Mansoura Engineering Journal

Volume 36 | Issue 2 Article 8

11-11-2020

Web-Based Online Monitoring and Control for Economic Load Dispatch.

M. Tantawy
Electrical Engineering Department., Faculty of Engineering., El-Mansoura University., Mansoura., Egypt.

M. El-Said
Electrical Engineering Department., Faculty of Engineering., El-Mansoura University., Mansoura., Egypt.

A. Hassan
Electrical Engineering Department., Faculty of Engineering., El-Mansoura University., Mansoura., Egypt.

Sinar Shebl
Electrical Engineering Department., Faculty of Engineering., El-Mansoura University., Mansoura., Egypt.

Recommended Citation

Tantawy, M.; El-Said, M.; Hassan, A.; and Shebl, Sinar (2020) "Web-Based Online Monitoring and Control for Economic Load Dispatch.," *Mansoura Engineering Journal*: Vol. 36 : Iss. 2 , Article 8.
Available at: https://doi.org/10.21608/bfemu.2020.122781

WEB-BASED ONLINE MONITORING AND CONTROL FOR ECONOMIC LOAD DISPATCH

المراقبة و التحكم على شبكة الانترت لمشكلة التوزيع الاقتصادي للأحمال

M. A. Tantawy*, M. I. El-Said*, A. E. Hassan* and Sinar K. Shebl
* Electrical Engineering Department, Faculty of Engineering, Mansoura University

ملخص البحث

إن خدمات الشبكة العنكبوتية وما يتوفر من تطوير مستمر لتطبيقاتها يساعد و يعمل على التوسع و الانتشار في إمكانية الحصول على البيانات و متابعتها و تحديثها دون الارتباط بالمكان. و تتطور التطبيقات على الشبكة لتوفر إمكانية تنفيذ نماذج معمارية البرمجيات الموزعة لأداء عمليات القوى الكهربائية كعملية مراقبة سريان الأحمال لنظام كهربي متعدد المناطق بطريقة آمنة الى حد كبير و موزعة و ذات منصة برمجيات مستقلة. إن عملية توزيع الأحمال اقتصاديا هي عملية حيوية للنظام الكهربائي لتوزيع القدرة الفعالة للحمل على جميع وحدات التشغيل بشكل مناسب بغرض تقليل تكاليف التشغيل لوحدات التوليد مع تحقيق قيود تشغيل النظام. و في هذا البحث يتم تقديم مقترح برمجي تطبيقي للمراقبة و التحكم في توزيع الأحمال اقتصاديا. يقدم هذا المقترح نظام بنائي جديد مطبق على مشكلة توزيع الأحمال اقتصاديا من خلال تطبيقه على الشبكة العنكبوتية. هذا البحث يهدف الى التأكيد على أهمية الاتجاه الى الغاء المركزية و لعملية المراقبة و التحكم و أيضا لنشر الحاجة الى سهولة التصفح من أي طرف متاح

Abstract -- **Web services, and more in general service-oriented architectures, are emerging technologies of choice for implementing distributed architectural models for performing power system operations such as on-line load dispatch and load flow monitoring of multi-area power systems in a complete secure, distributed and platform independent environment. On-line load dispatch monitoring requires the calculation of economic load dispatch solutions by using real-time data obtained from the power system clients. The power system client applications are running in a heterogeneous environment, in this paper an attempt is proposed to apply a web-based online monitoring and control for the economic load dispatch (ELD) problem. The work is presenting a new system architecture applied to the ELD problem for monitoring and controlling on a web-based application. The work here aims to emphasis the importance of decentralization of the monitoring and controlling procedures and to spread the need for anywhere accessibility of the electrical power system.**

Keywords: web-based, economic load dispatch, online monitoring and control

I. INTRODUCTION

The rapid progress of Internet-based networking technologies enables a remote access to electrical system equipment and instruments. Remote access power system network is gradually emerging with the advance of technology. Therefore the use of advanced technology software systems is required to improve the performance and reliability of the electrical power systems, by making the maximum available benefit of what the technology is providing every day. Basically the remote web based real time system is based on the concept of using the web to monitor and control of the power system network. The existing Web enabled models for power system operations are mainly concerned with exchange of information and do not provide a reliable environment to solve power system problems [1]. The exchange of computing power is inherent in the grid environment and hence quicker and fault tolerant models can be proposed for solving power system problems.

The improving of Internet based applications, and the rapid change of technology used, not only has helped engineers developing more featured implementations solving their sophisticated problems, but also helped with providing less cost hardware to be used. Meaning that for such a process of monitoring through the web for an electrical power system, all what you need is a Personal Computer (PC), a web browser and a reliable Internet connection.

The use of huge servers or high specifications computers isn't needed as it was before. Web applications make it completely easy to access it from anywhere with any kind of computer as long as it is connected to the World Wide Web.

II. STATE-OF-THE-ART

The rapid growth of Internet applications greatly changes the appearance and operation of industrial and commercial products in many aspects. Modern instrumentation integrating control, communication and measurement for laboratory or factory automation will ultimately be dependent on its Internet technology base. Chung-Ping et. al. [2] demonstrated an advanced Virtual Instruments (VI) application on multiple client PC's bound to a dedicated Intranet environment. They constructs an innovative power monitoring software architecture, applying the IP-multicast concept on real-time implementation. The conventional client/server architecture is good for a client to retrieve documents from the Web server; however, its performance is insufficient for real-time interactive multiple accesses [3-5]. Instead the IP-multicast technique, allowing multi-point-to-multipoint data exchange in an optimized

Accepted June 30, 2011.

fashion, generates the data exactly once and delivers it to a group of end users [6,7]. It demonstrates advantages in saving consider- able bandwidth and allows passive reception, so the real-time transmission is readily accomplished.

Web-based services are commonly used in the trade of products and in sharing information. In the utility sector, many enterprises provide services that allow customers to analyze their energy consumption and billing data. Web-based services can also be used for the collection of measurement data and for the control of devices. The implementation of services connected to actual processes of an industrial company sets quite many new questions concerning data management, server side communications and data security [8]. A complete Web based, platform independent power system simulation package with various analyses distributed in a clustered environment has been modeled by Irving et.al. [9]. Sando et. al. [10] demonstrated through experimental results that online security analysis could be executed in lesser period of time even for large power system. In future every electrical generator will be equipped with computational and communication facilities. Grid computing can provide a relatively inexpensive new technology allowing the output of embedded generators to be monitored. The ability of grid enabled systems to interact autonomously will be vital for small generators where manned operation is unlikely to be viable.

All the previous models are working on creating a web-based model for monitoring and controlling of electrical power system, yet Economic Load Dispatch trails are too few, and this is the problem we are focusing on in this paper. All the models used database as the only way to store and retrieve the data, and single components to process the whole functionality of the system. In this paper we are discussing other different approaches to provide more reliable and scalable systems. Previous stated researches created the web user interface for the monitoring consisting of blocks showing results and changing on request of updates, the proposed architecture web user interface, uses high level tools for viewing more enhanced and pretty interface for an easier usability of the system.

III. ONLINE MONITORING AND CONTROL

Web applications are popular due to the ubiquity of web browsers, and the convenience of using a web browser as a client, sometimes called a thin client. The ability to update and maintain web applications without distributing and installing software on potentially thousands of client computers is a key reason for their popularity, as is the inherent support for cross-platform compatibility. Common web applications include webmail, online retail sales, online auctions, wikis and many other functions. When developing Web applications, the data security and the performance of the application must be taken into consideration by different way. For example, the performance of the application is limited by the speed of data communication connection the user is using.

At device level, there are two basic ways to implement Web-based services. The first and the most straightforward method is to implement Transmission Control Protocol-Internet Protocol (TCP-IP) interface and services (Hyper Text Transfer Protocol "HTTP"-service) in the device itself. In this implementation method, it is required that the TCP-IP network is available in the field level. Since this is not the normal situation so far, it is often more convenient to use an existing field level bus or other existing remote reading system and to construct Web-services at the process control level or at the management level [8]. The power system operation and control needs huge volume of data where a new approach is needed to enable the power system data to be processed, analyzed and interpreted by different power system clients. Existing power system simulations are primarily desktop applications with a small number of exceptions implemented on parallel processing computers.

The existing Web enabled models for power system operations are mainly concerned with exchange of information and do not provide quick and reliable solutions to power system problems. The exchange of computing power is inherent to grid environment and hence quicker and fault tolerant models for power system operations can be proposed for design and implementation [11].

The electric power system monitoring process is the continuous measurement of the power system network information such as voltages (V), currents (I), phase angle between V&I, frequency, active power, reactive power, harmonics, and so on. That is to assure optimum operation and control of power system under normal and abnormal conditions.

Recently the continuously measured data are displayed on monitors of computers that exist in the control and monitoring room that is SCADA system where real-time measurements are displayed for the electric power system network operators.

The traditional architecture of a web-based monitoring application shown in Fig.1 consists of three basic tiers:

a. Presentation Tier
b. Application Tier
c. Database Tier

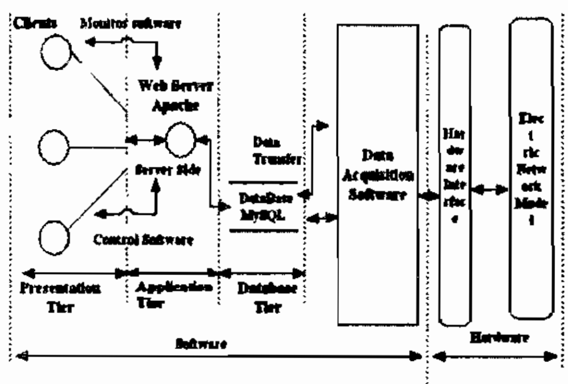


Fig.1 Three-Tier System Architecture

a. Presentation Tier

This tier enables user to interact with the database through the web browser in a user-friendly manner from remote location. It is also called user services layer. It provides user to access the application. This layer optionally permits to feed and manipulate data. This tier centralization of any monitoring and controlling systems. Being online by the older researches only included the accessing of data through the network on-time, but doesn't come through the real problems of being centralized, unreliable for any failures of that central system, the high cost of used servers and the problems of scaling the system or reconfiguring it.

In this paper, the proposed architecture system is designed to overcome problems of older researches and trials to improve the performance and reliability of the online monitoring and control process to be highly accessible and scalable, and to get the most available use of the advantages of web based concepts used with open source technologies. In the coming sections a detailed explanation of the system design and implementation. The architecture consists of 6 main components, they are:

1. Data Acquisition System (DAS)
2. Advanced Messaging Queuing Protocol (AMQP)
3. Service Back Ground (ELD Component)
4. Logging System
    - Distributed data base
    - Web Server
5. Web User Interface
6. Power Station
    - Network Generators
    - Queue consumer
    - Dispatcher

Fig. 2 shows the full architecture components.

Each component of the system has an independent role in the architecture, components' details and work flow are explained below.

a. Data Acquisition System

Data acquisition system (DAS) is the process of sampling signals that measure real world physical conditions and converting the resulting samples into digital numeric values that can be manipulated by a computer. Data acquisition systems typically convert analog waveforms into digital values for processing. Data acquisition applications are controlled by software programs developed using various general purpose programming languages such as Basic, C, Java, List. COMEDI is an open source Application Program Interface (API) used by applications to access and control the data acquisition hardware. For an electrical network, transferring the data online is being processed through the DAS.

In this system, the required input data would be the electrical load demand at a certain moment, the initial state of the generating units, the max and min power of each generator, the cost coefficients and the transmission line parameters.

b. Advanced Messaging Queuing Protocol

The Advanced Message Queuing Protocol (AMQ Protocol or AMQP) creates full functional interoperability between conforming clients and messaging middleware servers (also called "brokers").

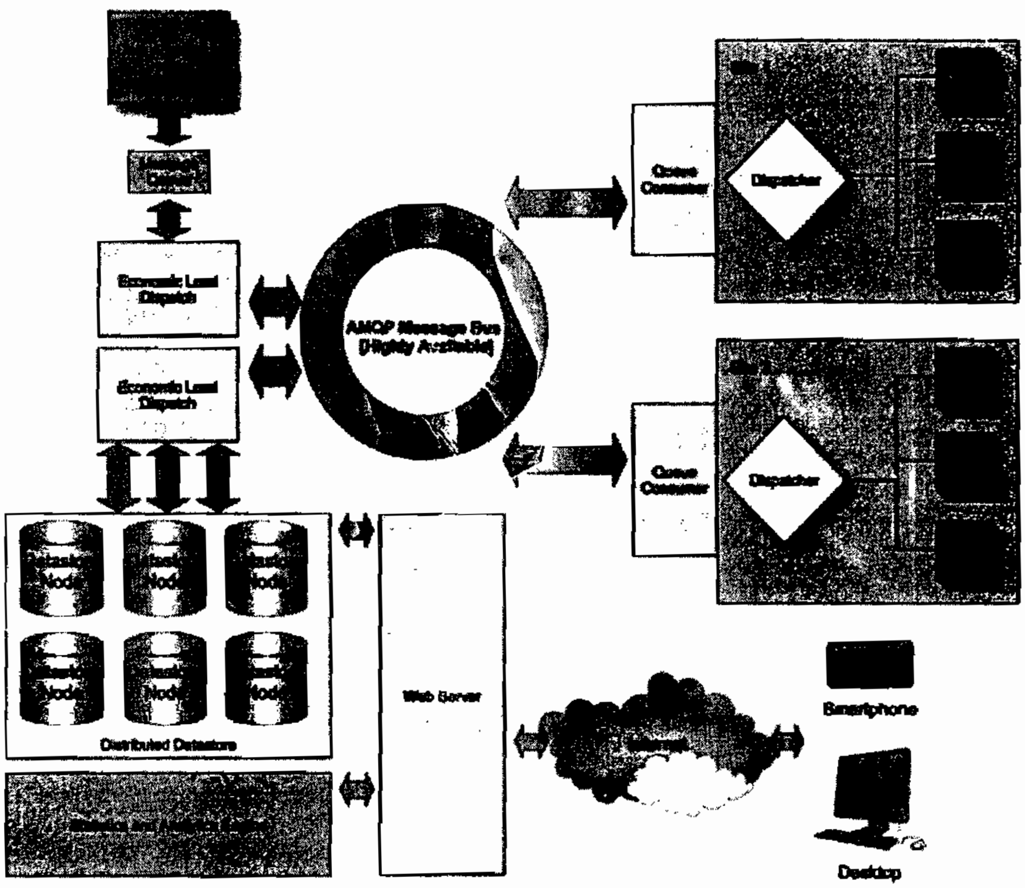


Fig. 2 Proposed System Architecture

could be build using java [12], python technology as a client and server side programming or PHP and JavaScript [13] as a client and server side programming.

b. Application Tier

Application tier consists of web server and application logic for data monitoring and control. This tier is also called application service layer. The logics and rules are separately stored in the files using Web scripts. These logics and rules are properly interfaced with the main Web server in this tier. This provides management services that are shared by multiple applications.

c. Database Tier

This tier is also called data service layer. It concerns with persistent data usually stored in a database or in permanent storage. This is the actual Relational Data Base Management System (RDBMS) access layer. It can be accessed either through the application services layer or on event created by the user services layer.

The major benefits of three-tier architecture are reusability, flexibility, manageability, maintainability, and scalability. Yet the disadvantages of it is using one database for storing all the data of the system. Database uses poll (that is, an automatic page refresh is periodically requested) technology which requires a new request from the server every and each time the client needs to update the data, besides central database is not very reliable for a large and huge data. Also the three-tier architecture lacks the feature of delegation between components and the asynchronous communication between them also.

All the previous researches and trails used that architecture. This paper presents a trial to overcome the previous mentioned disadvantages and provide a more reliable consistent architecture.

IV. Economic Load Dispatch

The web-based online monitoring and control provided systems were all solving problems regarding the distribution of electrical power energy, power quality monitoring and energy management [14-16].

The proposed system is focusing on economic load dispatch monitoring and control problem with an architecture system that uses more reliable techniques and open source technologies in a trial to apply these new concepts to improve the process of web-based online monitoring and control for the economic load dispatch.

Economic operation is very important for a power system to return a profit on the capital invested. Rates fixed by regulatory bodies and the importance of conservation of fuel place pressure on power companies to achieve maximum possible efficiency. Maximum efficiency minimizes the cost of a KWh to the consumer and the cost to the company of delivering that KWh in the face of constantly rising prices for fuel, labor, supplies, and maintenance.

Operational economics involving power generation and delivery can be subdivided into two parts-one dealing with minimum cost of power production called economic dispatch and the other dealing with minimum-loss delivery of the generated power to the loads. For any specified load condition economic dispatch determines the power output of each plant (and each generating unit within the plant), which will minimize the overall cost of fuel, needed to serve the system load. Thus, economic dispatch focuses upon coordinating the production costs at all power plants operating on the system. The economic load dispatch (ELD) is an important real time problem, in properly allocating the real power demand among the online generating units. Traditional optimization techniques [17] such as the lambda iteration method, gradient method, the linear programming method and Newton's method are used to solve the ELD problem with monotonically increasing cost function. But they are highly sensitive to starting points and often converge to local optimum or diverge altogether. In reality, the input-output characteristic of generating units are non-linear due to valve-point loading and more advanced algorithms are worth developing to obtain accurate dispatch results.

The prime objective of the ELD problem is to minimize the total generation cost in power system (with an aim to deliver power to the end user at minimal cost) for a given load demand with due regard to the system equality and inequality constraints [18]. To date, various investigations on ELD problems have been undertaken, as better solutions would result in more saving in the operating cost.

a. Fuel Cost Function

The components of the cost that fall under the category of dispatching procedures are the costs of the fuel burnt in the fossile plant because nuclear plants tend to be operated at constant output levels and hydro plants have essentially no variable operating costs. The total cost of operation includes the fuel cost, costs of labour, supplies and maintenance. The objective function of the economic load dispatch problem is to minimize the cost function.

$$F_i(P_i) = \sum_{i=1}^{NG} (a_i P_i^2 + b_i P_i + c_i)$$

F_i = fuel cost of generator i. ($/hr)
P_i = power output of generator i (MW)
a_i = measure of losses in the system. ($/(MW)²hr)
b_i = represents the fuel cost. ($/MWhr)
c_i = includes salary and wages, interests and depreciation and is independent of generation ($/hr)
NG = number of generation buses

b. System Constraints

The solution of economic load dispatch problem is subjected to some constraints such as the system constraints, generating units constraints, transformers constraints and transmission line constraints. The most applied constraints in the problem are:

1. Generator constraints
2. Voltage constraints
3. Running space capacity constraints
4. Transmission line constraints
5. Transformers tap settings constraints
6. Network security constraints

V. PROPOSED ARCHITECTURE

The proposed architecture system aims to apply the concepts of "web-based" online monitoring and control for an electrical power system. The system is well designed to totally eliminate the localization and

The classic Web paradigm (known as "pull") is synchronous: it has the client (browser) solicit data from the server in a synchronous manner. This means that every time the client needs a data update, it has to ask the server expressly to find out if the data has changed and obtain the new value. In other words, for every request from a client there is a corresponding reply from a server. When a Web page is visualized, the data contained within it is static on the user's browser and is not updated until a page refresh is made (manual or automatic). To fix these problems at the source requires a change in paradigm, from synchronous to asynchronous. To guarantee a very low latency between the generation of fresh data and its presentation to the end user within a common browser, a dedicated solution is necessary, namely Push Technology. In the push (or streaming) model, the client receives updates in an asynchronous manner at the server's discretion, in the form of a continuous data flow. The client becomes a passive part of the system, receiving updated information as soon as it is available on the server, without having to ask for it periodically. In this sense, e-mail could be considered the oldest and most widespread form of push technology on the Internet, see Fig. 3.

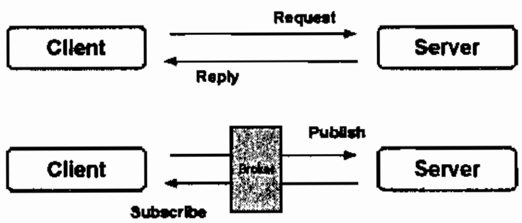


Fig. 3 Pull vs. Push

The general architecture of the AMQP model shown in Fig.4 mainly consists of a middleware server that is a data server that accepts messages and does two main things with them, it routes them to different consumers depending on arbitrary criteria, and it buffers them in memory or on disk when consumers are not able to accept them fast enough. The AMQ Protocol Model takes the approach of smaller, modular pieces that can be combined in more diverse and robust ways. It starts by dividing these tasks into two distinct roles:

- The exchange, which accepts messages from producers and routes them message queues
- The message queue, which stores messages and forwards them to consumer applications

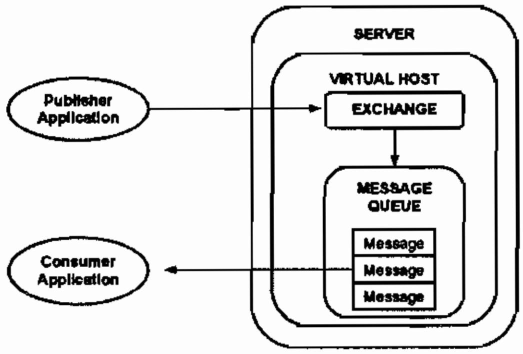


Fig. 4 AMQP General Model Architecture

A message queue stores messages in memory or on disk, and delivers these in sequence to one or more consumer applications. Message queues are message storage and distribution entities. Each message queue is entirely independent and is a reasonably clever object.

- The message can be in any defined format such as Extensible Markup Language (XML)

```
<message>
       <timestamp> 1307378039 </timestamp>
       <load> 750 </load>
</message>
```

or JavaScript Object Notation (JSON)

```
" {
    '1307378039': 750,
}
"
```

A message curries the required information to be transferred.

c. Service Back Ground (ELD component)

The constructed binding in this proposed system is the shared queue binding, the shared queue here is storing the load demand in MW for each defined period of time and the nomination system data (max, min power, cost coefficients of the generating units), and it is being shared with each consumer by turn, the consumer is represented by the service background (Economic Load Dispatch) component which by turn reads the published load and perform the load flow calculations first to get the Beta-loss matrix and input it to the load dispatch calculations using the lambda iteration method according to the data (number of units, max. and min. power for each unit, cost coefficients ...etc.) transmitted from the sites (generating units). The load dispatch daemon is a software component, which is written and coded using Python programming language.

Python [19] isn't a mathematical or simulation tool as Matlab. It is a full functionality programming language that is very easy to perform all the mathematical operations and calculations for any problem using NumPy library. NumPy is the fundamental package needed for scientific computing with Python, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined. This allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

The advantages of Python as a programming language are stated as:

1. Readability
2. Simple to get support
3. Fast to code
4. Reusability
5. Portability
6. Object Oriented Programming
7. Open source programming language
8. Web-based programming language

This software component (Load Dispatch deamon) is completely responsible of doing the calculations of the ELD problem, for the generating units in the power station serving the load demand transferred through DAS.

The load dispatch component is divided among many nodes, represented by white boxes machines (traditional personal computers), in order to divide the operation between those several points, and to overcome the problem of centralization, by making sure that if a single point fails, there are other ones working and will replace this failure immediately. In this case there will be no single-point of failure, we are guaranteeing the consistency of the calculations operation.

The loosely coupled components in the architecture design, makes it easy to replace each component without affecting any other ones in the whole system. According to this feature, the ELD component can be easily replaced by any other software methodology for solving the load dispatch problem e.g. Swarm optimization method, Fuzzy Logic, Neural Networks ....etc

d. Logging System

The forth component of the proposed architecture is the logging system component. After acquiring the required demand load data from the electrical network, being processes through the AMQP to the economic load dispatch calculation component, the results obtained are to be stored in a Database for further processes.

For the small-scale system –practically applied– the used database will be a traditional relational database (SQL database). A relational database is a collection of data items organized as a set of formally-described tables from which data can be accessed or reassembled in many different ways without having to reorganize the database tables. A relational database is a set of tables containing data fitted into predefined categories. Each table contains one or more data categories in columns. Each row contains a unique instance of data for the categories defined by the columns. For example, a typical business order entry database would include a table that described a customer with columns for name, address, phone number, and so forth. Another table would describe an order: product, customer, date, sales price, and so forth.

On the other hand, for large-scale system (i.e. a highly frequency reading for the data, and/or a long-term calculations) a non-relational (NoSql) database will be used.

For such a large data, the statistics and analytics engine will be responsible for creating such a reliable statistics of the system, e.g. the load history for the past years, the performance and reliability of the generating units of the network.... etc.

Distributed and Scalable Non-Relational Database Systems, Non-SQL oriented distributed databases are designed to scale from day one and offer reliability in the face of failures. In the proposed work, MONGO DB is used for logging the system data.

e. User Interface

The viewing component of the architecture is the web user interface. Accessing it using any web-browser (Firefox, Google Chrome, Safari or Internet Explorer)

The python application (service background) was applied to a 6-Bus system (3 generators) and a 14-bus system (5 generators), while the user interface (section d) was applied to the 6-Bus system only.

The web framework used for the proposed work is django (pronounced JANG-goh) an open source high-level Python Web framework that encourages rapid development and clean, pragmatic design. The system representation image is drawn in SVG format, and embedded in HTML5 tags, styled using Cascading Style Sheets (CSS3) and data is rendered using jQuey (a javaScript library)

The default view of the pages is for the system representation with its latest condition (latest load, generated power and change in the power sent to the dispatcher unit), as shown in Fig. 5 and When a new load is demanded, a notification bar appears for an update with the demanded load.

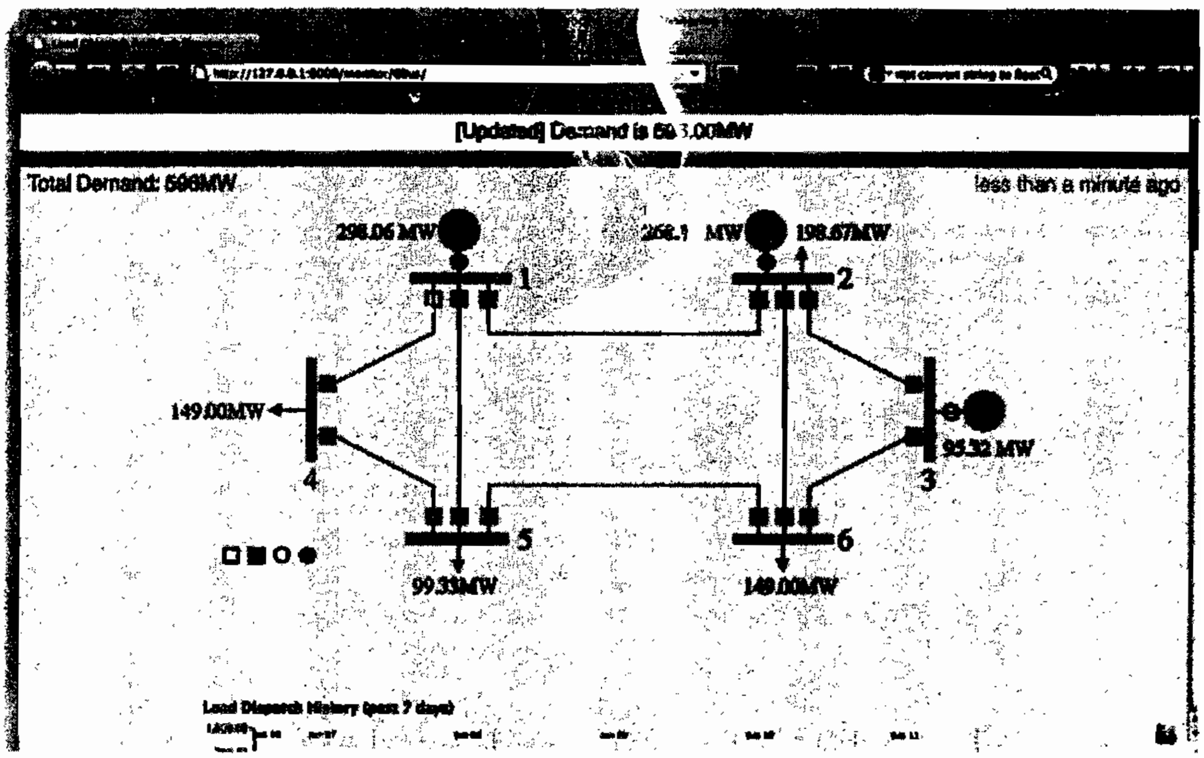


Fig. 5 Home Page Preview with an Update of the Load

By hovering any of the generating units, a tooltip will show-up. A tooltip is a box that appears when you hover your cursor over an element like a hyperlink or the generating unit in the project. It provides supplementary information about it; information displayed is as follows:

- Alpha (c_i), Beta (b_i) and Gamma (a_i) of the generating unit
- Maximum and Minimum power of the unit.
- The state of the generating unit (On/Off)
- The generated power by the unit.
- The required (dispatched) power.
- The reserve power from the generating unit.
- The change in the power (ΔP) with an Up Green arrow for an increase in power or a Down Red one for a decrease.
- The fuel cost of the generating unit.

And for each generating unit, there's a meter graph representing the generated power from the unit (in Yellow), the reserved power (in Green) and the limit of the maximum power (in Red), as shown in Fig. 6,7.

Alpha: 78 **Beta**: 7.97 **Gamma**: 0.00482
pMax: 200 **pMin**: 50
Generated Power (current): 111.48
Reserved Difference (current): 88.52

Power Difference: -26.98 ↓

Dispatch Power: 84.503
Fuel Cost: 785.9
State: On

Fig 6. Tooltip Information

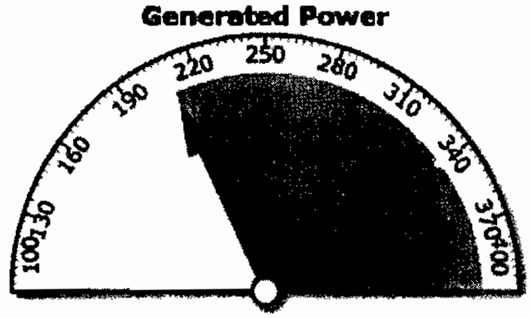


Fig. 7 Generator Meter Graph

The second part of the page view, is the "Load Dispatch History Graph", this is a line graph showing the generated power of each unit versus the time, with the load demand curve for the past seven days, and there is a capability of zooming in (see Fig. 8) for a certain period of time viewing each generating unit power and load demand at this specified period of time.

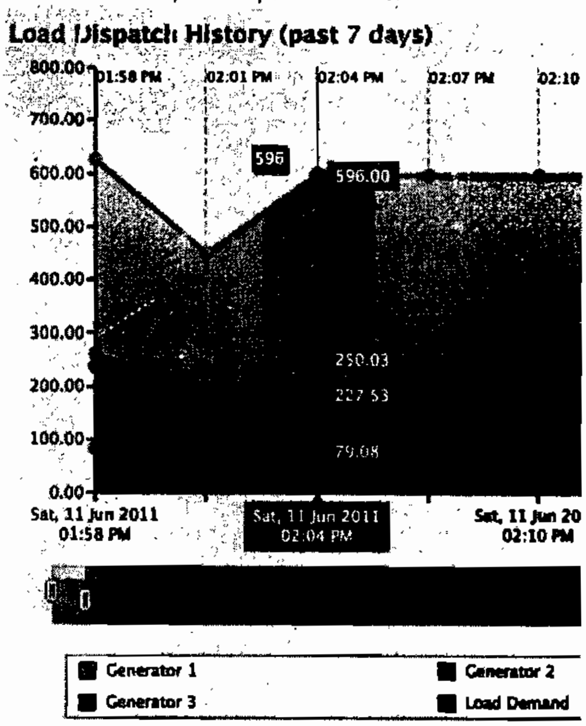


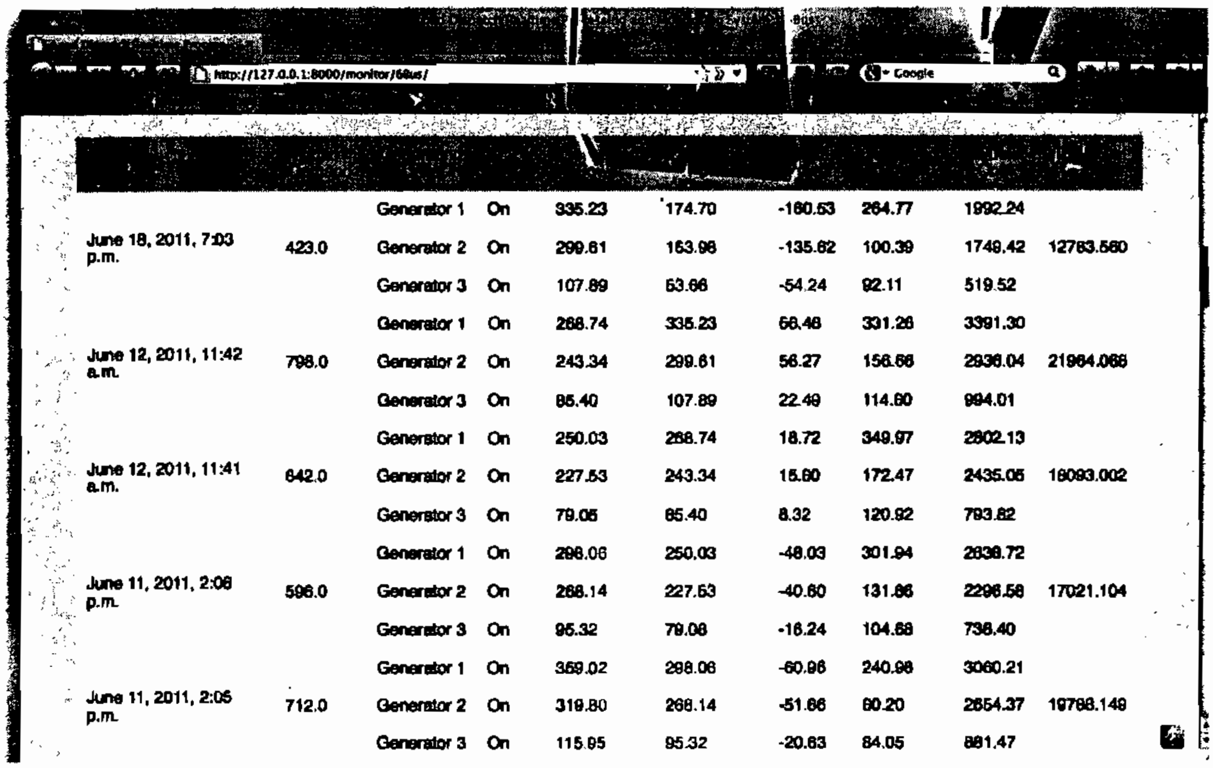
Fig. 8. Line Graph of generated power

A history table is viewed in the main page, for easy monitoring procedure, and the data is clickable in the way that choosing any state can be viewed on the system image preview in the main page Fig. 9.

f. Power Station

The power station is delegated the authority of its internal management and control, the load dispatch system doesn't have to either understand or even know the internals of the power station's management software components. The Power station will have a unified software external interface that communicates with the redundant message bus to retrieve the necessary commands in realtime and that interface will communicate with the internal management system to invoke the necessary actions on the power generation units. The communication between the power station and the load dispatch system is TCP/IP based, thus any network infrastructure can be used to connect the power station to the load dispatch system. High-bandwidth is not needed in the communication infrastructure so any cheap leased line or ADSL line can work to connect the power station to the load dispatch system, thus improving the overall economics and feasibility of implementing this system in production.

The power station includes two components, the first one is "The Command Consumer/Dispatcher" that acts as an AMQP consumer waiting for messages containing commands for the power station. The consumer processes the messages in real-time and will redirect the errors and feedback to the message bus if needed. The other one is "Power Generation Unit Management Interface" the management interface is a generic standard software interface to power generation units that abstracts the underlying hardware control system used for a particular power generation unit.



		Generator 1	On	335.23	174.70	-160.53	264.77	1992.24	
June 18, 2011, 7:03 p.m.	423.0	Generator 2	On	299.61	163.96	-135.62	100.39	1749.42	12763.560
		Generator 3	On	107.89	53.66	-54.24	92.11	519.52	
		Generator 1	On	268.74	335.23	66.48	331.26	3391.30	
June 12, 2011, 11:42 a.m.	798.0	Generator 2	On	243.34	299.61	56.27	156.66	2936.04	21964.066
		Generator 3	On	85.40	107.89	22.49	114.60	994.01	
		Generator 1	On	250.03	268.74	18.72	349.97	2802.13	
June 12, 2011, 11:41 a.m.	642.0	Generator 2	On	227.53	243.34	15.80	172.47	2435.05	18093.002
		Generator 3	On	79.05	85.40	8.32	120.92	793.82	
		Generator 1	On	298.06	250.03	-48.03	301.94	2836.72	
June 11, 2011, 2:06 p.m.	596.0	Generator 2	On	268.14	227.53	-40.60	131.86	2296.58	17021.104
		Generator 3	On	95.32	79.08	-16.24	104.68	736.40	
		Generator 1	On	359.02	298.06	-60.96	240.98	3060.21	
June 11, 2011, 2:05 p.m.	712.0	Generator 2	On	319.80	268.14	-51.66	60.20	2654.37	19768.149
		Generator 3	On	115.95	95.32	-20.63	84.05	881.47	

Fig. 9 History Table Preview

The management unit also takes care of reporting failures or any kind of outage (circuit breaker status) in the power generation unit to the load dispatch system through the command consumer/dispatcher component. The dispatcher will then enqueue a failure message to the message bus so that the load dispatch calculator handle the situation correctly and prevent sending signals to this unit in the future until further notice from the operator.

VI. System Features and Goals

The proposed architecture is designed in order to overcome most of the disadvantages of traditional online monitoring and control systems and to provide a better, more reliable strategy to monitor and control electrical power system online.

The architecture features, are subjected to provide a highly available, scalable and loosely coupled components for obtaining a reliable online system with no independences.

The main goals and objectives of this design are:
1. Asynchronous communication between components.
2. Highly available system.
3. Design to scale.
4. Auto detection of state and configuration.
5. Lazy distribution analysis of recorded events.
6. Multi-site action delegation.

VII. CONCLUSION

The proposed system aimed to apply the capability of monitoring and controlling through a web-based application of the ELD problem and avoiding the localization and centralization of the controlling units.

Monitoring the electrical power system through a graphical user interface on the web makes it easier to access data from any place, as the user interface also provides all the required information and history logs of the system. Moving from the traditional programming languages and data storing technologies to a more open sourced and efficient ones was achieved in the proposed work through using Python programming language over Matlab, and by using the Advanced Messaging Queuing System (AMQP) over traditional relational databases. Scalability, delegation and asynchronous communication between components and the high availability and reliability were presented as the main features and advantages of the proposed system.

The rapid progress of web-based applications enables the contributions of applying the technologies to the electrical power system for monitoring and controlling starting from the smallest equipment to the largest complicated networking system.

REFERENCES

[1] B. Qui and H. B. Gooi, "Web based SCADA display system for access via Internet," IEEE Trans. on Power Systems, Vol. 15, No. 2, May 2000.

[2] Chung-Ping Young, Wei-Lun Juang, and Michael J. Devaney, "Real-Time Intranet-Controlled Virtual Instrument Multiple-Circuit Power Monitoring " IEEE, VOL. 49, NO. 3, JUNE 2000

[3] A. Ferrero and V. Piuri, "A simulation tool for virtual laboratory experiments in a WWW environment," IEEE Trans. Instrum. Meas., vol. 47, June 1998.

[4] R. Giannetti, "An Internet interactive client/server analog oscilloscope emulator," Proc. IEEE IMTC'97, May 1997.

[5] M. Bertocco, F. Ferraris, C. Offelli, and M. Parvis, "A client-server architecture for distributed measurement system," IEEE Trans. Instrum. Meas., vol. 47, Oct. 1998.

[6] D. Koslur, IP Multicasting. New York: Wiley, 1998.

[7] G. Fortino, D. Grimaldi, and L. Nigro, "Multicast control of mobile measurement systems," IEEE Trans. Instrum. Meas., Vol. 47, Oct. 1998.

[8] Tuomo Lindh, Jero Ahola, Jarmo Partanen. Web-Based Monitoring of Electrical Systems in Industrial Plants, International Toshiba Web Conference 2001

[9] Malcolm Irving and Gareth Taylor , 2003 "Prospects for Grid-Computing in Future Power Network's" Discussion Paper circulated by Brunel Institute of Power Systems 2003

[10] M. Di Santo, N. Ranaldo, D.Villacci and E. Zimeo , " Performing Security Analysis of Large Scale Power Systems with a Broker- based Computational Grid "IEEE Proceedings of ITCC2004, 2,77-82, 2004

[11] R. Ramesh and V.Ramachandran. "Web Service Model for On-Line Load Flow Monitoring of Multi-Area Power Systems." Engineering Letters, 13:3, EL_13_3_24 November 2006

[12] J. Gosling, B. Joy, G. Steele, and G. Bracha, "The Java Language Specification", book, Sun Microsystems, Inc., 2005

[13] D. Flanagan, "JavaScript: The Definitive Guide", 5th Edition O'Reilly Media, 20[illegible]

[14] G. A. Taylor, M. R. Irving Member, IEEE, P. R. Hobson, C. Huang, P. Kyberd and R. J. Taylor, "Distributed Monitoring and Control of Future Power Systems via Grid Computing," IEEE, 2006

[15] Ringo P. K. Lee, L. L. Lai, Norman Tse, "A Web- Based Multi-Channel Power Quality Monitoring System for a Large Network," Power System Management and Control, 17-19 April 2002 Conference Publication No. 488 0 IEE 2002

[16] D .Y. Raghavendra Nagesh, Sowjanya A and Dr. S.S. "Real Time Decision Support for Energy Management" TulasiRam Proceedings of the World Congress on Engineering 2008 Vol.I WCE 2008, July 2 - 4, 2008, London, U.K.

[17] S.Hemamalini and Sishaj P. Simon. "Economic Load Dispatch With Value Point Effect Using Artificial Bee Colony Algorithm". NATIONAL SYSTEMS CONFERENCE, NSC 2008,

[18] B. Shaw , S. Ghoshal, V. Mukherjee, and S. P. Ghoshal "Solution of Economic Load Dispatch Problems by a Novel Seeker Optimization Algorithm" International Journal on Electrical Engineering and Informatics - Volume 3, Number 1, 2011

[19] Magnus Lie Hetland, "Beginning Python From Novice to Professional," Second Edition, ISBN-10: 9781590595190, Apress, 2008