

A Service-Oriented Architecture for Electric Power Transmission System Asset Management*

Jyotishman Pathak¹, Yuan Li², Vasant Honavar¹, and James McCalley²

¹ Department of Computer Science

² Department of Electrical & Computer Engineering

Iowa State University

Ames, IA 50011-1040 USA

{jpathak, tua, honavar, jdm}@iastate.edu

Abstract. In electric power transmission systems, the assets include transmission lines, transformers, power plants and support structures. Maintaining these assets to reliably deliver electric energy at low prices is critical for a nation's growth and development. Towards this end, we describe a novel service-oriented architecture for sensing, information integration, risk assessment, and decision-making tasks that arise in operating modern high-voltage electric power systems. The proposed framework integrates real-time data acquisition, modeling, and forecasting functionalities provided by relatively autonomous, loosely coupled entities that constitute the power industry to determine operational policies, maintenance schedules and facility reinforcement plans required to ensure reliable operation of power systems.

1 Introduction

Modern electric power systems comprising of power transmission and distribution grids consist of a large number of distributed, autonomously managed, capital-intensive assets. Such assets include power plants, transmission lines, transformers, and protection equipment. Over the past 15 years, the investment in acquiring new assets has significantly declined causing many such assets to be operated well beyond their intended life with unavoidable increase in stress on the system. Typically, a single power transmission company has its own centralized control center and is responsible for maintaining different types of equipment. The failure of critical equipment can adversely impact the entire distribution grid and increase the likelihood of additional failures.

Avoiding catastrophic failures and ensuring reliable operation of such a complex network of assets presents several challenges in data-driven decision making related to the operation, maintenance and planning of assets. Specifically, decision-makers must anticipate potential failures before they occur, identify alternative responses or preventive measures, along with their associated costs, benefits and risks. Effective decision-making in such a setting is critically dependent on gathering and use of information characterizing the *conditional*, *operational* and *maintenance* histories of the

* This research is funded in part by the NSF DDDAS-TMRP grant# 0540293 and the ISU Center for CILD (<http://www.cild.iastate.edu>)

assets, e.g., equipment *age* and *time* since the last inspection and maintenance. Recent advances in sensing, communications, and database technologies have made it possible, at least in principle, for decision-makers to access operating/maintenance histories and asset-specific real-time monitoring data, which can be used to ensure reliable and cost-effective operation of modern power systems so as to reduce (if not eliminate) the frequency and severity of catastrophic failures such as blackouts[1].

However, effective acquisition and use of condition data, operating and maintenance histories, and asset-specific real-time monitoring data presents several challenges in practice: (i) the assets that constitute a modern power system are geographically distributed (ii) the data sources differ in terms of data semantics (due to differences in *ontologies*), and spatial and temporal granularity of data (iii) the development of models that use this information to reliably predict the ways the asset may deteriorate and fail (and recommend counter-measures) requires integration of results of several types of analysis.

Service-Oriented Architecture (SOA) [2] and Web services [3] offer a flexible and extensible approach to integration of multiple, often autonomous, data sources and analysis procedures. The recent adoption of Web services in the power industry [4,5,6,7] and the support for interoperability with other frameworks (e.g., SCADA) through the use of emerging Web services standards make SOA an especially attractive framework for designing software infrastructure to address the challenges outlined above. Against this background, we propose a service-oriented software architecture for power systems asset management. The design of this architecture which is being outlined in this paper is currently being implemented.

The rest of the paper is organized as follows: Section 2 presents an overview of the electric power system asset management problem along with our proposed solution; Section 3 discusses the proposed service-oriented architecture for power systems asset management; Section 4 describes some of the modules (services) comprising the proposed system; Section 5 briefly discusses related work; and Section 6 concludes with a summary and an outline of some directions for further research.

2 Electric Power System Asset Management

In electric power systems, asset management decision problems are characterized by: (1) strong interdependencies between physical performance of individual assets, physical performance of the overall system, and economic system performance; (2) limited resources; (3) important uncertainties in individual component performance, system loading conditions, and available resources; (4) multiple objectives. These problems can be classified into one of four types which involve resource allocation with the objective to minimize cost and risk. These specific asset management decision problems include (a) Operations, (b) Short-term maintenance selection and scheduling, (c) Long-term maintenance planning, and (d) Facility planning. The problems differ primarily in their time scale but are linked by a common focus on the interactions between the equipment condition and the decisions taken. The operational decision problem of how to meet demand in the next hour to week treats facilities available and their deterioration levels as given (though the deterioration is not known precisely).

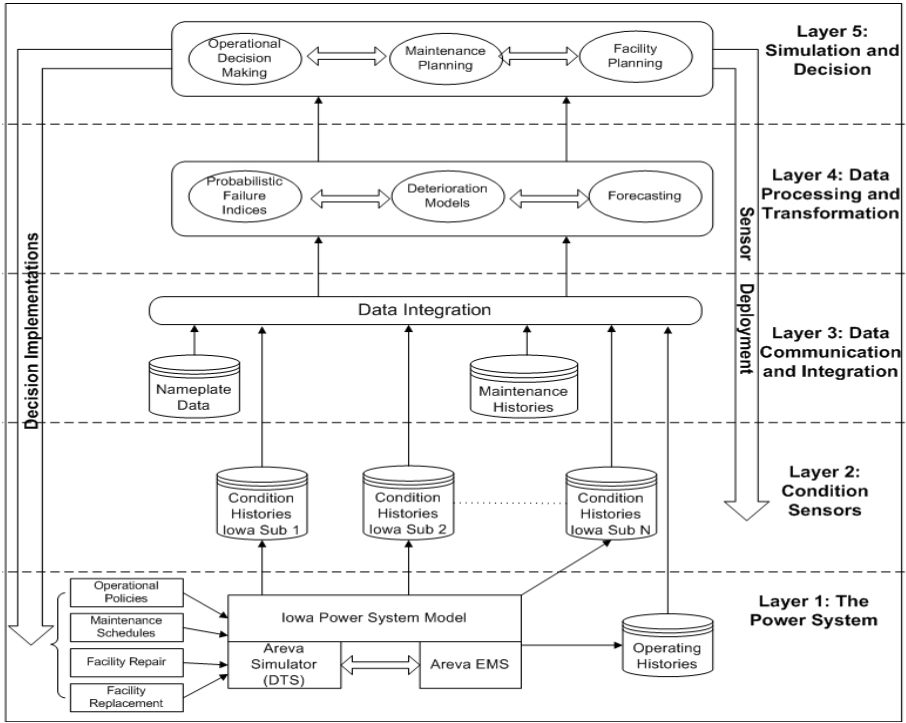


Fig. 1. Structure of the Asset Management Decision Problem [8]

Figure 1 illustrates the structure of the asset management problem and facilitates description of how we intend to address the problem. Essentially, our approach focuses on the use of equipment condition measurements to estimate short-term failure probabilities along with the deterioration effects of loading each piece of equipment at various levels, and the use of such estimates to guide dispatch and unit commitment decisions.

Layer 1, The Power System: This layer consists of a continuously running model of the Iowa power system with network data provided by local utility companies using a commercial-grade (Areva¹) Dispatcher Training Simulator (DTS) and Energy Management Software (EMS). The DTS and EMS were developed to simulate the environment as seen by a control center operator.

Layer 2, Condition Sensors: This layer consists of databases (one for each substation) that capture condition data and operational and maintenance histories of equipment in substations. Different substations may be owned by different utility companies.

Layer 3, Data Communication & Integration: This layer models communication between each substation and the respective substation server (typically through wireless links) together with integration of data. This layer needs to provide dependable, efficient and secure mechanisms for connecting the data sources with analysis mechanisms (Layer 4).

¹ <http://www.areva.com>

Layer 4, Data Processing & Transformation: This layer operates on the integrated data from Layer 3 to produce, for each piece of equipment, an estimate of the short-term probability of failure at any given time. The estimation of such failure probabilities relies on deterioration models (e.g., models [9] of chemical degradation processes in power transformer insulating materials such as oil and cellulose), driven by on-line sensors which measure levels of certain gases in the oil, gases that are produced by these deterioration processes.

Layer 5, Simulation & Decision: This layer utilizes the component probabilistic failure indices from Layer 4 together with short and long-term system forecasts to drive stochastic simulation and decision models. The resulting operational policies, maintenance schedules, and facility reinforcement plans will then be implemented in the power system (as modeled by the Areva simulator).

In what follows, we describe a service-oriented software architecture for power systems asset management that realizes the framework outlined above.

3 SOA-Based Framework for Power System Asset Management

A Service-Oriented Architecture (SOA) is a component model that supports interaction between multiple functional units, called *services*. A service is a software module that has a well-defined interface specifying a set of named operations that the service provides and a set of messages that the service receives/sends, an implementation of the interface, and if deployed, a binding to a documented network address [2]. An SOA can be implemented using several alternative technologies including Web services [3]. A Web service is a service that defines its interface using the Web Services Description Language². Such a service can be accessed using a protocol that is compliant with the Web Services Interoperability³ standards. Web service interfaces are platform and language independent, thereby allowing Web services running on different platforms to interoperate.

Our framework, PSAM-s, for Power System Asset Management shown in Figure 2 employs a Web services-based SOA. The core of the framework⁴ is the PSAM-s engine comprising of multiple services that are responsible for enabling interaction between the users and other services that offer specific functionality (e.g., prediction of power transformer failure-rates). These services can be broadly categorized into *internal* and *external* services.

3.1 PSAM-s Internal Services

These are the services which are part of the PSAM-s engine. They include:

Submission Service. This service is responsible for handling job requests from the user. Such a request would typically initiate the execution of one or more *information processing services* (described in Section 3.2), and the results after the execution will be

² <http://www.w3.org/TR/wsdl>

³ <http://www.ws-i.org>

⁴ In the context of PSAM-s, we use the terms “service” and “Web service” interchangeably.

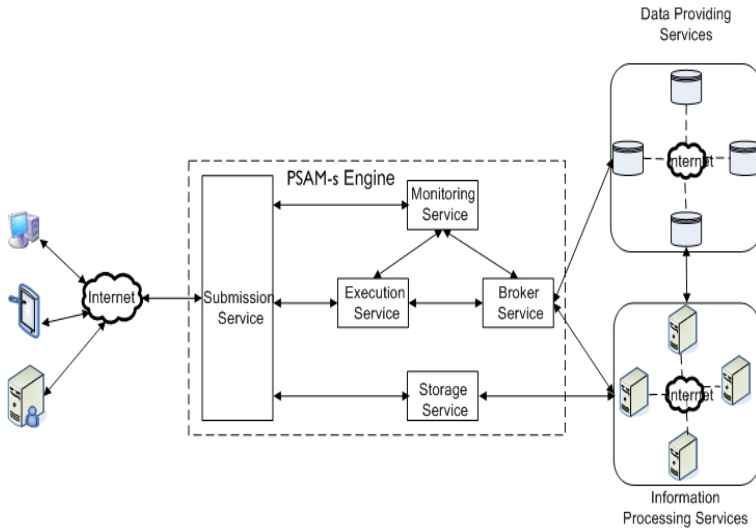


Fig. 2. PSAM-s System Architecture

returned to the user as well as stored in a repository via the *storage service* (described below). The submission service expects the job requests to be defined using the Web Services Business Process Execution Language (WS-BPEL⁵). WS-BPEL specifies a model and a grammar for describing the behavior of a process based on its interactions with other processes (also called partners in WS-BPEL terminology). The interaction with the partners happens through the Web service interfaces and the WS-BPEL process defines the coordination, along with the state and logic necessary for the coordination, to realize a specific goal or requirement. We assume that there exists an “integration specialist” in PSAM-s who is responsible for assembling WS-BPEL process templates/documents for some of the routinely requested user jobs in power system asset management (e.g., failure-rate prediction) that could be used by a common user (e.g., a control center operator). Over time, new templates could be designed or the existing ones modified depending on the user requirements.

Execution Service. Once the user submits a job request (i.e., a WS-BPEL document) to the submission service, the job is sent to the execution service which is responsible for executing the *information processing service(s)* specified in the WS-BPEL document to fulfill the user job requirement. Usually, this document will specify a composition of multiple services whose execution needs to be orchestrated according to a specified control flow (defined as part of the workflow). For example, executing a composite service might involve executing an information processing service that will predict the failure-rate indices for a particular equipment (Layer 4 in Figure 1) which in turn is used by another information processing service to determine if short-term maintenance is required for the equipment under consideration (Layer 5 in Figure 1).

⁵ http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsbpel

Broker Service. The information processing services mentioned above use equipment condition and historical data to determine the information needed to improve maintenance scheduling. To facilitate the access to this data, we introduce a broker service that is responsible for establishing “dynamic data links” between the information processing and the data providing services (described in Section 3.2). The broker service enables the information processing services dynamically access and interact with the data providing services that are online and contain information that is of interest.

The broker service is based on the WS-Brokered Notification (WSN) specification [10]. It implements an event-driven publish/subscribe protocol. Here, an event represents an occurrence or something that has happened, e.g., a database being updated with new data; resulting in event-messages that are produced and consumed explicitly by the applications. The broker service acts as an intermediary which allows potential message publishers and consumers to interact. In our context, the message publishers correspond to the data providing services and the consumers correspond to the information processing services. Essentially, a data providing service sends an one-way notification message to the broker whenever an event occurs (e.g., a database update). These messages are wrapped in the WSN notify message and are associated with a *topic*, which corresponds to a concept (e.g., `DatabaseStatus`) used to categorize kinds of notification. These topics belong to the same topic namespace (a combination of an unique uniform resource identifier and the topic name), which also contains the meta-data associated with the topics. This meta-data includes information about the type or types of messages that the notification publisher will send on a given topic.

The notification consumers (in our case, the information processing services), are registered with the broker service that is capable of distributing the information provided by the notification producers. Thus, the broker acts as a ‘matchmaker service’ and identifies the consumers that have registered for specific types of notifications (or topics) and disseminates the relevant information when available (from the publisher). Since the consumer recognizes the topic and its associated meta-data, it knows how to handle the notification. Thus, depending on the message contained in the notification message, the consumer (information processing services) may or may not interact (or establish a ‘data link’) with the producer (data providing services).

Monitoring Service. Once the job has been submitted, the user can monitor its status via the monitoring service. The idea is to allow users to observe the behavior of the information processing services (during execution) for purposes such as fixing problems or tracking usage. Essentially, the monitoring service maintains an index which automatically registers all the information processing services (notification consumers) that are registered with the broker service mentioned above. These information processing services are WSRF-compliant [11], and thereby readily make their status and state information available as WSRF resource properties (more details in Section 3.2). Whenever the values of these properties change (e.g., the status of a service changing from `idle` to `active`), a notification is pushed via WSN [10] subscription methods to the monitoring service. This information is provided to the user dynamically at run-time.

Storage Service. The results of the computations done by the information processing services are stored in the storage service along with additional meta-data about the

computations themselves (i.e., the workflow-related information stating which services were executed and in what fashion). The computation results are provided to the user, whereas the meta-data is used by the storage service as follows: in many scenarios, the users might be interested in executing a workflow, comprising of a set of information processing services, multiple times in a periodic fashion. Obviously, this is a very compute- and I/O-intensive process. Hence, when a job is first submitted by the user via the submission service, the description of the job is matched with the computational meta-data (previously stored) by the storage service. If there is no match (i.e., this job or workflow has not been executed yet), the job is sent to the execution service as described above. However, if there is a match (i.e., the same job has been executed before), the storage service communicates with the broker service to identify the relevant data providing services that will potentially take part during the execution of the job under consideration, and then analyzes to determine if there has been any change/update to the data represented by these services since the previous execution of the job. If not, then the results from the previous computations are returned to the user, otherwise, computations are executed on this new/updated data. We believe that such an optimization approach will potentially result in saving significant time and computational power, specially for periodically executed jobs. However, in principle, this approach can be substituted by more sophisticated sampling techniques that are used to scale the performance of traditional data-driven decision making algorithms [12].

3.2 PSAM-s External Services

These services interact with the PSAM-s internal services to do useful analysis, and in principle, can be provided by external vendors. These external services include:

Data Providing Service. As mentioned earlier, analysis of equipment-related data play an important role in the decision-making process, and in our framework, this data is provided by the components in layers 2 & 3 (Figure 1). There are at least 4 types of information that is captured in these two layers: *equipment data* consists of “nameplate” information including manufacturer, make, model, rated currents, voltages and powers etc.; *operating histories* refer to the loading and voltage conditions, and through faults, to which the equipment has been subjected in the past; *maintenance histories* records inspections and maintenance activities performed on each piece of equipment; and finally *condition histories* are comprised of measurements providing information about the state of the equipment with respect to one or more failure modes. For example, common condition data information for a transformer includes tests on: oil (dissolved gas, moisture, hydrogen, and furan), power factor, winding resistance, partial discharge (acoustic emissions, spectral decomposition of currents), and infrared emissions.

Except the condition data, all the above data are usually collected manually and recorded in multiple database systems distributed across the substation and corporate headquarters of the utility companies. For our PSAM-s project, we are collaborating with few such companies⁶ across the mid-western US, and Iowa in particular, for accessing these databases. At the same time, for the equipment condition data, we have

⁶ The company names are withheld due to confidentiality issues.

deployed multiple sensors in one of the substation test sites in central Iowa to monitor: (i) anomalous electrical activity within transformer via its terminals, (ii) anomalous chemical changes within the transformer oil, and (iii) anomalous acoustic signals generated by partial discharge events within the transformer. The data collected by the sensors is fed at regular intervals in multiple condition-monitoring databases maintained at our university.

To model the various databases as data providing services, we “wrap” the databases into WSRF-compliant [11] (Web Services Resource Framework) Web services. WSRF provides a generic framework for modeling and accessing persistent resources (e.g., a database) using Web services. WSRF introduces the notion of *Resource Properties* which typically reflect a part of the resource’s state and associated meta-data. For example, one of the resource properties for PSAM-s data providing services is `DBstatus`, which has sub-topics `offline`, `online` and `updated`, that can be assigned a value `true/false`. Whenever there is a change in the value of the resource property⁷, an appropriate notification associated with a particular topic is sent to the broker service mentioned above. This information is appropriately handled by the broker to establish “dynamic data links” between the data providing and information processing services.

Information Processing Service. The information processing services are the most important set of components in PSAM-s as they provide insights into the asset management decision problem. Similar to the data providing services, they are also WSRF-compliant and publish various resource properties (e.g., whether a service is `idle` or `active`) that are monitored by the monitoring service. Furthermore, the dynamic data links (with the data providing services) that are established by the broker service, allow the information processing services to communicate with them in a *federated* fashion [13], where the information needed (by the information processing services) to answer a query is gathered directly from the data sources (represented by the data providing services). There are two advantages of such an approach: (a) the information is always up-to-date with respect to the contents of the data sources at the time the query is posted; (b) the federated approach avoids a single point-of-failure in query answering, as opposed to a centralized architecture (e.g., a data warehouse), where once the central data warehouse fails, no information can be gathered. We divide the set of information processing services modeled in PSAM-s into two categories corresponding to layers 4 & 5 in Figure 1: (i) data transforming services, and (ii) simulation & decision-making services.

The data transforming services (part of Layer 4) interact with the data providing services to gather and utilize equipment condition information collected from inspections, testing and monitoring, as well as maintenance history to estimate probabilities of equipment failure in some specified interval of time [9]. The underlying probabilistic model captures the deterioration in equipment state as influenced by past loading, maintenance and environmental conditions.

The simulation & decision-making services (part of Layer 5) utilize the failure probabilities determined by the data transforming services together with the short and long term system forecasts to drive integrated stochastic simulation and decision models [14,15]. The resulting operational policies, maintenance schedules, and facility

⁷ The topics `offline` and `online` cannot have similar values at the same time.

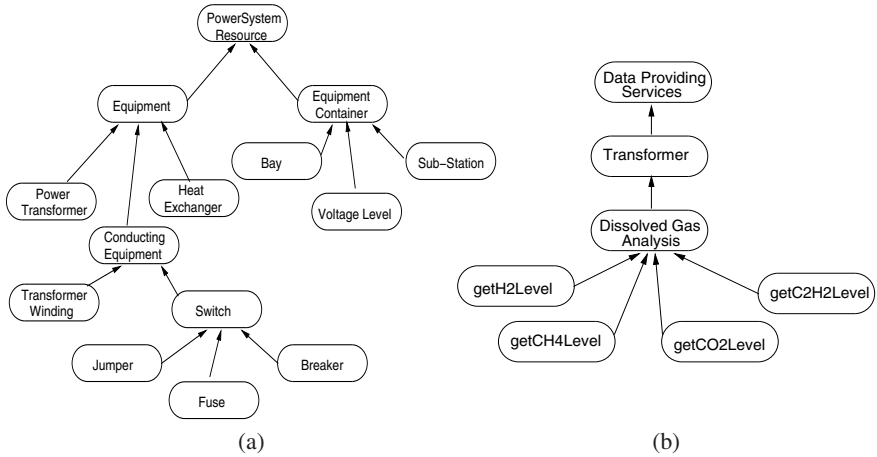


Fig. 3. (a) A Partial Data Ontology (b) A Partial Process Ontology

reinforcement plans are then implemented on the power system (as represented by the Areva simulator in Figure 1). Furthermore, the decision models help discover additional information which drive the deployment of new as well as re-deployment of existing sensors in Layer 2.

3.3 Semantic Interoperability in PSAM-s

As noted earlier, the data providing services in PSAM-s model multiple data repositories (e.g., condition data, maintenance histories) as WSRF-compliant Web services. Typically, these data repositories are autonomously owned and operated by different utility companies. Consequently, the data models that underlie different data sources often differ with respect to the choice of attributes, their values, and relations among attributes (i.e., data source ontologies). Thus, effective use of this data (by the information processing services) requires flexible approaches to bridging the syntactic and semantic mismatches among the data sources. To address this issue in PSAM-s, we model two different types of ontologies: *data ontology* and *process ontology*.

The *data ontology* provides a reference data model that will be used by the software entities and applications, and is based on the Common Information Model (CIM) [16]—a widely used language for enabling semantic interoperability in the electric energy management and distribution domain. The basic CIM data model comprises of twelve different packages structuring the data model and allows representation of various power system related information (e.g., dynamic load data, flow of electricity). In PSAM-s, all the services provide their internal data according to this data ontology and expose CIM-compliant interfaces (based on the *process ontology* described below) thereby allowing multiple services to exchange system data. For example, Figure 3(a) shows a partial data ontology which corresponds to the equipment class hierarchy adopted from the CIM. Each node in this ontology corresponds to an equipment (or an equipment category) and the edges represent sub-class/category relationships. However,

in certain cases, it may not be possible to readily map the system's internal data into the common model. In such cases, we propose to use custom-adapters or mappings [13] for the required translation.

The *process ontology* provides a reference functional model that focuses on the interfaces that the compliant services have to provide. This ontology is also based on CIM and specifies the functionalities that the services must deliver, where the formal definition of those functions is understood in terms of CIM semantics. The process ontology allows us to create a standardized service interface that is insulated from the change in the implementation of the service itself. This provides significant flexibility in terms of system integration and pragmatic advantage in the modeling of existing large systems (e.g., SCADA) as services in PSAM-s, which are predominantly non-CIM at their core. For example, Figure 3(b) shows a partial process ontology corresponding to the data providing services which expose information about transformers, and in particular, the dissolved gas concentrations in the transformer oil. Thus, any data providing service that complies to provide this information must implement an interface that defines functions such as *getH₂Level* and *getCH₄Level* to extract the hydrogen and methane concentrations in the transformer oil, respectively.

4 Implementation Status

We have implemented an early prototype of the proposed PSAM-s framework for the problem of power transformer failure rate estimation based on condition monitoring data [9], a sub-problem of the larger power systems asset management problem (Figure 1). The prototype [9] integrates power transformer data from multiple sources to train Hidden Markov Models [17] for predicting the failure rate of transformer oil deterioration.

A more complete implementation of the PSAM-s framework is currently underway. A major part of this effort lies in modeling the needed data analysis services. We are currently porting our code for transformer failure rate prediction [9] to WSRF-compliant information processing services. We are developing WSRF-compliant data providing service models based on condition, maintenance and operational data repositories for transformers. The development of the data and process ontologies as well as inter-ontology mappings needed for semantic interoperability between multiple services is also underway [13].

5 Related Work

In recent years, service-oriented architectures for solving problems in electric power systems are beginning to receive attention in the literature. Lalanda [4] have proposed an e-Services based software infrastructure for power distribution. This infrastructure is based on the Open Services Gateway Initiative (OSGi) model and comprises of various components that are integrated and correlated to enable effective monitoring and management of electrical equipment and optimization of power usage. Marin et al. [5] have outlined an approach for adopting e-Services for power distribution. Their proposal involves designing a meta model containing the business logic for services-based power distribution which is then used to generate the code needed to realize the model,

thereby providing an increased flexibility in building the underlying software infrastructure. Morante et al. [6] have proposed a framework for power system analysis based on Web and Grid services. Their architecture integrates a set of remotely based controlled units responsible for acquisition of field data and dynamic loading of power equipments, a grid computing based solution engine for on-line contingency analysis, and an interface for reporting the results of analysis. Later, the authors in [6] also present a Web services based framework for power system security assessment [7]. Here, the proposed approach integrates multiple services such as real-time data acquisition, high performance computational and storage services to efficiently perform complex on-line power system security analysis.

Our work on *PSAM-s*, is focused on a flexible, distributed software architecture for determining operational policies, maintenance schedules and facility reinforcement plans for power systems assets management. The framework builds on and extends our previous work [14,15] on the use of transformer condition data to assess probability of failure [9]. A major focus of *PSAM-s* is on integrating disparate information sources and analysis services, drawing on our work on data integration [13] and service composition [18] to advance the current state-of-the-art in electric power system asset management.

6 Summary and Discussion

We have described *PSAM-s*, a service-oriented software architecture for managing operations, maintenance and planning in modern high-voltage electric power transmission systems. The adoption of the service-oriented architecture in *PSAM-s* provides for interoperability of multiple, autonomously operated data sources and analysis services along with the much needed flexibility and agility to respond to changes in the power transmission and distribution network that call for new data sources or analysis services to be incorporated into the system.

We have completed implementation of an initial prototype of the proposed framework (focused on condition assessment and failure prediction of transformers). Work in progress is aimed at a more complete implementation, incorporating additional data sources and services needed to support a more comprehensive approach to power systems assets management.

Some directions for future work include: (a) development of approaches that allow incremental update of analysis results and predictions based on new data (as opposed to calculating the predictions from scratch); (b) better support for service selection and composition, e.g., incorporation of non-functional attributes of the services (such as Quality of Service) during the process of selection and establishment of dynamic data links by the broker service, and (c) performance evaluation of *PSAM-s* under a range of operational scenarios.

References

1. Pourbeik, P., Kundur, P., Taylor, C.: The Anatomy of a Power Grid Blackout - Root Causes and Dynamics of Recent Major Blackouts. *IEEE Power and Energy Magazine* 4(5), 22–29 (2006)

2. Ferguson, D., Stockton, M.: Service-Oriented Architecture: Programming Model and Product Architecture. *IBM Systems Journal* 44(4), 753–780 (2005)
3. Alonso, G., Casati, F., Kuna, H., Machiraju, V.: *Web Services: Concepts, Architectures and Applications*. Springer, Heidelberg (2004)
4. Lalanda, P.: An E-Services Infrastructure for Power Distribution. *IEEE Internet Computing* 9(3), 52–59 (2005)
5. Martin, C., Lalanda, P., Donsez, D.: A MDE Approach for Power Distribution Service Development. In: Benatallah, B., Casati, F., Traverso, P. (eds.) *ICSOC 2005*. LNCS, vol. 3826, pp. 552–557. Springer, Heidelberg (2005)
6. Morante, Q., Vaccaro, A., Villacci, D., Zimeo, E.: A Web based Computational Architecture for Power System Analysis. In: *Bulk Power System Dynamics and Control - VI*, pp. 240–246 (2004)
7. Morante, Q., Ranaldo, N., Zimeo, E.: Web Services Workflow for Power System Security Assessment. In: *IEEE International Conference on e-Technology, e-Commerce and e-Service*, pp. 374–380. IEEE Computer Society Press, Los Alamitos (2005)
8. McCalley, J., Honavar, V., Ryan, S., et al.: Auto-Steered Information-Decision Processes for Electric System Asset Management. In: Alexandrov, V.N., van Albada, G.D., Sloat, P.M.A., Dongarra, J.J. (eds.) *ICCS 2006*. LNCS, vol. 3993, pp. 440–447. Springer, Heidelberg (2006)
9. Pathak, J., Jiang, Y., Honavar, V., McCalley, J.: Condition Data Aggregation with Application to Failure Rate Calculation of Power Transformers. In: *39th Annual Hawaii Intl. Conference on System Sciences*, IEEE Computer Society Press, Los Alamitos (2006)
10. Niblett, P., Graham, S.: Events and Service-Oriented Architecture: The OASIS Web Services Notification Specifications. *IBM Systems Journal* 44(4), 869–886 (2005)
11. Snelling, D.: Web Services Resource Framework: Impact on OGSA and the Grid Computing Roadmap. *GridConnections* 2(1), 1–7 (2004)
12. Ghoting, A., Otey, M.E., Parthasarathy, S.: LOADED: Link-Based Outlier and Anomaly Detection in Evolving Data Sets. In: *4th IEEE International Conference on Data Mining*, pp. 387–390. IEEE Computer Society Press, Los Alamitos (2004)
13. Caragea, D., Zhang, J., Bao, J., Pathak, J., Honavar, V.: Algorithms and Software for Collaborative Discovery from Autonomous, Semantically Heterogeneous, Distributed Information Sources. In: Balcázar, J.L., Long, P.M., Stephan, F. (eds.) *ALT 2006*. LNCS (LNAI), vol. 4264, pp. 13–44. Springer, Heidelberg (2006)
14. Dai, Y., McCalley, J., Vittal, V.: Annual Risk Assessment for Overload Security. *IEEE Transactions on Power Systems* 16(4), 616–623 (2001)
15. Ni, M., McCalley, J., Vittal, V., Greene, S., et al.: Software Implementation of On-Line Risk-based Security Assessment. *IEEE Transactions on Power Systems* 18(3), 1165–1172 (2003)
16. McMorran, A., Ault, G., Morgan, C., Elders, I., McDonald, J.: A Common Information Model (CIM) Toolkit Framework Implemented in Java. *IEEE Transactions on Power Systems* 21(1), 194–201 (2006)
17. Rabiner, L.R., Juang, B.H.: An Introduction to Hidden Markov Models. *IEEE ASSP Magazine* 3(1), 4–15 (1986)
18. Pathak, J., Basu, S., Honavar, V.: Modeling Web Services by Iterative Reformulation of Functional and Non-Functional Requirements. In: Dan, A., Lamersdorf, W. (eds.) *ICSOC 2006*. LNCS, vol. 4294, pp. 314–326. Springer, Heidelberg (2006)