Power System Control Centers: Past, Present, and Future

FELIX F. WU, FELLOW, IEEE, KHOSROW MOSLEHI, MEMBER, IEEE, AND ANJAN BOSE, FELLOW, IEEE

Invited Paper

In this paper, we review the functions and architectures of control centers: their past, present, and likely future. The evolving changes in power system operational needs require a distributed control center that is decentralized, integrated, flexible, and open. Present-day control centers are moving in that direction with varying degrees of success. The technologies employed in today's control centers to enable them to be distributed are briefly reviewed. With the rise of the Internet age, the trend in information and communication technologies is moving toward Grid computing and Web services, or Grid services. A Grid service-based future control center is stipulated.

Keywords*—Computer control of power systems, control center, energy management system, SCADA.*

I. INTRODUCTION

The control center is the central nerve system of the power system. It senses the pulse of the power system, adjusts its condition, coordinates its movement, and provides defense against exogenous events. In this paper, we review the functions and architectures of control centers: their past, present, and likely future.

We first give a brief historical account of the evolution of control centers. A great impetus to the development of control centers occurred after the northeast blackout of 1965 when the commission investigating the incident recommended that "utilities should intensify the pursuit of all opportunities to expand the effective use of computers in power system planning and operation. . .. Control centers should be provided with a means for rapid checks on stable and safe capacity limits of system elements. . .through the use of digital computers." [1] The resulting computer-based control center, called the Energy Management System (EMS), achieved a quantum jump in terms of intelligence and application software capabilities. The requirements for data acquisition devices and systems, the associated communications, and the computational power within the control center were then stretched to the limits of what computer and communication technologies could offer at the time. Special designed devices and proprietary systems had to be developed to fulfill power system application needs. Over the years, information technologies have progressed in leaps and bounds, while control centers, with their nonstandard legacy devices and systems that could not take full advantage of the new technologies, have remained far behind. Recent trends in industry deregulation have fundamentally changed the requirements of the control center and have exposed its weakness. Conventional control centers of the past were, by today's standards, too centralized, independent, inflexible, and closed.

The restructuring of the power industry has transformed its operation from centralized to coordinated decentralized decision-making. The blackouts of 2003 may spur another jump in the applications of modern *information and communication technologies* (ICT) in control centers to benefit reliable and efficient operations of power systems. The ICT world has moved toward distributed intelligent systems with Web services and Grid computing. The idea of Grid computing was motivated by the electric grids of which their resources are shared and consumers are unaware of their origins. The marriage of Grid computing and service-oriented architecture into Grid services offers the ultimate decentralization, integration, flexibility, and openness. We envision a Grid services-based future control center that is characterized by:

- an ultrafast data acquisition system;
- greatly expanded applications;

Manuscript received October 1, 2004; revised June 1, 2005. This work was supported in part by the Research Grant Council of Hong Kong under Grant 7176/03E), in part by National Key Basic-Research Funds of China under Grant 2004CB217900), in part by EPRI Worldwide under Contract P03-60005), and in part by the DOE CERTS Program under Contract DE-A1-99EE35075.

F. F. Wu is with the Center for Electrical Energy Systems, University of Hong Kong, Hong Kong (e-mail: ffwu@eee.hku.hk).

K. Moslehi is with ABB Network Managements, Santa Clara, CA 95050 USA (e-mail: Khosrow.Moslehi@us.abb.com).

A. Bose is with the College of Engineering and Architecture, Washington State University, Pullman, WA 99164-2714 USA (e-mail: bose@wsu.edu).

Digital Object Identifier 10.1109/JPROC.2005.857499

- distributed data acquisition and data processing services;
- distributed control center applications expressed in terms of layers of services;
- partner grids of enterprise grids;
- dynamic sharing of computational resources of all intelligent devices;
- standard Grid services architecture and tools to manage ICT resources.

Control centers today are in the transitional stage from the centralized architecture of yesterday to the distributed architecture of tomorrow. In the last decade or so, communication and computer communities have developed technologies that enable systems to be more decentralized, integrated, flexible, and open. Such technologies include communication network layered protocols, object technologies, middleware, etc. which are briefly reviewed in this paper. Control centers in power systems are gradually moving in the directions of applying these technologies. The trends of present-day control centers are mostly migrating toward distributed control centers that are characterized by:

- Separated supervisory control and data acquisition (SCADA), energy management system (EMS), and business management system (BMS);
- IP-based distributed SCADA;
- common information model (CIM)-compliant data models;
- Middleware-based distributed EMS and BMS applications.

Control centers today, not surprisingly, span a wide range of architectures from the conventional system to the more distributed one described above.

The paper is organized as follows: Section II provides a historical account of control center evolution. Section III presents the functions and architecture of conventional control centers. Section IV describes the challenges imposed by the changing operating environment to control centers. Section V presents a brief tutorial on the enabling distributed technologies that have been applied with varying degrees of success in today's control centers. Section VI describes desirable features of today's distributed control centers. Section VII discusses the emerging technology of Grid services as the future mode of computation. Section VIII presents our vision of future control centers that are Grid services-based, along with their data acquisition systems and expanded functions. Section IX draws a brief conclusion.

II. Control Center Evolution

In the 1950s analog communications were employed to collect real-time data of MW power outputs from power plants and tie-line flows to power companies for operators using analog computers to conduct load frequency control (LFC) and economic dispatch (ED) [2]. Using system frequency as a surrogate measurement of power balance between generation and load within a control area, LFC was used to control generation in order to maintain frequency and interchange schedules between control areas. An ED adjusts power outputs of generators at equal incremental cost to achieve overall optimality of minimum total cost of the system to meet the load demand. Penalty factors were introduced to compensate for transmission losses by the *loss formula*. This was the precursor of the modern control center. When digital computers were introduced in the 1960s, remote terminal units (RTUs) were developed to collect real-time measurements of voltage, real and reactive powers, and status of circuit breakers at transmission substations through dedicated transmission channels to a central computer equipped with the capability to perform necessary calculation for automatic generation control (AGC), which is a combination of LFC and ED. Command signals to remotely raise or lower generation levels and open or close circuit breakers could be issued from the control center. This is called the SCADA system.

After the northeast blackout of 1965, a recommendation was made to apply digital computers more extensively and effectively to improve the real-time operations of the interconnected power systems. The capability of control centers was pushed to a new level in the 1970s with the introduction of the concept of system security, covering both generation and transmission systems [3]. The *security* of a power system is defined as the ability of the system to withstand disturbances or contingencies, such as generator or transmission line outages. Because security is commonly used in the sense of against intrusion, the term *power system reliability* is often used today in place of the traditional *power system security* in order to avoid causing confusion to laymen. The security control system is responsible for monitoring, analysis, and real-time coordination of the generation and the transmission systems. It starts from processing the telemetered real-time measurements from SCADA through a *state estimator* to clean out errors in measurements and communications. Then the output of the state estimator goes through the contingency analysis to answer "what-if" questions. Contingencies are disturbances such as generator failure or transmission line outages that might occur in the system. This is carried out using a steady-state model of the power system, i.e., power flow calculations. Efficient solution algorithms for large nonlinear programming problem known as the *optimal power flow* (OPF) were developed for transmission-constrained economic dispatch, preventive control, and security-constrained ED (SCED). Due to daily and weekly variations in load demands, it is necessary to schedule the startup and shutdown of generators to ensure that there is always adequate generating capacity on-line at minimum total costs. The optimization routine doing such scheduling is called unit commitment (UC). Control centers equipped with state estimation and other network analysis software, called Advanced Application Software, in addition to the generation control software, are called *energy management systems* (EMS) [4].

Early control centers used specialized computers offered by vendors whose business was mainly in the utility industry. Later, general purpose computers, from mainframe to mini, were used to do SCADA, AGC, and security control. In the late 1980s minicomputers were gradually replaced by a set of UNIX workstations or PCs running on an LAN [5]. At

the same time, SCADA systems were installed in substations and distribution feeders. More functions were added step by step to these *distribution management systems* (DMS) as the computational power of PCs increased.

In the second half of the 1990s, a trend began to fundamentally change the electric power industry. This came to be known as industry restructuring or deregulation [6]. Vertically integrated utilities were unbundled; generation and transmission were separated. Regulated monopolies were replaced by competitive generation markets. Transmission, however, remained largely regulated. The principle for the restructuring is the belief that a competitive market is more efficient in overall resource allocation. While suppliers maximize their profits and consumers choose the best pattern of consumption that they can afford, the price in a market will adjust itself to an equilibrium that is optimal for the social welfare.

Two types of markets exist in the restructured power industry. One is the bilateral contracts between suppliers and consumers. The other one is an auction market in which generators submit bids to a centralized agent which determines the winning bids and the price. The price could be determined by a uniform pricing scheme (in the United States) based on the highest bid price that is deployed to serve the load, or a nonuniform pricing scheme (in the U.K.) based on the bid price (pay-as-bid). A market operator is needed to run the auction market. The market operator may be an *independent system operator* (ISO), a *regional transmission organization* (RTO), or other entities with similar functions. With the introduction of electricity markets, some control centers, e.g., that of an ISO or an RTO, have had the responsibility of running the market operation, as well as maintaining system reliability. The two aspects are usually separated but with close coordination.

The electricity market is different from any other commodity market in that power has to be balanced at all times. This requirement leads to a more complex market structure. Most electricity markets have a day-ahead energy market, a real-time balancing market, and an ancillary service market. The day-ahead market is a forward market in which hourly clearing prices are calculated for each hour of the next day based on generation and demand bids, and bilateral transaction schedules. If the reliability of the transmission system imposes limits on most economical generation causing transmission congestion to occur, *congestion management* is required. One of the approaches to congestion management is based on the pricing differences, called *locational marginal prices* (LMP), between the nodes of the network. The LMPs are obtained from the nodal shadow prices of an OPF or SCED. The day-ahead market enables participants to purchase and sell energy at these binding day-ahead LMPs. A *security-constrained unit commitment* (SCUC) is conducted, based on the bids submitted by the generators, to schedule the startup and shutdown of generators in advance. The transmission customers may schedule bilateral transactions at binding day-ahead congestion charges based on the difference in LMPs at the generation and the demand sides. The balancing market is

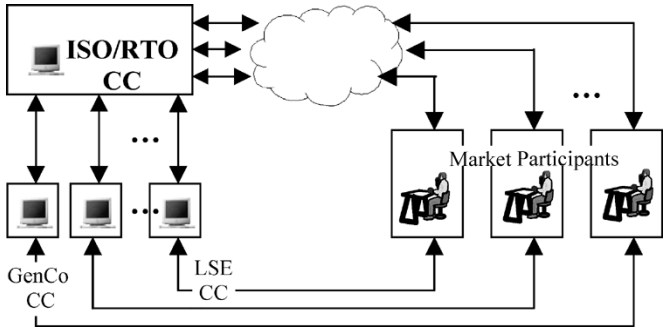

Fig. 1. Control centers (CC) in the market environment.

the real-time energy market in which the market clearing prices (a new set of LMPs) are calculated using SCED every 5 min or so based on revised generation bids and the actual operating condition from state estimation. Any amount of generation, load, or bilateral transaction that deviates from the day-ahead schedule will pay the balancing market LMP.

There is a host of supporting functions to ensure reliable delivery of electricity to consumers. To ensure against possible generator failure and/or sudden load increase, additional generation capacity has to be provided. This real power reserve is always ready to take on the responsibility instantaneously. Adequate reactive power resources are needed to maintain voltage at an acceptable level for proper operation of the system. These are all grouped under the name of *ancillary services*. An ancillary service can either be self-provision by users of the transmission system or system-wide management by the ISO/RTO. Markets have also been established to manage ancillary services.

Industry restructuring has so far brought two major changes in control centers structures. The first one is the expansion of the control center functions from traditional energy management, primarily for reliability reasons, to business management in the market. The second one is the change from the monolithic control center of traditional utilities that differed only in size to a variety of control centers of ISOs or RTOs, transmission companies (Transcos), generation companies (Gencos), and load serving entities (LSEs) that differ in market functions. The control centers in the market environment are structured hierarchically in two levels, as shown in Fig. 1.

In Fig. 1, the ISO/RTO control center that operates the electricity market of the region coordinate the LSE and other control centers for system reliability in accordance with market requirements. All entities, ISO, RTO, LSE, Genco, etc., are market participants. Their control centers are equipped with business functions to deal with the market. The part of control center functions that is responsible for business applications is called the *business management system* (BMS). The ISO or RTO is usually the market operator; therefore, its BMS is also called the *market operations system* (MOS). There are close interactions between the functions of EMS and BMS as shown in Fig. 2. For other types of control centers that do not operate a market, a BMS is added to the traditional EMS to *interact* with the market.

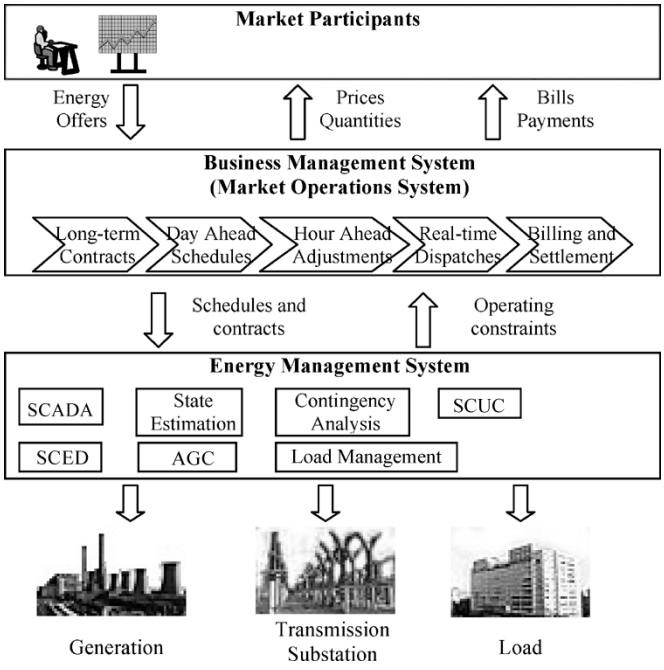

Fig. 2. EMS and BMS interactions.

III. Conventional Control Centers

Control centers have evolved over the years into a complex communication, computation, and control system. The control center will be viewed here from functional and architectural perspectives. As pointed out previously, there are different types of control centers whose BMS are different. From the functional point of view, the BMS of the control center of ISO/RTO is more complex than the others. Our description below is based on a generic control center of ISO/RTO, whereas specific ones may be somewhat different in functions and structure.

A. Functions

From the viewpoint of the system's user, a control center fulfills certain *functions* in the operation of a power system. The implementations of these functions in the control center computers are, from the software point of view, called *applications*.

The first group of functions is for power system operation and largely inherits from the traditional EMS. They can be further grouped into data acquisition, generation control, and network (security) analysis and control. Typically, data acquisition function collects real-time measurements of voltage, current, real power, reactive power, breaker status, transformer taps, etc. from substation RTUs every 2 s to get a snapshot of the power system in steady-state. The collected data is stored in a real-time database for use by other applications. The sequence of events (SOE) recorder in an RTU is able to record more real-time data in finer granularity than they send out via the SCADA system. These data are used for possible post-disturbance analysis. Indeed, due to SCADA system limitations, there are more data bottled up in substations that would be useful in control center operations.

Generation control essentially performs the role of *balancing authority* in NERC's functional model. Short-term *load forecasts* in 15-min intervals are carried out. AGC is used to balance power generation and load demand instantaneously in the system. Network security analysis and control, on the other hand, performs the role of *reliability authority* in NERC's functional model. *State estimation* is used to cleanse real-time data from SCADA and provide an accurate state of the system's current operation. A list of possible disturbances, or contingencies, such as generator and transmission line outages, is postulated and against each of them, power flow is calculated to check for possible overload or abnormal voltage in the system. This is called *contingency analysis or security analysis*.

The second group of functions is for business applications and is the BMS. For an ISO/RTO control center, it includes market clearing price determination, congestion management, financial management, and information management. Different market rules dictate how the market functions are designed. The determination of market clearing price starts from bid management.

Bids are collected from market participants. A bid may consist of start-up cost, no-load cost, and incremental energy cost. Restrictions may be imposed on bids for market power mitigation. Market clearing prices are determined from the acceptable bids. SCUC may be used to implement day-ahead markets. In a market with uniform pricing or pay-as-bid, the determination is done simply by the stack-up of supply versus demand. If the LMP that incorporates congestion management is applied, an OPF or SCED will be used. Other congestion management schemes such as uplift charges shared by all for the additional charges resulting from congestion are employed in some markets. To manage the risk of congestion charge volatility, a hedging instrument called transmission right is introduced. Transmission rights can be physical or financial. Physical rights entitle the holder the rights to use a particular portion of the transmission capacity. Financial rights, on the other hand, provide the holder with financial benefits equal to the congestion rent. The allocation and management of transmission rights are part of market operations and require running OPF. Financial management functions in the electricity market include accounting and settlement of various charges.

Fig. 3 highlights some major functions of BMS and EMS in today's control centers. In the deregulated environment, the AGC set points and the transaction schedules are derived from BMS, or the MOS, instead of the traditional EMS ED and interchange scheduling (IS). The BMS uses the network model, telemetry data and operating constraints from EMS to clear the market. We will explain the other blocks (ERP and data warehouse) outside the dotted lines of EMS and BMS in Fig. 3 in Section IV.

For a fair and transparent use of the transmission system, certain information needs to be available to the public and such information is posted, in the United States, through the Internet at the Open Access Same-time Information System (OASIS). Also, in compliance with NERC requirements, the tagging scheduling and checkout functions are used by control centers to process interchange transaction schedules.

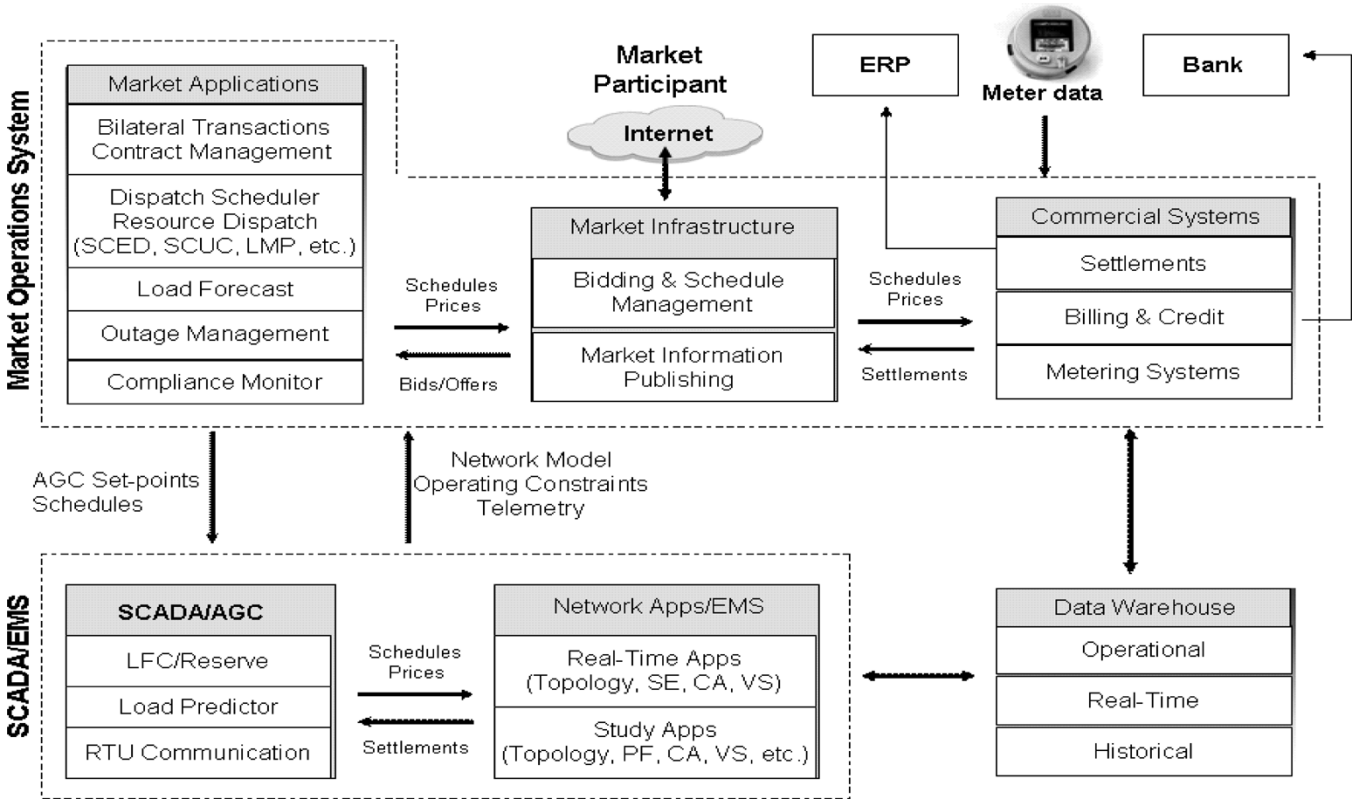

Fig. 3. CC functions.

B. Architecture

The SCADA system was designed at a time when the power industry was a vertically integrated monopoly. The centralized star configuration in which data from several remote devices were fed into a single computer was a ubiquitous configuration in the process control industry. This architecture fit the needs of the power system then. Over the years, networking and communications technologies in the computer industry have progressed significantly. But in the power industry they had not changed much and the SCADA system had served its needs well until the recent onset of deregulation. Substation automation in recent years, however, has introduced digital relays and other digital measurement devices; all called *intelligent electronic devices* (IEDs) [7]. An RTU could become another IED. The IEDs in a substation are linked by an LAN. The computer in the control center or the EMS, serving generation control and network analysis applications, has advanced from mainframes, to minis, to networks of workstations or PCs. A dual-configured LAN with workstations or PCs is commonly adopted in the control centers. The inter control center connections are enabled through point-to-point networks for data transfer. The BMS, on the other hand, communicate through the Internet. The control center thus has several networks: a star master-slave network from RTUs to the control center with dedicated physical links, a LAN for EMS application servers, a point-to-point network for inter control center connections, and the Internet for BMS market functions. The substation control center has a LAN for its SCADA and distribution feeder and other automation functions (Fig. 4).

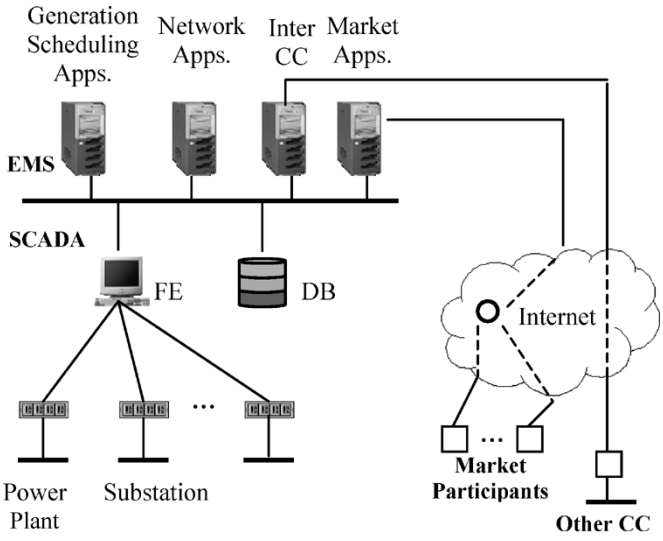

Fig. 4. Conventional CC architecture.

Large amounts of data are involved in a control center. In a conventional control center, real-time data are collected from RTUs. Historical data and forecasted data are stored in storage devices. Different sets of application data are used by different application servers. Display file data are used by GUI (graphical user interface) workstations. Various copies of data have to be coordinated, synchronized, and merged in databases. Historically, proprietary data models and databases were used as a result of proprietary RTU protocols and proprietary application software to which the databases were attached. Power systems are complex; different models

of the same generator or substation with varying degrees of granularity and diverse forms of representations are used in different applications, e.g., state estimation, power flows, or contingency-constrained economic dispatch.

IV. Changing Environment

The control center evolved from SCADA and EMS that were developed for an industry that was a vertically integrated monopoly with franchised service territory that for all practical purposes stayed stationary. The deregulation has unleashed changes in the structure of the industry. Divestitures, mergers and acquisitions continue to change the boundaries between companies. Consequently, the existing control centers have to be re-arranged both in terms of their geographic coverage and functionalities. Retail competition alters supply–demand alignment. The formation and reformation of ISOs and RTOs alter the alignment of companies and the relationships among their control centers. The rearrangement of ISOs and RTOs may result in control centers with noncontiguous territories under their jurisdiction. Frequent modifications of market and regulatory rules require changing functionalities of control centers. Some new market participants come and some old ones go. Some control center functions may be shifted away to companies dedicated to sell services to control centers and new functions and services may emerge as innovation runs its course. Control centers must be able to deal not only with their peer control centers, but also with a large number of new actors in the market environment, such as regulatory agencies, energy markets, independent power producers, large customers and suppliers, control center service providers, etc. The result of all of these is that the relations between a control center and the entities (other control centers, RTUs, market participants, new actors) below it, above it, or next to it are constantly undergoing changes. Thus, modern control centers have to be able to cope with changing business architecture.

In the early 1990s advocates from the technical community pushed from below for the integration of various "islands of automation," as well as various management information systems in the power industry in order to further enhance operational efficiency and reliability. They felt that the computer, communication, and control technologies had advanced to a point where this was possible. The list of the islands of automation and management information systems included EMS, SCADA, PPCS (power plant control systems), DA (distribution automation including substation automation and feeder automation), automated mapping and facility management, geographic information system (AM/FM/GIS), management information system (MIS), customer information system (CIS), etc. After some modest successes, a decade later, advocates from the management community are now pushing from above for digitization and integration of all operational and business processes in the enterprise, as the convergence of forces emanating from deregulation on the one hand and bursting Internet and e-business on the other, into an *enterprise architecture* [8]–[10]. This time around the efforts are much more compelling. The enterprise architecture effectively defines

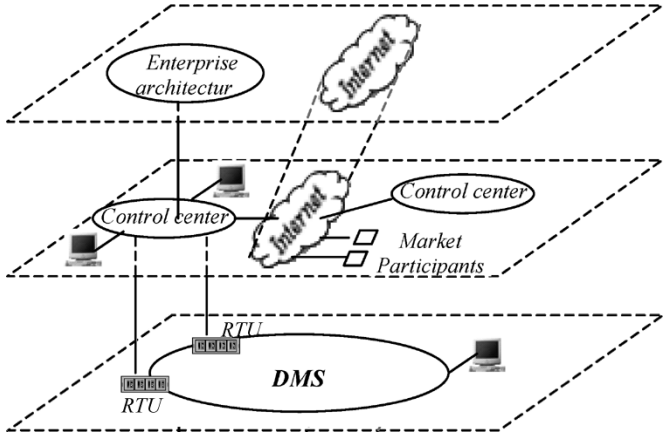

Fig. 5. Integration needs of control centers.

the business, the information necessary to operate the business, the technologies necessary to support the business operations, and the transitional processes necessary for implementing new technologies in response to changing business or regulatory requirements. Further, it allows a utility to analyze its internal processes in new ways that are defined by changing business opportunities or regulatory requirements instead of by preconceived system design.

It has become increasingly apparent that the control center EMS and SCADA systems, once the exclusive domain of operations, possess a wealth of technical as well as commercial information that could be used for many business applications to improve their responsiveness and precision, provided that EMS/SCADA can be fully integrated with enterprise-level systems. This has become imperative with advent of markets. Of particular interest is the integration of operational data from SCADA, EMS, and BMS into the *enterprise resource planning* (ERP) [11] or *enterprise resource management* (ERM) systems, such as SAP or Oracle. Therefore, control centers not only have to integrate "horizontally" with other control centers, market participants, etc., but also to integrate "vertically" with other functions in the enterprise (Fig. 5).

The ERP system manages all aspects of the business, including production planning, material purchasing, maintaining inventories, interacting with suppliers, tracking transactions, and providing customer service. ERP systems therefore need to have a rich functionality integrating all aspects of the business. By digitizing all these business processes, the company will be able to streamline the operation and lower the cost in the supply chain. The efficiency of an enterprise depends on the quick flow of information across the complete supply chain from customer to production to supplier. Many companies, including power companies, have begun to use a *data warehouse* [12] to support decision-making and other business processes in the enterprise. A data warehouse is a copy of the enterprise data relevant to business activities that are specifically structured for query and analysis. It evolved from the previously separate decision support and executive information systems. In such systems, the data from multiple sources are logically and physically transformed to align with the business structure.

Imbedded with analytical tools (e.g., SAS for statistical analysis), data mining provides customized views of the data to meet the needs of different players at all levels of business such as high-level views for the executives and more detailed views for others. Fig. 3, shown previously, indicates that today's control centers are linked to ERP and data warehouse in the enterprise.

Additionally, as the markets expand and the power grid becomes more congested, operational reliability is becoming more crucial. Maintaining system reliability requires more robust data acquisition, better analysis and faster coordinated controls. A distributed system is essential for meeting the stringent timing and reliability requirements [13].

To summarize, in a competitive environment, economic decisions are made by market participants individually and system-wide reliability is achieved through coordination among parties belonging to different companies, thus the paradigm has shifted from centralized to decentralized decision making. This requires data and application software in control centers to be *decentralized* and *distributed*. On the other hand, for efficient operation in the new environment, control centers can no longer be independent of other systems within and outside the enterprise. The operational data of BMS/EMS/SCADA are important for enterprise resource planning and business decision making, as well as data exchange with other control centers for reliability coordination. Control center functions must be *integrated* as part of the enterprise architecture, as well as integrated in the regional cooperation. Participants in a market are free to join and leave, and for various reasons and markets themselves are changing. Functions in control centers and control center configurations are changing. *Flexibility*, therefore, is important in this dynamic and uncertain environment. Control center design has to be *modular* so that modules can be added, modified, replaced, and removed with negligible impact on other modules to achieve maximum flexibility. Another aspect of flexibility in design is *scalability* and *expandability*, i.e., the ability to efficiently support the expansion of the control center resulting from either growth in the system or inclusion of new functionality. An *open* control center [14] becomes a necessity; dependence on specific vendors is no longer acceptable. Control center software must be *portable* to be able to run on heterogeneous hardware and software platforms. Different hardware, operating systems, software modules, can be *interoperable* within the system, all being part of the same control center solutions.

The changing environment therefore demands that the control centers be distributed and be fully:

- decentralized;
- integrated;
- flexible;
- open.

V. Enabling Technologies

The distributed control centers are evolving today with varying degrees of success. But the trends are unmistakable. We introduce in this section the basic concepts in the modern software technologies enabling such evolution.

A. Communications Protocols

Computer communications in the Internet, as well as in LANs, use standard protocols [15]. Protocols are agreed rules. Standard protocols are based on the open system interconnection (OSI) layered model, in which the upper layers rely on the fact that the functions in the lower layers work flawlessly, without having to know how they work, hence reducing the complexity in overall standardization. The *link layer* is responsible for the network access. Typically protocols for the link layer for LAN are Fiber Distributed Data Interface (FDDI) and Ethernet. The *network layer* is responsible for data addressing and the transmission of information. The protocols define how packets are moved around on the network, i.e., how information is routed from a start node to the end node. The typical protocol used for the network layer is the Internet Protocol (IP). The *transport layer* is responsible for the delivery of data to a certain node. It ensures whether and how the receipt of complete and accurate messages can be guaranteed. The Transmission Control Protocol (TCP) is the key protocol in this layer. The *application layer* ensures the delivery of data to a certain application from another application, which is located on the same or on another node in the network. This layer uses messages to encapsulate information. The protocols on this level include the Hypertext Transfer Protocol (HTTP) and File Transfer Protocol (FTP).

Protocols in a packet-switching shared communication network allow efficient allocation of bandwidth. TCP/IP is the protocol suite developed for the Internet and universally adopted. It can be used over virtually any physical medium [16]. The use of widely accepted standard IP protocol provides a high degree of interoperability. The power industry, through the Electric Power Research Institute (EPRI), has developed an Inter Control Center Protocol (ICCP), based on the OSI model and adopted as an International Electrotechnical Commission (IEC) standard, which is widely used as the protocol for inter control center communications. The RTU–control center communications, on the other hand, were developed when the communication channels had very limited bandwidth. Proprietary serial protocols were used and are still being used in most cases. The communications for market operations, which were introduced more recently, adopt e-commerce standards like XML (eXtensible Markup Language) [17] to mark up documents containing structured information. Document here refers to data forms such as e-commerce transactions, vector graphics, etc. Structured information means that it contains both content (words, pictures, etc.) and some indication of the role played by that content. A markup language adds formatting directives to plain text to instruct the receiver regarding how to format the content. XML is a platform-independent language for interchanging data for Web-based applications.

More on enabling technologies in communication networks for distributed computing in power systems can be found in [18], [19].

B. Distributed Systems

In the last 20 years, rapid evolution has been made in distributed systems, including distributed file systems, distributed memory systems, network operating systems, middleware, etc. As a result of the recent advent of high-speed networking, the single processor-computing environment has given way to distributed network environment. A *distributed system* here refers to a collection of independent computers that appears to its users to be a single coherent system [20]. The important characteristic is that to the user a distributed system presents no difference whether there is a single computer or multiple computers. A distributed system attempts to hide the intricacies and heterogeneity of the underlying hardware and software by providing a virtual machine on which applications can be easily executed. A distributed system is supported by both hardware and software, and its architecture determines its system functions. The most important element of the architecture is the operating system, which acts as the resource manager for applications to share resources such as CPUs, memories, peripheral devices, the network, and data.

A *multiprocessor operating system* provides more CPUs to support high performance and is transparent to application users. A *multicomputer operating system* extends the multiprocessor operating system to a network of homogenous computers with a layer of software that implements the operating system as a virtual machine supporting parallel and concurrent execution of various tasks. Each node has its own kernel that manages local resources and a separate module for handling interprocessor communications. Programming multicomputers may involve the complexity introduced in specifying communications through message passing. In contrast, *network operating systems* do not assume that the underlying hardware is homogeneous and that it should be managed as if it were a single system. Instead, they are generally constructed from a collection of uniprocessor systems, each with its own operating system. The machines and their operating systems may be different, but they are all connected to each other in a computer network. Also, network operating systems provide facilities with the ability to allow users to make use of *specific services* (such as file transfer, remote login, etc.) available on *a specific machine*.

Neither a multicomputer operating system nor a network operating system really qualifies as a distributed system according to the definition above. A multicomputer operating system is not intended to handle a collection of *independent* computers, while a network operating system does not provide a view of a *single coherent system*. A middleware-based distributed system is a solution to combining the scalability and openness of network operating systems and the transparency and related ease of use of distributed operating systems. It is accomplished by adding an additional layer of software that is used in network operating systems to more or less hide the heterogeneity of the collection of underlying platforms and also to improve distribution transparency. This additional layer is called *middleware* (Fig. 6).

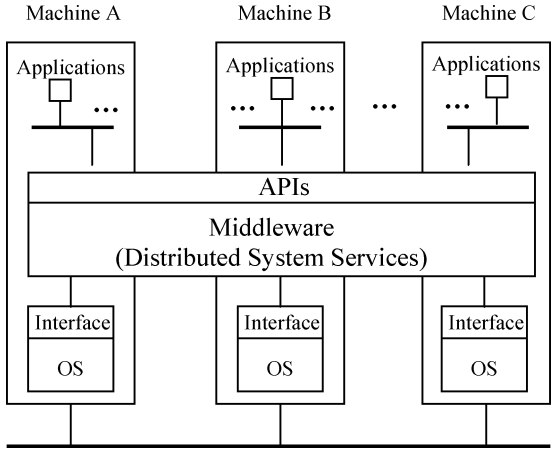

Fig. 6. Middleware-based distributed system.

C. Object Technology

Middleware is based on distributed object technology. We first review object-oriented methodology [21]. Object-oriented programming was developed in the late 1980s as an attempt to shift the paradigm of software design and construction from an art to engineering. The traditional procedural programming approach separated data from instructions, and every software package had to be constructed and comprehended in its totality. As applications become more involved, software has grown to be ever more complex and unwieldy, causing nightmares for its verification and maintenance.

Object-oriented programming is a modular approach to software design. Each module, or *object*, combines data and procedures (sequence of instructions) that act on the data. Each object is denoted by a name and has its state. The data or variables within the object express everything about the object (state) and the procedures or methods specify how it can be used (behavior). A method, or a procedure, has access to the internal state of the object needed to perform some operation. A group of objects that have similar properties, operations, and behaviors in common is called a *class*. It is a prototype that defines the variables (data) and methods common to all objects of a certain kind. A class may be derived from another class by inheriting all the data descriptions and methods of the parent class. *Inheritance* in object-oriented programming provides a mechanism for extending and/or specializing a class. Objects are invoked by sending messages (input) which in return produce output. Object-oriented languages provide a well-defined interface to their objects through classes. The concept of decoupling of the external use of an object from the object itself is called *encapsulation*. The interface is designed in such a way as to reveal as little as possible about its inner workings. Encapsulation leads to more self-contained and hence more verifiable, modifiable, and maintainable software. By reusing classes developed for previous applications, new applications can be developed faster with improved reliability and consistency of design. In this new paradigm, objects and

classes are the building blocks, while methods, messages, and inheritance produce the primary mechanisms.

C++ added classes to C in the late 1980s and became market leader in object-oriented programming language in the 1990s. Java was created as a simplification of C++ that would run on any machine and is now a major player among object-oriented languages. Java is innately object-oriented in contrast to the hybrid approach of C++. Java also has several advanced capabilities for distributed programming, distributed middleware, and the World Wide Web, respectively, such as RMI, EJB, and Applets, which will be discussed later. A major milestone in the development of object technology was the creation of the *Unified Modeling Language* (UML) in 1996. UML is now the industry standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems. It simplifies the complex process of software design, making a blueprint for construction.

There is an international organization called the Object Management Group (OMG), supported by most vendors and dedicated to maximizing the portability, reusability, and interoperability of software using object technology. The Common Object Request Broker Architecture (CORBA) specification is the result of input from a wide range of OMG members, making its implementations the most generally applicable option. Microsoft's DCOM, and the Sun Microsystems Remote Method Invocation (RMI) are examples of other models that enable software objects from different vendors, running on different machines, and on different operating systems, to work together.

D. Component Technology

Object-oriented programming in its early development has failed to meet expectations with respect to reusability. Component technologies build on the idea of providing third-party components that isolate and encapsulate specific functionalities [22], [23]. Components usually consist of multiple objects and this characteristic enables them to combine functionalities of the objects and offer them as a single software building block that can be adapted and reused without having to change them programmatically. The objective of the development of software components is to move toward a world in which components can be independently developed, assembled and deployed, like hardware. Reusable components are supposed to be plugged together in a distributed and inter-operable environment. Components vary in their granularity. A component can be very small, such as a simple GUI widget (e.g., a button), or it can implement an entire complex application, such as a state estimation. In the latter case, the application could be designed from scratch as a single component, a collection of components [24], or it could comprise a legacy application wrapped to conform to component interface standards.

A component must provide a standard interface that enables other parts of the application to invoke its functions and to access and manipulate the data within the component. The structure of the interface is defined by the component model. The component model provides guidelines to create and implement components that can work together to form a larger application. A component builder should not have deal with implementation of multithreading, concurrency control, resource-pooling, security, and transaction management. Furthermore, if these services were implemented in each component, achieving true plug-and-play application assembly would be very difficult. A component model standardizes and automates the use of these services.

The primary component models with wide acceptance within the software industry include: Enterprise JavaBeans (EJB), CORBA Components, and Microsoft COM/DCOM. Different models can be utilized to its greatest advantage of all available features of the underlying container and execution system to facilitate performance enhancement. Component adapters can be used to achieve to some degree the plug-and-play capability with different systems. XML Web Services, which will be discussed in Section VII, provide an Internet-based integration model for any-to-any integration and allow applications to communicate and share data over the Internet, regardless of operating system or programming language. They are like components.

E. Middleware

The objective of distributed object technology is to break complex applications into small components. Each component is in charge of a specific task and may run on a different machine in a network and all components may be seamlessly integrated into a common application. The need for interaction between the software objects led to the specification of middleware models to deal with communication between multiple objects that may reside on different network hosts. Middleware allows remote requests to invocate methods from objects located on other machines in the network [23].

Middleware is responsible for providing transparency layers that deal with distributed systems complexities such as location of objects, heterogeneous hardware/software platforms, and different object implementation programming languages. Shielding the application developer from such complexities results in simpler design and implementation processes. Besides the core task of transparency of object invocations, some middleware technologies offer additional services to the application developer such as security, persistence, naming, and trading.

Middleware provides generic interfaces for messaging, data access, transactions, etc. that enable applications and end users to interact with each other across a network. It represents a diverse group of software products that function as an integration, conversion, or translation layerIn essence, the term middleware denotes a set of general-purpose services that sits between platforms (various types of hardware and operating systems) and applications. A standardized interface allows applications to request services over the network without knowing how or even where the service is implemented. Middleware therefore facilitates the design of distributed systems whose configuration may dynamically change and where multiple applications implemented in

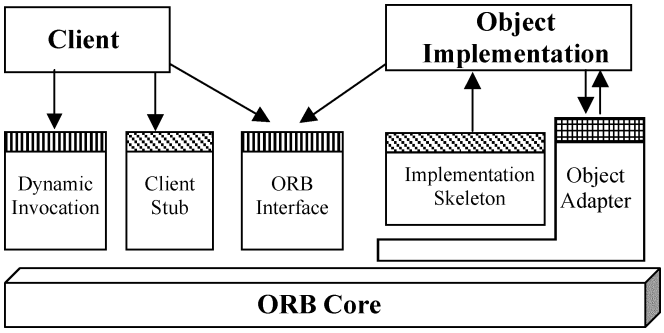

Fig. 7. ORB interface.

different languages and running on different systems communicate with each other.

There are four main classifications of middleware.

- Transactional middleware—supports distributed synchronous transactions.
- Message-oriented middleware—enables communication through messages.
- Procedural middleware—primarily used in point-to-point communication.
- Object-oriented middleware—includes object-oriented concepts and supports messaging and transactions.

F. CORBA

CORBA belongs to the object-oriented middleware classification. It is an open standard specifying a framework for transparent communication between applications and application objects [23], [25]. CORBA is a distributed object architecture that allows objects to interoperate across networks. A CORBA client asks for some services from an object, its request is transferred to the object request broker (ORB), which is responsible for forwarding the request to the right object implementation. This request contains all the information required to satisfy the request, such as target object, operations, parameters, etc. A client can request a service without knowing what servers are attached to the network. The various ORBs receive the requests, forward them to the appropriate servers, and then hand the results back to the client. Clients therefore never come in direct contact with the objects, but always with the interfaces of these objects (Fig. 7), which are determined through interface definition language (IDL). In addition, the communication of a client with an object running as a different process or on a different machine uses a communication protocol to portably render the data format independently from the ORB.

The ORB provides a mechanism for transparently communicating client requests to target object implementations. The ORB simplifies distributed programming by decoupling the client from the details of the method invocations. This makes client requests appear to be local procedure calls. When a client invokes an operation, the ORB is responsible for finding the object implementation, transparently activating it if necessary, delivering the request to the object, and returning any response to the caller. The interfaces provided by ORB include domain-independent interfaces, such as the discovery of other available services, security, transactions, event notification, and other common facilities, and domain-specific interfaces that are oriented toward specific application domains, such as power systems, as well as nonstandard interfaces developed specifically for a given application.

The ORB is the middleware that handles the communication details between the objects. CORBA is a mature technology suitable for tightly coupled transaction processing systems in high-volume applications within an enterprise.

G. Agents Technology

Agent technology is an extension of object technology. An agent is a software entity that is situated in some environment and can sense and react to changes in that environment [26]. Agents do not just act in response to changes that have occurred in their environment, they have their own goals and also can initiate action to achieve them, i.e., an agent is capable of independent action on behalf of its user or owner. In other words, an agent can figure out for itself what it needs to do in order to satisfy its design objectives, rather than being explicitly told what to do at any given moment. Agents are loosely coupled and can communicate via messaging. New functions can easily be added to an agent-based system by creating a new agent, which will then make its capabilities available to others. A multiagent system is one that consists of a number of agents which interact with one another, typically by exchanging messages through some computer network infrastructure. Tasks are carried out by interacting agents that can cooperate with each other. Agents are thus required to cooperate, coordinate and negotiate with each other. The agent technology makes it possible to build extensible and flexible distributed cooperation systems.

H. Industry Efforts

EPRI has been leading the power industry in standardizing communication protocols and data models. The Utility Communication Architecture (UCA) launched in the late 1980s was an attempt to define a set of comprehensive communication protocols based on the OSI reference model for use in electric utilities. Notable success out of that effort is the ICCP mentioned in Section V-A, which became an IEC standard (IEC TASE-2) for communications among control centers. For effective communication, the protocol is only one aspect; the semantics of the data to be exchanged is just as important. The *Common Information Model* (CIM) [27]–[29], again led by EPRI, is to specify common semantics for power system resources (e.g., a substation, a switch, or a transformer) used in EMS, their attributes and relationships and is described in the UML in recognition of object-oriented component technology. The objective of CIM is to support the integration of independently developed applications between vendor-specific EMS systems, or between an EMS system and other systems that are related to EMS operation, such as generation or distribution management. CIM has been extended to support exchange of market information both within and between ISOs and RTOs. The CIM market extension is called CME, and it expedites

e-transactions and market performance measurement. XML for CIM model exchange has been developed [30].

CIM together with component interface specification (CIS) forms the core of EPRI's Control Center Application Programming Interface (CCAPI) project. CIM defines the essential structure of a power system model whereas CIS specifies the component interfaces. CCAPI has since been adopted as an IEC standard: the IEC 61 970 (Energy Management System Application Programming Interface) [31]. IEC 61 970 normalizes a set of application programming interfaces (APIs) for the manipulation of both real-time critical and near real-time EMS/SCADA data, as well as a data model and a configuration data exchange format.

Other IEC standards [32] that are relevant to control center operation are IEC 61 968 (System Interfaces for Distribution Management) and IEC 61 850 (Communication Networks and Systems in Substations). IEC 61 968 extends the IEC 61 970 model for both modeling and APIs to distribution management systems. The APIs are meant for inter-application messaging at the enterprise level. IEC 61 850 primarily specifies abstract communication service interfaces (ACSI) and their mappings to concrete protocols, but it also defines an elaborate data model and configuration data exchange format, independent of CIM.

VI. Modern Distributed Control Centers

The term "distributed control center" was used in the past to refer to the control center whose applications are distributed among a number of computers in a LAN [33]. By that standard almost all control centers today that are equipped with distributed processing capability in a networked environment would be called "distributed." That definition is too loose. On the other hand, if a control center that is fully decentralized, integrated, flexible, and open, as the Grid services-based future control center to be described in Section VIII-C, is counted as a distributed control center, the definition is perhaps too stringent. We adopt the definition of the distributed system in Section V-B to control centers and call it a distributed control center if it comprises of a set of *independent* computers that appears to the user as a single *coherent* system. A distributed control center typically has some or all of its data acquisition and data processing functions distributed among independent computers and its EMS and BMS applications also distributed. It utilizes the distributed system technologies to achieve some level of decentralization, integration, flexibility, and openness: the characteristics that is desirable in today's power system operating environment.

Current trends in the development of distributed control centers from the previous multicomputer networked system to a flexible and open system with independent computers are moving in the following directions:

- separation of SCADA, EMS, and BMS;
- IP-based distributed SCADA;
- standard (CIM)-based distributed data processing;
- middleware-based distributed EMS and BMS applications.

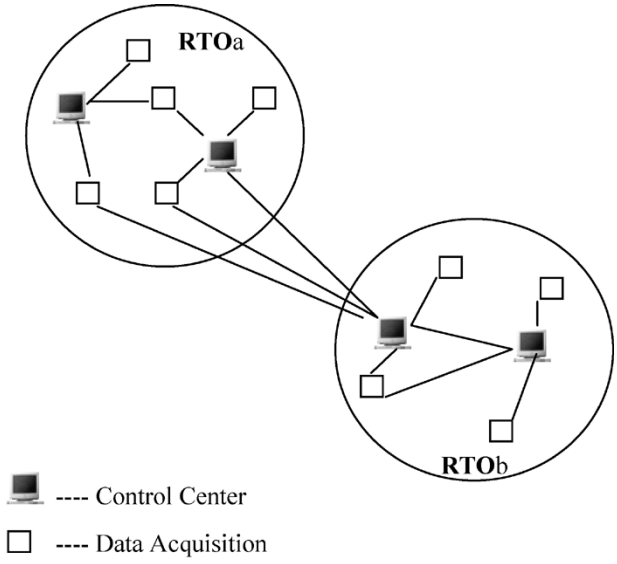

Fig. 8. Distributed data acquisition.

The data acquisition part of SCADA handles real-time data, and is very transaction-intensive. The applications in EMS involve mostly complex engineering calculations and are very computation-intensive. These two dissimilar systems are tightly bundled together in a conventional control center because proprietary data models and databases are used due to historical reasons. With proprietary data models and database management systems to handle data, such data could not be easily exchanged, and it prevents effective use of third-party application software. A separate SCADA system and EMS system would serve a control center better by expediting the exploitation of new technologies to achieve the goals of decentralization, integration, flexibility, and openness. The separation of SCADA and EMS is a logical thing to do.

The SCADA function in a conventional control center starts with the RTUs collecting data from substations and, after simple local processing (e.g., data smoothing and protocol specification), the data is then sent through a dedicated communication channel with proprietary protocol to the appropriate data acquisition computer in the control center where a TCP/IP based computer network is used. An interface is therefore needed. The data acquisition computer converts the data and prepares it for deposition in the real-time database. The real-time database is accessed and used by various applications. For reasons of efficiency, the interface may be handled by a telecontrol gateway, and more gateways may be used in a control center. The location of the gateway may move to the substation if standard IP protocol is used. The gateway is then connected to the control center. If the RTU is TCP/IP based, it can be connected directly to the control center resulting in a distributed data acquisition system. In this way, the gateway serves as a data concentrator and communication processor (Fig. 8). RTUs or IEDs may be connected to a data concentrator or connected directly to the control center [34].

The use of fixed dedicated communication channels from RTUs to control center leaves no flexibility in RTU-control center relationship which was not a problem in the past. When, for example, another control center requires real-time data from a particular substation not in the original design, there are only two ways to do it. In the first case, the other control center has to acquire the data through the control center to which the RTU is attached. In the second case, a new dedicated channel has to be installed from the RTU to the other control center.

The dedicated channels with proprietary protocols for SCADA were developed for reasons of speed and security [35], [36]. Communication technologies have advanced tremendously in the last couple of decades in both hardware and software, resulting in orders of magnitude increase in transmission speed and sophistication in security protection. Modern communication channels with metallic or optical cables have enormous bandwidths compared to the traditional 9600 kb/s or less available for RTU communications. Responsibility to guarantee timely delivery of specific data in a shared communication network such as intranet or Internet falls to the QoS function of communication network management and is accomplished through protocols for resource allocation. With further advancement in QoS, more and more stringent real-time data requirements may be handled through standard protocols. Network security involves several issues: confidentiality, integrity, authentication, and nonrepudiation. Cryptography is used to ensure confidentiality and integrity, so that the information is not available to and can not be created or modified by unauthorized parties. Digital hash is used for authentication and digital signature for nonrepudiation. Network security has advanced rapidly in recent years [37].

There is no reason that the new SCADA communications should not be using standard protocols such as IP-based protocols. Indeed, inability of SCADA to take advantage of recent progress in cyber security is considered a serious security risk by today's standards [38]. SCADA should be IP-based and liberated from dedicated lines by tapping into an available enterprise WAN or as a minimum use Internet technology to enable the use of heterogeneous components. In the future, as Internet QoS performance and encryption protocols improve, there will be little difference between a private-line network and *virtual private network* (VPN) on the Internet when standard protocols are used. A VPN is a network that is constructed by using public wires to connect nodes. The system uses encryption and other security mechanisms to ensure that only authorized users can access the network and that the data cannot be intercepted. VPN can be used to augment a private-line enterprise network when a dedicated facility can not be justified. In other words, the physical media and the facilities used for the network will become less of a concern in a control center when standard protocols are used.

If standard communication protocols are used, a control center (i.e., its application software) may take data input from data concentrators situated either inside or outside its territory. The only difference between data from an internal data concentrator or an external one is the physical layer of the communication channel. The former may be through the intranet and the latter needs special arrangement for a communication channel [38], [40] (Fig. 9).

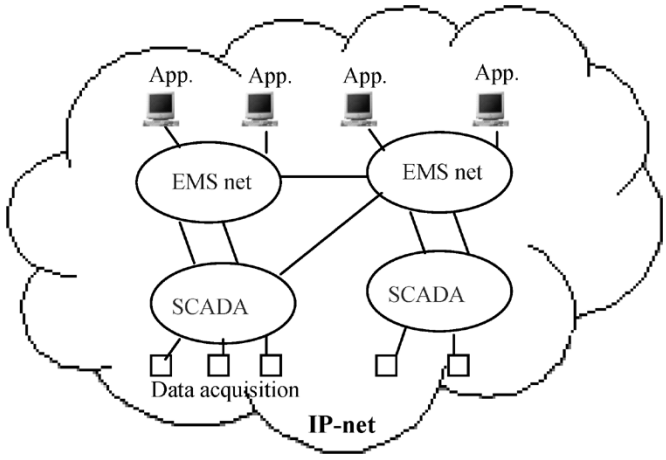

Fig. 9. IP-based distributed SCADA.

With IP-based distributed SCADA that uses standard data models, data can be collected and processed locally before serving the applications. Again using standard data models, databases can also be distributed. Search engines may be utilized for fast and easy access to relevant information. Intelligent agents with learning capability may be deployed for data management and data delivery.

Once the data is deposited in the real-time database, it can be used by various applications to serve the required functions of EMS or BMS. The output of an application may be used by another application. As long as an application has access through the network to the database with sufficient speed and ensured security, the physical location of the application server and the data will be of little concern. The network should be capable of hosting any kind of applications and supporting intelligent information gathering through it [41]. Component and middleware technologies enable such distributed architecture.

Present-day control centers are mostly provided with CIM data models and middleware that allow distributed applications within the control center. Only a few of them use CIM-based data models and middleware-based applications as their platforms. Specific applications of distributed technologies include Java [42], component technology [43], [44], middleware-based distributed systems [45], CORBA [46], [47], and agent technology [48], [49].

A comprehensive autonomous distributed approach to power system operation is proposed [24]. Deployment of agents responsible for specific temporally coordinated actions at specific hierarchical levels and locations of the power system is expected to provide the degree of robust operation necessary for realizing a *self-healing* grid [13].

VII. Emerging Technologies

The information and communication technologies have converged into *Grid services* that are based on *Web services*

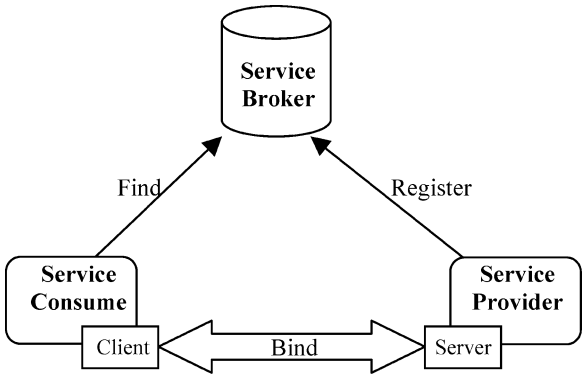

Fig. 10. Service-oriented architecture.

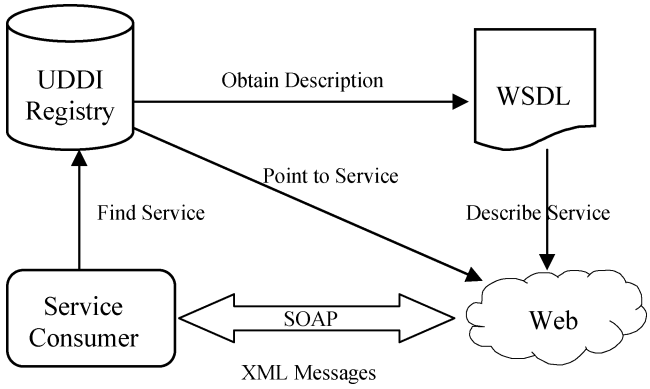

Fig. 11. Web service.

and *Grid computing*. Future control centers should embrace this development and build an infrastructure on Grid services. In this section, we introduce the service-oriented architecture, Web services and Grid computing.

A. Service-Oriented Architecture (SOA)

SOA has evolved over the last ten years to support high performance, scalability, reliability, and availability in computing. Applications have been designed as services that run on a cluster of centralized servers. A *service* is an application that can be accessed through a programmable interface. With that definition, an agent can be viewed as providing a service. The service concept is a generalization of the component concept. The following figure depicts the conceptual roles and operations of a SOA. The three basic roles are the service *provider*, the service *consumer*, and a service *broker*. A service provider makes the service available and publishes the contract that describes its interface. It then *registers* the service with a service broker. A service *consumer* queries the service broker and finds a compatible service. The service broker then gives the service consumer directions regarding where to find the service and its service contract. The service consumer uses the contract to bind the client to the server (Fig. 10).

Most standard distributed computing systems implement a SOA. For example, clients may access SOA services using a middleware, such as DCOM, CORBA, or RMI. ORB in CORBA functions as a service broker (Section V-F). While these tightly coupled protocols are very effective for building a specific application, their flexibility in the reusability of the system is still limited, compared to Web services, to be introduced below, that have evolved from such systems. Because they are not fully independent of vendor implementations, platforms, languages, and data encoding schemes, SOA based on middleware has limitation on interoperability as well.

B. Web Services

Web services [50], [51] are a particular type of SOA that operates effectively over the Web using XML-based protocols. Web services enable interoperability via a set of open standards to provide information about the data in a document to users on various platforms. Web services are built on service-oriented architecture, Internet/intranet technologies, and other technologies like information security. The core components of Web services consist of:

- Simple Object Access Protocol (SOAP) for cross-platform inter-application communication;
- Web Services Description Language (WSDL) for the description of services;
- Universal Description, Discovery, and Integration protocol (UDDI) for finding available Web services on the Internet or corporate networks.

Web service providers describe their services using WSDL (Fig. 11) and register with UDDI registry. UDDI can point to services provided by service providers and obtain descriptions through WSDL. For a service client, a typical invoking process would be the following:

- locate a Web service that meets the requirements through UDDI;
- obtain that Web service's WSDL description;
- establish the link with the service provider through SOAP and communications with XML messages.

The Web services architecture takes all the best features of the service-oriented architecture and combines it with the Web. The Web supports universal communication using loosely coupled connections. Web protocols are completely vendor-, platform-, and language-independent. Web services support Web-based access, easy integration, and service reusability. With Web service architecture, everything is a service, encapsulating behavior and providing the behavior through an interface that can be invoked for use by other services on the network. Services are self-contained, modular applications that can be described, published, located, and invoked over the Internet. Promises of Web services for power system applications have been pointed out [52]–[55].

C. Grid Computing and Grid Services

Online power system dynamic analysis and control for future control centers will demand computational power beyond what is currently available. It also requires distribution of intelligence at all hierarchical levels of the power grid to enable sub-second coordinated intelligent control actions (Section VIII-B). In future control centers, applications need to be more intelligent and computation needs to be more intelligent too. In this subsection, we look at recent progress

and future promises in distributed computing that facilitate distributed intelligence and fast computing.

In recent years, progress has been made in distributed high-performance computing. High-performance computing, traditionally called supercomputing, is built on different, but co-located processors. It is expensive and used by only special customers for special purposes. Cluster computing is based on clusters of high-performance and massively parallel computers built primarily out of commodity hardware components. It is popular and has been applied to control centers.

There is a new paradigm, called *Grid computing* [56]–[58], that has emerged out of cluster computing. It is a clustering of a wide variety of geographically distributed resources (computer CPUs and memories) to be used as a unified resource, yet it provides seamless access to and interaction among these distributed resources, applications and data. A virtual organization is formed when an application is invoked. This new paradigm is built on the concept of services in the service-oriented architecture. However, Grid computing has generalized the concept of software services to resources. In other words, resources in Grid computing are provided as services. Grid computing renders Grids resources and Grid applications to consist of dynamically composed services.

The motivation and the vision of Grid computing are to develop:

- a world in which computational power (resources, services, data) is as readily available as electrical power and other utilities, in which computational services make this power available to users;
- in which these services can interact to perform specified tasks efficiently and securely with minimal human intervention.

More specifically, the idea of grid computing is to provide:

- universal access to computing resources;
- seamless global aggregation of resources;
- seamless composition of services.

To enable the aggregation of geographically distributed resources in grid computing, protocols and mechanisms to secure discovery of, access to, and aggregation of resources for the realization of virtual organizations and the development of applications that can exploit such an aggregated execution environment are necessary. In 1996 the Advanced Research Projects Agency (ARPA) launched the successful Globus Project with the objective to create foundation tools for distributed computing. The goal was to build a system that would provide support for resource discovery, resource composition, data access, authentication, authorization, etc. Grid computing is making progress to become a practical reality. And it is the future to come.

Advocates of Grid computing are pushing for the grand vision of global grids over the Internet. The other extreme, however, is the cluster grids of small managed computer cluster environments that are popularly employed today. In between, we may view various sub-grids of the Global grid in Grid computing as consisting of:

- enterprise grids;
- partner grids.

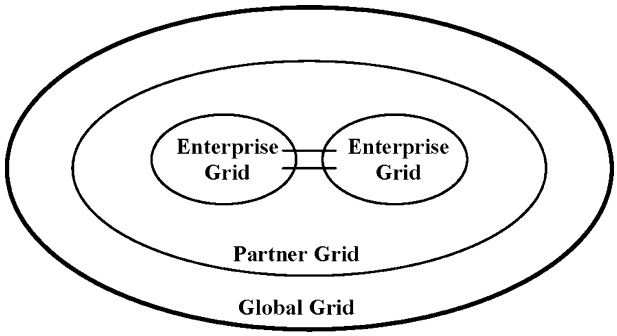

Fig. 12. Enterprise grids and partner grids.

Enterprise grids are meant for multilocation enterprises to share their resources. Partner grids are extensions of enterprise grids to facilitate collaboration and access to share resources between sister organizations (Fig. 12). They are of particular interest in the context of control centers.

Grid service is a convergence of Grid computing and Web services. Grid services offer dependable, consistent, and pervasive access to resources irrespective of their different physical locations or heterogeneity, using open standard data formats and transport protocols. Grid services can be viewed as an extension of Web services. A standard called Open Grid Services Infrastructure (OGSI) was developed using this approach. Globus Toolkit 3.2 (GT3.2) is a software toolkit based on OGSI that can be used to program Grid-based applications. Another standard, the Web Services Resource Framework (WSRF), was presented in 2004, to substitute OGSI. WSRF aims to integrate itself into the family of Web services. Globus Toolkit 4 (GT4) is a full implementation of WSRF [59].

D. Web and Grid Services Security

Security has become a primary concern for all enterprises exposing sensitive data and business processes through a shared environment. In fact, the biggest obstacle to wider adoption of Web services today has been the security concern. It is thus important for Web services and Grid services to have impermeable security mechanisms. Web services security has several dimensions: it requires *authentication* (establishing identity), *authorization* (establishing what a user is allowed to do), *confidentiality* (ensuring that only the intended recipient can read the message), and *integrity* (ensuring the message has not been tempered with). Encryption and digital signatures are the means to accomplish cyber security. We mentioned in what follows several recent developments in Web services security [60]Security is obviously one of the most challenging aspects of Grid computing and great progress is being made [61].

Security assertion markup language (SAML), developed by OASIS, defines security-related schemas for structuring documents that include information related to user identity and access or authorization rights. By defining how this information is exchanged, SAML lets companies with different internal security architectures communicate. It functions as a

framework for exchanging authentication, attribute, and authorization assertions across multiple participants over the Internet using protocols such as HTTP and SOAP. SAML can also indicate the authentication method that must be used with a message, such as a password, Kerberos authentication ticket, hardware token, or X.509 digital certificate. Another development is the Web Services Security Protocol (WS-Sec), developed by IBM and Microsoft, that let applications attach security data to the headers of SOAP messages. It can include security algorithm metadata that describe the process for encrypting and representing encrypted data and define syntax and processing rules for representing digital signatures. It is under consideration as a standard by OASIS. Traditional network firewalls are not suitable for Web services because they have no way of comprehending the messages crossing their posts. Other traditional security techniques, such as virtual private networks or secure sockets layer (SSL) technology, can not secure the large number of transactions of Web services. XML firewall has been developed to intercept incoming XML traffic and take actions based on the content of that traffic.

VIII. Future Control Centers

Future control centers, as we envision, will have much expanded applications both in power system operations and business operations based on data that are collected in a much wider and faster scale. The infrastructure of the control center will consist of large number of computers and embedded processors (e.g., IEDs) scattered throughout the system, and a flexible communication network in which computers and embedded processors interact with each other using standard interfaces. The data and data processing, as well as applications, will be distributed and allow local and global cooperative processing. It will be a distributed system where locations of hardware, software, and data are transparent to the user.

The information technology has evolved from objects, components, to middleware to facilitate the development of distributed systems that are decentralized, integrated, flexible, and open. Although significant progress has been made, it is still not fully distributed. Today's middleware is somewhat tightly coupled. For example, the ORB in CORBA, which provides interface between objects, is not fully interoperable. Recent progress in the ease and popularity with XML-based protocols in Internet applications has prompted the development of the Web services architecture, which is a vital step toward the creation of a fully distributed system. The concept of *services* represents a new paradigm. A software service was originally defined as an application that can be accessed through a programmable interface. Services are dynamically composed and distributed, and can be located, utilized, and shared. The developers in Grid computing have extended the concept of service from software to resources such as CPU, memory, etc. Resources in Grid computing can be dynamically composed and distributed, and can be located, utilized, and shared. Computing, as a resource service, is thus distributed and shared. The idea of service-oriented architecture, Grid computing and open standards should be embraced for adoptions not only in future control centers, but also in other power system functions involving information and communication technologies.

In a Grid services environment, data and application services, as well as resource services, are distributed throughout the system. The physical location of these services will be of little concern. It is the design of the specific function in an enterprise that dictates how various services are utilized to achieve a specific goal. The control center function, i.e., to ensure the reliability of power system operation and to manage the efficient operation of the market, represents one such function in the enterprise. In this new environment, the physical boundary of any enterprise function, such as the control center, may no longer be important and indeed become fuzzy. It is the collective functionality of the applications that represents the control center makes it a distinct entity.

Grid services-based future control centers will have distributed data services and application services developed to fulfill the role of control centers in enhancing operational reliability and efficiency of power systems. Data service will provide just-in-time delivery of information to applications that perform functions for power system control or business management. The computer and communication infrastructure of future control centers should adopt standard Grid services for management of the resources scattered around the computers and embedded processors in the power system to support the data and application services.

The concepts we just brought up about future control centers: extended data acquisition, expanded applications, Web services, Grid computing, etc., will be elaborated in more details in the remaining subsections.

A. Data Acquisition

For power system reliability, the security monitoring and control function of the control center is actually the second line of defense. The first line of defense is provided by the protective relay system. For example, when a fault in the form of a short circuit on a transmission line or a bus occurs, measurement devices such as a current transformer (CT) or potential transformer (PT) pick up the information and send it to a *relay* to initiate the tripping (i.e., opening) of the appropriate circuit breaker or breakers to isolate the fault. The protective relay system acts in a matter of one or two cycles (one cycle is 1/60 of a second in a 60-Hz system). The operation of the protective relay system is based on local measurements. The operation of security monitoring and control in a control center, on the other hand, is based on system-wide (or wide-area) measurements every 2 s or so from the SCADA system. The state estimator in EMS then provides a snapshot of the whole system. Different time scales driving the separate and independent actions by the protective system and the control center lead to an information and control gap between the two. This gap has contributed to the missed opportunity in preventing cascading outages, such as the North American blackout and the Italian blackout of 2003. In both cases, protective relays operated according to their designs by responding to local measurement, whereas the control center

did not have the system-wide overall picture of events unfolding. During that period of more than half an hour, control actions could have been taken to save the system from a large-scale blackout.

The security monitoring and control functions of today's control center, such as state estimation, contingency analysis, etc., are based on steady-state models of the power system. There is no representation of system dynamics that govern the stability of a system after a fault in control center's advanced application software. The design philosophy for security control is that of preventive control, i.e., changing system operating conditions before a fault happens to ensure the system can withstand the fault. There is no analytical tool for emergency control by a system operator in a control center. All of these are the result of limitations imposed by: 1) the data acquisition system and 2) computational power in conventional control centers. The issue of computational power has already been addressed by Grid computing in Section VII-C. We will discuss the issue of measurement system in what follows.

Although RTUs, IEDs, and substation control systems (SCSs) in substations sample power measurements in a granularity finer than a second, SCADA collects and reports data (by exception) only in the interval of several seconds. The system-wide measurements, strictly speaking, are not really synchronized, but their differences are in the order of the time-scale of the window of data collection, which is approximately 2 s. But because the model is a steady-state model of the power system, such a discrepancy is tolerated. As mentioned earlier, the bandwidth limitation problem in SCADA is a legacy problem from the past. Future communication networks for SCADA using WAN will have much wider bandwidth and will be able to transmit measurement data in finer resolutions. However, the data needs to be synchronized. This can be done by using synchronization signals from the global positioning system (GPS) via satellites. Modern GPS-based *phasor measurement units* (PMU) [62] are deployed in many power systems to measure current and voltage phasors and phase angle differences in real time. GPS in PMU provides a time-tagged one pulse-per-second (pps) signal which is typically divided by a phase-locked oscillator into the required number of pulses per second for sampling of the analog measurements. In most systems being used at present, this is 12 times per cycle or 1.4 ms in a 60-Hz system. In principle, system-wide synchronous data in the order of milliseconds or even finer can be collected by PMU-like devices in the future to be used for monitoring system dynamic behavior. PMUs and the future generation of PMU-class data acquisition devices can augment existing RTUs, IEDs, and SCSs to provide a complete picture of power system dynamical conditions and close the gap between today's protective relay operations and control center functions.

The electricity market is a new experiment. As the market matures and our understanding of market operation strengthens, new measurement requirements and devices, and new market information or data acquisition systems that will ensure an efficient and fair market will definitely emerge. Real-time measurements are needed and techniques should be developed for market surveillance to mitigate market power and for enforcement of contract compliance. Real-time data, properly collected and judiciously shared, can also assist regional cooperation, such as regional relay coordination or regional transmission planning among interconnected systems, that benefits all parties involved [8].

B. Functions

Market functions of a control center will expand once new measurement systems are available and our understanding of market behavior increases. We mentioned market surveillance and contract compliance and there will be more in the future. On the power system operation side, the new data acquisition systems, such as PMUs that provide measurements in the order of milliseconds, offer new opportunities for dynamic security assessment and emergency control that would greatly enhance system reliability. A great deal of research has already begun along these directions.

Current control centers provide analysis and control of power systems based on the steady-state models of the power system. For system dynamic effects, such as transient stability, the approach has always been to conduct simulation studies based on postulated future conditions and the results are used to design protective system response and to set operational limits on transmission lines and other apparatuses. This approach is becoming more and more difficult to continue due to increasing uncertainty in system operation conditions in the market environment. Online monitoring and analysis of power system dynamics using real-time data several times a cycle will make it possible for appropriate control actions to mitigate transient stability problems in a more effective and efficient fashion [63]. Other system dynamic performance, including voltage stability and frequency stability, can also be improved with the assistance of PMUs [64]–[66]. A comprehensive "self-healing power grid" framework for coordinating information and control actions over multiple time-scales ranging from milliseconds to an hour employing distributed autonomous intelligent agents has been defined in [13].

Another function in control centers that has developed rapidly in the last couple of years and will become even more important in the future is the visualization tools [67], [68] to assist power system operators to quickly comprehend the "big picture" of the system operating condition. As technology progresses, more and more data will become available in real time. The human-machine aspect of making useful information out of such data in graphics to assist operators comprehend the fast changing conditions easily and timely and be able to respond effectively is crucial in a complex system such as the power system as long as human operators are still involved.

Developing new functions to utilize enhanced data acquisition systems to greatly improve power system reliability and efficiency will be a great challenge to the research community. Successful research results will be valuable in bringing power system operations to a new level of reliability and efficiency.

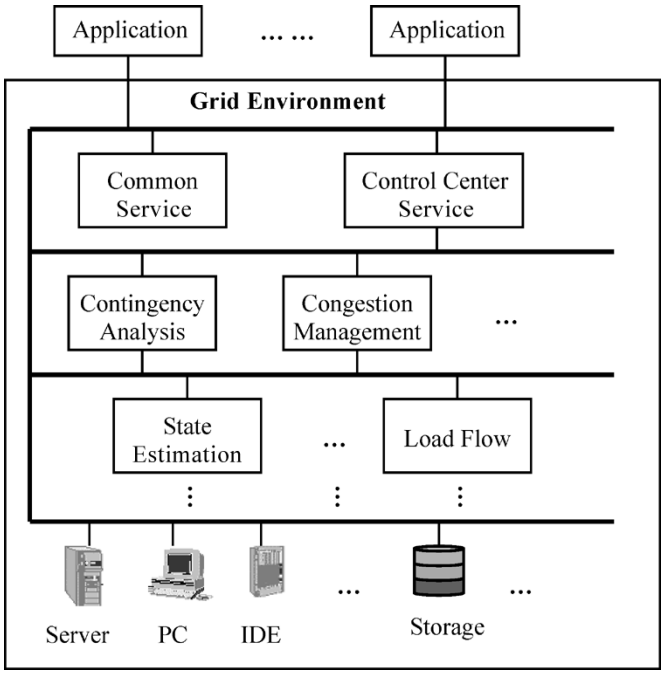

Fig. 13. Control center application services.

C. *Grid Services-Based Future Control Centers*

A future Grid service-based control center will be an ultimate distributed control center that is decentralized, integrated, flexible, and open. In a Grid-service environment, everything is a service. Future control centers will have data services provided throughout the power system. Data acquisition services collect and timestamp the data, validate and normalize them, and then make it available. Data processing services process data from various sources for deposition into databases or high level applications. Applications will call data services and data will be delivered just in-time for critical applications. Various functions serving the needs of control centers are carried out as application services. Traditional applications, such as contingency analysis, congestion management, may be further decomposed into their constituent components, for example, power flows, OPF, etc. Application services may have different granularity and may rely on other services to accomplish its job (Fig. 13). Data and application services are distributed over the Grids. The Grids can use the intranet/Internet infrastructure in which sub-networks are formed for different companies (enterprise grids) with relatively loose connection among cooperative companies (partner grid). The computer and communication resources in the Grids are provided and managed by the standard resource services that deliver distributed computing and communication needs of the data and application services.

The designer of control centers develops data and application services, and no longer needs to be concerned with the details of implementation, such as the location of resources and information security, provided the services are properly registered in the Grid environment. The new business model is that the software vendors will be service providers and power companies as service integrators. Power companies focus on information consumption and vendors focus on software manufacturing, maintenance, and upgrading. Computer and communication infrastructure will be left to the ICT professionals. This clear separation of responsibility would simplify and accelerate the delivery of new technology.

We envision future control centers based on the concept of Grid services include among others the following features:

- an ultrafast data acquisition system;
- greatly expanded applications;
- a partner grid of enterprise grids;
- dynamic sharing of computational resources of all intelligent devices;
- use of service-oriented architecture;
- distributed data acquisition and data processing services;
- distributed control center applications expressed in terms of layers of services;
- use of standard Grid services architecture and tools to manage ICT resources.

IX. Conclusion

A control center uses real-time data to support the operation of a power system to ensure a high level of reliability and an efficient operation of the market. In this paper, we have briefly reviewed the evolution of control centers from its past to present. An elementary tutorial on the enabling technologies, from object to middleware technologies that help in making today's control centers more decentralized, integrated, flexible, and open is included. The power industry is catching up in the application of the latest ICTs to control centers. With the rise of the Internet age, the trend in ICT is moving toward Grid services. The introduction of PMUs, on the other hand, may usher in a new generation of data acquisition systems and enabling of more advance applications in controlling dynamic performance of power systems in real time. We have attempted to outline a development direction for future control centers utilizing Grid services architecture.

Control centers involve extremely complex systems with intricate linkages of hardware, software, and devices. The presentation of this paper aims to simplify a great deal of the complex issues involved in implementation for the sake of conceptual clarity. Every step in the implementation is a challenge. However, challenges should not deter us from taking actions for change. The more we resist change to revamp the monstrous complex system today, the more difficult it will become to take advantage of technology advancement to improve power system operations in the future. By not tapping into the mainstream of the ICT, the maintenance cost of the custom system will eventually outweigh the investment cost of new technologies.

The focus of this paper has been on the technology and the closing of the technology gap between power system control centers and ICT. Closing the gap in technology is relatively easier compared to another gap we would like to highlight here before the closure. This is the gap between applications and technology. The promises of new data acquisition devices and systems, Grid computing and boundless-bandwidth communications offer tremendous opportunities in

the development of new functions and new approaches to improve power system reliability and efficiency. The advances in research into innovative theories and methods to effectively utilize new technologies are much slower than the technology advancement itself. Approaches and methodologies for power system analysis and control have changed very little in the past few decades despite of the fast changes in technology and environments. Technology gaps can be closed by materials supplied by hardware, software and devices. Application gap can only be closed by visions powered by brains. Human resource development should be high on the agenda for the leaders of the community for this purpose.

ACKNOWLEDGMENT

The authors would like to thank Mr. H. Zhou and Dr. A. B. R. Kumar for their assistance in the preparation of this paper.

REFERENCES

[1] U.S. Federal Energy Commission, "Final report on 1965 blackout," July 19, 1967.

[2] T. E. Dy-Liacco, "Control centers are here to stay," *IEEE Comput. App. Power,* vol. 15, no. 4, pp. 18–23, Oct. 2002.

[3] F. F. Wu, "Real-time network security monitoring, assessment and optimization," *Elect. Power Energy Syst.,* vol. 10, pp. 83–100, Apr. 1988.

[4] F. F. Wu and R. D. Masiello, Eds., "Computers in power system operation," *Proc. IEEE (Special Issue),* vol. 75, no. 12, Dec. 1987.

[5] T. E. Dy-Liacco, "Modern control centers and computer networking," *IEEE Comput. App. Power,* vol. 7, pp. 17–22, Oct. 1994.

[6] P. Joskow, "Restructuring, competition and regulatory reform in the U.S. electricity sector," *J. Econ. Perspectives,* vol. 11, no. 3, pp. 119–138, 1997.

[7] M. Kezunovic, T. Djoki, and T. Kosti, "Automated monitoring and control using new data integration paradigm," in *Proc. 38th Annu. Hawaii Int. Conf. System Sciences* 2005, p. 66a.

[8] F. Maghsoodlou, R. Masiello, and T. Ray, "Energy management systems," *IEEE Power Energy,* vol. 2, no. 5, pp. 49–57, Sep.–Oct. 2004.

[9] A. F. Vojdani, "Tools for real-time business integration and collaboration," *IEEE Trans. Power Syst.,* vol. 18, pp. 555–562, May 2003.

[10] N. Peterson, T. A. Green, and A. deVos, "Enterprise integration via data federation," presented at the DA/DSM DistribuTECH Europe 99 Conf., Madrid, Spain, 1999.

[11] D. Amor, *The E-Business Revolution*. Upper Saddle River, NJ: Prentice Hall PTR, 2000.

[12] M. Jacke, M. Lenzerini, Y. Vassilion, and P. Vassiliadis, *Fundamentals of Data Warehouses*, 2nd ed. New York: Springer, 1998.

[13] K. Moslehi, A. B. R. Kumar, D. Shurtleff, M. Laufenberg, A. Bose, and P. Hirsch, "Framework for a self-healing power grid," presented at the 2005 IEEE PES General Meeting, San Francisco, CA, 2005.

[14] G. P. Azevedo and A. L. Oliveira Filho, "Control centers with open architectures," *IEEE Comput. App. Power,* vol. 14, no. 4, pp. 27–32, Oct. 2001.

[15] J. Walrand and P. Varaiya, *High-Performance Communication Networks*. San Francisco, CA: Morgan Kaufmann, 1996.

[16] D. J. Marihart, "Communications technology guidelines for EMS/SCADA systems," *IEEE Trans. Power Del.,* vol. 16, no. 2, pp. 181–188, Apr. 2001.

[17] Extensible Markup Language (XML) W3C [Online]. Available: http://www.w3.org/xml

[18] K. Tomsovic, D. Bakken, V. Vankatasubramanian, and A. Bose, "Designing the next generation of real-time control, communication and computations for large power systems," *Proc. IEEE (Special Issue on Energy Infrastructure Systems),* vol. 93, no. 5, pp. 964–965, May 2005.

[19] C. C. Liu, J. Jung, G. T. Heydt, V. Vittal, and A. G. Phadke, "The strategic power infrastructure defiense (SPID) system," *IEEE Control Syst. Mag.,* vol. 20, no. 4, pp. 40–52, Aug. 2000.

[20] A. S. Tanenbaum and M. V. Steen, *Distributed Systems: Principles and Paradigms*. Upper Saddle River, NJ: Prentice-Hall, 2002.

[21] Object FAQ [Online]. Available: http://www.objectfaq.com/oofaq2/body/basics.htm

[22] W. Emmerich and N. Kaveh, "Component technologies: Java Beans, COM, CORBA, RMI, EJB and the component model," in *Proc. 24th Int. Conf. Software Engineering (ICSE)* 2002, pp. 691–692.

[23] W. Emmerich, *Engineering Distributed Objects*. New York: Wiley, 2000.

[24] K. Moslehi, A. B. R. Kumar, E. Dehdashti, P. Hirsch, and W. Wu, "Distributed autonomous real-time system for power system operations—a conceptual overview," presented at the IEEE PES Power System Conf. Exhibition, New York, 2004.

[25] Object Management Group [Online]. Available: http://www.omg.com

[26] M. Klusch, Ed., *Intelligent Information Agents* Berlin, Germany: Springer, 1999.

[27] "Final Report, Common Information Model (CIM): CIM 10 Version," Nov. 2001.

[28] D. Becker, H. Falk, J. Billerman, S. Mauser, R. Podmore, and L. Schneberger, "Standards-based approach integrates utility applications," *IEEE Comput. App. Power,* vol. 14, no. 4, pp. 13–20, Oct. 2000.

[29] J. P. Britton and A. N. deVos, "CIM-based standards and CIM evolution," *IEEE Trans. Power Syst.,* vol. 20, no. 2, pp. 758–764, May 2005.

[30] A. deVos, S. E. Widergren, and J. Zhu, "XML for CIM model exchange," in *IEEE Proc. Power Industry Computer Applications* 2000, pp. 31–37.

[31] *Energy Management System Application Programming Interface (EMS-API), Draft IEC Standard* , IEC 61 970, Oct. 2002 [Online]. Available: ftp://epriapi.kemaconsulting.com/downloads

[32] C. Hoga and G. Wong, "IEC 61 850: open communication in practice in substations," in *Proc. IEEE PES 2004* 2004, vol. 2, pp. 618–623.

[33] L. Murphy and F. F. Wu, "An open design approach for distributed energy management systems," *IEEE Trans. Power Syst.,* vol. 8, no. 3, pp. 1172–1179, Aug. 1993.

[34] G. Daniëls, G. Beilßer, and B. Engel, "New tools to observe and control large networks," presented at the CIGRE Session 2002, Paris, France.

[35] C. H. Hauser, D. E. Bakken, and A. Bose, "A failure to communicate," *IEEE Power Energy,* vol. 3, no. 2, pp. 47–55, Mar.–Apr. 2005.

[36] B. Qiu, Y. L. Liu, and A. G. Phadke, "Communication infrastructure design for strategic power infrastructure defence (SPID) system," in *Proc. IEEE Power Engineering Soc. Winter Meeting* 2002, vol. 1, pp. 672–677.

[37] C. Landwehr, "Computer security," *Int. J. Inf. Security,* vol. 1, pp. 3–13, Aug. 2001.

[38] J. E. Dagle, S. E. Widergren, and J. M. Johnson, "Enhancing the security of supervisory control and data acquisition (SCADA) systems: the lifeblood of modern energy infrastructures," in *Proc. IEEE Power Engineering Soc. Winter Meeting* 2002, vol. 1, p. 635.

[39] H. Hayashi, Y. Takabayashi, H. Tsuji, and M. Oka, "Rapidly increasing application of intranet technologies for SCADA," in *Proc. IEEE T&D Conf. Exhibition: Asia Pacific* 2002, vol. 1, pp. 22–25.

[40] J. Corera, J. Martí, J. Arriola, W. Lex, A. Kuhlmann, and W. Schmitz, "New SCADA/DMS/EMS integrated control system architecture for Iberdrola," in *CIGRE Session 2002* Paris, France.

[41] A. Diu and L. Wehenkel, "EXaMINE—experimentation of a monitoring and control system for managing vulnerabilities of the European infrastructure for electrical power exchange," in *Proc. IEEE Power Engineering Soc. Summer Meeting* 2002, vol. 3, pp. 1410–1415.

[42] X. P. Wu, Y. Zhang, and X. W. Wang, "A new generation EMS," in *IEEE Proc. PowerCon Int. Conf.* 2002, vol. 1, pp. 190–194.
[43] X. B. Qiu and W. Wimmer, "Applying object-orientation and component technology to architecture design of power system monitoring," in *Proc. PowerCon International Conf.* 2000, vol. 2, pp. 589–594.
[44] X. L. Li, M. Y. Gao, J. S. Liu, Z. H. Ding, and X. Z. Duan, "A software architecture for integrative utility management system," in *Proc. IEEE Power Engineering Soc. Winter Meeting* 2001, vol. 2, pp. 476–480.
[45] K. Kawata, T. Yamashita, Y. Takahata, and M. Ueda, "A large-scale distributed control system on multi-vendor platform," in *Proc. IEEE T&D Conf. Exhibition: Asia Pacific* Oct. 2002, vol. 1, pp. 37–42.
[46] X. L. Li, D. Y. Shi, Z. H. Ding, X. Z. Duan, M. Y. Gao, and Y. Z. He, "Study on MAS architecture of EMS," in Chinese, *Autom. Elect. Power Syst.,* vol. 25, pp. 36–40, Jun. 2001.
[47] Y. Q. Yan, W. C. Wu, B. M. Zhang, Z. N. Wang, and C. R. Liu, "Preliminary research and implementation of soft-bus for EMS supporting component interface specification," in Chinese, *Power Syst. Tech.,* vol. 28, pp. 11–16, Oct. 2004.
[48] G. P. Azevedo, B. Feijo, and M. Costa, "Control centers evolve with agent technology," *IEEE Comput. App. Power,* vol. 13, no. 3, pp. 48–53, Jul. 2000.
[49] S. Katayama, T. Tsuchiya, T. Tanaka, R. Tsukui, H. Yusa, and T. Otani, "Distributed real-time computer network architecture: power systems information model coordinated with agent applications," in *Proc. IEEE T&D Conf. Exhibition: Asia Pacific* 2002, vol. 1, pp. 6–11.
[50] Systinet Corp. , "Web Services: A practical introduction," [Online]. Available: http://www.systinet.com
[51] I. Foster, C. Kesselman, J. M. Nick, and S. Tuecke, "Grid services for distributed system integration," *Computer,* vol. 35, pp. 37–46, Jun. 2002.
[52] J. Zhu, "Web services provide the power to integrate," *IEEE Power Energy,* vol. 1, no. 6, pp. 40–49, Nov.–Dec. 2003.
[53] K. Matsumoto, T. Maruo, N. Mori, M. Kitayama, and I. Izui, "A communication network model of electric power trading systems using Web services," presented at the IEEE Proc. Power Tech. Conf., Bologna, Italy, 2003.
[54] Q. Morante, N. Ranaldo, and E. Zimeo, "Web services workflow for power system security assessment," in *Proc. IEEE Int. Conf. e-Technology, e-Commerce and e-Service* 2005, pp. 374–380.
[55] W. Zhang, C. Shen, and Q. Lu, "Framework of the power gird system," in Chinese, *Autom. Elect. Power Syst.,* vol. 28, no. 22, pp. 1–4, Nov. 2004.
[56] I. Foster, C. Kesselman, and S. Tuecke*, The Anatomy of the Grid* [Online]. Available: http://www.globus.org/alliance/publications/papers/anatomy.pdf
[57] M. Parashar and C. A. Lee, Eds., "Special issue on grid computing," *Proc. IEEE,* vol. 93, no. 3, Mar. 2005.
[58] I. Foster and C. Kesselman, Eds.*, The Grid: Bluprint for a New Computing Infrastructure*, 2nd ed. New York: Elsevier, 2005.
[59] Globus Group*, A Globus Toolkit Primer* [Online]. Available: http://www.globus.org
[60] D. Geer, "Taking steps to secure Web services," *Computer,* vol. 36, no. 10, pp. 14–16, Oct. 2003.
[61] M. Humphrey, M. R. Thompson, and K. R. Jackson, "Security for grids," *Proc. IEEE,* vol. 93, no. 3, pp. 644–652, Mar. 2005.
[62] A. G. Phadke, "Synchronized phasor measurements in power systems," *IEEE Comput. App. Power,* vol. 6, no. 2, pp. 10–15, Apr. 1993.
[63] C. W. Liu and J. Thorp, "Application of synchronized phasor measurements to real-time transient stability prediction," *IEE Proc. Gener. Transm. Distrib.,* vol. 142, no. 4, pp. 355–360, Jul. 1995.
[64] C. W. Carson, D. C. Erickson, K. E. Martin, R. E. Wilson, and V. Venkatasubramanian, "WACS-Wide-area stability and voltage control systems: R&D and online demonstration," *Proc. IEEE,* vol. 93, no. 5, pp. 892–906, May 2005.
[65] R. F. Nuqui, A. G. Phadke, R. P. Schulz, and N. Bhatt, "Fast on-line voltage security monitoring using synchronized phasor measurements and decision trees," in *Proc. IEEE Power Engineering Soc. Winter Meeting* 2001, vol. 3, pp. 1347–1352.
[66] M. Larsson and C. Rehtanz, "Predictive frequency stability control based on wide-area phasor measurements," in *Proc. IEEE Power Engineering Soc. Summer Meeting* 2002, vol. 1, pp. 233–238.
[67] T. Overby and J. Weber, "Visualizing the electric grid," *IEEE Spect.,* vol. 38, no. 2, pp. 52–58, Feb. 2001.
[68] G. Krost, T. Papazoglou, Z. Malek, and M. Linders, "Facilitating the operation of large interconnected systems by means of innovative approaches in human-machine interaction," presented at the CIGRE Shanghai Symposium 2003, paper 440-05 [Online]. Available: http://www.cigre.org

Felix F. Wu (Fellow, IEEE) is the Philip Wong Wilson Wong Professor in Electrical Engineering at the University of Hong Kong, Hong Kong, where he served as Pro Vice Chancellor (Vice President) from 1997 to 2001. He is also a Professor Emeritus at the University of California, Berkeley, where he has been on the Faculty since 1974.

Khosrow Moslehi (Member, IEEE) received the Ph.D. degree from the University of California, Berkeley.

He is the Director of Product Development at ABB Network Management/Central Markets, Santa Clara, CA. He has over 20 years of experience in power system analysis, optimization, system integration, and architecture.

Anjan Bose (Fellow, IEEE) is the Dean of the College of Engineering and Distinguished Professor at Washington State University, Pullman. He has over 30 years of industrial and academic experience in power system engineering.

Dr. Bose is a Member of the National Academy of Engineering.