Grid Service Model for Real Time Economic Load Dispatch

R. Ramesh and V. Ramachandran

Abstract—The main objective of this paper is to develop a distributed model using grid environment through which the economic load dispatch solutions of multi-area power systems can be obtained continuously. Grid computing is a viable solution in order to exploit the enormous amount of computing power available across Internet to solve large interconnected power system problems. A grid service model is proposed for economic load dispatch, which provides the solutions at specific intervals of time. The proposed model is designed in such a way that any node in the grid can provide the economic load dispatch (ELD) solution. The node which serves as a server node at a specific instance in the grid can obtain the power system data from other client grid nodes and responds with economic load dispatch solutions. The proposed model is highly distributed and has inherent features of scalability, reliability and also uses all available computing power, hence economic feasibility of analyzing power system problems was taken care implicitly.

Index Terms—Distributed services, grid computing, real time economic load dispatch.

I. Introduction

The power system and control need huge volume of data and hence a new approach is required to enable the power system data to be processed, analyzed and interpreted by different power system clients. The power system economic dispatch problem is dealt with determination of optimal combination of power outputs for all generating units, which minimizes the total fuel cost while satisfying the operational constraints. 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 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. In the proposed Grid Service model for solving power system problems, an attempt is made to create a grid environment for solving economic load dispatch problem in which the grids are clusters of interconnected computers that can collectively provide quicker access to very large power system data.

Manuscript received December 31, 2005; revised August 3, 2006.

R. Ramesh is with the Department of Electrical and Electronics Engineering, College of Engineering, Gunidy, Anna University, Chennai, India (e-mail: rramesh@ annauniv.edu).

V. Ramachandran is with the Department of Computer Science Engineering, Anna University, Chennai, India (e-mail: rama@ annauniv.edu)

Publisher Item Identifier S 1682-0053(07)0454

II. State-of-the-Art

The rapid development of the Internet and distributed computing has opened the door for feasible and cost effective solutions for multi-area power system problems. Nithiyananthan and Ramachandran proposed a mobile agent model for solving on-line economic load dispatch problem to over come the overheads associated with sequential power system economic load dispatch computation [2]. Chen and Lu demonstrated the potential advantages of the Web as the platform for developing and deploying complex power system simulations by using distributed technologies and model-view controller concepts [3]. Since the existing supervisory control And data acquisition systems (SCADA) were developed based on different platforms using different languages, the author Qui et al proposed an Internet based SCADA display system and tried to solve the legacy issues using Java native interface (JNI), which is really a tedious process [4]. Nithiyananthan and Ramachandran developed EJB based component model for distributed load flow monitoring, which incorporates multi-tier architecture that enhances reusability [5]. A complete web based and platform independent power system simulation package with various analyses distributed in a clustered environment has been modeled by Irving *et al*. [6]. Sando *et al.* demonstrated through experimental result that on-line security analysis could be executed in lesser period of time even for large power system [7]. 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 [8].

The existing Web enabled models for solving multi-area power system problems are highly distributed and are operating in a heterogeneous environment, need to have an access to massive dataset and to do complex computations. Even though, there are many architectural models for solving multi-area power systems, the proposed grid service model is more powerful and flexible since it provides standard technology and platform for solving power system problems in heterogeneous environments.

III. Grid Service Model for Power System Operations

Open-standard web technologies, such as XML (eXtensible Mark-up Language) and web Services, offer new opportunities for the construction of grid infrastructure

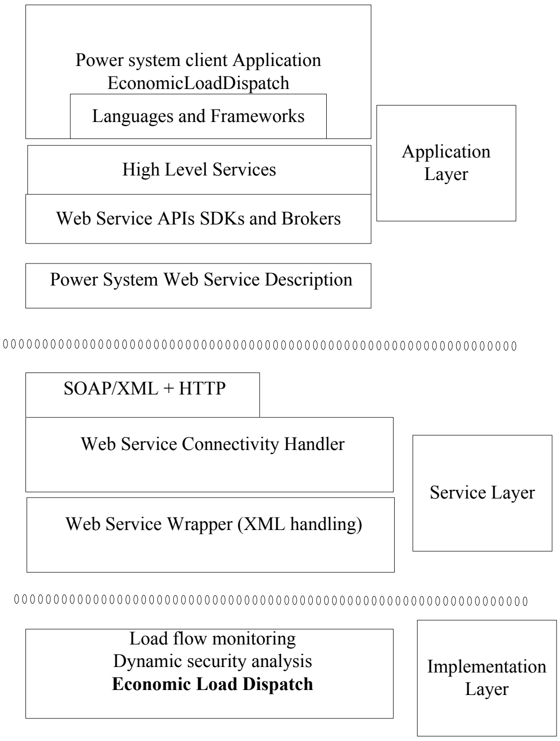

Fig. 1. Layered structure of the Grid Service model for power system operations.

that can be adopted for solving multi-area power system problems. A layered structure of the proposed grid service model for power system operations is shown in Fig. 1.

The problem of economic load dispatch has been taken into consideration to test the proposed model. A grid environment has been set up to carry out the economic load dispatch for different sub-systems of an integrated power system. Each sub system has been considered as a power system client and has been associated with a grid node in which the client need not send the request every time. In the proposed model, the power system data has been represented in XML which is a simple tag-based approach provides a flexible extensible mechanism that can handle the range of digital data from highly structured database records to unstructured. This feature is quite suitable for power system data exchange, since power applications are normally based on various platforms from different vendors and also needs large amount of data to be transferred for monitoring the multi-area power systems in real time. The power system data have been represented in XML and the XMLised power system data is packaged and enveloped using SOAP and sent over HTTP transport. The proposed grid service model exchanges the power system data and response. The proposed model is highly interoperable since the power system data and the operational behaviors are represented in XML and Java respectively, thus making it language and platform independent. The pictorial representation of the grid service model that consists of power system client framework, Web service component and the grid container is given in Fig. 2. The Gird container encapsulates the ELD service.

The power system client has a priori knowledge on the runtime binding information about the economic load dispatch service. Service facilitators have been created through power system web services description (PWSDL) module in order to implement proxies between power

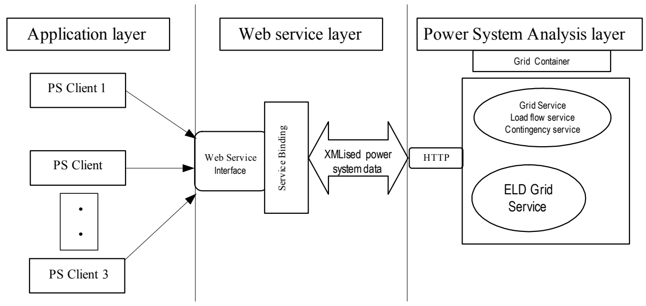

Fig. 2. Client Side framework and container model for ELD service.

power system client and the ELD server. The Web service interface adopts simple object access protocol (SOAP) for transmitting XMLised power system data. The proposed grid service model for economic load dispatch is plugged into the open grid services architecture (OGSA) framework along with the power system grid service description module. The economic load dispatch service has been deployed within the Grid Container.

IV. Economic Load Dispatch Service Description

The proposed grid service implementation is addressable and potentially stateful. An instance of the Economic Load Dispatch object as shown in the application layer of Fig. 1 implements an interface. The grid service model maintains a PortType named as eldPortType through which power system web service is being rendered. eldPortType is an interface for EconomicLoadDispatch object through which the power system client knows about the method signature and invokes the ELD service.

The power system web service description is extracted from the proposed GWSDL economic load dispatch service module [9]. The economic load dispatch service and its bindings are then created. The default behaviors of the open grid service infrastructure (OGSI) are imported to enhance the proposed service. The economic load dispatch grid service port defines an individual endpoint by specifying a single address for the binding. The binding attribute refers the linking rules as defined by power system web service description portType. The binding mechanism and operations are drawn as a class diagram and it is shown in Fig. 3

The power system Web Services Description (PWSDL) defines the remote interface that describes the remote method available in EconomicLoadDispatchService through which the power system client interacts with the ELD service. The PWSDL declares one method, which receives input data as powersystemdata from the power system client and returns a result as ELDsolution representing the economic load dispatch solution of the respective power system client. The PWSDL code for the economic load dispatch grid service is shown in Fig. 4. This module describes that a SOAP request may be sent to EconomicLoadDispatch through the SOAP-HTTP binding.

A. Economic Load Dispatch Service Implementation

The ELD grid service is implemented and hosted in the grid service container. The ELD service also extends from the default grid service interface and is imported from OGSI that facilitates the creation of proxy for the

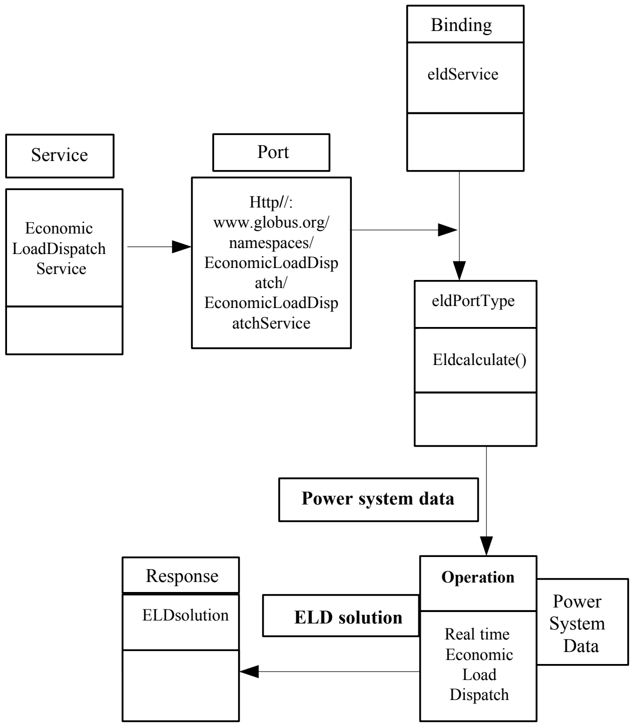

Fig. 3. Class diagram of ELD service.

```
<? xml version =”1.0” encoding=”UTF-8”?>
<types>
<xsd:schematargetNamespace=
http://www.globus.org/namespaces/
EconomicLoadDispatch/
EconomicLoadDispatchService
<xsd:element name=”Request”>
        <xsd:complextype>
                  <xsd:sequence>

<xsd:elementname=”powersystemdata”
type=”xsd:string”/>
                   </xsd:sequence>
         </xsd:complexType>
</xsd:element>
<xsd:element name=”Response”>
        <xsd:complextype>
                  <xsd:sequence>
                           <xsd:element
name=”ELDsolution” type=”xsd:string”/>
                   </xsd:sequence>
         </xsd:complexType>
</xsd:element>
</xsd:schema>
</types>
```

Fig. 4. Sample GWSDL code for the economic load dispatch grid service.

conversion of grid service handle (GSH) to a grid service reference (GSR). GSR describes how a power system client can communicate with ELD grid-service instance. It uses the SOAP binding for invoking the economic load dispatch service implementation. A data virtualization service has been utilized to register the power system component using the built-in naming context and to identify the location of the server grid node in order to obtain the economic load dispatch service.

```
<? xml version=”1.0”?>
<service name=”economic load dispatch /Economic
LoadDispatchServiceFactory” provider =”
        Handler” style=”wrapped”>
<parameter name=”name”
value=”EconomicLoadDispatchServiceFactory”/>
<parameter name=”instance”
value=”EconomicLoadDispatchServiceInstance/>
<parameter name=”instance-schemaPath”
 value = ”schema /  EconomicLoaddispatch
/EconomicLoadDispatchService/EconomicLoad
Dispatch_service.wsdl”/>
<parameter name=”instance-baseclassName”
 value=  ”org.globus.EconomicLoadDispatch.services.
impl.EconomicLoadDispatch.Impl”/>
<parameter name=”inatance-className”
value=  ”org.globus.EconomicLoadDispatch. stubs.
EconomicLoadDispatchService.EconomicLoadDispatch
PortType”/>
```

Fig. 5. WSDD for economic load dispatch service deployment.

B. Economic load dispatch service parameters description

The various parameters needed for the economic load dispatch service deployment are described using Web Service Deployment Descriptor (WSDD).

EconomicLoadDispatchServiceFactory is a service factory for economic load dispatch which is created within the grid service container and used to create instances of a ELD service. Each ELD grid service instance (EconomicLoadDispatchServiceInstance) has a unique identity with respect to the other instances in the container. The parameters used for economic load dispatch grid service deployment in the container are defined in WSDD as shown in Fig. 5.

After deploying the economic load dispatch grid service, it is possible to use the service browser to view all the grid services deployed in the container and also to view the power system web service description documents in the grid service registry. OGSA enables power system clients to create transient services as well as to discover and evaluate the properties of available economic load dispatch services.

C. Economic load dispatch service stubs creation

The stub, which is the client side proxy of the economic load dispatch service remote object has been built and it is added to the Grid Archive (GAR) file. The GAR file contains all the stubs necessary for the economic load dispatch service implementation. The economic load dispatch service can be deployed into the grid service container using this GAR file.

V. Power System Client Application

The power system client application sends "create economic load dispatch grid service" requests on the economic load dispatch factory interface to create a new service instance. The newly created ELD service instance will be automatically assigned a globally unique name called GSH, which is used to distinguish this specific service instance from other grid service instances. In

TABLE I
SAMPLE 20-BUS BUS POWER SYSTEM DATA IN XML FORMAT


```
<? xml version =”1.0” encoding=”UTF-8”?>
<no_genbuses> 20</no_gen.buses>
<power_MW>2500.0</power_MW>
<abc_values>
  <!--- a,b,c parameters---- >
   <busno>
      1
      <a>0.00068></a>
      <b>18.19></b>
      <c>1000></c>
    </busno>
        .
        .
        .
    <busno>
      20
      <a>0.0073></a>
      <b>19.79></b>
      <c>850.0></c>
    </busno>
</abc_values>
        .
```


addition to using a GSH, economic load dispatch grid service instances are made accessible to client applications via the use of a GSR. The power system client uses the GSR to send request directly to the specific instance and hence the economic load dispatch solution are obtained as response.

VI. THE ELD SERVICE - ROCEDURE

The steps involved in the execution of the economic load dispatch grid service are detailed as follows:

- Start the grid service container.
- Start the power system client (Any number of power system clients may be considered).
- Power system client creates a new instance of EconomicLoadDispatchGridService from the central of EconomicLoadDispatchServiceFactory.
- Grid server node uses the power system client’s reference to receive the power system data from the client and it carry outs the economic load dispatch.
- Power system client obtains the result and provides a view of the results through an applet.
- For every specific interval of time, the server grid node automatically receives power system data from the client.

In the coming years the power system will have large number of small generators rather than small number of large generators [6]. It will be necessary to monitor their generation at distribution level. The proposed grid service model can provide the information about the total production of power from the large number of generators and also can help in its scheduling in remote areas. It is necessary to monitor them in real time situation because a small disturbance in the system can result in black out. The proposed grid service model can also be helpful in monitoring the generation status for scheduling the maintenance.

VII. POWER SYSTEM CLIENT APPLICATION

The proposed Grid service model makes use of globus [9] toolkit for enabling Grid Architecture and operates in Linux environment. The supporting packages needed are Java SDK 1.4.2 and ANT 1.6.1 [10]. The grid environment provides the basis for an open source OGSA implementation and supports existing APIs as well as PWSDL interfaces.

VIII. TEST CASES

The economic load dispatch was carried out using the proposed grid model on a 3-bus system, 5-bus system and 20-bus system. The sample 20-bus system data in XML format are given in Table I.

Any number of power system clients can be served without limit. When the server receives data from the power system client, it runs the economic load dispatch solution automatically and the result is sent back to the respective power system client. Using this model, different power system clients can obtain economic load dispatch solutions continuously at regular time intervals.

IX. CONCLUSION

An efficient grid service model has been developed to carry out the economic load dispatch of multi-area power systems. The grid environment provides excellent scalability, high security and has a capacity to meet the huge computation requirement, which is suitable to carry out economic load dispatch for large interconnected multi area power systems. The proposed model can be suitably implemented for a large power system network spread over a large geographical area.

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, pp. 681-686, May 2000.

[2] K. Nithiyanatan and V. Ramachandran, "Distributed mobile agent model for multi-area power system on-line economic load dispatch," *J. of Electrical Engineering*, vol. 54, no. 11-12, pp. 316-319, 2003.

[3] S. Chen and F. Y. Liu, "Web Based simulations of power systems," *IEEE Trans. on Computer Applications in Power*, vol. 15, no. 1, pp. 35-40, Jan 2002.

[4] B. Qiu and H.B. Gooi, "Internet based SCADA display system," *IEEE Trans. on Computer Applications in Po*wer, vol. 15, no. 1, pp. 14-19, Jan 2002.

[5] K. Nithiyananthan and V. Ramachandran, "EJB based component model for distributed multi-area power systems load flow monitoring," *Int. J. for Engineering Modeling*, vol. 15, no. 1-4, 2002. http://www.gradst.hr/engmod/2002/index.html

[6] M. Irving, G. Taylor, and P. Hobson, "Plug in to grid computing," *IEEE Trans. on Power and Energy Magazine*, vol. 2, no. 2, pp. 40-44, Mar./Apr. 2004.

[7] M. Di Santo, N. Ranaldo, D.Villacci, and E. Zimeo, "Performing security analysis of large scale power systems with a broker-based computational grid," in *Proc. of IEEE ITCC2004*, vol. 2, pp.77-82, Apr. 2004.

[8] M. Irving and G. Taylor, "Prospects for grid-computing in future power networks," *Discussion Paper Circulated by Brunel Institute of Power Systems*, Sept. 2003.

[9] http://www.globus.org

[10] http://www.Java.sun.com

R. Ramesh received his M.E. degree in Power system engineering form the Faculty of Engineering and Technology, Annamalai University, Chidambaram, India in 2001. He is currently working as a lecturer in department of Electrical and Electronics Engineering, College of Engineering, Gunidy, Anna University, Chennai, India. His research interests include power system analysis and modeling, grid computing and Internet technologies.

V. Ramachandran received his M.E. degree and Ph.D. in Electrical Engineering form College of Engineering, Gunidy, Anna University, Chennai, India. He is currently working as a Professor of Computer Science and Engineering and Director of Ramanujan Computing Centre, Anna University, Chennai. His research interests include power systems reliability engineering, network security, soft computing and Web technology.