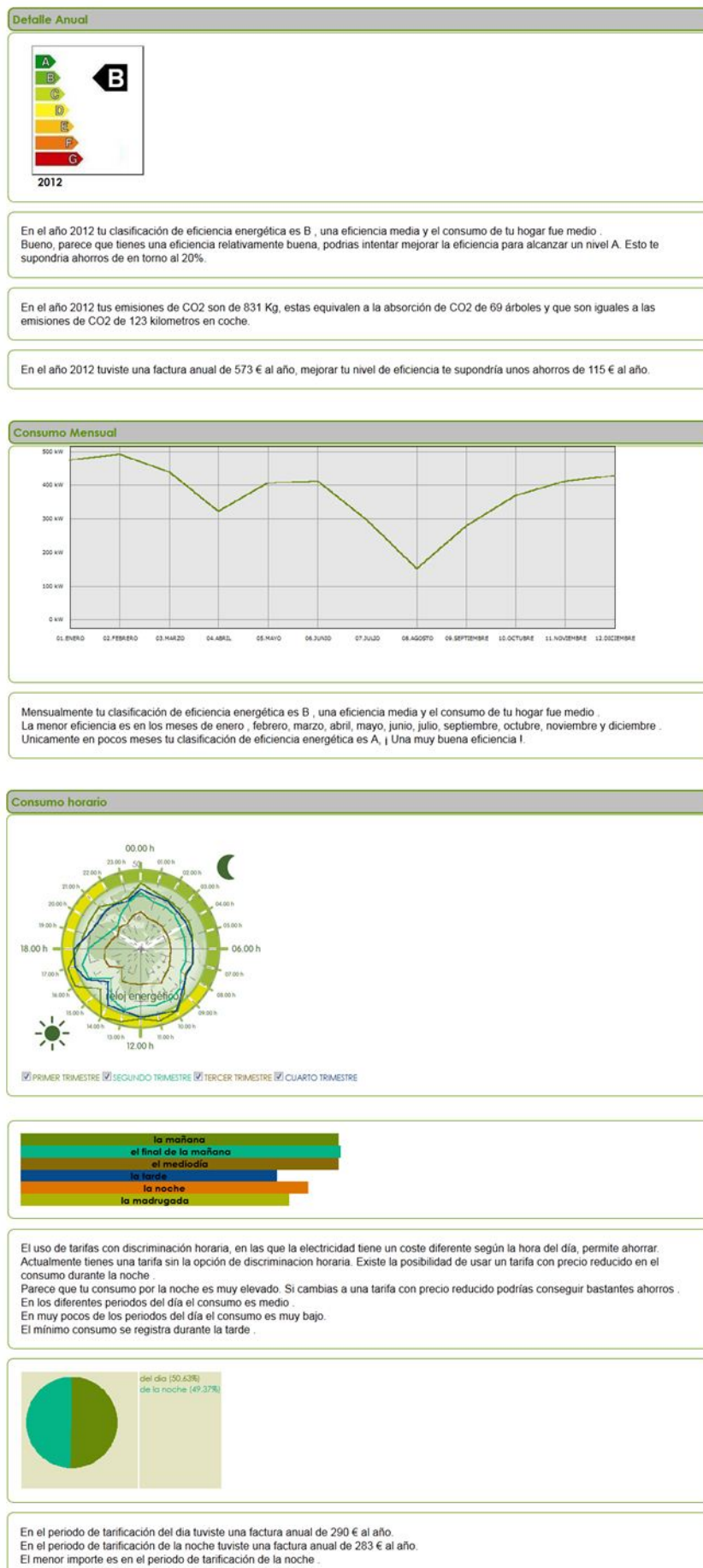


Figure 9.9: Efficiency report definition for Matias.

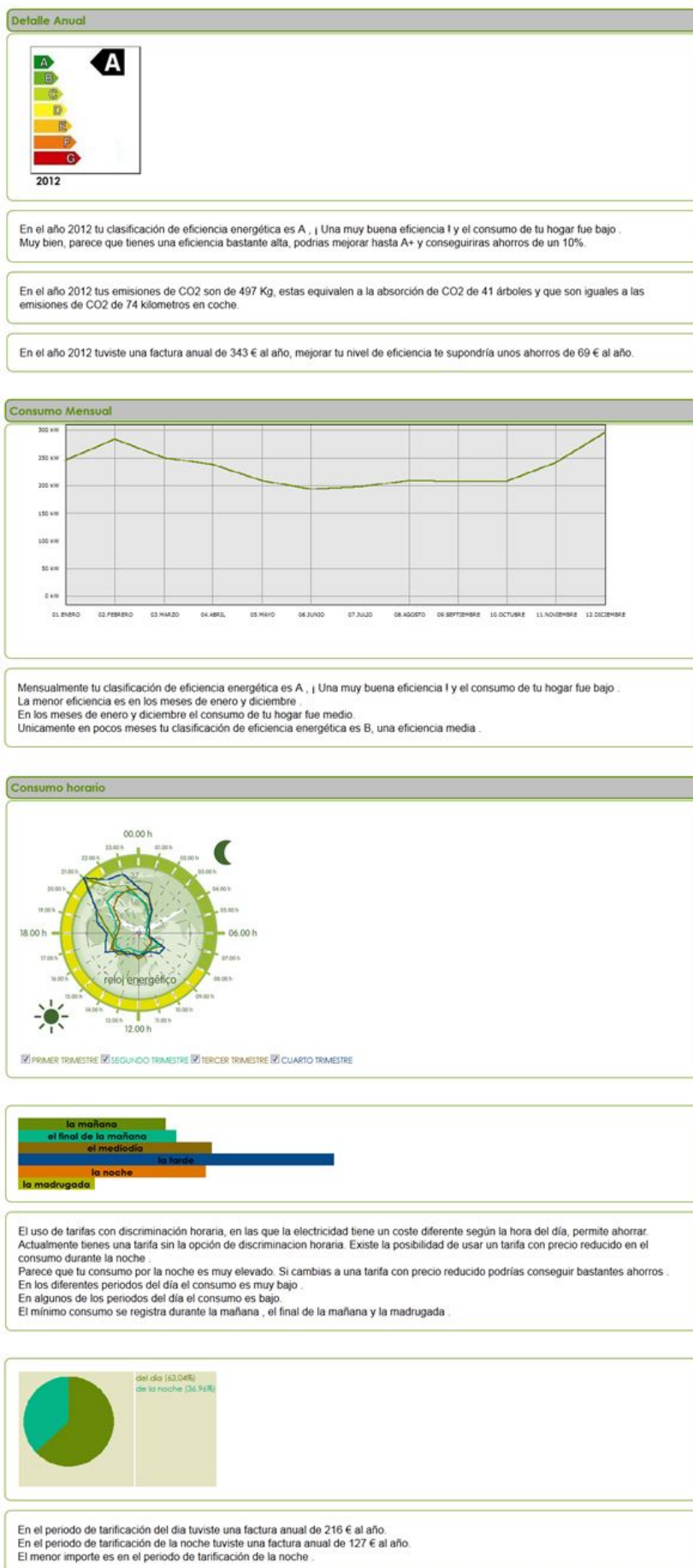


Figure 9.10: Efficiency report definition for Angel.

Chapter 10

Concluding Remarks

10.1 Review

The increase of energy efficiency is a prior objective to achieve a sustainable development. The company EDP launched different programs to promote energy efficiency. One of the initiatives is the project “*Ecofamilias*”. This project has the objective of providing useful information about consumption patterns to households. “*Ecofamilias*” is an informative program to promote energy efficiency at households with the digital electricity smart meters installed, capitalizing the information gathered.

For companies, having personalized contact with customers suppose a step forward to increase customer satisfaction and would like to move to more intelligent interaction with the users, e.g., virtual agents and natural language communication. There are some examples of research focused on virtual agents but they are closer to a string pattern matching process of questions and answers and lack the ability to deal with numeric information and databases.

To interpret energy consumption information is not trivial, the acquisition of knowledge for an expert advisor is a complex task. Information about consumption is affected by external factors like sociodemographic characteristics of the household. A relevant step in the thesis, in order to develop the Personal Energy Advisor, is the acquisition of knowledge from the environment and the translation of this knowledge into an implementable design. The ability to transform this data in easily interpretable natural language, extracting remarkable information, will be a key in the process of identifying paths to improve efficiency.

In the thesis was presented a computational system able to simulate the actions of an expert in energy advice, looking for logical solutions, the ones more appropriate for each household in search of efficiency. An application framework for the definition of the personal energy advisor was defined. The Personal Energy Advisor was analyzed in terms of functionality.

10.2 Contributions

In the research line of Computational Theory of Perceptions, this thesis was developed based on the Granular Linguistic Model (GLM), OLAP data access, Affective Computing and Virtual Interactive Agents.

The contribution of the thesis is to define a system to assist users in achieving energy efficiency, the Personal Energy Advisor. The advisor is able to interpret user queries requested by users and reason answers and advice about the energy consumption patterns using a knowledge domain. To define the architecture different contributions have been defined over existing paradigms. These contributions allow generating dynamically, based on perceptions, natural language interaction for users with a virtual energy advisor or access to advisory reports.

The list of detailed contributions can be organized in two main blocks, a) a set of proposed paradigms, namely OLAP-GLMP, Discourse OLAP-GLMP, Emotional OLAP-GLMP, and Question-Answer OLAP-GLMP and b) a software framework implementation of the paradigms, a methodology for development over the framework and the implementation of the Personal Energy Advisor and test of the solutions proposed.

a) The proposed paradigms are described as follows:

- OLAP-GLMP. Based on previous paradigms, it is proposed an On-line analytic processing model for the navigation through granular linguistic models in order to retrieve dynamically data from a database. The data is translated into linguistic expressions exploring possibilities of obtaining from natural language queries, reasoned responses using the GLM for modeling the knowledge base. This proposal “*OLAP Navigation in the Granular Linguistic Model of a Phenomenon*” has been published in (85).
- Discourse OLAP-GLMP. During text generation several tasks are performed, e.g., analyze the possibilities to automatically generate coherent and cohesive linguistic sentences based on natural language. It is done using acquired constraints to reorganize the structure of an initial data obtained from a defined input domain. It is performed by a machine driven process of reorganization of the GLM. The reorganization is performed by pruning non relevant perceptions and by aggregating related ones avoiding redundancy. This proposal “*Selection of the Best Suitable Sentences in Linguistic Descriptions of Data*” has been published in (83).
- Emotional OLAP-GLMP. In order to increase the effectiveness and acceptance while communicating information, it is needed to express a description in a natural way showing human-like emotions. When language is produced, as speakers, text is tailored to the hearer and the actual situation. This enables the possibility to include more information than is contained in the literal meaning of linguistic expressions.

This proposal “*Generating advices with emotional content for promoting efficient consumption of energy*” has been accepted for publication in the International Journal of Uncertainty, Fuzziness and Knowledge-based Systems (84).

- Question-answer OLAP-GLMP. Using perceptions is defined the procedures to interpret user questions and dynamically compute the answers. Some developments are proposed for interpretation of text in terms of perceptions, allowing to establish a dynamic dialog with the user facilitating communication.

b) The software framework and the development of the Personal Energy Advisor is described as follows:

- In order meet the desired requirements and to test the proposed paradigms it has been developed a software tool to implement the energy advisor. It consists in an application framework for the definition of an the Personal Energy Advisor based on requirements reducing design and implementation gaps and facilitating development.
- It is also proposed a methodology that using the software framework allows developing this kind of computational applications. Formal methods in software engineering are an important application area for intelligent systems. A description of software methodologies are presented, followed by the particularization of a methodology for covering the process involved in the development of the Personal Energy Advisor system using the framework of the solution. The methodology is developed step by step to define the Personal Energy Advisor.

Finally the proposed extensions and the system framework are used to test the functionality of the Personal Energy Advisor. The resulting advisory reports and interaction between the Personal Energy Advisor and the user is presented showing the viability of the approach including a practical demonstration.

10.3 Articles

Along the thesis have been published the following papers and articles:

Menendez, C., Eciolaza, L. and Trivino, G. (2014). Generating advices with emotional content for promoting efficient consumption of energy. International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems. (Accepted but not published yet).

Menendez, C. and Trivino, G. (2012). Selection of the best suitable sentences in linguistic descriptions of data. In Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., and Yager, R., editors, *Advances in Computational Intelligence*, volume 298 of *Communications in Computer and Information Science*, pages 295-304. Springer

Berlin Heidelberg.

Menendez C. and Trivino G.(2011). OLAP navigation in the Granular Linguistic Model of a Phenomenon. In Proceedings of the 5th IEEE International Symposium on Computational Intelligence and Data Mining (CIDM 2011), 11-15 April. Paris (France), pp. 260- 267.

10.4 Future work

Within the results of the thesis are considered some future research lines from the conclusions obtained. There are several directions in which this research could continue. In order to provide access to other information sources, i.e., input data not stored in a data base but existing in sources of free text could be included like Web contents. Also the capabilities for the generation of natural language can be improved in order to be more human-like, i.e., natural and flexible.

Bibliography

- [MIT] Mityc, ministerio de industria turismo y comercio.
- [2] (1990). IEEE Standard Glossary of Software Engineering Terminology. Technical report.
- [3] Agrawal, R., Gupta, A., and Sarawagi, S. (1997). Modeling multidimensional databases. In *Data Engineering, 1997. Proceedings. 13th International Conference on*, pages 232–243. IEEE.
- [4] Alhajj, R. and Kaya, M. (2003). Integrating fuzziness into olap for multidimensional fuzzy association rules mining. In *Data Mining, 2003. ICDM 2003. Third IEEE International Conference on*, pages 469–472. IEEE.
- [5] Anderson, K. and Kerr, C. (2002). *Customer Relationship Management*. McGraw-Hill.
- [6] Arbib, M. A. and Fellous, J.-M. (2004). Emotions: from brain to robot. *TRENDS IN COGNITIVE SCIENCES*, 8(12):554–561.
- [7] Arguelles, L. and Trivino, G. (2013). I-struve: Automatic linguistic descriptions of visual double stars. *Engineering Applications of Artificial Intelligence*, (0):-.
- [8] Bagley, P. R. (1968). Extension of programming language concepts. Technical Report 0518086, University City Science Center.
- [9] Barbat, B. E. and Cretulscu, R. (2003). Affordable affective avatars. persuasion, emotions and language (s). In *Proc. of the 1st Balkan Conference in Informatics (BCI'2003)*.
- [10] Bartneck, C. (2002). Integrating the occ model of emotions in embodied characters.
- [11] Barzilay, R. and Elhadad, M. (1997). Using lexical chains for text summarization. In *In Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17.
- [12] Bateman, J. A. (1997). Enabling technology for multilingual natural language generation: the kpml development environment. *Natural Language Engineering*, 3(1):15–55.
- [13] Beckert, B., Hoare, T., Hahnle, R., Smith, D. R., Green, C., Ranise, S., Tinelli, C., Ball, T., and Rajamani, S. K. (2006). Intelligent systems and formal methods in software engineering. *IEEE Intelligent Systems*, 21(6):71–81.

- [14] Bergmair, R. (2006). Syntax-driven analysis of context-free languages with respect to fuzzy relational semantics. Technical Report UCAM-CL-TR-663, University of Cambridge, Computer Laboratory.
- [15] Bien, Z. Z. and Lee, H.-E. (2007). Effective learning system techniques for human-robot interaction in service environment. *Know.-Based Syst.*, 20(5):439–456.
- [16] Boehm, B. (2011). Towards richer process principles. In *Proceedings of the 2011 International Conference on Software and Systems Process*, ICSSP '11, pages 234–234, New York, NY, USA. ACM.
- [17] Bohnet, B. and Dale, R. (2004). Referring expression generation as a search problem. In *Proceedings of the 2nd Australasian Language Technology Workshop*. pdf.
- [18] Bordogna, G. and Pasi, G. (1993). A fuzzy linguistic approach generalizing boolean information retrieval: A model and its evaluation. *JASIS*, 44(2):70–82.
- [19] Breazeal, C. (2003). Emotion and sociable humanoid robots. *Int. J. Hum.-Comput. Stud.*, 59(1-2):119–155.
- [20] Brooks, R. A. (1986). A robust layered control system for a mobile robot. *IEEE Journal on Robotics and Automation*, 2(1):14–23.
- [21] Buttle, F. (2004). *Customer Relationship Management, Concepts and Tools*. Elsevier Butterworth-Heinemann.
- [22] Cabibbo, L. and Torlone, R. (1998). A logical approach to multidimensional databases. *Advances in Database Technology—EDBT'98*, pages 183–197.
- [23] Cao, Y.-G., Ely, J., Antieau, L., and Yu, H. (2009). Evaluation of the clinical question answering presentation. In *Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing*, BioNLP '09, pages 171–178, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [24] Castillo-Ortega, R., Marín, N., and Sánchez, D. (2011). A fuzzy approach to the linguistic summarization of time series. *Journal of Multiple-Valued Logic and Soft Computing*, 17(2-3):157–182.
- [25] Castillo-Ortega, R., Marín, N., and Sánchez, D. (2009). Linguistic summary-based query answering on data cubes with time dimension. In Andreasen, T., Yager, R., Bulskov, H., Christiansen, H., and Larsen, H., editors, *Flexible Query Answering Systems*, volume 5822 of *Lecture Notes in Computer Science*, pages 560–571. Springer Berlin Heidelberg.
- [26] Cerpa, N. and Verner, J. M. (2009). Why did your project fail? *Commun. ACM*, 52(12):130–134.
- [27] Codd, E., Codd, S., and Salley, C. (1993). Providing olap (on-line analytical processing).

- [28] Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387.
- [29] Codd, E. F. (1979). Extending the database relational model to capture more meaning. *ACM Transactions on Database Systems (TODS)*, 4(4):397–434.
- [30] Codd, E. F. (1990). *The relational model for database management: version 2*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [31] Coll, C. (1991). Psicología y currículum: Una aproximación psicopedagógica a la elaboración del currículum escolar. *Cuadernos de pedagogía. Paidós*.
- [32] Cook, D. J. and Das, S. K. (2007). How smart are our environments? an updated look at the state of the art. *Pervasive Mob. Comput.*, 3(2):53–73.
- [33] Costanza, E., Ramchurn, S. D., and Jennings, N. R. (2012). Understanding domestic energy consumption through interactive visualisation: a field study. In *UbiComp 2012, 14th ACM International Conference on Ubiquitous Computing*, pages 216–225.
- [34] Cowie, R., Douglas-Cowie, E., Tsapatsoulis, N., Votsis, G., Kollias, S., Fellenz, W., and Taylor, J. G. (2001). Emotion recognition in human-computer interaction. *Signal Processing Magazine, IEEE*, 18(1):32–80.
- [35] Darby, S. (2006). The effectiveness of feedback on energy consumption. *A Review for DEFRA of the Literature on Metering, Billing and direct Displays, April*.
- [36] Delgado, M., Molina, C., Sanchez, D., Vila, A., and Rodriguez-Ariza, L. (2004). A fuzzy multidimensional model for supporting imprecision in olap. In *Fuzzy Systems, 2004. Proceedings. 2004 IEEE International Conference on*, volume 3, pages 1331 – 1336 vol.3.
- [37] Delgado, M., Verdegay, J. L., and Vila, M. A. (1992). Linguistic decision-making models. *International Journal of Intelligent Systems*, 7(5):479–492.
- [38] Dyck, S. and Majchrzak, T. A. (2012). Identifying common characteristics in fundamental, integrated, and agile software development methodologies. In *Proceedings of the 2012 45th Hawaii International Conference on System Sciences, HICSS '12*, pages 5299–5308, Washington, DC, USA. IEEE Computer Society.
- [39] Eciolaza, L., Pereira-Faria, M., and Trivino, G. (2012). Automatic linguistic reporting in driving simulation environments. *Applied Soft Computing*, (0):–.
- [40] Eisman, E. M., López, V., and Castro, J. L. (2009). Controlling the emotional state of an embodied conversationalagent with a dynamic probabilistic fuzzy rules based system. *Expert Syst. Appl.*, 36(6):9698–9708.
- [41] Ekman, P. (1992). An argument for basic emotions. *Cognition and Emotion*, 6:169–200.
- [42] El-Nasr, M. S., Yen, J., and Ioerger, T. R. (2000). Flame-fuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-Agent Systems*, 3(3):219–257.

- [43] Elhadad, M. (1996). Fuf manual. Technical report, Dept of Mathematics and Computer Science, Ben Gurion University.
- [44] Elhadad, M. and Robin, J. (1992). Controlling content realization with functional unification grammars. In *Aspects of Automated Natural Language Generation*, pages 89–104. Springer Verlag.
- [45] Elliott, C. D. (1992). *The affective reasoner: a process model of emotions in a multi-agent system*. PhD thesis, Evanston, IL, USA. UMI Order No. GAX92-29901.
- [46] Fayyad, U., Piatetsky-shapiro, G., Smyth, P., and Widener, T. (1996). The kdd process for extracting useful knowledge from volumes of data. *Communications of the ACM*, 39:27–34.
- [47] Feng, L. and Dillon, T. (2003). Using fuzzy linguistic representations to provide explanatory semantics for data warehouses. *Knowledge and Data Engineering, IEEE Transactions on*, 15(1):86 – 102.
- [48] Fleischman, M. and Hovy, E. (2002a). Emotional variation in speech-based natural language generation. In *International Natural Language Generation Conference*.
- [49] Fleischman, M. and Hovy, E. (2002b). Towards emotional variation in speech-based natural language generation. In *Proceedings of the Second International Natural Language Generation Conference (INLG02)*, pages 57–64.
- [50] Fragopanagos, N. F. and Taylor, J. G. (2005). Emotion recognition in human-computer interaction. *Neural Networks*, 18(4):389–405.
- [51] Fraunhofer (2012). Study on energy savings potentials in eu member states, candidate countries and eea countries. preliminary results.
- [52] Fujita, H., Mark, V., Fujita, H., and Marik, V. (2009). *New Trends in Software Methodologies*. IOS Press, Amsterdam, The Netherlands, The Netherlands.
- [53] Graesser, A., McMahan, C., and Johnson, B. (1994). *Question Asking and Answering*. San Diego: Academic Press.
- [54] Gratch, J. and Marsella, S. C. (2001). Tears and fears: Modeling emotions and emotional behaviors in synthetic agents. In *Proceedings of the 5th International Conference on Autonomous Agents*, page 278–285, Montreal, Canada.
- [55] Grice, H. P. (1975). Logic and conversation. In Cole, P. and Morgan, J. L., editors, *Syntax and Semantics: Vol. 3: Speech Acts*, pages 41–58. Academic Press, San Diego, CA.
- [56] Grice, P. (1991). *Studies on the way of words*. Harvard University Press.
- [57] Halliday M.A.K., H. R. (1976). *Cohesion in English*. English language series. Longman.

- [58] Hargreaves, T., Nye, M., and Burgess, J. (2010). Making energy visible: A qualitative field study of how householders interact with feedback from smart energy monitors. *Energy Policy*, 38(10):6111–6119.
- [59] Hess, U. and Thibault, P. (2009). Why the same expression may not mean the same when shown on different faces or seen by different people. In Tao, J. and Tan, T., editors, *Affective Information Processing*, pages 145–158. Springer London.
- [60] Hovy, E. (1988a). *Generating Natural Language Under Pragmatic Constraints*. UMI order. Erlbaum.
- [61] Hovy, E. and Laboratory, G. A. F. B. R. (1992). *Natural Language Generation*. RL-TR-. Rome Laboratory Public Affairs Office.
- [62] Hovy, E. H. (1988b). Planning coherent multisentential text. In *Proceedings of the 26th annual meeting on Association for Computational Linguistics*, ACL '88, pages 163–169, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [63] Hovy, E. H. (1990). Pragmatics and natural language generation. *Artif. Intell.*, 43(2):153–197.
- [IDAE] IDAE. Idae, instituto para la diversificacion y ahorro de la energia.
- [65] Jacobson, I., Booch, G., and Rumbaugh, J. (1999). *The unified software development process*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [66] Johnson-Laird, P. (1983). *Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness*. Cognitive science series. Cambridge University Press.
- [67] Kasinadh, D. P. V. and Krishna, P. R. (2007). Building fuzzy olap using multi-attribute summarization. In *Proceedings of the International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007) - Volume 01*, ICCIMA '07, pages 370–374, Washington, DC, USA. IEEE Computer Society.
- [68] Kimball, R. and Ross, M. (2004). *The Data Warehouse Toolkit-The Complete Guide to Dimensional Modeling*. John Wiley and Sons (Asia) Pte. Limited.
- [69] Kruchten, P. (2003). *The Rational Unified Process: An Introduction*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3 edition.
- [70] Kukich, K. (1983). Design of a knowledge-based report generator. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics*, ACL '83, pages 145–150, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [71] Kurgan, L. A. and Musilek, P. (2006). A survey of knowledge discovery and data mining process models. *Knowl. Eng. Rev.*, 21(1):1–24.
- [72] Larsen, H., Kacprzyk, J., Zadrozny, S., Andreasen, T., and Christiansen, H. (2000). *Flexible Query Answering Systems: Recent Advances*. Advances in soft computing. Physica-Verlag HD.

- [73] Laurent, A. (2001). Generating fuzzy summaries from fuzzy multidimensional databases. In Hoffmann, F., Hand, D., Adams, N., Fisher, D., and Guimaraes, G., editors, *Advances in Intelligent Data Analysis*, volume 2189 of *Lecture Notes in Computer Science*, pages 24–33. Springer Berlin Heidelberg.
- [74] Laurent, A., Bouchon-Meunier, B., Doucet, A., Gancarski, S., and Marsala, C. (2000). Fuzzy data mining from multidimensional databases. *Studies in Fuzziness and Soft Computing*, 54:278–283.
- [75] Leidner, J. L. (2003). Current issues in software engineering for natural language processing. In *Proceedings of the HLT-NAACL 2003 workshop on Software engineering and architecture of language technology systems - Volume 8*, SEALTS '03, pages 45–50, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [76] Lenz, H. and Shoshani, A. (1997). Summarizability in olap and statistical data bases. pages 132–143. IEEE Computer Society.
- [77] Mahamood, S. and Reiter, E. (2011). Generating affective natural language for parents of neonatal infants. In *Proceedings of the 13th European Workshop on Natural Language Generation*, pages 12–21. Association for Computational Linguistics.
- [78] M.A.K.Halliday (1994). *An Introduction to Functional Grammar*. Arnold, London, 2 edition.
- [79] Mann, W. C. (1983). An overview of the penman text generation system. In Gensereith, M. R., editor, *AAAI*, pages 261–265. AAAI Press.
- [80] Mann, W. C. and Thompson, S. A. (1987). Rhetorical structure theory: Description and construction of text structures. In Kempen, G., editor, *Natural Language Generation: New Results in Artificial Intelligence, Psychology, and Linguistics*, pages 85–95. Nijhoff, Dordrecht.
- [81] Mansouri, I., Newborough, M., and Probert, D. (1996). *Energy Consumption in UK Households: Impact of Domestic Electrical Appliances*, volume 54, pages 211–285. Elsevier.
- [82] McKeown, K. (1985). Discourse strategies for generating natural-language text. *Artif. Intell.*, 27(1):1–41.
- [83] Menendez, C. and Trivino, G. (2012). Selection of the best suitable sentences in linguistic descriptions of data. In Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., and Yager, R., editors, *Advances in Computational Intelligence*, volume 298 of *Communications in Computer and Information Science*, pages 295–304. Springer Berlin Heidelberg.
- [84] Menendez, Carlos, E. L. and Trivino, G. (2014). Generating advices with emotional content for promoting efficient consumption of energy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*.

- [85] Menendez-Gonzalez, C. and Triviño, G. (2011). OLAP navigation in the Granular Linguistic Model of a Phenomenon. In *IEEE Symposium Series on Computational Intelligence*. Paris, France.
- [86] Michael Elhadad, J. R. (1996). Surge: a comprehensive plug-in syntactic realization component for text generation. Technical report, Dept of Mathematics and Computer Science, Ben Gurion University.
- [87] Moldovan, D., Badulescu, A., Tatu, M., Antohe, D., and Girju, R. (2004). Models for the semantic classification of noun phrases. In *Proceedings of the HLT-NAACL Workshop on Computational Lexical Semantics*, CLS '04, pages 60–67, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [88] Norman, D. A., Ortony, A., and Russell, D. M. (2003). Affect and machine design: Lessons for the development of autonomous machines. *IBM Systems Journal*, 42(1):38–44.
- [89] Ortony, A., Clore, G., and Collins, A. (1990). *The Cognitive Structure of Emotions*. Cambridge University Press.
- [90] Pierce, J., Schiano, D. J., and Paulos, E. (2010). Home, habits, and energy: examining domestic interactions and energy consumption. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1985–1994, New York, NY, USA. ACM.
- [91] Pomerantz, J. (2005). A linguistic analysis of question taxonomies: Research articles. *J. Am. Soc. Inf. Sci. Technol.*, 56(7):715–728.
- [92] Popescu, A.-M., Etzioni, O., and Kautz, H. (2003). Towards a theory of natural language interfaces to databases. In *Proceedings of the 8th international conference on Intelligent user interfaces*, IUI '03, pages 149–157, New York, NY, USA. ACM.
- [93] Portet, F., Reiter, E., Gatt, A., Hunter, J., Sripada, S., Freer, Y., and Sykes, C. (2009). Automatic generation of textual summaries from neonatal intensive care data.
- [94] Pylyshyn, Z. W. (1986). *Computation and Cognition: Toward a Foundation for Cognitive Science*. Bradford Books. Mit Press.
- [Qin et al.] Qin, Z., Thint, M., and Beg, M. M. S. Deduction engine design for pnlbased question answering system. In *World Congress of the International Fuzzy Systems Association*, 2007.
- [96] Quirk, R. (1985). *A comprehensive grammar of the English language / Randolph Quirk ... [et. al.] ; index by David Crystal*. Longman London ; New York.
- [97] Reeves, B. (2000). The benefits of interactive online characters. *Center for the Study of Language and Information, Stanford University*.
- [98] Reigeluth, C. M. (1980). The elaboration theory of instruction: A model for sequencing and synthesizing instruction. *Instructional Science*, page 9:195–219.

- [99] Reilly, W. (1996). Believable social and emotional agents. Technical Report CMU-CS-96-138, Carnegie-Mellon University. Computer science. Pittsburgh (PA US), Pittsburgh.
- [100] Reiter, E. (2007). An architecture for data-to-text systems. In *Proceedings of the Eleventh European Workshop on Natural Language Generation, ENLG '07*, pages 97–104, Stroudsburg, PA, USA. Association for Computational Linguistics.
- [101] Reiter, E. and Dale, R. (1992). A fast algorithm for the generation of referring expressions. In *Proceedings of the 14th International Conference on Computational Linguistics*, pages 232–238, Nantes, France.
- [102] Reiter, E. and Dale, R. (2000). *Building Natural Language Generation Systems*. Studies in Natural Language Processing. Cambridge University Press.
- [103] Rennolls, K. and AL-Shawabkeh, A. (2008). Formal structures for data mining, knowledge discovery and communication in a knowledge management environment. *Intell. Data Anal.*, 12(2):147–163.
- [104] Robin, J., Favero, E., and Informtica, C. D. (2001). Hyssop: Natural language generation meets knowledge discovery in databases.
- [105] Robinson, W. and Rackstraw, S. (1972). *A question of answers*. Number v. 5, parte 1 in Primary socialization, language and education. Routledge and K. Paul.
- [106] Rogova, E. and Chountas, P. (2007). On imprecision intuitionistic fuzzy sets and olap - the case for knolap. In Castillo, O., Melin, P., Montiel, O., Sepúlveda, R., Pedrycz, W., and Kacprzyk, J., editors, *IFSA (2)*, volume 42 of *Advances in Soft Computing*, pages 11–20. Springer.
- [107] Sanchez-Valdes, D., Alvarez-Alvarez, A., , and Trivino, G. (2012). Linguistic description of temporal traffic evolution in roads. In *Actas XVI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF)*, Valladolid, Spain.
- [108] Schröder, M. (2001). Emotional speech synthesis: A review.
- [109] Schröder, M. (2004). *Speech and emotion research: an overview of research frameworks and a dimensional approach to emotional speech synthesis*. Phonus : reports in phonetics. Institut für Phonetik, Universität des Saarlandes.
- [110] Shove, E. (2003). *Comfort, Cleanliness and Convenience: The Social Organization of Normality*. New Technologies / New Cultures. Bloomsbury Academic.
- [111] Shove, E. (2010). Beyond the abc: climate change policy and theories of social change. *Environment and Planning A*, 42(6):1273–1285.
- [112] Solaimani, S., Bouwman, H., and Baken, N. (2011). The smart home landscape: a qualitative meta-analysis. In *Proceedings of the 9th international conference on Toward useful services for elderly and people with disabilities: smart homes and health telematics*, ICOST'11, pages 192–199, Berlin, Heidelberg. Springer-Verlag.

- [113] Sripada, S. and Gao, F. (2007). Linguistic interpretations of scuba dive computer data. In *IV*, pages 436–441. IEEE Computer Society.
- [114] Sripada, S. G., Reiter, E., Davy, I., and Nilssen, K. (2004). Lessons from deploying nlg technology for marine weather forecast text generation. In *in: Proceedings of PAIS-2004, 2004*, pages 760–764.
- [115] Taylor, A. S., Harper, R., Swan, L., Izadi, S., Sellen, A., and Perry, M. (2007). Homes that make us smart. *Personal Ubiquitous Comput.*, 11(5):383–393.
- [116] Thomas, H. and Datta, A. (2001). A conceptual model and algebra for on-line analytical processing in decision support databases. *Info. Sys. Research*, 12(1):83–102.
- [117] Trivino, G., Alvarez-Alvarez, A., and Bailador, G. (2010a). Application of the computational theory of perceptions to human gait pattern recognition. *Pattern Recognition*, 43(7):2572 – 2581.
- [118] Trivino, G., Alvarez-Alvarez, A., and Bailador, G. (2010b). Application of the computational theory of perceptions to human gait pattern recognition. *Pattern Recogn.*, 43(7):2572–2581.
- [119] Trivino, G. and Bailador, G. (2007). Linguistic description of human body posture using fuzzy logic and several levels of abstraction. In *Proceedings of the IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA)*, pages 105–109.
- [120] Trivino, G., Mengual, L., and van der Heide, A. (2009). Towards an architecture for semiautonomous robot telecontrol systems. *Inf. Sci.*, 179(23):3973–3984.
- [121] Triviño, G., Mengual, L., and van der Heide, A. (2009). Towards an architecture for semiautonomous robot telecontrol systems. *Information Sciences.*, 179(23):3973–3984.
- [122] Triviño, G., Sánchez, Á., Montemayor, A. S., Pantrigo, J. J., Cabido, R., and Pardo, E. G. (2010). Linguistic description of traffic in a roundabout. In *Proceedings of the IEEE Fuzzy 2010*. Barcelona, Spain.
- [123] Trivino, G., Sanchez, A., Montemayor, A. S., Pantrigo, J. J., Cabido, R., and Pardo, E. G. (2010). Linguistic description of traffic in a roundabout. In *Proc. of the 2010 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Jul. 18-23, Barcelona, Spain*, pages 2158–2165.
- [124] Trivino, G. and Sugeno, M. (2012). Towards linguistic descriptions of phenomena. *International Journal of Approximate Reasoning*, (0):–.
- [125] Trivino, G. and van der Heide, A. (2008a). An experiment on the description of sequences of fuzzy perceptions. *Hybrid Intelligent Systems, International Conference on*, 0:228–233.

- [126] Trivino, G. and van der Heide, A. (2008b). Linguistic summarization of the human activity using skin conductivity and accelerometers. In *Proceedings of the Conference Information Processing and Management of Uncertainty in Knowledge Based Systems (IPMU)*, pages 1583–1589.
- [127] van der Heide, A. and Triviño, G. (2010). A prototype of a machine speaking with emotions. *Actas del XV Congreso Español sobre Tecnologías y Lógica Fuzzy, ESTYLF 2010. Huelva, Spain: 2010*, page 6.
- [128] Wallace, R. (2004). The elements of AIML style. ALICE AI Foundation.
- [129] Wallace, R. (2009a). The anatomy of a.l.i.c.e. In Epstein, R., Roberts, G., and Beber, G., editors, *Parsing the Turing Test*. Springer.
- [130] Wallace, R. (2009b). The anatomy of a.l.i.c.e. In Epstein, R., Roberts, G., and Beber, G., editors, *Parsing the Turing Test*. Springer.
- [131] Waltz, D. L. (1978a). An english language question answering system for a large relational database. *Commun. ACM*, 21(7):526–539.
- [132] Waltz, D. L. (1978b). An english language question answering system for a large relational database. *Commun. ACM*, 21(7):526–539.
- [133] Yu, J., Reiter, E., Hunter, J., and Mellish, C. (2007). Choosing the content of textual summaries of large time-series data sets. *Nat. Lang. Eng.*, 13(1):25–49.
- [134] Zadeh, L. A. (1975). The concept of linguistic variable and its application to approximate reasoning. *Information sciences*, 8:199–249.
- [135] Zadeh, L. A. (1999a). From computing with numbers to computing with words – from manipulation of measurements to manipulation of perceptions. *IEEE Transactions on Circuits and Systems*, 45(1):3–40.
- [136] Zadeh, L. A. (1999b). From computing with numbers to computing with words - from manipulation of measurements to manipulation of perceptions. *IEEE Transactions on Circuits and Systems*, 45(1).
- [137] Zadeh, L. A. (2001). A new direction in ai: towards a computational theory of perceptions of measurements to manipulation of perceptions. *AI Magazine*, 22(1):73–84.
- [138] Zadeh, L. A. (2004). Precisiated natural language (pnl). *AI Mag.*, 25(3):74–91.
- [139] Zadeh, L. A. (2006a). *From Search Engines to Question Answering Systems â“ The Problems of World Knowledge, Relevance, Deduction and Precisation*. Elsevier B.V.
- [140] Zadeh, L. A. (2006b). Granular computing the concept of generalized constraint-based computation. *Rough Sets and Current Trends in Computing (Lecture Notes in Computer Science)*, 4259:15–46.

-
- [141] Zadeh, L. A. (2008). Toward human level machine intelligence – is it achievable? the need for a paradigm shift. *IEEE Computational Intelligence Magazine*, 3(3):11 – 22.
- [142] Zheng, Z. (2002). Answerbus question answering system. In *Proceedings of the second international conference on Human Language Technology Research, HLT '02*, pages 399–404, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.