CHAPTER 1

INTRODUCTION

1.1 GENERAL

Computer applications used in power systems analysis have undergone profound changes since the last decade. The interconnected bulk electric power systems are becoming integrated with vast networked information systems. The existing power system operations are primarily desktop applications with a small number of exceptions implemented in parallel processing super computers. Many of today’s desktop computing applications are being distributed to take advantage of the shared resources and information available in a clustered environment. Traditionally on-line power system applications are, for the most part self-contained monolithic programs that have a limited access to one another’s procedure and data. They are usually cumbersome to build and expensive to maintain because simple changes in the power system operation logic may cause the entire application to be rewritten, recompiled and retested.

The present conventional client-server architecture for power system analysis is complicated, memory management is difficult, source code is bulky, and exception-handling mechanism is not so easy. In the conventional power system operation and control, it is assumed that the information required for the monitoring and controlling of power systems is centrally available and all
computations are to be done sequentially at a single location. With respect to sequential computation, the server has to be loaded every time for each client's request and the time taken to deliver the solution is also relatively high. An attempt has been made in this thesis to create a complete platform-independent analysis of power systems in a distributed environment.

As the complexity and programming burden of today's software system increases due to the expectancy of distributed capabilities and seamless networked operations, middleware computing is becoming an increasingly popular way of providing helpful and useful software support abstractions in a coherent manner. Currently adequate support for power system analysis within a distributed environment remains to be provided. Successful implementation of a power system application in a distributed environment is itself a formidable task as distributed environment has combinatorial explosion problem. This problem is addressed by the encapsulation of communication mechanism between the client and server technologies such as Remote Method Invocation (RMI) and .Net Remoting. These technologies allow a collection of distributed components to interact with each other in such a way as if they are running in the same machine. Efforts to utilize distributed components are more concerned with functionality addition than with critically examining the performance. This thesis proposes novel ideas for the development of power system applications in distributed environment using open standards and facilitates the performance evaluation of power system applications in distributed environment.
1.2 STATE OF THE ART

A good deal of literature is available pertaining to on-line power system analysis through conventional client/server architecture. In spite of the good amount of publications in this area, it was felt that there is a need for research enhancement and the development of effective architectures to make the power system analysis in a complete platform independent environment. The theoretical literature on this subject is now becoming overwhelming; yet applications in power engineering have only been evident in the last few years.

Distributed object computing has emerged and gained acceptance for the development of a wide variety of applications. Networked computing has spawned the area of distributed computing, which has enabled the realistic investigation of component oriented computing as a promising approach to distributed application development. However there is little support for power systems within either distributed computing platform or component systems addressing issues related to power system applications.

The Power System Analytical Data Task Force (Miller et al 2002) had developed a data dictionary containing definitions of power system analytical data. The task force proposed a protocol for exchanging power system information contained in a database. Even though the protocol provides a standard way of representing power system data, it does not provide a solution to the legacy issues associated with on-line power system analysis in a distributed environment.

Over a decade, the power system researchers concentrate on web based Supervisory Control and Data Acquisition (SCADA) system to control
power system operations. Contributions had been made (Qiu and Gooi 2000) to make SCADA system web enabled, which gathers incoming power system data for further processing by a number of distributed processes. Technological advances in Internet and WWW had made it possible to develop a low cost WWW display system for accessing information via tabular displays and one-line diagrams on the net. Qiu and Gooi had developed a real-time power system simulator, which is Java based three-tier client/server application that can run across different platforms without any modification. An Internet based SCADA display system had been proposed for one-line diagram of auto generation (Bin qiu et al 2002) using Very Large Scale Integration (VLSI) placement and routing algorithms. Since the existing SCADA systems were developed based on different platforms using different languages, the authors tried to solve the legacy issues using Java Native Interface (JNI) which is really a tedious process.

Web based conception has been elaborated and implemented by researchers all over the world since it was first introduced in the early 90s. Their (Gall and Hauck, Nilsen 1996) experience has indicated that Java can be successfully employed in the real-time system. The chairperson of the Global Internet Project (John Patrick 2000) said, “the Internet will change everything - the way technology is created, deployed and the way we use it”. It is foreseeable that many services and applications including those of power system simulations will be implemented on the Internet and hence the existing power system simulation routines that have been already implemented on multiple platforms can share many common facilities and resources.
It has been proved that the performance of power system studies can be greatly enhanced by the use of Internet and E-business techniques (Sutherland and Gerald Irvine 2001). The Internet based power system data transfer is the critical factor as the power system applications are running in different platforms and the applications were developed using different programming paradigms. In order to exploit the full capabilities of Internet in solving the power system problems, it is recommended that a standard power system database has to be developed or an XMLised data conversion model is required that will solve the legacy issues associated with power system applications running in a distributed environment.

A complete web-based, platform independent, power system simulation package with various analyses distributed in a clustered environment had been developed by the researchers (Chen and Lu 2002) of Nanyang Technological University, Singapore. This contribution is achieved by leveraging heavily on the advanced distributed computing technologies and by using the model-view-controller (MVC) concept. They demonstrated the potential and advantages of embracing the Web as the platform for developing and deploying complex power system simulations by using the distributed technologies and MVC design concept.

An enhanced prototype model for Web browser based power distribution management system had been developed (Pedro Silva, Joao Tome Saraiva and Alexandre Sousa 2000) using object-oriented technology and a modular distributed multitask client/server architecture, which enables the user to connect to an Hyper Text Markup Language (HTML) page in the server, download the desired applet to the client’s machine establishing
communication, interfaces and requesting services to the main system. The authors had motivated to develop different architectures for providing automated solutions for power distribution problems.

A standardized framework for the performance assessment of power system applications in distributed environment is not yet available. A considerable work on the performance analysis of distributed object middleware was carried out (Matjaz et al 1999, Matjaz et al 2000) and an optimized thread management in RMI-IIOP that improves the performance.

Mobile agent technology has recently attracted attention in the field of distributed application system development. An agent is powerful in its ability to perform processing in a distributed network environment. Research work is focussed on the application of mobile agents in power system studies over a period of year or two. A real time mobile agent platform had been proposed (Takaya et al 2002) for power system protection and control system for which real time operation and advanced reliability are required. The authors proved that the agent technology is an effective means for carrying out cooperation among many protection and control devices through a network, for utilizing the abundant information from a power system stored in the protection and control equipments. Nagata (2002) had presented a multi-agent approach to power system restoration problem.

The rapid developments of the Internet and distributed computing have opened the door for feasible and cost-effective solutions for power system problems. The RMI Application Programming Interface (API), as representative of the distributed object paradigm, is an efficient tool for
building network applications. RMI can be used in lieu of Socket API to build a network application rapidly. RMI API provides the abstraction that eases the task of distributed application development. Because of the relative ease with which network applications can be developed using RMI, it is being considered for the rapid development of a prototype for power system applications.

1.3 OBJECTIVES

Keeping the above contributions in the background, the main aim of this research is to develop enhanced models for on-line power system analysis in a distributed and platform independent environment. In this thesis an analysis has been carried out to recognize the requirements for the power system applications in distributed environment and to focus on distributed object computing support for power system applications and its related issues that will alleviate better performance. The major areas of power system studies like load flow monitoring, economic load dispatch and dynamic security analysis are being considered for the implementation of the proposed models. Specifically, the main objectives of this thesis work are detailed as follows:

- To propose enhanced architectures for multi-area power system analysis in a distributed and platform independent environment.

- To propose a power system data conversion model in order to solve the overheads associated with the interaction of legacy power system applications in a distributed environment.
To propose group communication and active networking models in order to automate the entire process of power system analysis and to enhance the scalability of the system.

To develop component models for on-line power system analysis to achieve modularity and reusability in power system operation logic.

To propose an agent model for an on-line economic dispatch to achieve an uninterruptible service to the various subsystems with negligible bandwidth consumption.

To compare the performance of the proposed models with respect to the Round Trip Time (RTT) between the request and the response.

1.4 SCOPE OF THE THESIS

The thesis entitled “On-line Power System Analysis in Distributed Environment” delineates the development of enhanced distributed models for solving power system problems and provides a methodology for solving legacy issues associated with the interaction of power system applications in distributed and platform independent environment. This dissertation is divided into five chapters including the introduction.

In chapter 2, the concept of client/server architecture for solving power system applications and web based simulation of power system analysis are reviewed. The current trends in on-line power system analysis are
discussed in detail. The need for enhancement of on-line power system applications has been analysed. A power system data conversion model is proposed which represents the power system data in Extensible Markup Language (XML) as a flat file in order to solve the overheads associated with the interaction of legacy power system applications in a distributed environment. The standard IEEE load flow data for 3 bus, 9 bus and 13 bus power systems has been converted into XML form and stored in flat files. Later these flat files are serialized as input to remote applications when on-line power system analysis is being carried out in a distributed environment. A Document Type Definition (DTD) has been written to check the validity of the power system data that is referred by the power system clients before sending the data to the server application. Document Object Model (DOM) is implemented in the server side to dynamically access and to parse the power system data in XML form sent by the client.

The design and development of enhanced distributed models for on-line power system analysis in a completely platform independent environment are being dealt within chapter 3. Enhancements to the default RMI protocol are introduced in order to automate the process of power system applications, to reduce the execution time of power system applications in a distributed environment and to improve the scalability of the system. Dynamic class loading and self registry concepts are adopted to exploit a complete distributed environment and to minimize the resource consumption respectively.

The present work proposes an efficient technique that leads to a mechanism of client callback where the power system clients are also treated as
remote objects and are interacting with the remote load flow server object and vice versa. It allows each power system client to register itself with the remote load flow server object so that the server may initiate a remote method call to the power system client objects when the awaited time event occurs. A reference to an object that implements the power system client remote interface is accepted as an argument. It is necessary for the load flow server to employ a data structure to maintain a list of registered client interface references. A power system client computer basically does the distributed load flow monitoring through an applet for every specific period of time and frequently exchanges data with the server. The server does the load flow computation and then distributes the results. The total process has been automated by making the server to get the input data for every specific period of time. In this proposed method, the load flow server creates its own registry and it maintains the stubs for the remote objects by itself and hence the server no longer needs to depend on the bootstrap registry service provided by the RMI protocol. In RMI client-server architecture, clients can communicate with the remote object only when the server side stub is available with the client. The stub has been loaded on the power system client's side dynamically by an external web server. The proposed model has been extended to solve contingency ranking and economic load dispatch problems. Each power system client maintains the load flow data in its own database and hence a heterogeneous distributed database environment has been created. The major advantage of holding the database on the client side is that the database connectivity overhead is brought to the lowest possible.

RMI based group communication model is proposed for on-line power system analysis. In this proposed model each remote power system
client object can automate the entire process of power system analysis by generating the sequence of operations in a command list. The sequence of power system operations is controlled by this command list. Until the command list vector received by the power system client becomes null, power system client will send the request to the respective server based on the first IP address available in the command list. Once the command list vector becomes null, it indicates the completion of on-line power system analysis for that particular power system client.

An RMI based active network model is introduced for on-line power system analysis in a distributed environment and tested with on-line dynamic security analysis. This model overcomes the difficulties associated with the group communication model by making the load flow server and contingency server to communicate each other directly. The active network model has been developed in such a way that the load flow server becomes client remote object whenever there is a load flow request from the contingency server. The model has been developed such that once the power system client initiates the contingency analysis request and sends the data accordingly; it will finally receive the critical contingency list from the contingency server.

This chapter discusses the development of a component model for on-line load flow monitoring. Developed enterprise load flow components are pluggable, reusable and can simplify the complexity in the areas of synchronization, scalability, load flow monitoring integrity, networking and distributed object frameworks. A load flow bean once developed can be deployed on multiple platforms without recompilation or source code modification. In this proposed model, each power system client can access the
remote load flow Enterprise Java Bean (EJB) server through the servlets using data object serialization. The load flow server in turn computes and disseminates load flow solutions to all the power system clients simultaneously for every specific period of time based on the client’s requirement. In this proposed method, load flow monitoring by each client is achieved through an applet - servlet - EJB communication model for every specific period of time. The applet to servlet communication is enabled via HTTP tunneling.

A component model has also been proposed for on-line dynamic security analysis which allows each power system client to access the remote contingency bean. The contingency server computes and disseminates contingency ranking to all the power system clients simultaneously for every specific period of time based on client’s requirement. The load flow and contingency bean components are deployed within an EJB container. The EJB container runs the load flow bean automatically and sends the load flow solution as input to the contingency bean. In turn contingency bean calculates the performance index based on the voltage, real and reactive power. Finally critical contingency list has been sent back to respective power system client and hence the power system clients can monitor critical contingencies at regular time intervals. A component model for on-line economic load dispatch has also been developed and implemented.

The major factor that influences the performance of the on-line power system application in a distributed environment is Round Trip Time. The performance analysis of the proposed distributed models has been carried out with respect to load flow monitoring, dynamic security analysis and economic load dispatch for different power systems and the results are reported.
The variations of round trip time with respect to the number of power system clients are plotted as graph. RMI based model with callback mechanism performs better than all other models. CORBA based model consumes more RTT because every time the power system client has to search the CORBA service provider for the load flow service in the network.

In chapter 4, a remoting architectural model has been presented for on-line economic load dispatch in a distributed environment. The versatility of the proposed work is language independent based on common language runtime environment. The model is very flexible that provides the capability to extend power system application by adding custom components and different versions of economic load dispatch logic components that are serviced without affecting the power system client. The implementation of the proposed model and the steps for running the proposed application are discussed. Remoting provides a powerful and easy way to the power system clients to communicate with remote ELD server at different domains. External configuration files for both client and remote server are being used to organize and to control the deployment of distributed economic load dispatch application in .NET framework. The version and life time management of the remote objects are made possible using configuration file without recompiling the source code pertaining to client and server program.

A mobile agent based on-line economic load dispatch model has also been presented. The mobile agent framework provides a number of advantages including the saving of network bandwidth and increasing the overall performance by allowing the ELD agent to process the power system data in the client machine itself. Mobile agents have the unique ability to transport
themselves from one system in a network to another and operate asynchronously and independently of the process that created them. The proposed model is implemented in Java Agent Development Framework. The agent model has been implemented in such a way that for every specific period of time, the ELD agent migrates from one power system client to another to obtain the system data and the optimized economic load dispatch solutions with power loss have been sent back to the clients in a heterogeneous environment. The proposed architecture makes the economic dispatch application as more fault tolerant where network failure can influence only the migration of an ELD agent as the rest of the process is then done locally on the same node, thus continuous service and monitoring of economic load dispatch solutions are achieved.

In chapter 5, a review of work is presented major conclusions reached and contributions made are reported. Recommendations for further research are stated.