Optimal extraction and conditioning of historical information to support the operational decisions in a Smart Grid context

MARXA LENINA TORRES-ESPINDOLA, ALFREDO ESPINOSA-REZA
Dirección de Tecnologías Habilitadoras
Instituto Nacional de Electricidad y Energías Limpias
Reforma 113, Col. Palmira, Cuernavaca, Morelos
MEXICO

Abstract: - This article shows a proposal for the components architecture that allows, in an organized and collaborative way, the request, transport, and use of large amounts of historical information very effectively, without compromising the performance of the information systems and the technological platform that supports them. The architecture and some variants implemented with great success in semantic interoperability projects in the Smart Grid context are exposed, considering the use and adoption of the Common Information Model (CIM) defined in the set of standards IEC 61968 and IEC 61970 mainly.

Key-Words: - Optimal extraction, Common Information Model, Semantic Interoperability, Smart Grid.

1 Introduction

Traditionally, the information systems used for the operation of an electric utility, consider the handling of large amounts of information related to the operating status of the Electric Power System (EPS), including substations, feeders, transformers, switches, sectionalizers, reclosers. etc. This information is measured in the field by monitoring, protection, control, and automation devices, and is collected by monitoring and control systems, such as SCADA systems. The information includes digital values (states, alarms, locks) and analog values (voltage, current, real power, reactive power, power factor, imbalance, temperature, humidity, amount of dissolved gases, events intensity, operation counters, among other values), these values are generated in real-time in the EPS and are almost always stored in large databases that contain the memory of what happened every day and that, as a whole, includes knowledge of the behavior of the EPS under different operating conditions that occurred over a considerably long time, sometimes 10 years or more.

In this sense, recovering historical information, processing it, and using it in an agile and effective way for its analysis, allows operators and those responsible for the operation of the EPS, to capitalize on historical knowledge to improve current and future operations, prevent adverse situations in the event of failures and contingencies, improve the response to maneuvers required for maintenance and clearance, and therefore, improve the productivity, efficiency, safety, reliability, and quality indexes associated with the operation of the EPS.

When an electric utility has enough historical information collected directly from the devices installed in the EPS, then it has the ability to evolve the level of support and sustenance of each operational decisión in:

- Normal or steady-state operation, in meeting objectives such as productivity and efficiency.
- Emergency situations, to speed up recovery, as well as improve security, reliability, and quality.
- Unusual situations or cases, such as disturbances due to natural events, failures, unforeseen demand peaks, or special requirements for specific maintenance.

2 Problem Formulation

Once the volume of data is considerably high and the different operational actors use it in the electrical utility, a typical problem arises associated with the performance of the extraction and its ease of use since it is normal that more and more users require them to their daily processes, but the architecture of legacy systems do not consider the utility operational evolution.

2.1 Smart Grid context

In this sense, the Smart Grid and its strategies for adopting increasingly advanced analytical functions,

insert a new stress factor on technological platforms, since the evolution of the traditional grid is especially guided by data, communications, and the ability to make new and better decisions with the support of the information and the inherent knowledge, but if that knowledge cannot be retrieved in an agile way, it will remain stagnant in the information warehouses, thus wasting a great capacity ready into the utilities.

One of the most required functions in Smart Grid is effective information management to support operational decision-making, therefore, the need for more and better strategies to manage information in a unified way is evident. [1] [2]

2.2 Data quality

Another serious problem that information users is the consistency and quality of the data, since, derived from the acquisition processes themselves, sensors, device configuration in the field, communication interruptions, data channel speed, equipment, Among others, the raw data stored may not be of sufficient quality to be used correctly in high impact analytical functions, for example: power flow calculation for a feeder reconfiguration maneuver; design, and sizing of substations, protection devices setting, among others, highly sensitive to the information accuracy.

The data quality is often left as part of the acquisition and storage system that contains the data, so that the end-user assumes that the data always has the correct value, with the appropriate quality. This is not necessarily true and validation is left to the user.

2.3 Components architecture

Traditionally, the extraction, conditioning, and use of historical EPS information in the information systems of an electric utility are carried out directly, consulting the databases that contain the records (raw data), see figure 1. How the data will be used is delegated to the system or user who makes the request; the end-use for each extraction or analytical function in which they will be used is not analyzed.

In this traditional architecture, strategies are not implemented to avoid saturation of the technological platform, nor are optimization strategies established for multiple massive queries to serve all concurrent users and always give a consistent response, as fast as possible and valid for all.

In the authors' experience, there are information systems with more than 10 years of historical data stored for a few hundred or thousands devices in the field at EPS, which translates into millions of records; These systems can be easily affected if the

architecture of their components does not consider the situations described, that is, with a single direct query to the database administrator, the system can "fail" if the query is carried out without restrictions, for example:

*Select * from HISTORICAL_TABLE*

Under controlled conditions, it is very easy to carry out the corresponding validations to try to saturate the technological platform, as long as the precaution is taken of having data backup means and ease of restoring data and physical or virtual processing servers.
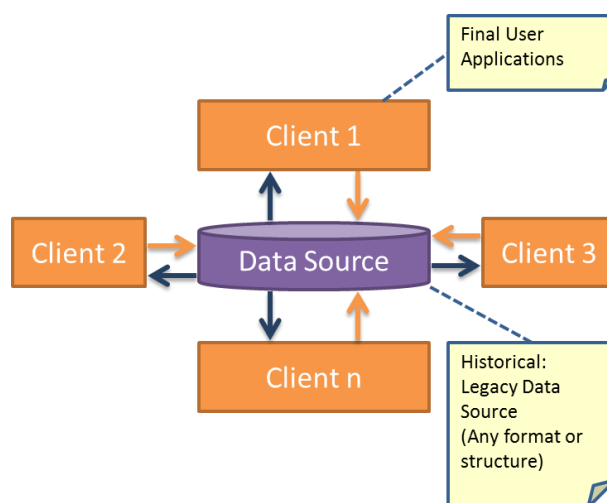

Fig. 1. Traditional components architecture for historical information extraction.

3 Problem Solution

The proposed solution is based on an **Optimal Extractor**, whose modular architecture, shown in figure 2, is easily adaptable to any specific situation since it includes several modules and each one solves one or a group of situations.

The main features of the solution are described below.

3.1 Users concurrency

When an electric utility begins to have reliable historical information from the EPS, a large number of users and needs arise naturally from its proper management that allows improving the business processes of the utility, from planning, construction, operation, maintenance, optimization, and reconfiguration, until its replacement and final disposal.

The most critical data use occurs in the area of operation, since adequate analytical management of historical information, with adequate response times, will allow EPS operators to make better

operational decisions with an approach to optimization associated technical and commercial processes.
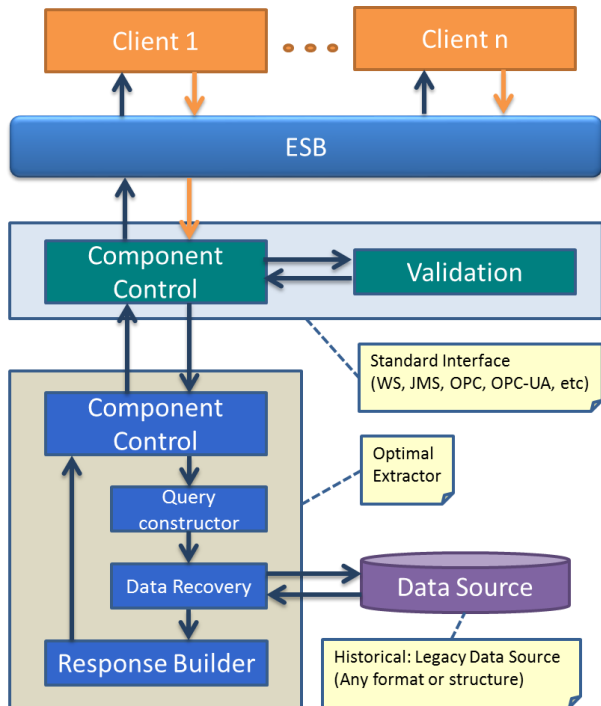

Fig. 2. Proposed components architecture for historical information extraction for syntactic interoperability.

For an EPS operator to be able to correctly use the information it requires, it must be on your screen as soon as possible, with maximum timeouts of 10 seconds for simple queries and 50 seconds for complex queries.

This is because, once you have the required information, you have a few minutes to use it correctly in each case that occurs during normal and emergency operations.

It should be noted that, in most cases, an EPS operator requires simple queries to improve its decision-making capacity. In general, must answer simple questions like:

Which was the maximum demand for this circuit yesterday? And last week? At what time does peak demand normally occur on circuits 1 and 2? Normally, which is the maximum current at circuit X? What is the typical hourly profile for circuit Y during the summer holidays? How does the voltage behave in the face of the demand decrease during the winter holidays? among others.

Eventually, an operator or responsible for the operation of the EPS requires slightly more elaborate queries, to make operational improvements with a focus on reducing technical losses, improving reliability or power quality, etcetera, in these cases The questions are, for example, which circuits in a substation has the greatest current imbalance in the last week? And, in the last month? How are the daily profiles for reactive power and power factor at circuit X? Which is the maximum capacity of the circuit Y to receive an energy transfer during the maximum daily demand in the last month? Which of the circuits 1, 2, or 3 can best receive half the power of circuit Z permanently? [3]

On the other hand, for an EPS analyst (planning and construction), the information queries are more complete and complex, for example, they must identify what is the Maximum Demand Peak (MDP) of a circuit in a year, and how it evolved in the last 5 years, in the same way, the Coincident Peak Demand (CPD) must be identified in a wide region or circuits set for a specific period [4], this type of query of great interest for this user, consumes many resources of the technological platform since the amount of data required can be from thousands to some millions, depending on the period and number of circuits, the remarkable thing is that normally the user only requires 1 to 100 significant data, but the obtain process it is very complex and time-consuming in computational terms.

To solve the concurrency situation, a very common alternative is an Enterprise Service Bus (ESB), which in addition to having a very efficient queue manager, allows the implementation of intermediation services to prioritize queries, according to the type of query and the user request. [5]

In the case of using an ESB, the **Optimal Extractor** functions are available for any application or client system that requires historical data, without the need to know the internal data structure of the source systems.

In case of not having an ESB, the **Component Control** module must carry out the sequencing and prioritization functions of the multiple simultaneous queries that are received, it can even partialize the queries to free up machine time of the entire technological infrastructure, as described in section 3.5.

3.2 Data quality verification
The historical data stored has the quality with which it was acquired at the moment in real-time, multiple factors can affect its quality and precision. A viable option to guarantee a response with high-reliability data is to integrate a **Validation** module, which is responsible for analyzing the data request in a query and applying specific validation, verification, and

completeness algorithms, and in case of detecting inconsistencies, perform an estimate of the corresponding replacement data and inform the applicant of the actions taken in the response calculation.

In this **Validation** module, the quality of the raw data can be verified in several ways, for example:

- Integrity: counting the number of records available for a data series in a defined period.
- Consistency: validating a data set according to the electrical or physical laws that model it.
- Accuracy: comparing a data set with external measurements, redundant measurements, or manual measurements, in the same period, or integrate measured values at different points in the ESP.
- Behavior: comparing the profile of a data set with the typical profile of that measurement.
- Validity: by cross-comparing a measure values with the similar measures values, geographically close measures, or calculated values.
- AI: Additionally, considering the data complexity, it is feasible to train artificial intelligence algorithms to perform much more complete validations, for example, identifying and applying typical profiles, autonomous autoregression, predictive models, correlation with exogenous variables, comparison with close members (case-based reasoning), automatic clustering algorithms, among others.

3.3 Handling large data sets

If the database does not have query restrictions, a request can have a large amount of data as a response, which could cause the saturation or collapse of the technological platform, which implies a delay in all other processes that are running simultaneously.

The **Component Control** module in conjunction with the **Response Builder** module can consider a strategy to prioritizing, sectioning, or partialize the queries and handle multiple responses so that the user who requested the data will eventually have a complete response, but it will be processed in packages appropriately, manageable by the technological platform so that all other concurrent users will be served and the waiting time will be distributed among all, thus, high priority users will have the answers in the required time and large volume users (who normally do not have high priority), will have the answer in a time only slightly

longer than if the query was made directly (in any case, it will take much more time than simple queries.).

3.4 Database operational security

Another specific problem in traditional architecture is that the operational stability of the technological platform is not guaranteed, since, as explained in section 2.3, it is relatively easy to affect it in uncontrolled use.

The solution proposed by the **Optimal Extractor** of figure 2 is to partialize the queries so that the machine time of the technological infrastructure is managed, that is, the **Component Control** module will be in charge of carrying out the following actions:

- Calculate the data amount that will be consulted in a user request.
- If the data amount exceeds an empirically defined limit (based on the technology platform harware resources and the data stored granularity), then the query will be sectioned or partialize and the **Query Constructor** and **Response Builder** modules will be informed.
- Multiple partial queries are issued.
- A waiting time is executed between queries (the time is also calculated empirically with the same criteria of the data limit).
- Partial responses are integrated into a single consistent response.
- The **Validation** Module algorithms are executed.
- The result is delivered to the user who requested it.

3.5 Standard data access

A serious problem with the traditional architecture shown in figure 1 is that each application or client system requires applying the data access standard to the technological platform and also requires knowing the database internal structure. More serious will be, when for specific reasons, the database changes technology, data access standard, or internal structure.

For solving this problem, the **Optimal Extractor** in its architecture proposed in figure 2, incorporates a data abstraction layer, therefore, if an ESB is used, all clients must use a single connectivity standard determined by the ESB, which is normally open and well known. In the case that an ESB is not available, the proposed architecture establishes the flexibility that access to the **Component Control** module

could be through one or more standard data interfaces, for example, Web Services (WS), Java Message Service (JMS), OLE for Process. Control (OPC), OLE for Process Control - Unified Architecture (OPC-UA), among others, or even, if necessary, a communications protocol such as DNP, Modbus or ICCP.

An additional advantage of this architecture is that, if the database changes its technology or internal structure, it will only be necessary to modify the **Data Recovery** module, without affecting the data clients in any way. [6] [7]

3.6 Standard data model

For Smart Grid applications, it is recommended to implement the semantic interoperability level between applications or systems, so a canonical data model based on standards must be used, which allows to unify and formalize the meaning of the data exchanged; it is recommended to adopt the Common Information Model (CIM) defined in the set of standards IEC 61968 and IEC 61970 mainly.

In the architecture proposed in figure 2, a wrapper must be added on the **Component Control** module side and another on the Client side, this will allow all the information that is transported between applications to be read and interpreted correctly and unified by any current or future application client, as shown in figure 3. Likewise, the adoption of standards will allow the use of any integration pattern, for example, those defined in standard IEC-61968-100.
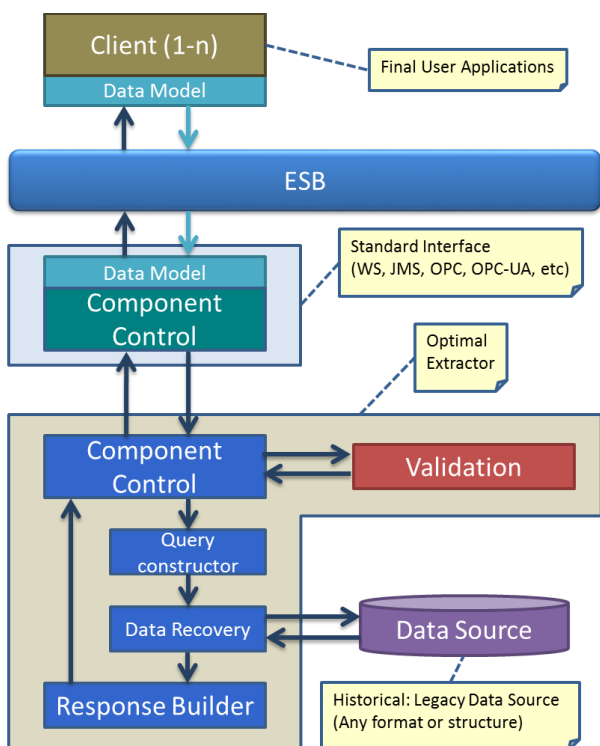

Fig. 3. Proposed components architecture for historical information extraction for semantic interoperability.

For a complete and advanced architecture, it is required to define the specific Profile based on the Canonical Data Model that will represent the particular data sets. If the CIM is used, then the established methodologies allow defining the CIM Profile that will be used for the data exchange in a semantic interoperability strategy for Smart Grid. [8]

The **Data Model** module of the architecture of figure 3 is in charge of implementing the wrapper to carry out the translation between the data of the source system and the client that needs to use the information.

Figure 4 shows a part of the CIM Profile proposed for the implementation of the **Optimal Extractor** developed.
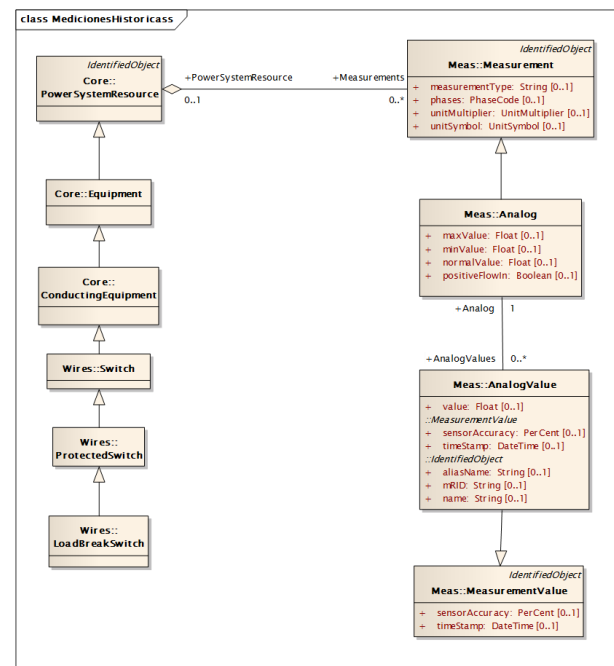

Fig. 4. CIM Profile (partial view) for Smart Grid semantic interoperability strategy.

4 Optimal extraction and conditioning

This section describes some of the main optimal conditioning functions for the efficient information transport between the source system and the client that needs to use the information.

4.1 Raw data

The following data can be consulted for any time range, for any variable registered in steady-state.

- Normal: Obtaining registered data without filters or validations. The use of this option is not recommended, as it will consume the most resources on the technology platform. The rules described in sections 3.3 and 3.4 apply.
- Discrimination: Obtaining data, eliminating invalid values. The measurement values consistency is verified according to the rules described in section 3.2.
- Missing rows: Sometimes, some causes can induce, in a period, specific measurement equipment cannot be requested for data gathering and therefore the historical database lacks information. This function reports the missing samples in the period consulted for each selected equipment. It includes two options: Only missing rows and Raw data with missing rows.

4.2 Statistical data
The following values can be calculated for any time range, at any of the data groupings by frequency, for any recorded steady-state variable. The rules described in section 3.2 apply.
- Average: Arithmetic average of the requested values.
- Maximum: The highest value of the requested values. It is used to identify an extreme value that could be an outlier or a data entry error.
- Minimum: The lowest value of the requested values. It is used to identify an extreme value that could be an outlier or a data entry error.
- Sum: The result of summing all the requested values. It is used to integrate values in a region, for example, the real power of substation circuits or a substations group.
- Standard deviation: It is the square root of the variance of the requested values. This measure of dispersion is the most characteristics.

4.3 Data grouping by frequency
The following grouping strategies of the calculated values of section 4.2 allow optimizing the queries, generating and transporting only the data that is really useful to the end-user, depending on the function in which it will be used.
- Hourly: Returns a single data for each hour requested. It allows generating of daily profiles of EPS electrical behavior.
- Daily: Returns a single data for each day requested. It allows generating of weekly or monthly profiles of EPS electrical behavior.

- Weekly: Returns a single data for each week requested. It allows generating of monthly profiles of EPS electrical behavior.
- Monthly: Returns a single data for each month requested. It allows generating of annual profiles of EPS electrical behavior.
- Annual: Returns a single data for each year requested. It allows comparing the annual EPS electrical behavior and annual trends.
- Period: Returns a single data for the entire time range requested. This function is used to compare the EPS electrical behavior in specific periods of interest.

4.4 Power Quality events
The following values can be requested for any time range, but metering devices must have power quality functions.
- Interruptions: It is an instantaneous change of frequency from the steady-state of the current, the voltage, or both. It has a unidirectional polarity and is characterized mainly by its rise and fall times and its maximum value.
  - Momentary. Obtains the values with a voltage percent less than or equal to 10% and duration less than or equal to 3,000 [ms].
  - Temporary. Obtains the values with a voltage percent less than or equal to 10% and duration greater than or equal to 3,000 [ms] but less than or equal to 60,000 [ms].
  - Sustained. Obtains the values with a voltage percent of 0% and duration greater than or equal to 60,000 [ms].
  - All. Gets all interrupts when the current flow stops for any reason in the selected time range.
- SAGS: Decrease in the effective voltage value between 0.9 and 0.1 P.U. and duration from 16 [ms] up to a few seconds.
  - Instant. Values with a voltage percent greater than or equal to 10% but less than or equal to 90%, and duration greater than or equal to 16 [ms] and less than or equal to 500 [ms].
  - Momentary. Values with voltage percent greater than or equal to 10% but less than or equal to 90%, and duration greater than 500 [ms] and less than or equal to 3,000 [ms].
  - Temporary. Values with a voltage percent greater than or equal to 10% but less than or equal to 90%, and duration greater than 3,000 [ms] and less than or equal to 60,000 [ms].

- o All. Gets all SAGS records stored for the selected time range.
- SWELL: Increase in the effective voltage value between 1.1 and 1.8 P.U. and duration from 16 [ms] up to a few seconds.
  - o Instant. Values with voltage percent greater than or equal to 110% but less than or equal to 180%, and duration greater than or equal to 16 [ms] and less than or equal to 500 [ms].
  - o Momentary. Values with voltage percent greater than or equal to 110% but less than or equal to 140%, and duration greater than 500 [ms] and less than or equal to 3,000 [ms].
  - o Temporary. Values with voltage percent greater than or equal to 110% but less than or equal to 120%, and duration greater than 3,000 [ms] and less than or equal to 60,000 [ms].
  - o All. Gets all SWELLS records stored in the selected time range.

4.5 Calculated data

- SCADA Equivalent Value: It is used when the real-time data is not available, normally from a SCADA, or it is necessary to compare the actual SCADA value with an estimated value based on historical data. It is obtained with the following sequence:
  - o Calculate the equivalent previous date, as the day of the previous week that is similar to the current day, or the day of the previous month that is similar to the current day.
  - o Take the current time without minutes.
  - o Request the average of historical values for the current time on the equivalent previous date.
- Last Stored Value: For any variable, it is requested which was the last value that was entered in the historical record and the corresponding timestamp. It allows to carry out validations of the historical record operational status to estimate the quality of the stored data.

5 Conclusion

The value that the **Optimal Extractor** has added to the processes of support the operational decisions in a Smart Grid context has been very relevant, and every time specialist users have found new ways to take advantage of this advantage of being able to visualize the EPS behavior over time.

For example, the **Optimal Extractor** allows you to graph or export to Excel the hourly average

measurements by Phase for each Substation Circuits in one year; this query will generate approximately 8,760 values per electrical parameter (365 * 24) regardless of the equipment sampling frequency; a normal extraction of raw data with a sampling frequency of 10-minutes implies transfer approximately 52,560 values for each parameter (365 * 24 * 6) and double if the sampling frequency is 5-minutes (365 * 24 * 12), in addition to the time and hardware resources on the client-side, required to process all the data obtained.

The integrated technology, the optimal data extraction strategies, and the adopted standards have made it possible to perform high-level functions with extraordinary performance, reducing even 95% of the waiting time to user response. For instance:

- An EPS operator can get the graph of the hourly real and reactive power profile for the last 24 hours in approximately 5 seconds.
- The graph or table of the hourly maximum values for voltage or real power measurements of a circuit for a year is generated in approximately 50 seconds.
- If the above query is made per day, the response time is less than 20 seconds.

A very representative and valuable query of the **Optimal Extractor** as support to the operational decisions for an EPS operator, during a failure and reestablishment event, is to have the ability to calculate on the fly (OLAP), the maximum hourly demand profile for the circuits involved, the circuit faulted and those that can support the restoration; the integrated function allows having the graph on screen in less than 3 seconds for each circuit.

Another integrated function that provides greater value for the users in charge of EPS operational analysis, is the computing of the Coincident Peak Demand (CPD) for all the circuits in a geographical region in a year [4]. Manually, this analysis for a geographical region with at least 500 circuits, can take from 2 to 3 months; the **Optimal Extractor** calculates the value in approximately 30 seconds and 2 minutes if the data quality algorithms are applied and generating the calculation memory required to support the results and operational decisions.

Table 1 shows results obtained comparing the three architecures performance:

- ARQ1: Traditional components architecture.
- ARQ2: Proposed components architecture for syntactic interoperability.
- ARQ3: Proposed components architecture for semantic interoperability.

In all the Test Cases, the ARQ2 obtains the better time response for the final user, and the improvements compared to the ARQ1 are very notable. Regarding ARQ3, when the data amount is relatively low, the time response for the final user can be greater than ARQ1, this is due to the metadata required by implementing the CIM Instances, in general, this effect is not presented when the data amount is increased, and the final user perception is not affected due the total time added is less than 2 [s].

The architecture and strategies proposed for the **Optimal Extractor** allow progress in the implementation of functions for the Smart Grid in the context of the operation of the EPS.

Table 1. Comparison results using the three architectures described.

Test Case			ARQ1		ARQ2			AQR3		
Case	Grouped by frequency	Period	Float values* transferred	Total time for user [s]	Float values* transferred	Total time for user [s]	Time % (ARQ2/ARQ1)	Float values* transferred	Total time for user [s]	Time % (ARQ3/ARQ1)
Statistical (AVG)	Hourly	1 Hour	420	0.15	35	0.09	60.0%	35	0.28	186.7%
		1 Day	10,080	0.47	840	0.16	34.0%	840	1.52	323.4%
		1 Week	70,560	0.80	5,880	0.14	17.5%	5,880	2.42	302.5%
		1 Month	302,400	6.73	25,200	0.86	12.8%	25,200	5.57	82.8%
		1 Year	3,679,200	71.36	306,600	6.92	9.7%	306,600	15.89	22.3%
	Daily	1 Day	10,080	0.37	35	0.13	35.1%	35	0.32	86.5%
		1 Week	70,560	0.70	245	0.29	41.4%	245	1.33	190.0%
		1 Month	302,400	5.93	1,050	1.36	22.9%	1,050	2.73	46.0%
		1 Year	3,679,200	70.86	12,775	2.62	3.7%	12,775	6.88	9.7%
	Monthly	1 Month	302,400	6.23	35	0.27	4.3%	35	0.43	6.9%
		1 Year	3,679,200	70.06	426	1.92	2.7%	426	3.06	4.4%
	Anualy	1 Year	3,679,200	69.26	35	1.74	2.5%	35	1.89	2.7%
CPD	Anualy	1 Year	18,396,000	^86729.55	4	3.90	0.004%	4	4.09	0.005%
* 32 Electrical measurements + Timestamp (date - time) + Circuit ID										
^ Includes 86,400 [s] for CPD processing in the client side										

References:

[1] A. Espinosa-Reza, A. Quintero-Reyes, R. Garcia-Mendoza, J.F. Borjas-Diaz, T.M. Calleros-Torres, B. Sierra-Rodriguez and R. Torres-Abrego, On-Line Simulator for Decision Support in Distribution Control Centers in a Smart Grid Context, *WSEAS Transactions on Systems and Control*, Vol. 5, No. 10, 2010, pp. 814-816.

[2] A. Espinosa-Reza, H.R. Aguilar-Valenzuela, M. Molina-Marin, M.L. Torres-Espindola, T.M. Calleros-Torres, E. Granados-Gomez, R. Garcia-Mendoza and C.F. Villatoro-Hernandez, Semantic Interoperability for Operational Planning for the Electric Power Distribution System, *WSEAS Transactions on Power Systems*, Vol. 11, 2016, pp. 289-298.

[3] M. Molina-Marin, E. Granados-Gomez, A. Espinosa-Reza and H. R. Aguilar-Valenzuela, CIM-Based System for Implementing a Dynamic Dashboard and Analysis Tool for Losses Reduction in the Distribution Power Systems in México, *WSEAS Transactions on Computers*, Vol. 15, 2016, pp. 24-33.

[4] A. Espinosa, S. Gonzalez and H.R. Aguilar, Results of applying a semantic interoperability strategy in Smart Grid applications for DSO in Mexico, *CIGRE Session 2016, General Meeting, Paris, France*, 2016.

[5] A. Espinosa-Reza, H. R. Aguilar-Valenzuela, M. Molina-Marin, M. L. Torres-Espindola, T. M. Calleros-Torres, E. Granados-Gomez, R. Garcia-Mendoza and C.F. Villatoro-Hernandez, Implementation of a CIM-Based Semantic Interoperability Strategy for Smart Grid in Mexico, *International Journal of Power Systems*, Vol 1, 2016, pp. 33-40.

[6] A. Espinosa-Reza, M.L. Torres-Espindola, M. Molina-Marin, E. Granados-Gomez and H.R. Aguilar-Valenzuela, Semantic Interoperability for Historical and Real Time Data Using CIM and OPC-UA for the Smart Grid in Mexico, *WSEAS Transactions on Cumputers*, Vol. 15, 2016, pp. 1-11.

[7] Jose Alfredo Sánchez-López, Eduardo Islas-Pérez, Alfredo Espinosa-Reza and Agustín Quintero-Reyes, Deploying SCADA Data to Web Services for Interoperability Purposes, *IEEE 2015 Global Information Infrastructure and Networking Symposium GIIS 2015*, Mexico, 2015.

[8] R. Rhodes, *Common Information Model Primer, Fourth Edition*, EPRI Technical Report, 2018.

Contribution of individual authors
Marxa Torres-Espindola, carried out the development of the components in C#.NET and implementation in many information systems.
Alfredo Espinosa-Reza, was responsible for the design of the architecture proposed, CIM Profile validation, and tester of the final products.