A Customer Management System for the Spanish Transport System Operator

Julia Díaz, José R. Dorronsoro, Ángela Fernández, Sandra Fresnillo, Teresa Huelin, Álvaro Valera

Instituto de Ingeniería del Conocimiento (IIC)
UAM-Cantoblanco, EPS Edificio B, P-5. 28049 Madrid (Spain)
Telef:+34 914972323, Fax:+34 914972334, e-mail: Julia.diaz@iic.uam.es

Abstract

The Active Demand Management (GAD) project is led by IBERDROLA´s Networks business area as part of the National Strategic Consortium for Technical Research (CENIT) initiative, included in the Ingenio 2010 Program for the promotion of research, development and innovation in Spain. This project pools the efforts of 15 Spanish companies and 14 research centres into a single stable Consortium.

The final goal of the Active Demand Management (GAD) project is to optimize the way electricity is used and, therefore, the cost associated with that usage, while meeting the consumer's needs with the same level of quality.

The active involvement of the consumers and usage elements in the demand management mechanisms and the communication of the energy generation costs in real time, including the environmental costs, will contribute to raising public awareness and to changing the users' energy consumption patterns, thus moving towards the much needed energy sustainability.

This paper presents the software that has been designed and developed for the Customer Management System (in Spanish, Sistema Gestor de Clientes – SGCL). The SGCL is a distributed system with three subsystems that result from its functional decomposition per agent involved: (1) Transport System Operator (TSO), referred as SGCL-OS; (2) Distribution System Operator (DSO), referred as SGCL-OD; (3) Reseller (CM), referred as SGCL-CM. The paper is focused in the SGCL-OS part.

Keywords: Customer Management System, Demand prediction, Transport System Operator (TSO), SaaS, SOA

1. Overall system description

The system design starts with the implementation of an agreed set of functions with the other participants in the GAD project. On this specification has developed a complete hardware and software system, which serves the TSO as a tool to support Operation.

Related to the software layers used to meet the original specifications mentioned in the introduction, the system has been programmed as a web application (or standard e-business server-based application):

Data layer. This layer has been implemented as a relational database. It stores any information associated with the GAD project and all necessary data to carry out the functionality of SGCL-OS.

Logic layer. This layer refers to any internal implementation required for the operation and development of the Network Analysis Application.

Presentation Layer. It has been designed as a web application and is responsible for displaying information and interacting with the user (TSO).

Communication with other agents. In the current scope of GAD project there only exists direct communication between the TSO and the DSO.

To provide the communication between both agents has been used Web Services technology.

Each of these layers is described in greater detail in subsequent sections.

2. Data Layer

The GAD database stores all necessary information generated by and for the *demand prediction* algorithm (see section 6), information from the Network Analysis Application, as well as other data derived from the GAD's context.

It has particular emphasis on mechanisms that preserve the integrity of data associated with network problems, identified and characterized by the Network Analysis Application, because they are linked to the changing conditions of the transport network. Also, very flexible mechanisms have been established for updating data due the multiplicity and the disparity between the different data sources. This entails loss of performance in data access, but greater stability for them.

Any impact identified during data reception is recorded and stored in the application logs.

3. Presentation Layer

The purpose of the web application is to present to the end user, TSO, any information that is helpful to operate with the tool GAD, and the interface to perform this operation.

The web application takes from the database all the necessary information to properly submit all the functionality available.

The burden of implementation is primarily in the server. The main functions performed by the web application are: (1) Briefing georeferenced of GAD databases: results of network analysis, results of the demand prediction algorithm and GAD potential technically available. (2) External interface with the DSO. It refers to GAD orders sending and GAD clients consumption request. (3) Internal interface of the network analysis on demand.

The communication interfaces included in this server are integrated directly into the web application. It does not pose any additional deployment of resources to those required by the application.

These interfaces include: (1) ASP module for SOAP requests by HTTP to communicate with DSO's Web Services. (2) ASP module to load files and invoke application on the server.

4. Communication

In the current scope of GAD project there only exists direct communication between the TSO and the DSO. Therefore, no direct communication exists with CM.

The communications between the TSO and the DSO have been implemented using a Service Oriented Architecture (SOA) due to the features that SOA offers:
(1) Functions reuse. A function is implemented in a system as a service and can be invoked internally or externally by any application.
(2) Service Orchestration. In order to perform an activity, this can be composed by a set of services, and these can make use of others.
(3) Transparency. A service defines WHAT, not HOW it does it.
(4) Encapsulation. The business process or logic process is hidden to external functions or services.
(5) Common language. A service provides a common language to guarantee interoperability.
(6) Flexibility. An internal change does not affect the service it provides.
(7) Scalability. It is easy to enclose new services, publishing the service they offer.
(8) Security. The service or the communication system where they are deployed can implement a policy to access to the service.

SOA has been built using Web Services. They allow exchanging data between different software applications developed in different programming languages and executed on any platform. They also allow that software and services of different companies located in different geographic places can be easily combined to provide integrated services. Thus, communication between TSO and DSO can be done easily and efficiently, ensuring interoperability.

Web Services is a technology based on standards:
(1) HTTP and SOAP for communication.
(2) XML and WSDL to describe and define the service.
(3) UDDI to make public, discover and recover the service.

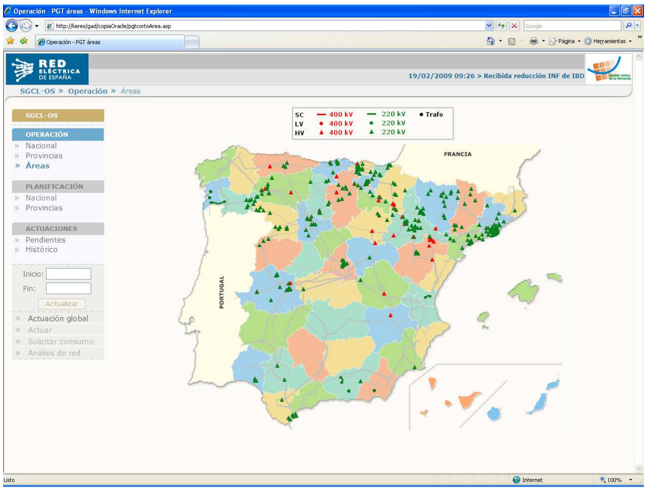
SOAP is a protocol specification relies on XML as its message format, so that the messages exchanged between the two agents are XML files.

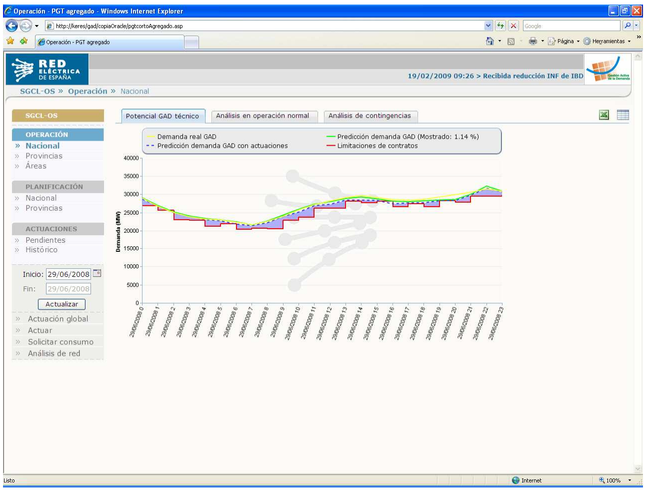
The information to be exchanged between the agents is:
(1) Estimated burden per node/province.
(2) To reduce demand per province.
(3) Aggregate consumption.
(4) Technical limitation.
(5) Order cancellation.

To provide and consume Web Services has been used Apache Axis2/C, a Web Services engine implemented in the C programming language and which is based on Axis2 architecture. This enables using C in SOA implementations.

Apache Axis2/C supports several versions in SOAP, which makes it ideal for use.

Next figures illustrate the application interface that summarizes the work described. The application options are in Spanish.





5. Demand prediction models

One TSO GAD goal is to flatten the customer demand curve in the long-term during the global planning that means a stable situation not only from the consumption point of view but also from the generation. Other goal is to improve the local optimization of demand peaks due to for example air conditioning in the Spanish east shore in summer.

So, TSO must be able to predict contingencies or grid inefficiency situations, to avoid grid overcharge and the subsequent malfunction. Thus, the demand behaviour prediction becomes necessary.

In order to achieve such goal, the TSO implements a *demand prediction* algorithm that is able to predict the demand on both a short and a long term, which is implemented by using a linear regression. This algorithm is able to perform electrical demand forecast under a known set of circumstances, such as historical demand values (see the following picture), the working days calendar and the meteorology.

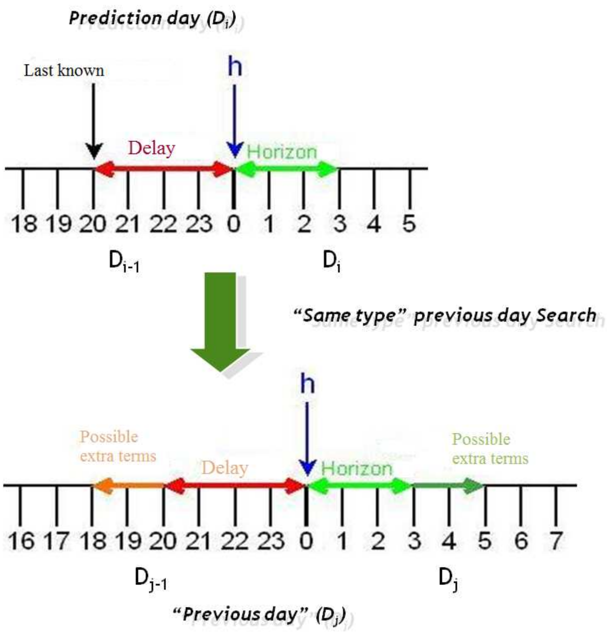


Figure 1. ***Demand prediction*** **algorithm diagram**

This prediction algorithm's goals are twofold: to be a useful tool not only for the operation in real time but also to help grid planning for mid and long term. Therefore two different variants exist: the first one obtains predictions with hourly resolution, extendable to daily or even weekly and the second has a longer resolution such as a three months period or even a year.

In the following **picture** the national demand curve is shown for a specific day: real demand (yellow) and prediction (green).

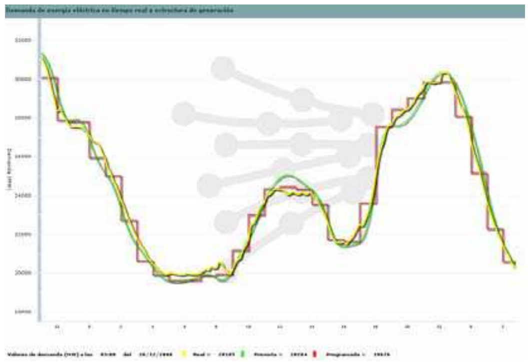


Figure 2. National demand curve example

6. Conclusions

Due the current scope of GAD project, the following paragraphs refer only to the *demand prediction* algorithm in the short term, within the transaction in real time.

The algorithm design is done following two steps: (1) Historical data preparation that involves demand differences calculation, hourly temperature data interpolation, etc. in order to obtain the model data inputs; (2) Algorithm execution, modeling and executing the regression in order to obtain the final prediction.
Using three years of data corresponding to demand and temperature, and calculating daily predictions for a whole year, the timing obtained is:
Historical Data Preparation (three years): 1m 23.757s
Algorithm execution (one year): 5.090s
Real time execution: 1.022s.

The formula that has been used in the algorithm is:

$$[D_{i,h+x} - D_{i,h-R}] = \alpha_0 + \sum_{z=1}^{Z-1+TD} \alpha_z \times [D_{j,h+z} - D_{j,h+(z-1)}] + \sum_{r=0}^{R-1+TA} \beta_r \times [D_{j,h-r} - D_{j,h-(r+1)}] + \sum_{z=0}^{Z-1} \delta_z \times T_{j,h+z}$$

$$x \in [0, Z-1]$$

Figure 3. ***Demand prediction*** **algorithm formula**

Where:

- i = day to predict
- j = the day before i
- $D_{i,j}$ = demand of the day *i* and the hour *j*
- α_i, β_i, δ_i = regresion coeficients
- α_0 = constant term
- Z = hourly prediction horizon
- R = hours after the last available real data
- TD = forward extra terms

TA = backward extra terms
$D_{i,h-R}$ = last known real data
$T_{i,j}$ = prediction of the day *i* and the hour *j*

Below it is described the scenario tests and the errors obtained with the best model:
(1) 1 hour delay: prediction time lapse starts at 0 hours, taking demand data of the last hour.
(2) Horizon: hourly predictions are issued for the following 24 hours.
(3) Regression terms: 29 differences in demand hourly of day earlier; 24 temperature estimations a day to predict; one constant term.
(4) Modelling period: 01/01/2004 - 30/06/2006.
(5) Prediction period: 01/07/2006 - 30/06/2007.

Validation methodology: In order to measure the models implemented, we have used the relative linear error (**RLE**), absolute linear error (**ALE**), Pearson's correlation coefficients (**PCC**) and mean quadratic error (**MQE**). In order, we show the Global Mean (**G**), Monday (**M**), Tuesday-Wednesday-Thursday (**T**), Friday (**F**), Saturday (**S**), Sunday- Vacations (**V**).

Table 1 shows the errors of the *demand prediction* algorithm that have been obtained as a result of applying the above scenario.

	G	**M**	**T**	**F**	**S**	**V**
RLE	2.03	2.46	1.96	1.98	1.64	2.21
ALE	603	744	624	634	445	544
PCC	98.55	98.07	98.12	98.21	98.24	97.62
MQE	884.69	1085.68	940.62	858.73	619.19	765.52

Table 1. *Demand prediction* algorithm errors

7. Acknowledgments

This system has been developed for Red Eléctrica de España. We want to thank the team involved in all phases of the project.

References

[1] "IEADSM Task XV Research Report 1_final revised_2006-10-27.pdf" from www.ideadsm.org
[2] "Carmen_Rodriguez_Spanish_DSO_FORUM" from http://dsm.iea.org (http://dsm.iea.org/Files/Exco%20File%20Library/Workshop%20Madrid%20October%202005/Carmen_Rodriguez_Spanish_DSO_FORUM.pdf)
[3] "EPRI Day-Ahead Hour-Ahead Forecasting for Demand Trading A Guidebook" from www.epri.com
[4] "Hippert Neural networks for short-term load forecasting a review and evaluation" from *ieeexplore.ieee.org*