

# Scalable and Usable Web Based Supervisory and Control System for Microgrid Management

E. Álvarez, A. M. Campos, R. García, S. González and C. Díez

R&D Department  
CTIC Foundation

Parque Científico- Tecnológico – Gijón, Asturias (Spain)

Phone/Fax number:+0034 984 291212/+:+0034 984390612, e-mail: {eduardo.alvarez, antonio.campos, rodrigo.garcia, saray.gonzalez, carlos.diez}@fundacionctic.org

**Abstract.** This work presents an Ad-hoc Web-Based SCADA for microgrid supervision and management. The SCADA was developed for centralized demand-driven microgrid control. It was designed for usability, incorporating a user-friendly human machine interface (HMI) based on the mash-up philosophy, which can be easily adapted by the user. This work presents a user-focused alternative for microgrid management.

## Key words

Microgrid, SCADA, mash-up, renewable energy, power systems.

## 1. Introduction

As microgrids are able to reduce problems related to the use of standard electrical power supply, they have become an effective and balanced tool to aid in the use of local sources of renewable and non-renewable energy. Microgrids offer distinct advantages, such as improved energy efficiency, minimization of overall energy consumption, reduced environmental impact, improvement of reliability and resilience, operational network benefits, and more cost-efficient replacement of electrical infrastructures.

In [1], microgrids are categorized in two modes. In the first, not island or connected mode, the microgrid is connected to the main network. In the second, island mode, the microgrid is not connected to the principal network in the case of planned events, such as maintenance tasks; or unforeseen circumstances, such as failures or sudden disconnections from the main network.

In not island mode there are two ways of managing the energy production of the microgrid. The first is based on the premise of a minimal exchange of energy with the main network. The microgrid generates energy according to demand, consuming as little energy as possible from the main network. In the second, the microgrid

participates in the market, buying and selling energy. In this case the microgrid produces energy to satisfy its own demands as well as to supply other microgrids. Alternatively, the microgrid can buy energy from other sources. This is often the case when energy is available at a relatively cheap rate, or when the energy generated is insufficient.

In order to manage the two microgrid categories, two architectures are proposed. They are described in [2]. In the hierarchical architecture there is a Central Controller element that governs the Local Controllers of the energy sources. In the decentralized architecture, the control is replicated in the Local Controllers of the energy sources.

In addition to the functioning mode and the type of control of the microgrid, it is necessary to set several parameters in order to optimize the management of the microgrid. It is necessary to know the power demand and the cost of producing this energy. With these parameters and using forecasting algorithms or real-time mathematical models, it is possible to determine how much energy is produced by each microgrid source and thus, to achieve the most efficient configuration. In both cases, several parameters must be inserted in the system in order to calculate the final distribution.

One of the principal advantages of microgrids is the reduction of production costs and of energy consumption. As the saying goes: "If you can't measure it, you can't improve it". The capacity to monitor the functioning of the system, and therefore to visualize any variable (such as production, demand or gas emission); as well as any abnormal state of the system (such as mal-functioning or raised consumption), will help to optimize energy management in microgrids.

The use of a SCADA (Supervisory Control and Data Acquisition) makes configuration and supervision of microgrids easier. Any layman can make modifications such as the introduction of a new energy source in the system. The user can adapt the system by simply

introducing information, (such as the efficiency curve or the cost of fuel), by means of intuitive interfaces and simple screens which the user can personalize.

Total knowledge of the functioning of every element allows the user to supervise the system at all times, and therefore to resolve unforeseen situations quickly. The user is able to control the system in functions such as starting or stopping sources of energy.

This research is focused on a connected mode microgrid, aiming for a minimal exchange of energy with the main network. The microgrid has hierarchical architecture and uses a real time system to calculate energy production. This type of microgrid increases the quantity of energy produced and reduces costs.

This paper proposes an innovative SCADA as a control and supervision system for connected mode microgrids with hierarchical control. This flexible and simple SCADA system is based on mash-up interfaces. One of its main advantages is that the components are based on free software and therefore require no economic outlay.

## 2. Related Work

There are various studies into microgrid control systems which differ in their architecture: distributed or hierarchical.

In distributed control systems each energy source is responsible for its own control. Each energy source has a Local Controller responsible for computing the actions necessary for smooth running of the equipment. In this type of architecture there is no master controller, that is to say, all controllers are at the same level. This type of architecture is explored in various works such as [3], focusing on solving problems related to electricity supply problems in populations located far from the power plant.

Long power transmission lines lose energy, power supply quality decreases and bottlenecks can appear in distribution and transmission. To solve these problems a microgrid is used to minimize the distance between the point of power generation and the loads. Power supply improvement is the only objective in [3], but production cost is not taken into account.

In hierarchical based control systems there is a Central Controller (MGCC) that heads the control system. At the second level, there are controllers located at microsources (MC) which exchange information with the MGCC and are in charge of the operation at each MG point.

As an example of this kind of architecture, a noteworthy work is proposed in [2]. This research focuses on the inclusion of microgrids in the power market, making it possible to buy or sell energy. The General Controller, which operates as the master of the system, includes several control functionalities: dispatching of active and reactive-power to microsources, demand management, and control of connection of different elements. It is based on the forecast of load and renewable production,

and information from the open market. The result is an economical energy production plan to set the microsources and the loads to be supplied over the following hours. However, this architecture does not allow control of economic or environmental costs or on-line running, which would allow the microgrid generation to adapt to variations in demand.

The research developed in [4], is focused on running and environmental cost. In this case, several mathematical methods are studied (SQP, MADS and genetic algorithms), which provide an economical energy production plan consisting of the settings of the microsources and loads to be supplied. These mathematics methods are used in real time to minimize fuel consumption and to control costs and emissions, while satisfying power demand. In this case the microgrid does not sell or buy energy. Although these methods are used in real time, they are excessively slow and they require high performance machines.

In [5], an alternative to these mathematical methods is proposed: an algorithm based on the generation of electrical cost functions of microsources. Using this algorithm, microgrids can be power controlled on-line, minimizing running cost and greenhouse emissions, and allowing seamless integration in the system. Furthermore, only low-cost, off-the-shelf logic controllers are necessary. This algorithm uses external data such as the number of microsources of the microgrid, fuel cost used by each energy source, and their efficiency functions. These efficiency functions are found in the manufacturer's technical specifications of the energy source. This curve is specified as a table of power-efficiency values which must be inserted in the system manually. The insertion of data may pose difficulties as program modification may be required each time a new energy source is plugged into the microgrid. An HMI, specifically a SCADA, facilitates the job, allowing easy introduction of configuration variables at any time. A SCADA also offers a supervision function. In a microgrid which aims for cost minimization, the variables that can affect this saving must be known at all times. The SCADA informs the user of each variable clearly and simply. The user can see the state of all of the system variables, as well as any alarm states, at any time. He can even modify the microsources.

There are SCADAs on the market that can be adapted to microgrids; however they offer many functions that increase cost without offering any added value. Moreover, they require the use of specific hardware and software. The SCADA proposed in this work offers the functions needed for supervision and control in this kind of microgrids at nearly zero cost, with complete freedom in terms of the hardware used.

There are various studies in which a SCADA is proposed for microgrid supervision and control without this being the main aim of study. Many of these SCADAs are developed for concrete systems but they lack the flexibility necessary to allow them to adapt to changes in the system. As an example, in [6] a web SCADA is

proposed for a microgrid in the energy market. However, as the microgrid considered in this work aims to minimize the energy exchange with the grid, a Web SCADA is not the best solution. Others, like [7], depend on specific hardware. The SCADA proposed is hardware independent; because of the use of a standard communication protocol, it is able to communicate with any device that uses this protocol. In this work a web SCADA is proposed for real-time control of the Hierarchical microgrid presented in [5]. This is a user-oriented web SCADA without the complexity of a general purpose SCADA and with user-friendly interfaces.

### 3. Architecture

The general architecture of the system is shown in Figure 1. It has a series of layers that give the user access to field variables. The OPC client accesses the OPC server and can obtain the variables from the general controller. The information is sent to the XML-DA server which allows REST requests, making it possible to visualise the information using a web interface.

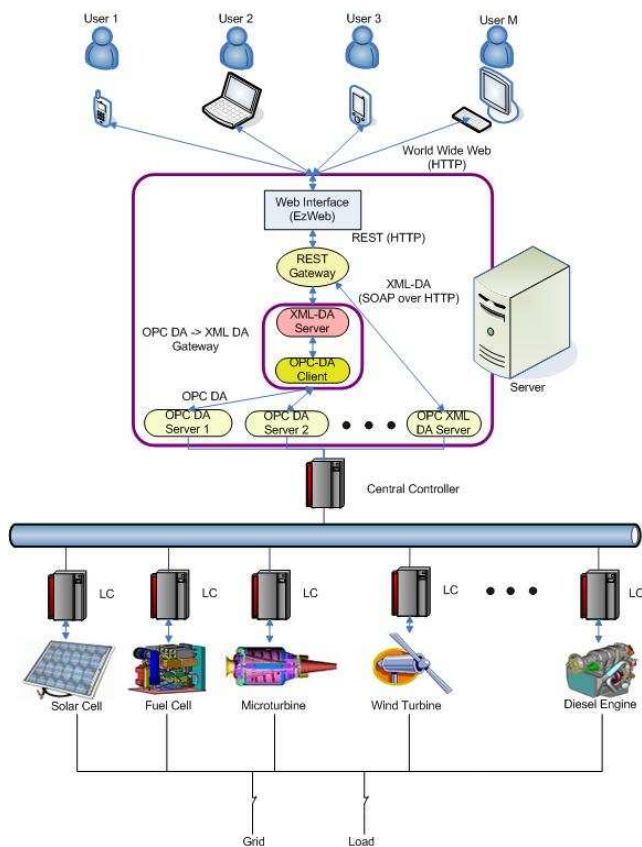


Fig. 1. General system architecture

Communication can travel in the opposite direction, so that the user can modify the field variables across the web interface. This architecture allows communication between the physical elements of the process and the Internet. Communication with the field elements is based on the OPC standard [8]. The OPC standard is a non-proprietary technical specification created by an industrial

automation task force and now maintained by the OPC Foundation. It consists of a standard interface system based on OLE/COM [9] by Microsoft. OPC allows inter-operation between information systems or desktop applications. Thus, applications which communicate with equipment controlled by PLC's can be developed.

The OPC-DA client is responsible for the interaction with the OPC-DA servers, which are usually provided by the equipment manufacturer. The OPC-DA client is built according to the OPC specification, OPC-DA [10] (OPC Data Access). OPC clients are usually used for transferring real time data from PLCs and other control devices to HMIs and other clients. In this case, OPC-DA client is used for transferring real time data from the Central Controller to the XML-DA server.

The XML-DA server communicates with the client and allows sharing and interchange of real time data from field devices of a wide range of platforms. The XML-DA server is based on the OPC XML-DA specification [11]. OPC XML data access is the adoption of XML technology [12] by the OPC Foundation to facilitate data interchange from the plant to Internet and to the business domain. The interfaces based on OPC XML-DA simplify the sharing and interchange of OPC data, supporting HTTP [13], SOAP [14], subscription based services, and OPC-DA data. In the future, PLC manufacturers will provide OPC XML-DA servers with their products and neither the OPC Client nor the XML-DA Server described here will be necessary.

The REST Gateway is a REST (Representational State Transfer) [15] component which allows access to the XMLDA Server data from the web in a RESTful way. The gateways from the web and from the XML-DA Server are communicated through the HTTP protocol using a subset from the HTTP operations. These operations are PUT, POST, GET and DELETE. This subset is the base of Internet and ensures connectivity and inter-operability. The invocation of this gateway through its URI (Universal Resource Identifier) allows the web interface to access and modify data. Also, this small application can invoke the XML-DA Server and repeat the corresponding command using the operation subset from the previous case. The REST Gateway acts as a translator between the whole system and the web architecture; it is the module that allows data requests from any system connected to the Internet.

The Web Interface is developed using the Web mash-up philosophy. These applications are web sites or web applications which make it easy to create new content in an intuitive way using Web-services. The free-software EzWeb[16] platform is used as a mash-up Web Interface. EzWeb allows the creation of an advanced operational environment built from simple elements (Gadgets). The Web Interface can access plant data through the OPC Gateway described above, through its URI. EzWeb allows visualization of data, and information exchange between the user and the field devices.

This architecture is a versatile system which is easy to use, install and configure. The use of the OPC standard makes it hardware-independent, as long as the manufacturer provides OPC-DA or OPC XML-DA compliant server. The proposed Web interface has been developed in a user-friendly way. As it is a mash-up web interface, it is truly customizable: the user can adapt the HMI to his preferences.

#### 4. Interface

These screens were developed for the supervision and control of the microgrid. They were designed using the mashup Web philosophy to improve usability.

Using these screens the user has access to all the information of the system and the user can control the following actions.

- View energy production and efficiency.
- View energy consumption.
- View gas emissions.
- View the state of the physical elements of the system.
- View alarms.
- Control energy sources (Run /Stop).
- Set efficiency curves of energy sources.

The configuration of screens shown is an example. As this is a mash-up environment, the user can adapt his environment by adding or removing gadgets, grouping them for tabs, changing the order, etc.

Below are the screens of the SCADA and the navigation between them.

The first screen is the login screen. On this screen, access of the user to the system is permitted or denied.



Fig. 2. Login screen

The following screenshot shows the production of current energy of every source of the microgrid and relevant information of the source selected by the user. Any alarm created by the system is shown on the screen. The user can control the system by starting or stopping the sources.

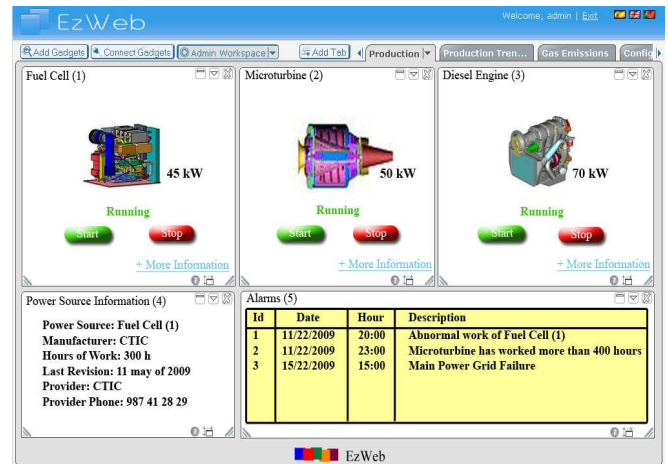


Fig. 3. Production screen

The screen configuration allows to the user to specify the function of efficiency of the source, information of interest about the source, and fuel cost. The screen displays the cost/Kw curve required in reply to the user's request.

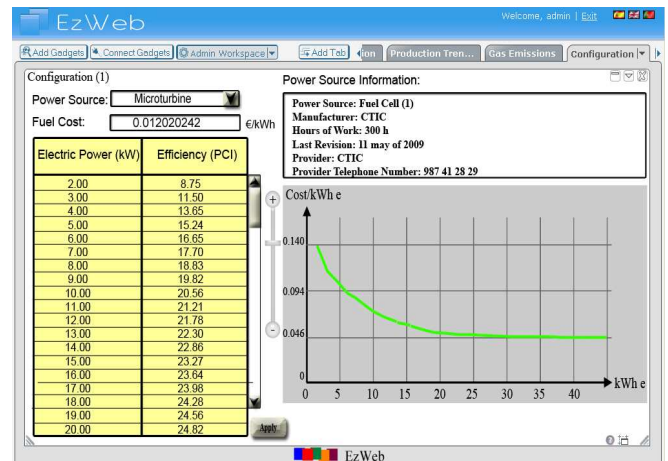


Fig. 4. Configuration screen

The user can visualize the production of energy by every source in real time. A comparative graph between energetic production and energetic demand is also shown.



Fig. 5. Production graphs screen

The following screen shows the graph of gas emission from each source and the total emissions from the

system. The gases shown are CO<sub>2</sub>, SO<sub>2</sub>, NO<sub>x</sub> and Particles.

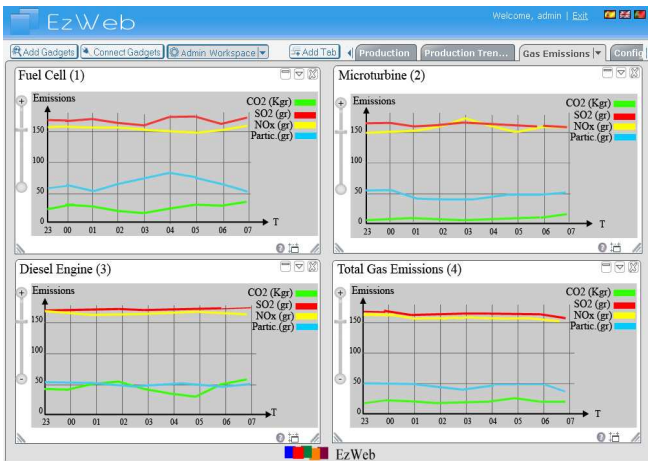


Fig. 6. Gas emissions screen

In this screen the user can select the OPC Server and specify the refresh time. The connection status (Connected or Disconnected) and relevant information are shown.

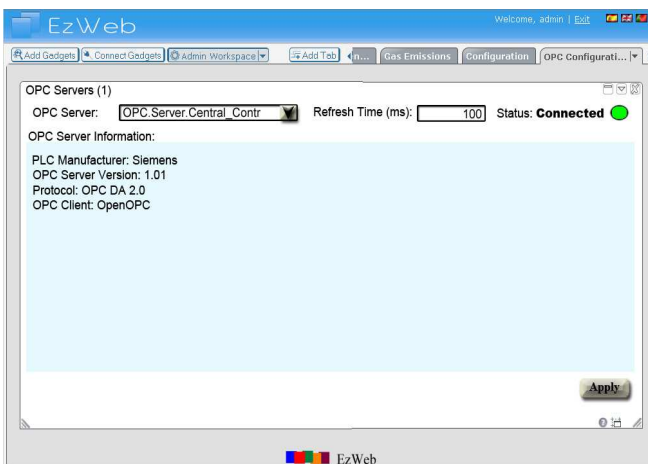


Fig. 7. OPC Configuration screen

In this screen a non-administrator user can see the energy consumption in real-time or the consumption between two dates.

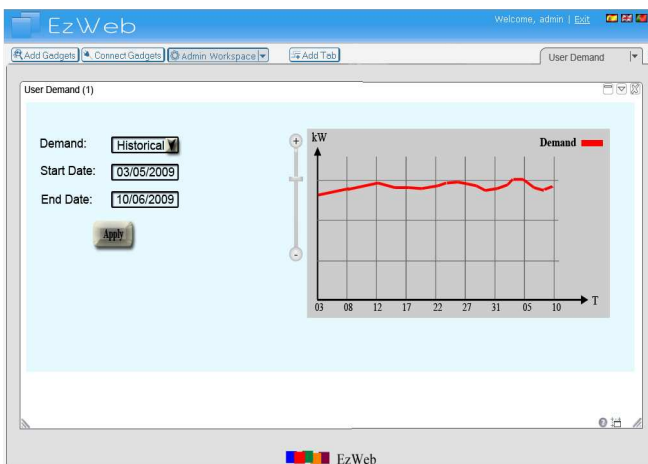


Fig. 8. User demand screen

## 5. Installation

The installation of this SCADA is easy for the user; he need only:

1. Install the PLC software (OPC Server).
2. Install the communications gateway between web and field devices.
3. Install EzWeb.
4. Add the desired gadgets to configure the work environment
5. Configure the OPC server using screen 7.
6. Configure the microsources using screen 4.

If new sources of energy are added, the user need only modify the microsource information provided by the manufacturer on screen 4 (step 6). The user can add gadgets related to the microsources.

## 7. Performance

Various experiments have been conducted in order to observe the performance of the application.

The tests were carried out on a 64 bits Windows 7 machine with a 1 Core 2 Quad processor Intel CPU and a RAM 8 GB. Softing OPC Demo Server was used.

Apache JMeter was used to test performance. Apache JMeter is open source software, a 100% pure Java desktop application designed to load test functional behavior and measure performance. It was originally designed for testing Web Applications but has since expanded to other test functions.

The following tests have been conducted. The results obtained in each are also shown below:

- Read request: 1 client, 2000 requests.  
Average: 4ms  
90 percentile=4  
Standard Deviation= 27.31ms  
Performance: 214.8request/s
- Read request: 2 clients, 2000 requests.  
Average: 4ms  
90 percentile=4  
Standard Deviation= 0.75ms  
Performance: 255.2request/s
- Read request: 5 clients, 2000 requests.  
Average: 4ms  
90 percentile=6  
Standard Deviation= 10.07ms  
Performance: 431.3request/s
- Read request: 10 clients, 2000 requests.  
Average: 9ms  
90 percentile=14  
Standard Deviation= 3.77ms  
Performance: 552.4request/s
- Read request: 20 clients, 2000 requests.  
Average: 23ms  
90 percentile=39  
Standard Deviation= 28.65ms  
Performance: 623.1request/s

- Read request: 50 clients, 2000 requests.  
Average: 71ms  
90 percentile=11  
Standard Deviation= 46.98ms  
Performance=608request/s

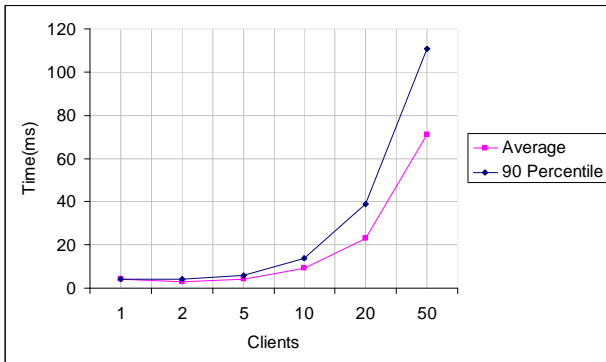


Fig. 9. Tests results (average,90 percentile)

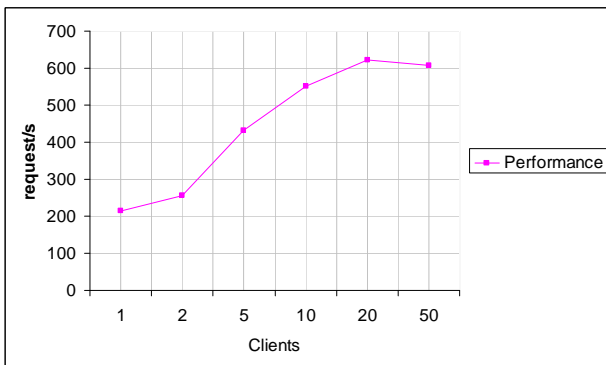


Fig. 10. Tests results (Performance)

## 8. Conclusion

In this work we propose a SCADA for centralized demand-driven microgrids. This kind of microgrid aims for a minimal exchange of energy with the power grid in order to satisfy demand with optimum cost.

The proposed SCADA is a simple architecture based on the OPC standard for communication with processes. This is what makes the SCADA hardware-independent. It is a control and supervision system that uses a mash-up web interface, which facilitates customization. Because it has a web architecture, it also facilitates access to the system with a simple web browser. It is also possible to use the system from various platforms or devices. The screens have been designed so that any user can supervise the system without extensive knowledge of the SCADA.

The proposed SCADA is a simple, usable alternative which solves the supervision and control needs of a microgrid.

## References

[1] J.A.P. Lopes, J.T. Saravia, N. Hatziaargyriou, N. Jenkins, "Management of MicroGrids" presented at

IEEC, Bilbao 28-29 October 2003.[Online] Available: [http://microgrids.power.ece.ntua.gr/documents/Microgrids\\_management\\_jeec2003.pdf](http://microgrids.power.ece.ntua.gr/documents/Microgrids_management_jeec2003.pdf)

[2] European Research Project "Large Scale Integration of Micro-Generation to Low Voltage Grids, DC1: MicroGrid Central Controller strategies and algorithms". [Online] Available: [http://www.microgrids.eu/micro2000/deliverables/deliverable\\_DC1\\_part1.pdf](http://www.microgrids.eu/micro2000/deliverables/deliverable_DC1_part1.pdf)

[3] P. Piagi, R.H. Lasseter, "Autonomous Control of Microgrids", Proc. IEEE Power Engineering Society General Meeting 2006, 18-22 June 2006, pp. 8.

[4] F. A. Mohamed: "Microgrid modelling and online management".Helsinki University of Technology (2008). Master's Thesis. [On line] Available: <http://lib.tkk.fi/Diss/2008/isbn9789512292356/>

[5] E. Álvarez, A.M. Campos, "Microgrid On-line Power Dispatch Algorithm based on Electrical Generation Cost Functions of Microsources", UPEC 2009 The 44<sup>th</sup> International Universities' Power Engineering Conference.

[6] Pipattanasomporn, M.; Rahman, S., "Intelligent distributed autonomous power systems (IDAPS) and their impact on critical electrical loads," Critical Infrastructure Protection, First IEEE International Workshop on , vol., no., pp. 9 pp.-, 3-4 Nov. 2005.

[7] Li Wang; Kuo-Hua Liu,"Implementation of a Web-based Real-Time Monitoring and Control System for a Hybrid Wind-PV-Battery Renewable Energy System," Intelligent Systems Applications to Power Systems, 2007. ISAP 2007. International Conference on vol., no., pp.1-6, 5-8 Nov. 2007.

[8] OPC Foundation Website [Online] <http://www.opcfoundation.org>, 2009

[9] OLE/COM Desc. [Online] Available [http://www.microsoft.com/](http://www.microsoft.com/com/)

[10] OPC Data Access Specification Version 2.0, [Online]. Available: <http://www.opcfoundation.org>, 2009

[11] OPC XML Data Access Specification Version 3.0, [Online] Available: <http://www.opcfoundation.org>, 2009.

[12] Extensible Markup Language (XML) 1.1 (Second Edition), C. M. Sperberg-McQueen, F. Yergeau, T. Bray, J. Paoli, E. Maler, J. Cowan, Editors, W3C Recommendation, 26 November 2008, <http://www.w3.org/TR/2006/REC-xml11-20060816/>. Latest version available at <http://www.w3.org/TR/xml>.

[13] T. Berners-Lee, R. Fielding, J. Gettys, J. Mogul, L. Masinter, P. Leach, H. Frystyk, Hypertext Transfer Protocol - HTTP/1.1, RFC 2616, June 1999. [Online] Available: <http://www.ietf.org/rfc/rfc2616.txt>.

[14] SOAP (Simple Object Access Protocol) Version 1.2, W3C Recommendation 2003/6/24, [Online] Available: <http://www.w3.org/TR/SOAP>.

[15] Vinoski, S., "RESTful Web Services Development Checklist," Internet Computing, IEEE , vol.12, no.6, pp.96-95, Nov.-Dec. 2008.

[16]EzWeb[Online] Available:<http://ezweb.morfeo-project.org/>