APPLICATION OF ADVANCED WIDE AREA EARLY WARNING SYSTEMS WITH ADAPTIVE PROTECTION

Final Project Technical Report
DOE Award: DE-OE0000120

March 2015

Prime Recipient:
California Institute for Energy & Environment
University of California, Berkeley

Principal Investigator:
Carl Blumstein

Project Manager:
Lloyd Cibulka

Subrecipients/Co-investigators:

Virginia Polytechnic Institute & State University
James Thorp & Virgilio Centeno

Mississippi State University
Roger King & Kari Babski-Reeves

Southern California Edison Co.
Frank Ashrafi

Pacific Gas & Electric Co.
Vahid Madani

DISCLAIMER

ACKNOWLEDGMENTS

This report is based upon work supported by the Department of Energy [National Energy Technology Laboratory] under Award Number DE-OE0000120. The authors wish to acknowledge the leadership and technical guidance provided for this project by the US Department of Energy, particularly Philip Overholt, Manager of the Transmission Reliability R&D Program in the Office of Electricity Delivery & Energy Reliability, and Brian Mollohan, Project Manager, National Energy Technology Laboratory.

The authors also extend special thanks to the California Energy Commission (CEC), Public Interest Energy Research (PIER) Program, for its co-funding of this project, and to the staff of the CEC for their support and technical guidance.

ABSTRACT

Recent blackouts of power systems in North America and throughout the world have shown how critical a reliable power system is to modern societies, and the enormous economic and societal damage a blackout can cause. It has been noted that unanticipated operation of protection systems can contribute to cascading phenomena and, ultimately, blackouts. This project developed and field-tested two methods of Adaptive Protection systems utilizing synchrophasor data. One method detects conditions of system stress that can lead to unintended relay operation, and initiates a supervisory signal to modify relay response in real time to avoid false trips. The second method detects the possibility of false trips of impedance relays as stable system swings "encroach" on the relays' impedance zones, and produces an early warning so that relay engineers can re-evaluate relay settings. In addition, real-time synchrophasor data produced by this project was used to develop advanced visualization techniques for display of synchrophasor data to utility operators and engineers.

EXECUTIVE SUMMARY

INTRODUCTION & OVERVIEW

A reliable and secure electric power system is vital to modern societies, and enormous economic and societal damage can be caused when the power system goes down, i.e., a blackout occurs. Post-mortem studies of blackouts have found that grid protection systems, which are typically set for normal operating conditions, are prone to incorrect operation during times of unusual stress, such as when loadings are extremely high or when unplanned outages of generators or transmission lines have occurred. Such undesirable operations have been cited as an important contributing factor in the sequence of events leading to cascading outages, so called because of the “domino effects” that can occur when a system component is unexpectedly removed from the grid, weakening the surrounding area and leading to further component outages. Therefore, if unintended relay operations can be avoided, blackouts will be less likely to occur.

Protection systems for the electric grid have evolved significantly in recent decades. Up until about the 1980s grid protection consisted of a collection of independent electromechanical relays, which were calibrated for an assumed set of normal conditions, together with power circuit breakers to “trip out” lines and generators to protect them when a fault (short circuit or other system disturbance) unexpectedly occurred. Relays were tasked simply with protecting a specific piece of equipment, and their settings (trip points) were static, i.e., once set they were the same for all operating conditions. The protection engineer used both technical calculations and judgment born of experience to set the relays for the best overall performance. But errors in relay operation, while hopefully minimized, nevertheless sometimes occurred.

Nowadays the electromechanical relays have mostly been supplanted with digital relays, which, being essentially microprocessor-based, are capable of being re-set quickly, and even remotely, if real-time system information is available and the prevailing conditions call for it. Also, given wide-area system information, special relaying schemes can be implemented that can coordinate and optimize relay operations for regional or area-wide objectives that were not possible with the older technologies. But, given that relay operations happen very quickly, sometimes as fast as a few milliseconds (a couple of cycles at the 60 Hz power frequency), it is necessary to time-synchronize the wide-area measurement data in order to overcome the problems of latency in data transmission, thereby ensuring correct analysis, implementation and operation of such a relaying scheme. Fortunately, the technology to achieve this time-synchronization has also become widely and economically available: global positioning systems (GPS). By time-stamping the system measurements via the GPS clock, the resultant data being labeled as “synchrophasor” data to distinguish it from non-time-correlated (“phasor”) measurements, a multitude of advanced applications becomes possible, including adaptive relaying, as the area of protection system technology described above has come to be known. These applications can provide substantial benefits to the electric system and its customers in the form of economical, safe, reliable and resilient electric service.

PROJECT GOALS & OBJECTIVES

The primary goal of this project is to demonstrate, in real-world utility systems and with the participation of practicing utility engineers, three specific high-value applications in adaptive protection technology.

First, synchrophasor data is used as input to an algorithm called Security/Dependability Balance. One of the defining characteristics of modern grid protection systems is redundancy: in order to ensure that when a fault occurs it is dependably cleared, more than one set of independent relays is employed in every relaying application; this guarantees that, if one of the relays fails to operate, the probability that all the relays will fail to operate is extremely small (Dependability). However, every relay system also has a small probability of operating when no fault is present: this is called a “false trip.” Redundant sets of relays will have a correspondingly higher probability of false trips, which fortunately are exceedingly rare, and when system conditions are normal, i.e., the system is “healthy,” false trips are usually just a nuisance and not a significant problem, and the “Security” of the system is considered satisfactory. However, when the system is not healthy, as when several lines or generators have gone out of service for unexpected reasons, or power flows are unusually high, or voltage resources are thin, false tripping needs to be avoided, because it will likely exacerbate the process of system collapse. By using synchrophasor data in an algorithm that is trained to recognize such abnormal system conditions, a supervisory logic can be quickly implemented to require a “voting” scheme for the relays. Simply put, this means that when system conditions are diagnosed as stressed, the supervisory logic requires that at least two of the multiple sets of relays must agree there is a fault before tripping is implemented. Thus, by using the intelligence available from synchrophasors, the balance between Dependability and Security can be maintained even when the system is in a weakened condition. The Midway – Vincent 500 kV transmission line, jointly owned and operated by PG&E and SCE, was chosen as the test bed for this application.

Second, synchrophasor data is used as input to an Impedance Relay Zone Encroachment algorithm. Some relays, called “impedance relays” (impedance is a quantity that combines electrical resistance and inductance), are set to detect conditions when the system impedance they “see” drops into an unacceptably low range, e.g., at a low point of a dynamic oscillation due to a major disturbance, or an unstable steady-state operating condition. The objective is to avoid system instability. Over time, as system conditions such as loading or equipment additions are made, the relay’s setting will no longer correspond to previous conditions, and the relay may begin to operate for stable swings, an undesirable outcome. The Zone Encroachment algorithm sets up a buffer zone around the relay’s normal tripping zone, and when system swings start to encroach on the buffer zone, an alarm and display message is sent to the system operator, who can relay the warnings to the protection engineer, who can then re-evaluate, and if necessary re-program, that relay’s settings to avoid undesired operations. The Midway terminal of the Midway – Vincent 500 kV line was chosen for this application.

And third, in order for utility engineers to absorb synchrophasor data quickly so as to facilitate the required real-time responses, this project will develop methods for display and visualization of protection system data and validate those methods with utility engineers in interactive interviews and workshops, keeping in mind the challenges of information overload that is all too common in utility control rooms these days.

Technical Approach

Each of the three applications of protection systems described above will be demonstrated using a three-part technical process: research and development; pilot demonstration; and field demonstration.

In the R&D phase, university researchers at Virginia Tech will further develop both the Security/Dependability Balance and Zone Encroachment algorithms (adapted from previously developed, non-real-time research versions), with the necessary modifications to allow them to run in real time with streaming synchrophasor data. University researchers at Mississippi State will develop new visualizations of synchrophasor data for protection applications.

In the Pilot Demonstration phase, the Security/Dependability Balance and Zone Encroachment algorithms will be implemented, first, in Virginia Tech's university laboratory using relaying and synchrophasor devices similar to that used by utilities, and then in the proof-of-concept laboratory facilities at the host utilities PG&E and SCE, to verify correct performance. A Data Evaluation Plan will be developed as a protocol for evaluating the performance of the Security/ Dependability Balance algorithm against real data, to be collected in the final, Field Demonstration Phase. The synchrophasor visualizations developed in the R&D phase will be presented to PG&E and SCE engineers and technicians in an interactive workshop setting, to elicit feedback for further improvements.

In the Field Demonstration phase, the two adaptive relaying algorithms will be implemented by the utilities themselves, using the same devices, equipment and systems that they use in operations; with the difference that the systems will be in "monitor" mode, i.e., there will be a period of data collection from the systems, which will be evaluated by the project team according to the Data Evaluation Plan to determine whether the systems will perform as intended. The data visualizations will be modified as needed according to the utility feedback received in the Pilot Demonstration phase, and a second round of interviews and workshops conducted with the revised visualizations to ensure operator and engineer acceptability.

PROJECT RESULTS AND CONCLUSIONS

The R&D phase of the project was completed successfully, and on time. The Security/ Dependability Balance and Impedance Relay Zone Encroachment algorithms were adapted by Virginia Tech from previous research (funded by the California Energy Commission's Public Interest Energy Research (PIER) program), and further developed to use real-time PMU data. Mississippi State developed preliminary data visualizations based on state-of-the-art Cognitive Task Analysis (CTA) methodologies, and the Principal Investigator traveled to SCE and PG&E for a number of operator and engineer interviews and workshops to present the initial data visualizations and collect a substantial amount of comments and feedback for further improvements. A Phase I Technical Report was submitted to DOE, detailing the work in this project phase [Centeno et al., December 2010].

The Pilot Demonstration phase of the project was also completed successfully, and on time. Virginia Tech implemented the two adaptive relaying algorithms in a laboratory environment, using industry-standard software and hardware and simulated PMU data, replicating actual utility system performance to the extent possible, and validating the performance of the algorithms. Utility participants from PG&E and SCE traveled to Virginia Tech to observe a demonstration of the pilot systems, and to discuss the differences in hardware and software that would be needed for the utility pilot demonstrations. The Virginia Tech graduate student who worked on the adaptive relaying development traveled to California for a summer to advise and assist the utilities with their pilot installations, which were successful and had very few, and minor, implementation issues. Mississippi State researchers used the results of the user feedback obtained in Phase I to develop advanced visualizations, using simulated PMU data to evaluate the revised designs against the updated performance criteria. Project accomplishments for this phase were detailed in a Phase II Technical Report submitted to DOE [Centeno et al., December 2011].

The Field Demonstration phase of the Security/Dependability Balance implementation, unfortunately, ran into unexpected issues and delays. The chief issue encountered was one of logistics: because of limited personnel and contracting resources available at the utilities, it took much longer than planned to schedule the resources necessary for the installations. Given that this was a research project, utility operations and infrastructure projects necessary for scheduled system enhancements took precedence in the allocation of personnel time. The adaptive relaying systems also needed to utilize the existing utility communications infrastructure, and additional delays were caused by the scheduling difficulties resulting from coordinating with routine operations and system outages, where necessary to install and calibrate equipment. In retrospect, it is clear that installation of such a system into existing operations is not a trivial task, and would best be accomplished where the application is considered and planned along with other system enhancements. It would also have been much more straightforward if the demonstration had been done on a transmission line owned by a single utility, thereby simplifying the amount of logistical coordination that was required.

From a technical standpoint, the field implementation of the Security/Dependability Balance system was, eventually, a success: both PG&E and SCE were able to set up and commission their respective systems, even though there were some differences in the specifics of the hardware and software used by the two utilities in their respective implementations of the system.

But, as a result of the scheduling difficulties, an additional two years' time extension to the project's original three-year plan was required. By the time the field data collection period began, significant grid enhancements had been made to the Western transmission network. In fact, many of these system enhancements were most likely responsible for the lack of resources to perform the research implementation, as capital projects had first priority on the utilities' budgets and personnel resources, and the research project had secondary priority. The net effect was that by the time the Security/Dependability Balance system was implemented, the Western grid was stronger and more stable than before, and periods of system stress were very few. Other contributing factors were relatively mild West Coast weather with line outages being less common, transient and dynamic events were rare and relatively mild, and overall system loading and power transfers were lower than in recent years (a combination of low hydro in the Pacific Northwest and lower loads in Southern California). As a result, very little useful event data was collected for validating the field performance of the voting scheme, although the limited data that were collected were positive in that regard.

In terms of lessons learned, it should be noted that the Security/Dependability Balance algorithm needs to be "trained" with system studies based on the expected conditions in real-life operation. In this case, Heavy Summer and Heavy Winter load flow cases were provided by SCE and PG&E at the start of the project (October 2009), and the algorithm developed from those cases. (Heavy Summer and Heavy Winter conditions are defined by specific time periods during the year to which they apply; the algorithm is not operational outside those specified times.) By the time the system was implemented in the field in late 2013, both base cases were in need of updating, as system configuration had changed due to additions and retirements of various transmission lines, generators, etc., and generation and load patterns had changed as well. Time and budget did not allow for another round of algorithm development in this research project, but in actual utility practice these base cases and the corresponding algorithm revisions would be routinely updated by the utility's planning engineers. It is the judgment of the project team, which included pragmatic utility engineers, that the cost and effort to implement the adaptive relaying algorithms is reasonable, and commensurate with other, similar types of applications in the operating environment.

The field implementation of the Impedance Relay Zone Encroachment algorithm encountered a different issue: R&D studies had pinpointed a 230 kV transmission line in PG&E's northern area as the best candidate for testing the system, in terms of the possibility of seeing potential zone encroachment. However, system studies showed that synchrophasor data would be needed at locations where no synchrophasors were currently installed, and surrogate PMUs were not deemed to be suitable replacements. Therefore, another line would have to be used, where synchrophasor data was available. The project team concluded that the Midway – Vincent 500 kV line would provide an adequate substitute; while there was little hope for zone encroachment, still, the implementation of the algorithm and the basic performance of the system could be demonstrated. The system was shown to be accurate, relatively straightforward in implementation, and could be used on any impedance relay for which PMU data were available.

Mississippi State was able to complete their visualization work as planned. A second round of interviews and workshops was conducted with the utilities to validate the revised visualizations and solicit further feedback. The resultant Protection Information Tool (PIT) was enthusiastically received by the utilities, whose personnel contributed substantially to its usefulness.

RECOMMENDATIONS FOR FURTHER RESEARCH

Both utilities that participated in this project, PG&E and SCE, recommend that the Security/Dependability system undergo further field validation for an additional year, so that more data can be gathered for analysis. This can be easily done by leaving the system in place and operating, continuing to monitor the system, and performing data evaluation as the data become available. Both utilities have engineering personnel that have the expertise to perform this work, it does not require significant effort, and they are willing to perform this work without any additional outside funding, as it falls within their line operations. Plus, both utilities are leaders in the field of synchrophasor technology implementation, and have a stake in definitively validating its performance and furthering the technology. Using the additional data from the extended field monitoring period, the utilities can re-assess the system models and associated decision tress and updated where required, with the results being instructive to utility engineers regarding how often this process should done in order to maintain sufficient relaying system accuracy.

The Impedance Relay Zone Encroachment system was successfully demonstrated to DOE and other interested parties at PG&E in the System Control Center in San Francisco in September of 2013. Using both existing PMU data and data from future PMU installations, the system can be implemented on any transmission lines of interest with a modest amount of additional programming. Very little, if any, additional research is considered necessary by PG&E or SCE in order to use the system as need dictates.

Another adaptive relaying algorithm, Generator Out-of-step Relaying, was developed at Virginia Tech in the CEC-funded precursor research [Centeno et al., September 2010]. This relaying application was considered by the CEC advisory groups to be the next-most important adaptive relaying application after the two addressed in this project, and a field trial could be undertaken, using an approach similar to the one used in this project.

Visualizations of synchrophasor data, especially for protection system data, were developed with the participation and technical guidance of the host utilities (PG&E and SCE), and conducted according to industry standards for such developments. The resulting displays represent an advance in the technology beyond the pure research phase, and closer to actual synchrophasor information representation that will enable enhanced operator decision-making in the control room. Visualizations of synchrophasor data are critical to many real-time applications, such as the Impedance Zone Encroachment system, and should be a critical part of their development and demonstration. The next step in research and development is for utilities to install these visualizations in their control room environment for hands-on experience; actual in-service use is necessary for optimization of the technology. Additional programming and adaptation of the PIT needs to be done by the utilities themselves, because of the custom engineering necessary to address the different hardware, software and data communications systems implemented by the different utilities.

Finally, this project represents an effort to reduce to practice just a few of the many potentially beneficial synchrophasor-based applications identified in the Kema "Phasor Business Case" Study [Novosel et al., 2007], and more such applications are likely to emerge in the future. Some, like the aforementioned Generator Out-of-step Relaying, are applications in the field of Adaptive Protection, which can benefit greatly from the accurate, high-resolution and time-synchronized data provided by PMUs. But there are a number of other applications in other areas of grid operations and planning that would also be enabled or greatly enhanced by synchrophasor data, and research efforts to reduce these promising applications to utility practice are urgently needed and strongly recommended.

BENEFITS TO THE PUBLIC

The primary benefit to the public from widespread use of the Security/Dependability Balance and Impedance Relay Zone Encroachment technologies is a more reliable and secure electric grid, one that suffers from fewer blackouts, and therefore fewer customer outages. Residential customers will experience less inconvenience from lack of electric service; commercial and industrial customers will experience significantly lower costs in terms of production not lost and personnel not idled. Electric rates will also be lower than would otherwise be the case.

This project was also, arguably, the first real-world utility demonstration of the use of synchrophasor data in advanced protective relaying applications, one that should inspire other researchers and utilities to implement the applications described here as well as others. The Phasor Business Case Study [Novosel et al., 2007] outlined numerous other potential applications that would benefit from synchrophasors, not just in the relaying area (Out of Step Relaying, Generator Protection, etc.) but in six other major areas of the utility business: Real-Time Monitoring and Control, System Benchmarking and Model Validation, Post-Disturbance Analysis, Power System Restoration, Protection and Control of Distributed Generation, and Overload Monitoring and Dynamic Rating of Transmission Lines. The use of synchrophasor data either enables or enhances applications in these areas, all of which contribute to improved power system reliability, security and economics.

Many utilities are currently in the process of procuring and installing synchrophasor infrastructure, studying the feasible business applications germane to their respective utilities, and estimating the economic benefits to be realized therefrom. This project demonstrated an economical and feasible approach to realizing the benefits of just three of the many applications that are enabled by synchrophasor data. It is hoped that the results described in this report will encourage similar efforts by others.

The research conducted in this project has resulted in a number of technical papers extending the project results into further investigations of the topics addressed in this report, as well as related areas of endeavor. These references can be found in the final section of this report, titled "Publications."

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1: INTRODUCTION

1.1 Background & Overview

Recent blackouts of power systems in North America and several other systems throughout the world have shown how critical a reliable power system is to modern societies, and the enormous economic and societal damage a blackout can cause. It has been noted that, during the cascading phenomena that can lead to blackouts, some protection systems may operate in an unanticipated fashion, and such undesirable operations are often an important contributing factor in the sequence of events leading to cascading outages [US DOE, August 1996; WSCC, October 1996; U.S.-Canada, April 2004]. Considering the very large number of relaying systems in existence (by a rough estimate over 5 million on the North American power grid), it is to be expected that some of these unanticipated operations are due to defective or improperly set relays. This has been documented as "the hidden-failure phenomenon in protection systems," and has been recently investigated by the present researchers. Other contributing factors to catastrophic failures are unexpected power system configurations which have not been foreseen when protection systems were set, errors in setting and calibration of relays, or undiscovered design flaws in the protection systems.

In recent years, innovations in the use of wide-area synchrophasor measurements of grid operational behavior, power system protection and power system operations have made possible new proactive approaches to the supervision of protection systems so that the likelihood of catastrophic failures of the power grid are significantly reduced, the regions of the power system affected by such events are limited, and the power system restoration process can be speeded up.

A 3-year project funded in 2006 by the California Energy Commission (CEC) through the Public Interest Energy Research (PIER) program has resulted in research indicating that the use of wide area synchrophasor measurements can be of significant value to reduce the likelihood of false trips by protection systems and reduce the likelihood of contributing to a cascading blackout [Centeno et al., Sept. 2010]. Synchrophasors, also sometimes referred to as phasor measurement units, or PMUs, are high-resolution monitoring devices that are time-synchronized to a master time source via GPS signals, so that the measurement data are time-stamped and can be correlated with other synchrophasor data over a wide area, enabling detailed analysis of events in quasi-real-time. The very positive results obtained to date have resulted in the selection of two specific synchrophasor-enabled applications for large-scale demonstration in this project.

The first application to be demonstrated is a supervisory relay scheme that is activated for specific conditions of system stress, as determined by synchrophasor measurements as input to a "voting scheme" implemented in the programming logic of a protective relaying system. The goal is to better balance the tradeoff between security (equipment protection) and dependability (grid service continuity) under conditions of system stress. This process, also referred to as "adaptive relaying," seeks to minimize the likelihood of false trips when such relay operations would exacerbate conditions contributing to cascading outages.

The second application is a monitoring and alarming application for distance relays. The optimum settings for such relays tend to evolve over time as the grid structure evolves. The purpose of this application is to monitor the impedance levels sensed by these relays and to set an alarm when they begin to encroach on the relay trip characteristics. The intended result would be an investigation into, and possible supervision of, relay settings by protection engineers.

It should be emphasized that simply presenting utility engineers and operators with data is not sufficient to address the variety of problems in power systems today. Information is preferable to data, and a component of this project plan is to take a step in this direction by presenting the information from the demonstration in the form of a third synchrophasor-enabled application, the Protection Information Tool (PIT). Information services are focused on providing the right information at the right moment to the right decision maker – human or machine. High-level operational information services (i.e., actionable intelligence) are often needed along with supportive sensor data or trends to provide context. The information services required by grid operators could vary from scenario development to estimates of socio-economic impacts of failures to quantitative statistics, trends and forecasts. These services also must be available in a geospatial context and at various temporal scales to support the needs of system operators, planners, and regulatory agencies. Information services must be characterized by a strong integration of grid data with ancillary data and information, which will require a knowledge-based approach for capturing the best practices of utilities and regulators.

1.2 PROJECT OBJECTIVES

The overall project objectives were to develop and demonstrate three synchrophasor-based methodologies for improved operation of protection systems:

a) Security-Dependability Adaptive Voting System: An adaptive protection system that will alter relay characteristics to adjust the security/dependability balance in response to changing power system conditions to reduce the likelihood of cascading outages in a stressed power system, as determined by real-time phasor measurements.
b) Alarms for Power Swing Encroachment on Relay Characteristics: The use of synchrophasor measurements in monitoring and reporting on encroachment on the operating zones of relays by non-fault events leading to inappropriate relay trips and a cascading process.
c) Visualization: A protection information tool to assist operators in the interpretation of synchrophasor data.

1.3 TECHNICAL APPROACH

The project was structured in three "phases": Research & Development, Prototype Testing, and Field Demonstration.

1.3.1 Phase I: Research and Development

The R&D phase of the project leveraged previous synchrophasor research to adapt theoretical research results to actual real-time synchrophasor applications, and comprised the following three technical tasks:

Algorithm Development for Adaptive Relay – The goal of this effort was to determine the analytical methods by which wide-area synchrophasor measurements can better balance the tradeoff between security (avoiding false trips) and dependability (reliably clearing actual faults) under conditions of system stress (high loading, component outages, etc.).

Algorithm Development for Encroachment Monitoring and Alarm – Determine the analytical methods by which wide-area measurements can be used to monitor system impedance conditions and provide an alarm for operators when stable system swings are beginning to encroach on the operating limits of impedance relays.

Protection Information Tool (PIT) Development – Apply web-based tools to an existing information system platform to develop visualizations of synchrophasor data.

1.3.2 Phase II – Prototype Testing

The prototype testing phase of the project comprised the following three technical tasks:

University Prototype Testing – Implementation and testing of the adaptive relay and encroachment monitoring algorithms developed in Phase I, using simulated PMU data and a PDC similar to one expected to be utilized in the field tests, in a laboratory environment.

Utility Prototype Testing – Implementation and proof-of-concept testing of the adaptive relay and encroachment monitoring algorithms developed in Phase I at the host utilities' laboratory facilities, using actual equipment closely replicating, to the extent possible, the actual anticipated field installations.

Protection Information Tool Evaluation – Synchrophasor visualizations developed in Phase I will be evaluated by utility engineers and technicians in a series of interviews and workshops; feedback from these activities will be used to further develop and refine the PIT visualizations.

1.3.3 Phase III – Field Demonstration

The field demonstration phase of the project comprised the following three technical tasks:

Field Installation – The host utilities will implement the Security/Dependability and Impedance Zone Encroachment algorithms as actual operating systems, installed in monitor mode, using the necessary synchrophasor devices and establishing the necessary communication links. Existing PMUs at utility substations will communicate with Phasor Data Concentrators, and data will be archived at the power companies' respective repositories.

Data Evaluation – Data collected will be evaluated to assess the performance of both protection system algorithms.

Protection Information Tool Evaluation – A second round of interviews and workshops with utility engineers and technicians will be conducted. The PIT will be reviewed and analyzed to make improvements to the PIT as needed based on the input and feedback from these activities.

CHAPTER 2: ADAPTIVE SECURITY/DEPENDABILITY BALANCE

2.1 ALGORITHM DEVELOPMENT FOR SECURITY/DEPENDABILITY BALANCE SYSTEM

2.1.1 Introduction

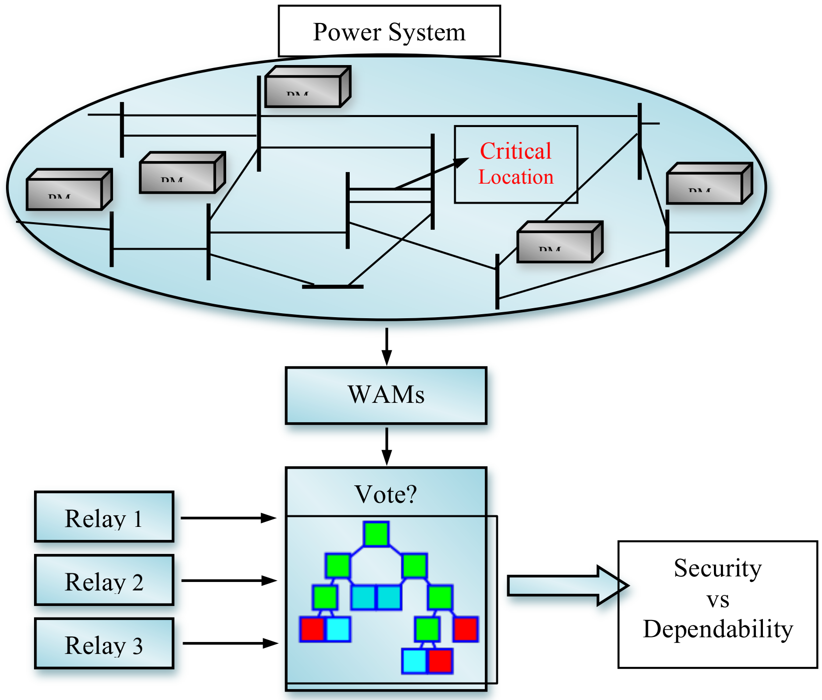
From past experiences of electric system blackouts, it has been noted that during the cascading electric system failures that lead to blackouts, some protection systems have operated in an undesirable fashion, which often were considered as major contributing factors in the sequence of events leading to cascading outages. Causes of these undesirable actions range from defective or improperly set relays (hidden-failures), configuration changes in power system, errors in setting and calibration of relays, and design flaws. In general, protection systems are designed to high reliability ("dependability") at the cost of somewhat increased probability of false trips (slightly reduced "security") [Horowitz et al., 2008]. Dependability can be defined as "the degree of certainty that a relay or relay system will operate correctly." Security, on the other hand, relates to the "degree of certainty that a relay or relay system will not operate incorrectly." In general, enhancing security implies an intrinsic loss of dependability and vice-versa, a classic case of engineering trade-offs. Protection engineers try to achieve an optimal balance between these two conflicting concepts, using experience and engineering judgment; this is why power system protection is often recognized as somewhat of an art rather than as a strict science.

To achieve high dependability, particularly in bulk power transmission systems, multiple (redundant) primary and backup relays are installed to protect a line. With this bias towards dependability, a fault will be cleared with a very high degree of probability. This is a desirable bias when the power system is in a normal (non-alert) state. However, when the system is in an alert (stressed) state, this is an unacceptable bias: a false trip is likely to cause greater damage to the system [Ibid.]. Therefore, it is desirable to alter the bias of the protection system in favor of increased security (with some negligible decrease in dependability) during stressed system conditions. The benefit is in avoiding cascading outages and improving the overall reliability and security of the system.

System topology and good stability margins justified such design since adequate transmission line redundancy entails a variety of alternative paths for power to flow. But it can be argued that due to the manner in which power systems have evolved, this philosophy needs to be reviewed and that, ***under stressed system conditions,*** a favorable bias towards security can be beneficial. An attractive solution to this problem is to "adapt" the security-dependability balance in response to changing system conditions. With the advancement of synchrophasor technology, the system conditions can readily be determined by real-time phasor measurements [Ibid.].

The methodology proposed for this project aims to reduce the likelihood of hidden failures and potential cascading events by adjusting the security/dependability balance of protection systems. The basic concept of the security/dependability adaptive voting scheme was developed in a previous CEC/PIER-funded project [Centeno et al, Sept. 2010]. Figure 1 provides an overview of the proposed system architecture for this project.

Figure 1. Conceptual Schematic of Adaptive Security/Dependability Voting Scheme



The voting scheme supervises the operation of a set of three independent and redundant relays for the protection of a transmission line. Wide area measurements are obtained with the aid of PMUs. The underlying hypothesis is that phasor measurements at strategic buses provide enough information to discriminate the need for a bias towards security. These measurements are used to infer the state of the power system, which is then classified as either "stressed" or "safe." If the system is determined to be in a stressed state, the proper course of action is to enable the voting scheme and therefore bias the protection system towards security. On the other hand, if the system is found to be safe, the voting scheme is disabled and any one relay can trip the breaker, i.e., a favorable bias towards dependability.

2.1.2 Results from Previous Adaptive Relaying Development Project

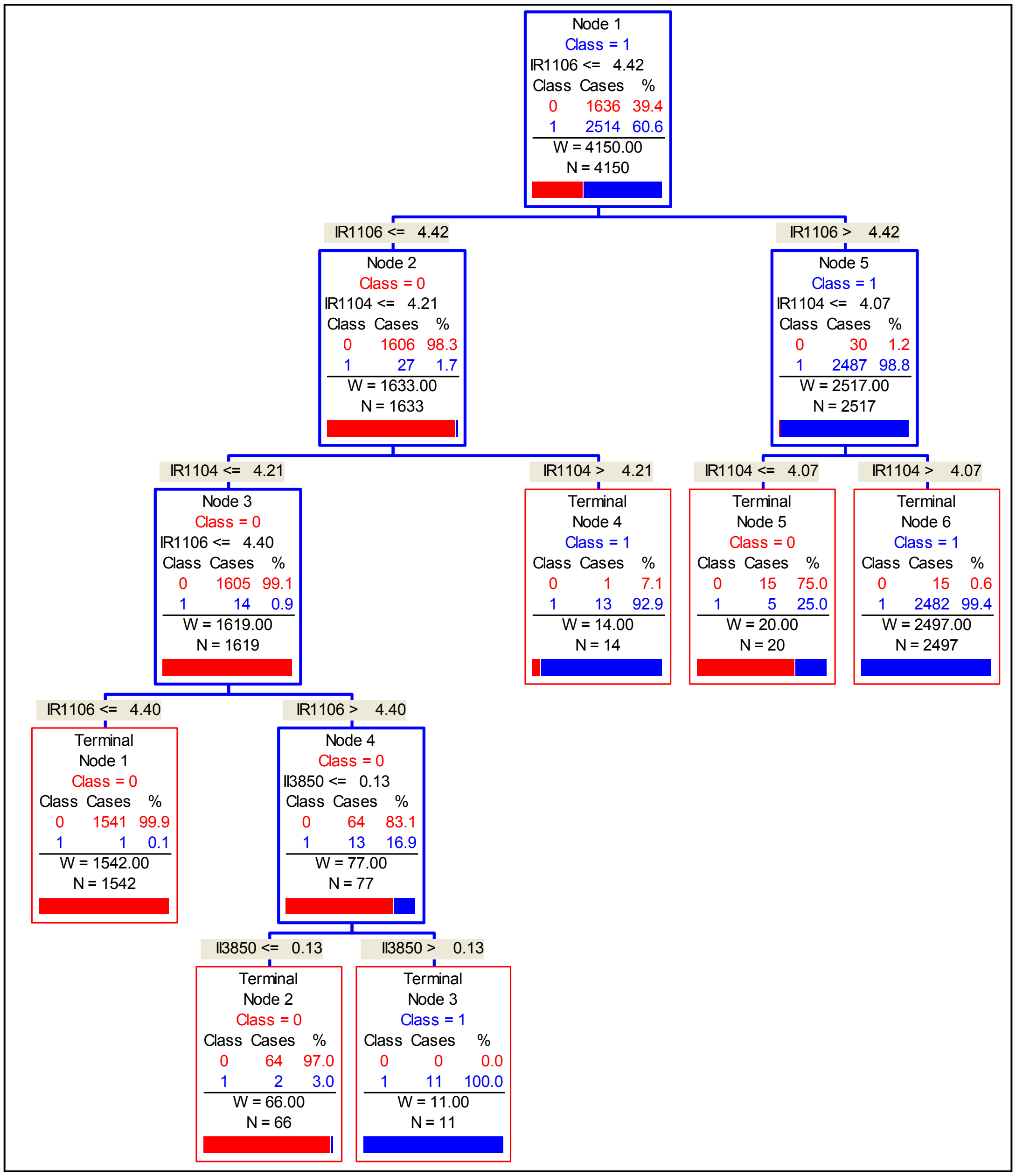
In the advocated methodology, decision trees are trained off-line to be used as an on-line application. An accurate model of the California power system was developed under the previous CIEE Project for this application [Centeno et al., Sept. 2010]. The methodology proposed for this project was tested using two seasonal models of the power system of California: heavy winter and heavy summer. PMU placement for each seasonal model was determined and the recommended PMU placement was based on the operation of the two seasonal models.

2.1.3 PMU Placement for Heavy Winter Model: Primary Splits and Surrogates

Splitting nodes of the Heavy Winter decision tree (Figure 2) indicates the desired location of the PMUs. Table 1 summarizes the attributes used to partition the sample space. PMUs are required at the following locations:

- Los Banos (current flows through two different transmission lines are measured)
- Devers
- Pittsburg (system reference)

Figure 2. Heavy Winter Decision Tree



To increase robustness and reliability, additional PMUs may be deployed to measure surrogate attributes. The objective of a surrogate is to maximize the predictive association with the primary split. Surrogates attempt to mimic the partition achieved by the primary split and are therefore handy in cases where the information of the primary split is missing, due to failure in the communication link, PMU malfunction, etc. Table 1 shows the best surrogate of each primary split. The higher the predictive association, the better the surrogate mimics the primary split. The predictive association measures given in Table 2 indicate how well the surrogate mimics the primary split.

Table 1. Splitting Attributes of the Heavy Winter Decision Tree

Attribute	PMU measurement
Ir1106	Real Current: Tesla – Los Banos
Ir1104	Real Current: Tracy – Los Banos
Ii3850	Imaginary Current: Palo Verde – Devers

Table 2. List of Surrogate PMUs

Node	Primary Split	Surrogate	Predictive Association
1	Ir1106 ≤ 4.42	Ir1104 ≤ 4.16 (Tesla – Los Banos)	0.93
2	Ir1104 ≤ 4.21	Angle Round MT ≤ 16.88	0.64
3	Ir1104 ≤ 4.07	Ii1115 ≤ -2.02 (Gates – Diablo)	0.75
4	Ir1106 ≤ 4.4	Ir1104 ≤ 4.15	0.52
9	Ii3850 ≤ 0.13	Ir87 ≤ 5.04 (Victorville – McCullough)	0.55

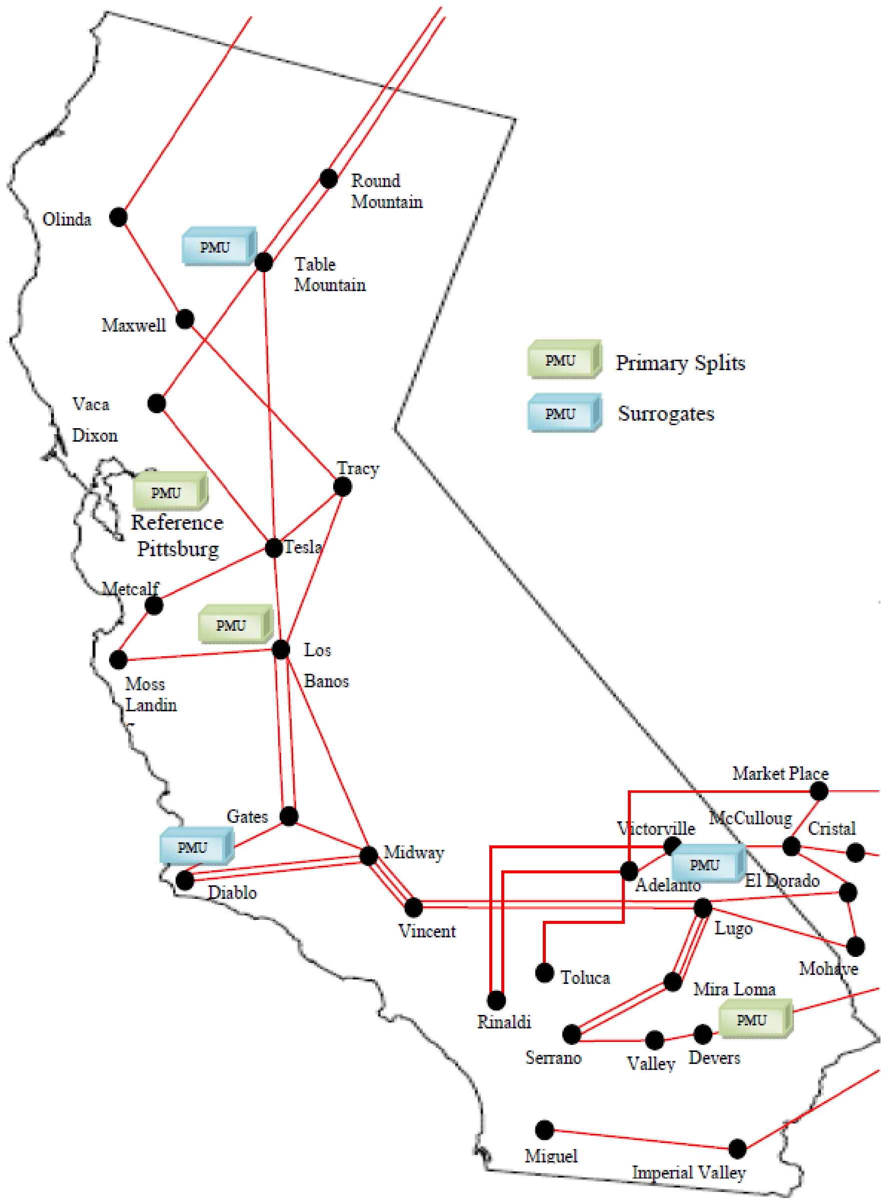
The schematic shown in Figure 3 depicts the final PMU placement. Primary splits are shown in green and surrogates in blue. As expected, a wide area perspective of the system is needed for an optimal performance of the decision tree.

In order to test the performance of the decision tree with out-of-sample data, additional test cases were created by simulating circuit element outages. The objective was to induce additional system operating points to assess the robustness of the tree to topology changes. The out-of-sample data consists of 660 system operating conditions obtained by simulating outages in:

- Generators delivering more than 200 MW
- Loads consuming more than 200 MW
- Transmission lines: 230 kV and 500 kV

These outages were simulated under diverse loading conditions. The results of the tests are summarized in Tables 3–6. Out of the 660 cases, 14 cases were misclassified by the decision tree, an error rate of approximately 2%. Out of those 14 cases, only 2 “stressed” states were misclassified as class zero. These results show an outstanding performance of the decision tree. As stated previously, if the system undergoes significant departures from the model assumptions, a new decision tree should be trained. The proposed out-of-sample tests only attempt to assess tree robustness under small departures.

Figure 3. PMU Placement for Heavy Winter Decision Tree



(PMUs in green represent primary splits. PMUs in blue represent surrogates.)

Table 3. Out-of-sample Test: Generator Outage

	Classified class 0	Classified class 1
True class: 0	30	5
True class: 1	0	45

Table 4. Out-of-sample Test: Load Outage

	Classified class 0	Classified class 1
True class: 0	117	1
True class: 1	0	50

Table 5. Out-of-sample Test: 230 kV Line Outage

	Classified class 0	Classified class 1
True class: 0	132	0
True class: 1	0	132

Table 6. Out-of-sample Test: 500 kV Line Outage

	Classified class 0	Classified class 1
True class: 0	62	6
True class: 1	2	78

2.1.4 PMU Placement: Heavy Summer Model

Splitting nodes of the Heavy Summer Decision tree (Figure 4) indicates the desired locations of the PMUs. Table 7 summarizes the attributes used to partition the sample space. PMUs are required at the following locations:

- Devers (current flows are measured through two different transmission lines)
- El Dorado
- Pittsburg (system reference)

To increase robustness and reliability, additional PMUs may be deployed to measure surrogate attributes. The objective of a surrogate is to mimic the partition achieved by the primary split. Table 8 shows the best surrogate of each primary split. The higher the predictive association, the better the surrogate mimics the primary split. The schematic shown in Figure 5 depicts the PMU placement. Primary splits are shown in green and surrogates in blue.

In order to test the robustness of the decision tree for small departures, test cases (not included in the learning sample) are created by simulating outages. As stated previously, the objective is to assess robustness against topology changes. The out-of-sample data consists of 1138 system operating conditions obtained by simulating outages in:

- Generators delivering more than 200 MW
- Loads consuming more than 200 MW
- Transmission lines: 230 kV and 500 kV

Figure 4. Heavy Summer Detailed Decision Tree

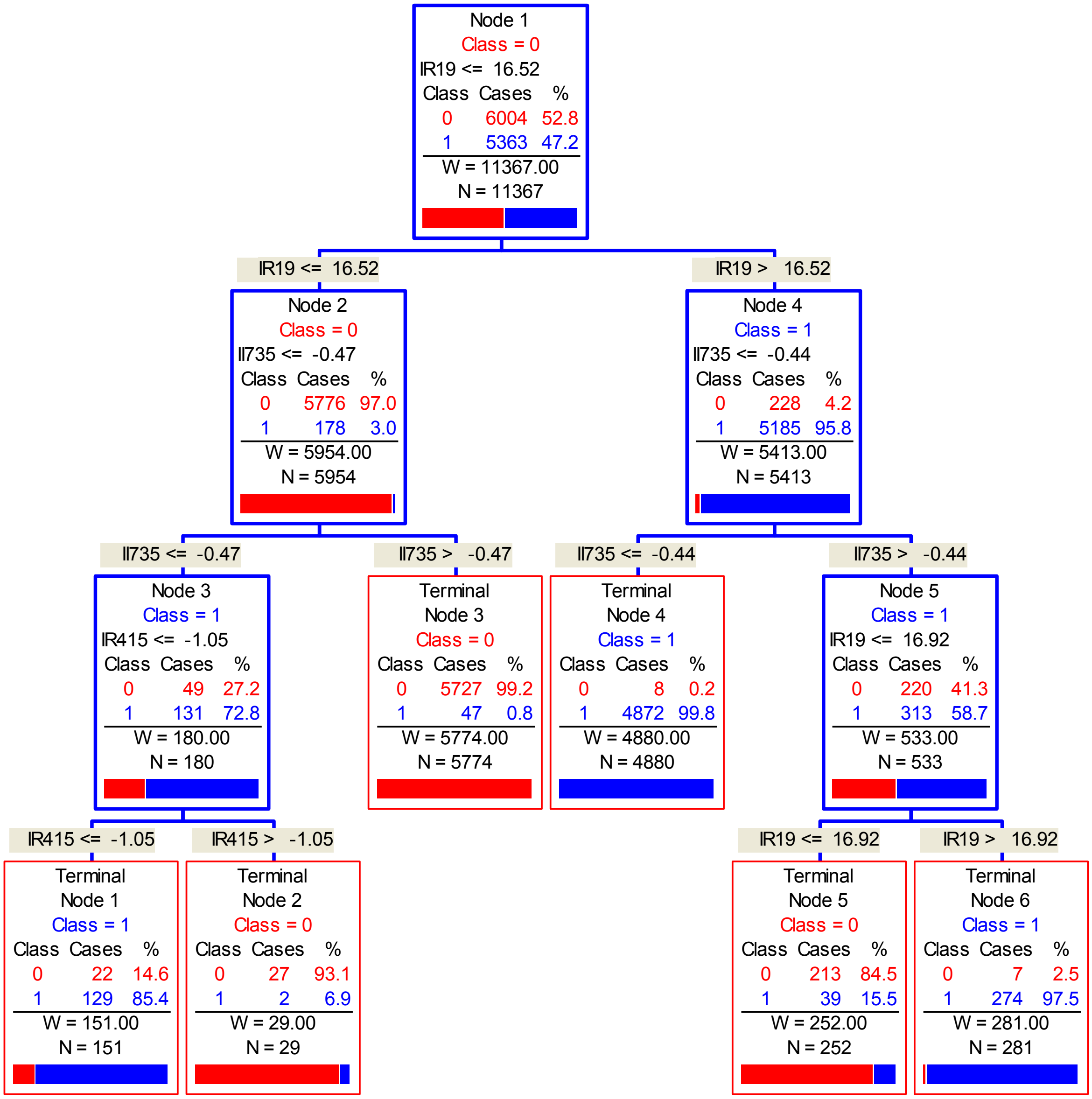


Table 7. Splitting Attributes of the Decision Tree

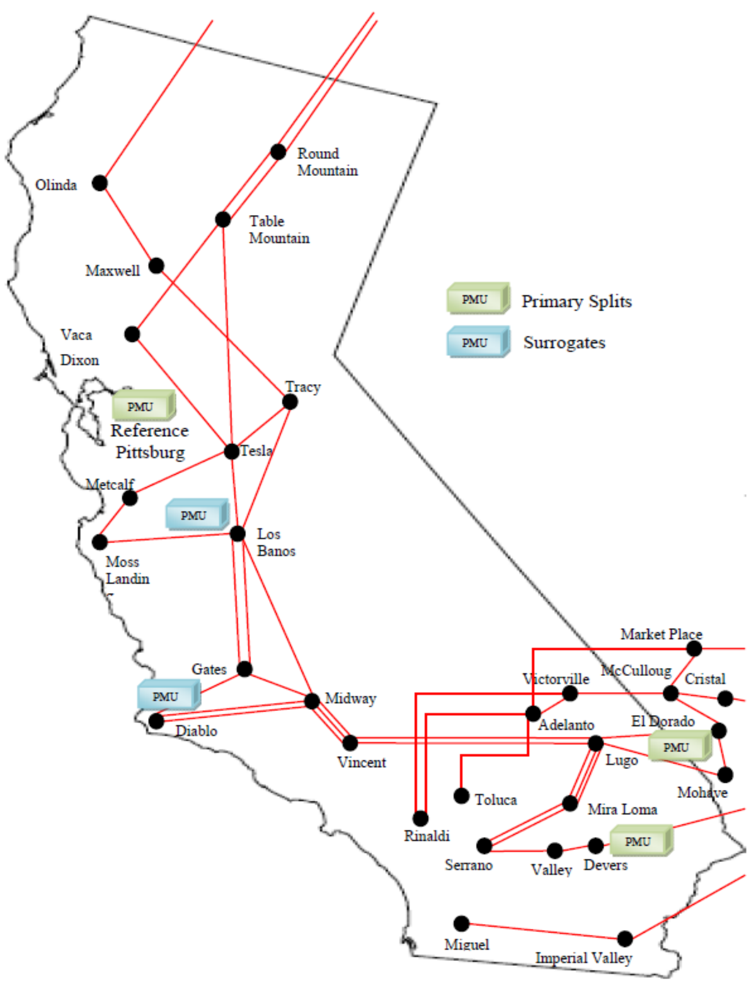
Attribute	PMU measurement
Ir19	Real Current: Palo Verde – Devers
Ii735	Imaginary Current: Devers – Valley SC
Ir415	Real Current: El Dorado – McCullough

Table 8. List of Surrogates

Node	Primary Split	Surrogate	Predictive Association
1	Ir19 ≤ 16.52	Ir472 ≤ -4.98 (Mohave – El Dorado)	0.93
2	Ii735 ≤ -0.47	Ii1033 ≤ 1.38 (Diablo – Midway)	0.17
3	Ii735 ≤ -0.4.4	Ii1033 ≤ 1.38 (Diablo – Midway)	0.72
4	Ir415 ≤ -1.05	Ii1022 ≤ 1.53 (Moss Landing – Los Banos)	0.79
7	Ir19 ≤ 16.92	Ir472 ≤ -5.26 (Mohave – El Dorado)	0.78

(The predictive association measures how well the surrogate mimics the primary split.)

Figure 5. PMU Placement for Heavy Summer Decision Tree



(PMUs in green represent primary splits. PMUs in blue represent surrogates.)

Each of these outages was simulated under diverse loading conditions. The results of the tests are summarized in Tables 9–12. Out of the 1,137 cases, 49 cases were misclassified by the decision tree; an error rate of approximately 4.3%. The tree has an adequate performance when subjected to topology changes.

Table 9. Out-of-sample Test: Generator Outage

	Classified class 0	Classified class 1
True class: 0	107	2
True class: 1	6	112

Table 10. Out-of-sample Test: Load Outage

	Classified class 0	Classified class 1
True class: 0	154	0
True class: 1	7	37

Table 11. Out-of-sample Test: 230 kV Line Outage

	Classified class 0	Classified class 1
True class: 0	278	0
True class: 1	25	284

Table 12. Out-of-sample Test: 500 kV Line Outage

	Classified class 0	Classified class 1
True class: 0	62	6
True class: 1	3	54

2.1.5 Change of Reference Results

The previous CEC research project [Centeno et al., Sept. 2010] used Pittsburg substation as a reference. The Technical Advisory Group (TAG) for this project advised against using Pittsburg substation as a reference due to lack of a 500 kV PMU measurement at that location. Two options were explored by the research team based on this development: a) estimation of the 500 kV voltage phasor from 230 kV PMU measurements at the same location; b) repeat decision tree implementation assuming reference at one of the given 500 kV known PMU locations. The research team decided to explore option b) and develop new decision trees with new reference locations.

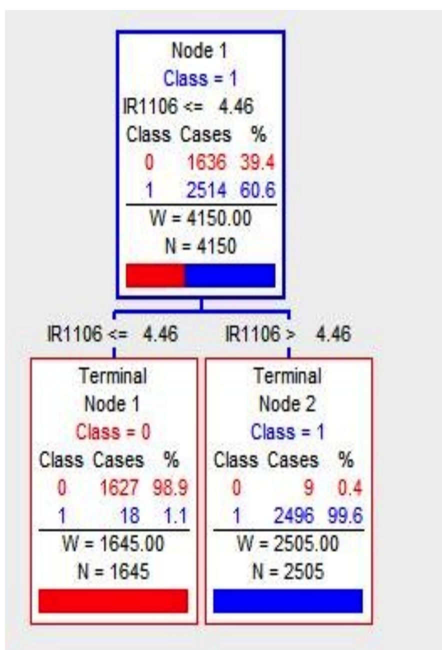
The system models and decision tree methodology developed for this project were used to develop new decision trees with a PMU reference at one of the PMU location from the list of 500 kV PMU locations provided by PG&E. The out-of-sample testing of the initial results revealed that the performance of the decision trees was reduced if a common reference was used for both the Heavy Winter and Heavy Summer system models.

The research team decided to use a variable reference setting where a difference reference was selected for each of the two system models. Tesla substation was selected as the reference for the Heavy Winter model and Vaca-Dixon substation as the reference for the Heavy Summer model. This selection of references resulted in an acceptable 1.01% error rate in the training cases and about a 3.5% error rate for out of sample cases.

2.1.6 Decision Tree: Heavy Winter Case

Figure 6 shows the decision tree selected for the Heavy Winter model. This decision tree uses the real part of the current in the Tesla – Los Banos line (IR1106) to determine the stable and unstable conditions for the Heavy Winter model. Tesla voltage angle is used as a reference and therefore no additional PMU measurements are required for this tree. The surrogate for the Tesla measurement is the voltage angle at Los Banos.

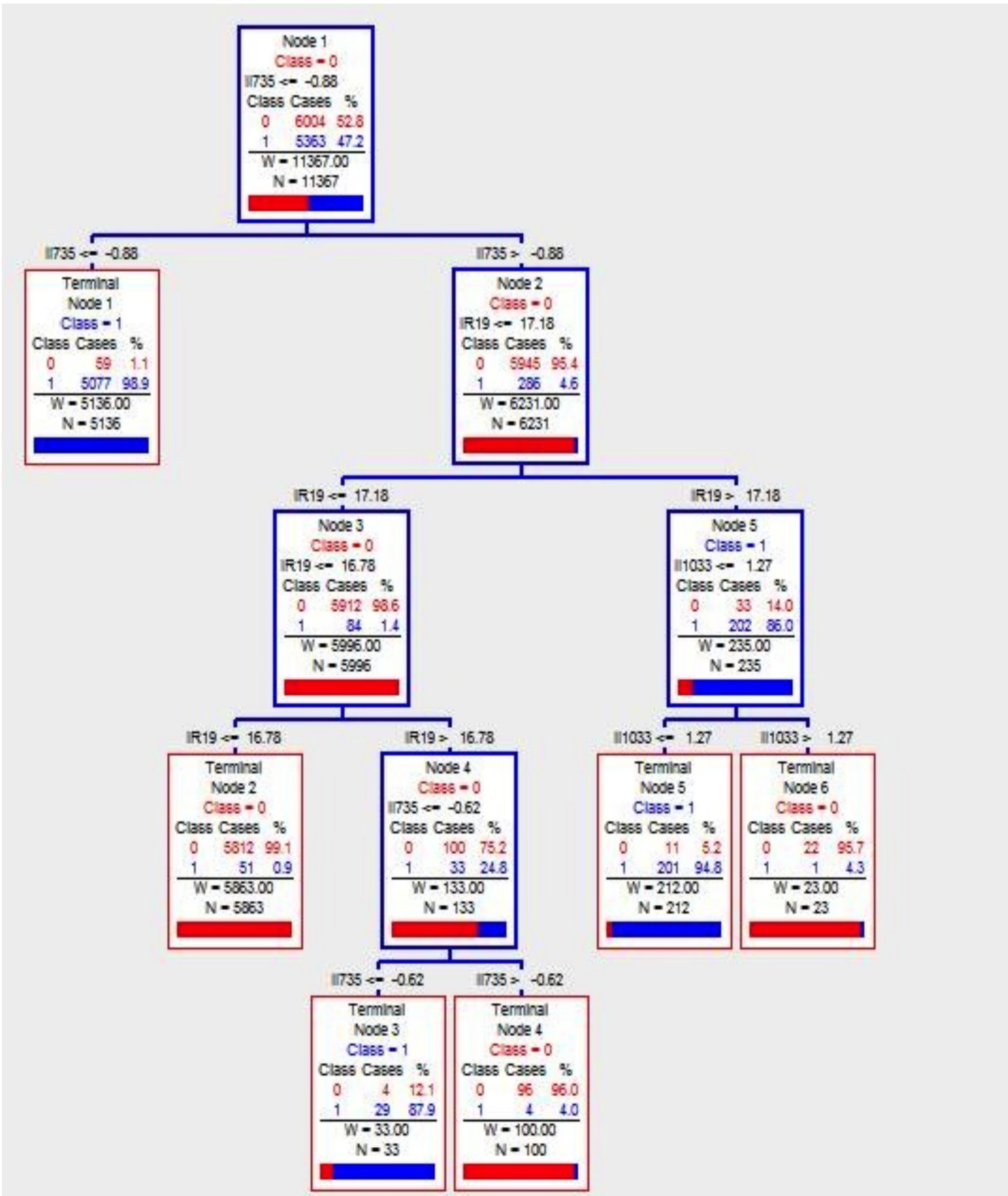
Figure 6. New Detailed Decision Tree for the Heavy Winter Model



2.1.7 Decision Tree Heavy Summer Model

Figure 7 shows the decision tree selected for the Heavy Summer model. This decision tree uses Vaca-Dixon substation as the reference and measurements at Devers and Diablo 500 kV substations to determine the stable and unstable conditions for the Heavy Summer model. The surrogates for this tree are located at El Dorado and Midway.

Figure 7. New Detailed Decision Tree for the Heavy Summer Model



The PMU at Devers monitors:

Imaginary part of the current on the Devers – Valley SC line (II735)

Real part of the current on the Palo Verde – Devers line (IR19)

The PMU at Diablo monitors:

The imaginary part of the current on the Diablo – Midway line (II1033)

The surrogates for the heavy summer case are at:

El Dorado monitors:

Imaginary part of the current of Mohave – El Dorado line (II472)

Surrogate for Imaginary part of the current at Devers – Valley SC line (II735)

Real part of the current at Mohave – El Dorado line (IR472)

Surrogate for Real part of the current at Palo Verde – Devers line (IR19)

Midway Monitors:

Imaginary part of the current at Diablo – Midway line (II1034)

Surrogate for Imaginary part of the current at Diablo – Midway line (II1033)

2.1.8 Performance of New Decision Trees

Both decision trees performed well for misclassification with their samples. The Winter model tree had 27 errors out of 4150 cases for an error rate of 0.65%. The Summer model tree had 130 errors out of 11,367 cases for an error rate of 1.14%. In order to test the robustness of the two decision trees to small departures, test cases, not included in the learning sample, were created by simulating outages. The objective of the out-of-sample tests is to assess robustness against topology changes. The out-of-sample data consists of system operating conditions obtained by simulating outages in:

- Generators delivering more than 200 MW
- Loads consuming more than 200 MW
- Transmission lines: 230 kV and 500 kV

Each of these outages was simulated under diverse loading conditions. The overall response of the two decision trees is given in Table 13 for the Heavy Winter tree and Table 14 for the Heavy Summer case. These tables also show the misclassification rates obtained when the reference from the other case is used.

The out-of-sample testing results in Tables 13 and 14 give a clear indication of the need to use separate references for the winter and summer models. The use of the summer reference on the winter case and the use of the winter reference in the summer case result in unacceptable misclassification rates of 16.7% and 44.8%, respectively. If Vaca-Dixon is used as a reference for both cases, the summer model will have an out-of-sample misclassification of 16.7% compared to the 3.63% error when each model used its own reference.

Table 13. Heavy Winter Model Tree Misclassifications by Outage Type and Reference Angle

		Tesla		Vacadixon	
		No. of cases	Mis-Classification	No. of cases	Mis-Classification
Gen		80	6	80	44
230KV lines		264	0	264	132
500KV lines		148	10	148	80
loads		168	5	168	40
		660	21(3.18%)	660	296(44.8%)

Table 14. Heavy Summer Model Tree Misclassifications by Outage Type and Reference Angle

		Tesla		Vacadixon	
		No. of Cases	Mis-Classification	No. of cases	Mis-Classification
Gen		227	15	227	7
230KV lines		604	157	604	26
500KV lines		125	8	125	4
loads		199	13	199	5
Total cases		1155	193(16.7%)	1155	42(3.63%)

2.2 Laboratory Implementation and Demonstration of Adaptive Relaying Algorithm

2.2.1 Overview

This section focuses on the laboratory implementation and testing of an adaptive voting scheme using two decision trees and two voting scheme configurations. The decision trees (DTs) were implemented into a software-based PDC and a hardware-based PDC. The voting scheme was implemented using a Master-Slave configuration and a PLC was used to perform voting for the relays. Section 2.2.2 elaborates on the application of decision trees in power system protection. It is shown that the decision trees can easily be converted to logic statements that can be implemented within a phasor data concentrator. Section 2.2.3 details how the decision trees were implemented into the phasor data concentrator. Section 2.2.4 discusses implementation of the relaying component of the adaptive voting scheme. The development of the voting schemes, testing of their functionality, and the results obtained show that all implementations are effective and accurate. Section 2.2.5 presents conclusions drawn from the research performed and possible future work.

Data Mining can be defined as the nontrivial extraction of implicit, previously unknown, and potentially useful information from data [Guoyin et al., 2008]. This process is used to transform a data base of information into a human-comprehensible format. Techniques for data mining include but are not limited to: artificial neural networks, fuzzy logic models, and decision trees. The most common application of data mining in power systems is decision trees [Mori, 2006]. Examples of decision tree applications to power system problems range from security assessment and fault detection to load forecasting and economic dispatch. This is primarily due to their efficiency in processing extensive amounts of data and the real-time processing time once the decision trees are implemented for determining power system operating conditions.

One of the latest uses of data mining techniques such as decision trees in adaptive protection of the power system is presented in [Bernabeu, 2009]. This work demonstrated that DTs provide a highly accurate, near real-time assessment of system conditions. Classification and Regression Trees (CART) are a recursive partitioning method used for building prediction models from data. Based on offline simulations of the power system, a decision tree was grown using CART® to classify the condition of the power system. CART® is a software program by Salford Systems that applies the mathematic theory of CART to generate decision trees based on large amounts of data.

Decision trees applied to power system security assessment provide a classification of “stressed” or “safe” operating conditions based on a predetermined set of measurements across the network. This classification can be used for arming a voting scheme that adapts its settings based on system conditions. Voting schemes in power system protection have been applied for many years but adaptation of the scheme based on the current state of the system is a relatively new concept. Should a single relay or relay scheme fail or misoperate, a majority vote of relays would restrain the protection system from sending a tripping signal to relays. Using the output of the real-time assessment of the system from the DT, the arming of the voting scheme can be adaptively applied to the wide-area system conditions.

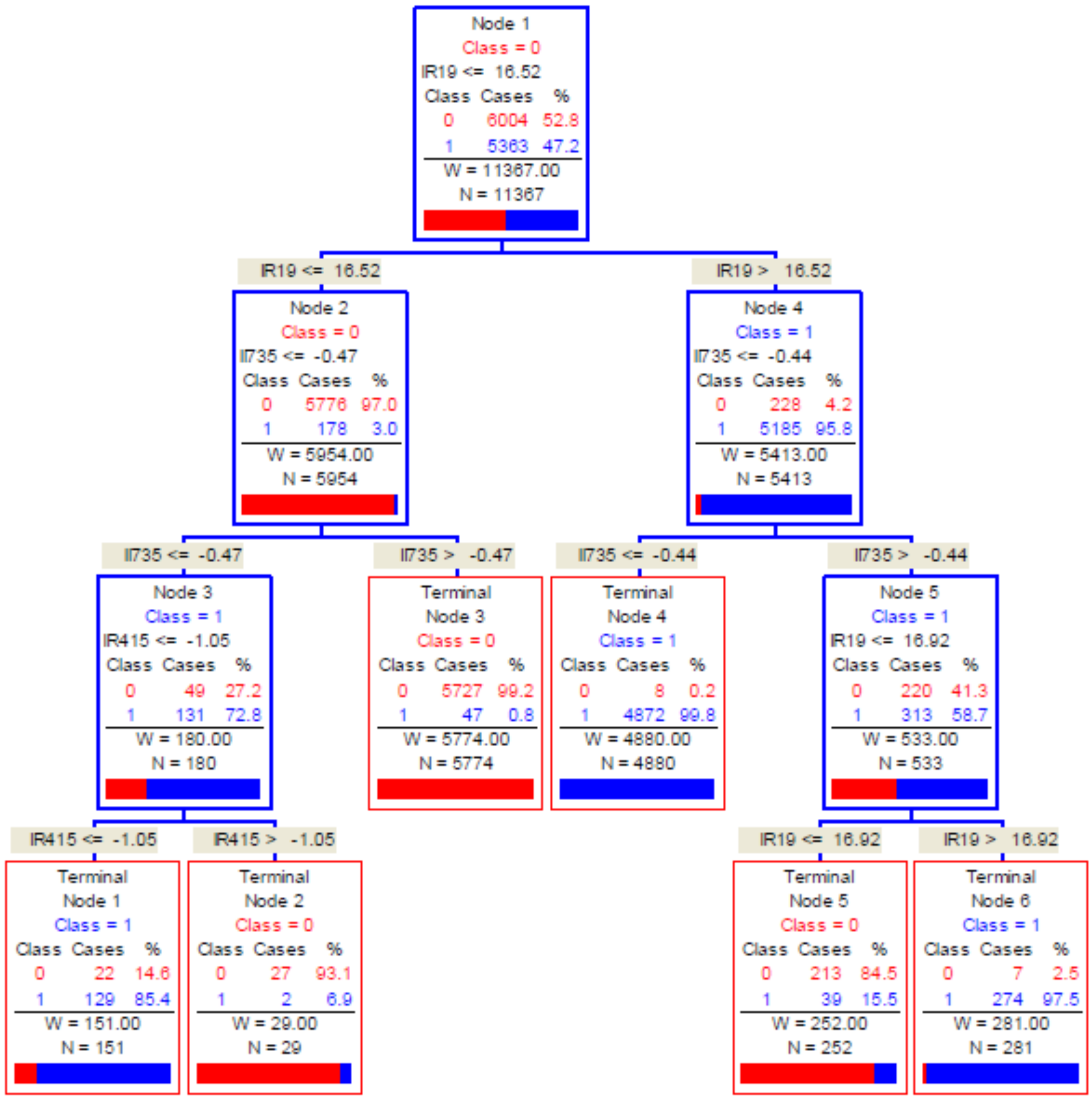
This chapter provides an understanding of how decision trees apply to power system protection. More specifically, how they are applied to the Security-Dependability Adaptive Voting scheme is explored.

2.2.2 Adaptive Protection Using Decision Trees

2.2.2.1 HW and HS Decision Trees in the Adaptive Voting Scheme

Bernabeu’s work [Ibid.] determined the optimal decision trees to be used for the adaptive voting scheme using both a Heavy Summer (HS) and Heavy Winter (HW) model. The HS and HW models are shown in Figure 8 and Figure 9, respectively.

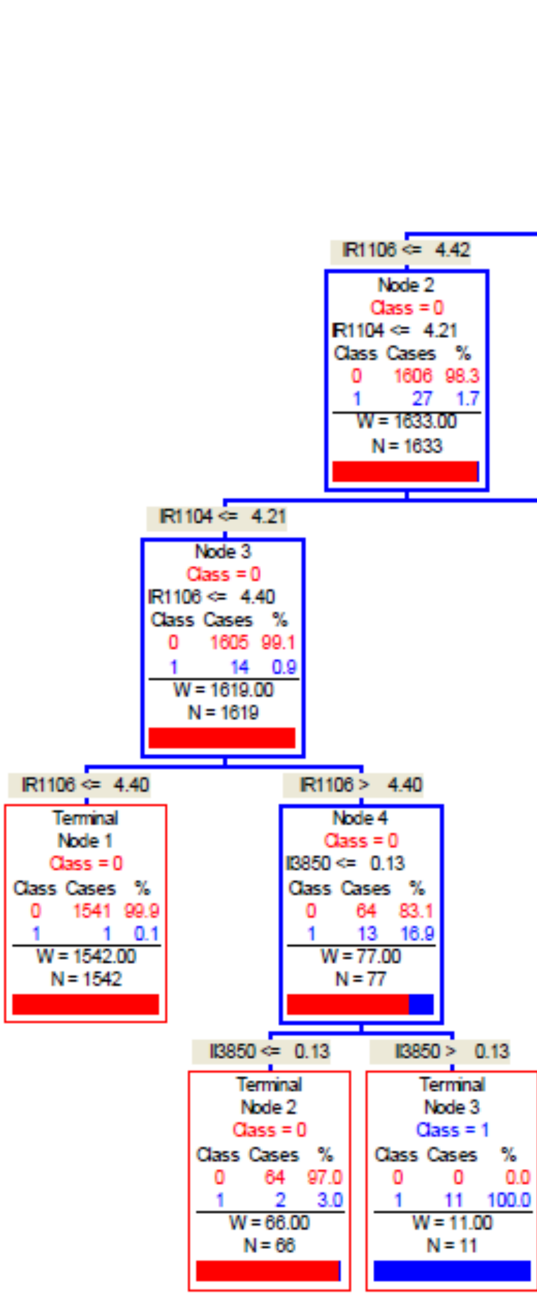
Figure 8. Heavy Summer Decision Tree



The tree's nodes are defined as the root node, splitting nodes, and classification nodes. The root node and splitting nodes for each tree describe those lines requiring PMU measurements (of positive sequence line currents) in the system. In addition, they also define the threshold values (in per unit) that the measurements are compared against.

Phasor measurement units (PMUs) are the only devices currently available with the capability to provide a wide-area view of the system at a given instant in time; they provide a snapshot picture of the power system measurements due to their time synchronization. The classification process using the decision tree requires all measurements to be time-aligned, the function of a phasor data concentrator (PDC). The decision tree then becomes an application in a wide-area measurement system, requiring time-aligned synchronized measurements of the power system.

Figure 9. Heavy Winter Decision Tree



At the time of this writing, two means of assessing the operating condition of the system exist: a PC connected to a PDC collecting the data or a PDC performing the assessment itself. Few manufacturers of PDCs allow user-programmable post-processing of the data, but the benefits are evident. With the PDC performing the assessment there is no need for an additional device such as a PC; this results in reduced latency and added reliability since there are less points of failure. Following aggregation and time alignment of the data, required measurements can be checked with their thresholds and the decision tree can be traversed and processed. Implementation of this setup is discussed in more detail in Section 2.2.4.

The location of the PDC is based on the time requirements of the system and the capabilities of the communications network. Ideally, the PDC would be installed at the critical location in the system, rather than at a centralized location such as a control center. The main advantage of using the critical location in the system is the reduced communications delays. Aggregation at the critical location would eliminate the transmission of the data following the alignment of the PMU packets; the classification status of the state of the system could be sent over the local substation Ethernet or hard-wired as a screw terminal input contact to the relays protecting the critical line. This signal is referred to as the arming signal of the protection scheme, and is the input of the voting scheme at the critical location.

2.2.2.2 CART® Decision Tree to If-Else Logic

CART® provides a graphic of the decision tree but this output must be converted to an implementable format. Any form of decision tree can be converted to "if-else" (binary tree) or "if-else/if-else" (multi-subset tree) statements for programming. Each splitting node in the tree represents an "if" statement; the left branch representing a true statement and the right branch representing a false statement. The terminal nodes in the tree provide a classification of the system state. The left decision node represents a safe condition and the right decision node represents a stressed condition.

A MATLAB program was created to produce a set of if-else statements modeling the trees generated, and is provided with the input files in Appendix A. The output of this program (in MATLAB format) is shown below for both the Heavy Summer and Heavy Winter decision trees, respectively. These if-else statements were implemented into the adaptive voting scheme at the PDC making the classification (discussed later).

Heavy Summer Decision Tree If-Else Logic

```
if( IR19 <= 16.520000 )
       if( II735 <= -0.470000 )
              if( IR415 <= -1.050000 )
                     STRESSED=1;
              else
                     STRESSED=0;
              end
       else
              STRESSED=0;
       end
else
       if( II735 <= -0.440000 )
              STRESSED=1;
       else
              if( IR19 <= 16.920000 )
                     STRESSED=0;
              else
                     STRESSED=1;
              end
       end
end
```

Heavy Winter Decision Tree If-Else Logic

```
if( IR1106 <= 4.420000 )
       if( IR1104 <= 4.210000 )
              if( IR1106 <= 4.400000 )
                     STRESSED:=0;
              else
```

```
                        if( II3850 <= 0.130000 )
                                STRESSED:=0;
                        else
                                STRESSED:=1;
                        end
                end
        else
                STRESSED:=1;
        end
else
        if( IR1104 <= 4.060000 )
                STRESSED:=0;
        else
                STRESSED:=1;
        end
end
```

The real-time classification of STRESSED = 0 or STRESSED = 1 was used to adapt the power system reliability setting in the voting scheme. Application of this arming signal is discussed in later chapters.

2.2.2.3 Relay Voting Scheme

The end-use application of a voting scheme seeks to increase the reliability of the protection system. During normal operating conditions of the power system, the voting scheme is disabled and the protective devices operate as expected. The decision is based on an 'OR' function; that is, any one relay or scheme can trip the breaker(s) protecting the equipment. This defines an increase in dependability for this system. Should a tripping condition occur, there exists a bias to remove that component from service with a greater chance of misoperation at any given time. A redundancy check is not present to verify that indeed a tripping condition has occurred. On the other end of the reliability spectrum, during stressed system conditions, the voting scheme is armed. During an armed state, the devices vote on tripping the breaker(s) prior to the action being taken. The voting process ensures that a misoperation does not occur, which defines an increase security on the system. During stressed conditions, a misoperation of a relay and breaker would cause heavier loading of the system and could potentially lead to a cascading failure or worse a blackout condition. The voting scheme is a means of increasing the certainty that an event such as this will not occur due to misoperation.

The arming signal generated at the real-time analysis stage of the adaptive protection system is provided to the field devices. This signal can be sent to a standalone device for arming or disarming the voting scheme, as in Figure 10, or the signal may be embedded into the existing code of the protective relays themselves, as in Figure 11.

With an additional device providing the voting mechanism, the existing relays can remain untouched because their functionality remains unchanged. Each device monitors the power system quantities and provides an output signal for tripping. The only required change would be changing where that output signal is sent. A problem with this setup is that during normal, safe operating conditions, the relay trip signal must be sent through an additional device increasing latency prior to breaker operation.

Rather than changing the configuration of the system, the relays can be set up in a Master-Slave approach. Only the internal logic of the relay trip settings is changed to account for signals coming from the other protective relays monitoring the common element. This setup does not include an additional device and also does not require a change in hardware configuration. It does, however, require the changes in the relay settings to be documented, and also requires a change in maintenance and testing procedures for the devices. This is often a difficult obstacle to overcome in the utility industry.

Figure 10. Voting Device Configuration

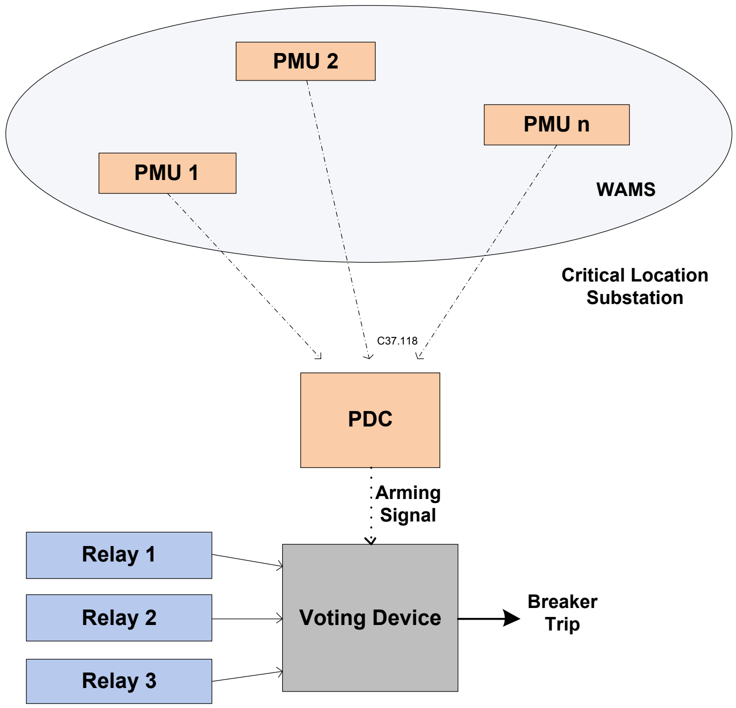
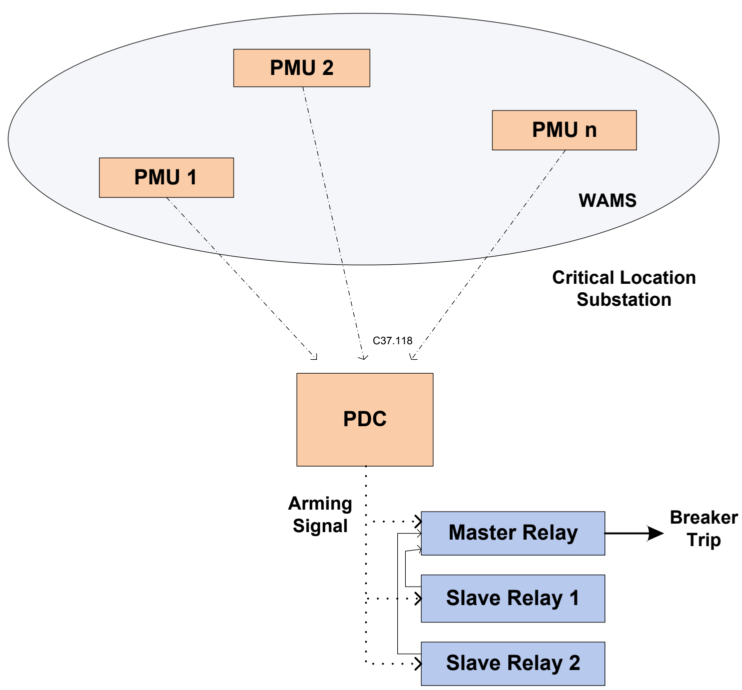


Figure 11. Embedded Master-Slave Configuration



2.2.3 Decision Tree Implementation in the PDC

Implementation of the security-dependability adaptive voting scheme uses decision trees to make a classification of the operating condition of the system. As previously described, classification was made at the PDC based on its ability to time align and aggregate system-wide measurements. PMU data collected by the PDC was processed through an additional user-defined set of functions that determined whether to arm or disarm the voting scheme based on stressed or safe operating condition, respectively. Many manufacturers will work with power engineers to develop firmware for their specific applications, but few PDCs are programmable by the user of the device. This chapter discusses two commercially available options for implementing this scheme at the PDC level, using equipment that provides the capability of programming user-defined functions into the device.

The first option explored was a hardware based PDC, the Schweitzer Engineering Laboratories SEL-3378 Synchrophasor Vector Processor (SVP) [Schweitzer SEL-3378 Manual]. The SEL-3378 was programmed using the accompanying SVP Configurator software from SEL. The other option explored was a software based PDC, OpenPDC, which is a software platform from Grid Protection Alliance, running on a Windows-based computer [http://OpenPDC.codeplex.com/]. This open source program was downloaded from the Internet and programmed using C# and Visual Studio 2010.

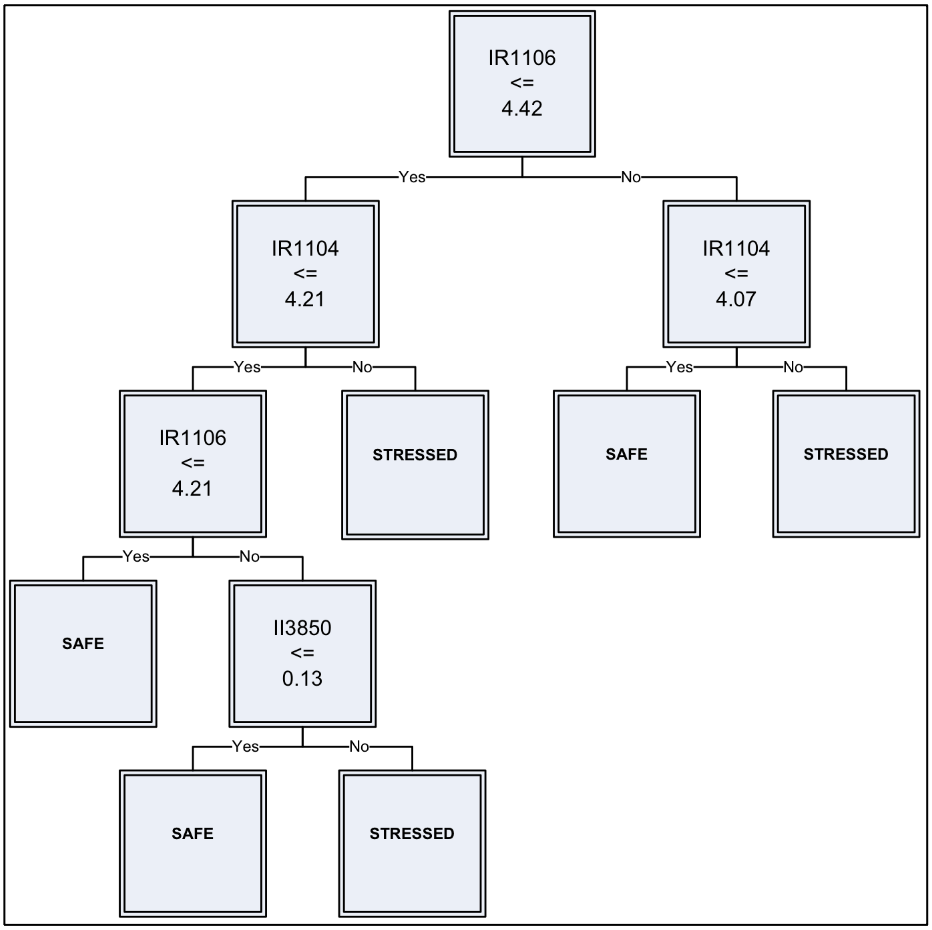
This chapter explains two independent configurations of the entire adaptive voting scheme. First, the decision tree functionality was tested using the hardware-based PDC. This work elaborates upon and completes the work performed in [Thomas, 2011]. Second, the decision tree was implemented in the software-based PDC. The ability to implement the decision trees into this type of PDC was also explored. Lastly, the adaptive voting scheme in its entirety was implemented using OpenPDC and a programmable logic controller (PLC). This configuration expands upon the work done thus far on the Security-Dependability Adaptive Voting Scheme. Topics discussed for both configurations include the network architecture, programming of the devices, testing methods, and results obtained.

2.2.3.1 Synchrophasor Vector Processor Implementation

The focus of the work performed using the hardware based PDC was to implement the Heavy Winter and Heavy Summer decision trees from [Bernabeu, 2009]. This chapter uses the Heavy Winter model for explanation of the setup, testing, and results, but the methodology holds for the Heavy Summer model also. A simplified version of the HW decision tree is shown in Figure 12 for clarity.

As the tree shows, there are three distinct line currents requiring monitoring by phasor measurement units (1104, 1106, and 3850). The physical location of these three lines is of no interest for testing purposes, but the number of measurements and required PMUs is of interest. The decision tree provided the measurement requirements for three distinct PMUs, each monitoring a power system quantity that is compared against a per unit threshold for classification. This requirement added the need of a per unit measurement system for testing. For field implementation of this system, an engineer would need the base value used to determine the quantity to enter into the tree in actual units rather than per unit values. For example, 4.21 may correspond to 2105A on a 500A base. The tree uses “IR” and “II” to refer to the real and imaginary components of currents in the system, where “IR” represents the real component of the complex current phasor and "II" represents the imaginary component of the complex current phasor.

Figure 12. Simplified Heavy Winter Decision Tree

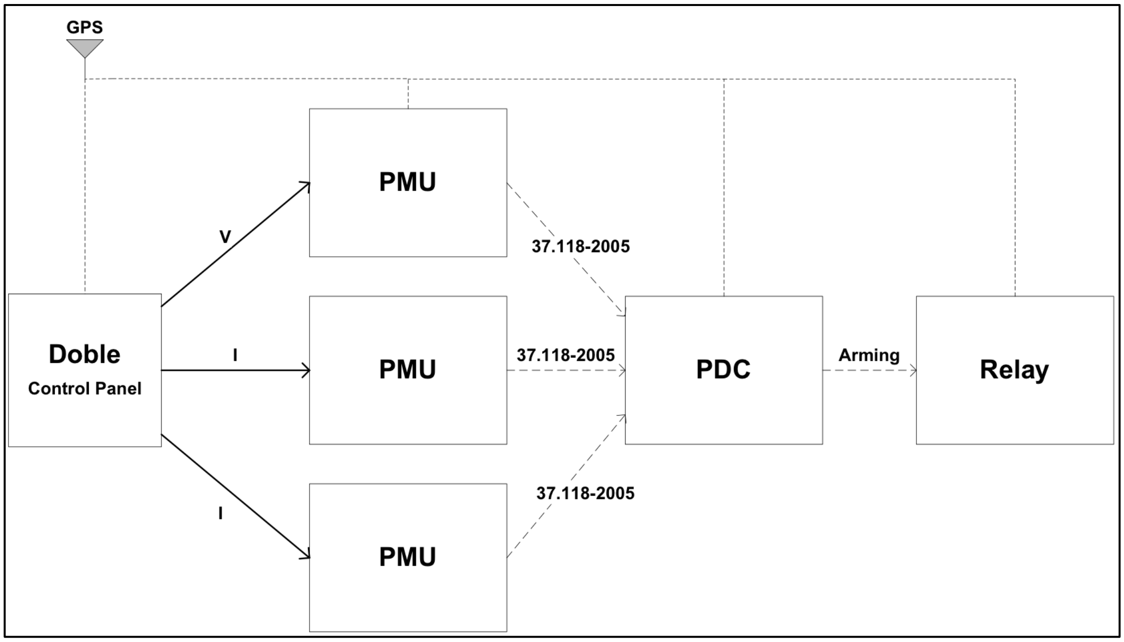


Network Architecture

A test system was set up to emulate how the devices would interact with each other when deployed in the field. The voting scheme was reduced to a single relay capturing and recording events from the PDC because the focus of this part of testing was on the functionality of the decision tree in the PDC. The adaptive protection scheme was tested using the network configuration shown in Figure 13.

The Doble 6150 Power System Simulator was used as a user-selectable signal generator, with simultaneous 3-phase voltage and current output channels. The Doble Control Panel is a computer user interface used to set and alter signals generated by the device. Since the device was time synchronized to UTC, it was used to select the appropriate voltage magnitude and phase angle to obtain the required real and imaginary components. Due to the structure of the tree and the limited output channels, the only way to test all possible conditions was to use a voltage as the origin node in the tree. All inputs were set to the same base values of V_{base} = 10V and I_{base} = 1A. The resulting tree used for testing the Heavy Winter decision tree is shown in Figure 14.

Figure 13. Hardware PDC Network Configuration

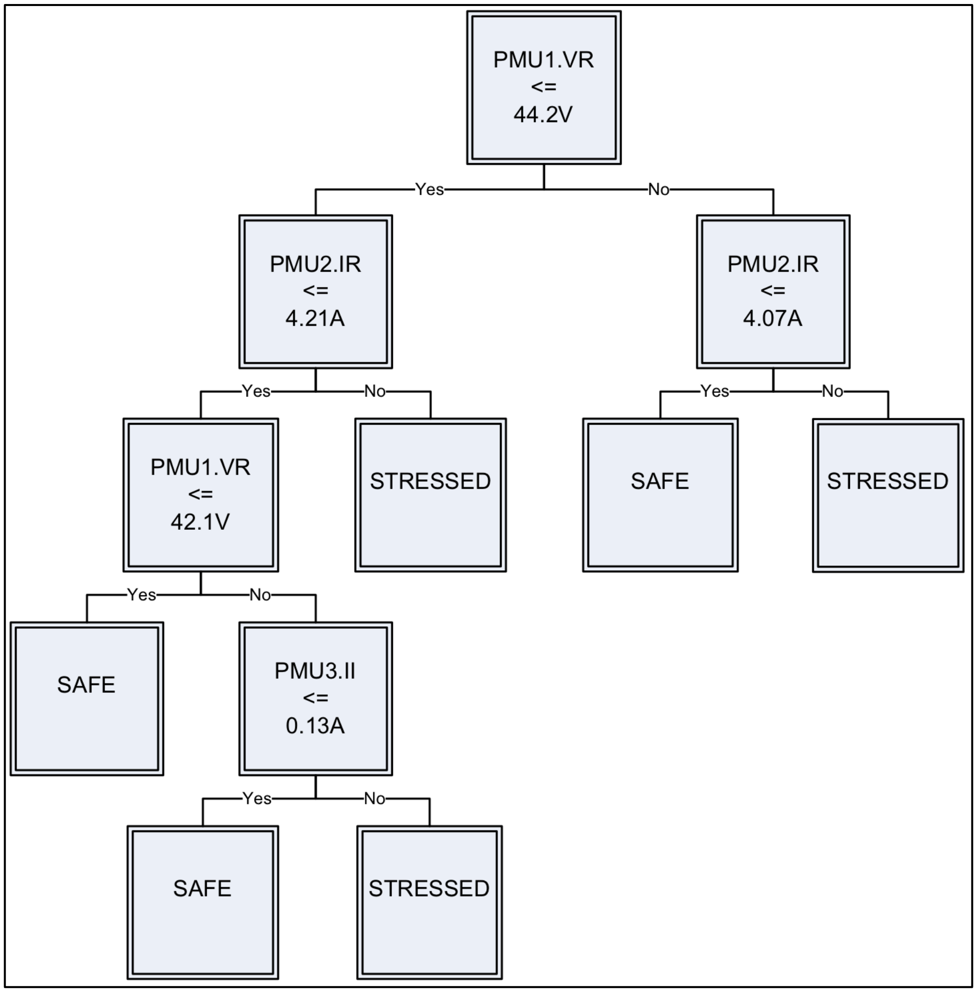


Three separate PMUs measured 3-phase current or voltage quantities, and reported the positive sequence phasors for each. These phasors were sent over Ethernet channels to the hardware PDC using C37.118-2005 protocol. The PDC aggregated and time aligned these PMU measurements and reported a single C37.118-2005 output stream to a computer for observation. Additionally, the PDC also performed the decision tree functionality and provided an arming output signal to a relay. The actual implementation of this system deploys a voting scheme between three relays to decide on a majority trip or single trip. The implementation for testing the hardware PDC functionality was limited to sending the arming signal to a single relay rather than a set of relays. Of interest for this testing was the complete and accurate operation of the PDC, not the voting scheme.

SEL-3378 and SVP Configurator Programming and Setup

The SEL-3378 Synchrophasor Vector Processor provides the features of a phasor data concentrator (PDC) as well as additional functions of a programmable logic controller (PLC). It aggregates and time aligns synchrophasor data as a traditional PDC would, and also allows for user-defined programmable logic. Up to 16 PMUs can be aggregated by the SEL-3378 using C37.118-2205 protocol or SEL's proprietary Fast Message protocol, and these messages can be time aligned at up to 60 messages per second. C37.118-2005 messages can be output to up to 6 external clients and one internal client with a throughput of 2 ms. The PDC can also send Fast Operate or SEL Mirrored Bits messages to perform control action. SEL offers interface hardware such as the SEL-2515 Remote I/O module for interfacing with non-SEL products [Schweitzer SEL-2515 Manual].

Figure 14. Hardware PDC Heavy Winter Testing Tree



SVP Configurator is the SEL software package for configuring the SEL-3378. It provides a visual platform for setting communication parameters and programming protection applications into the device. The SEL-3378 consists of different internal components performing operations in real-time; the SVP Configurator software allows for managing each of these components. Programming of the SEL-3378 is performed using standard IEC 61131-3 logic such as Structured Text, Function Block Diagrams, Ladder Logic, and Instruction Lists. These functions were created in the SEL SVP Configurator software provided with the device. Once the program is created in software by the user, it is uploaded to the SEL-3378 and run in real time.

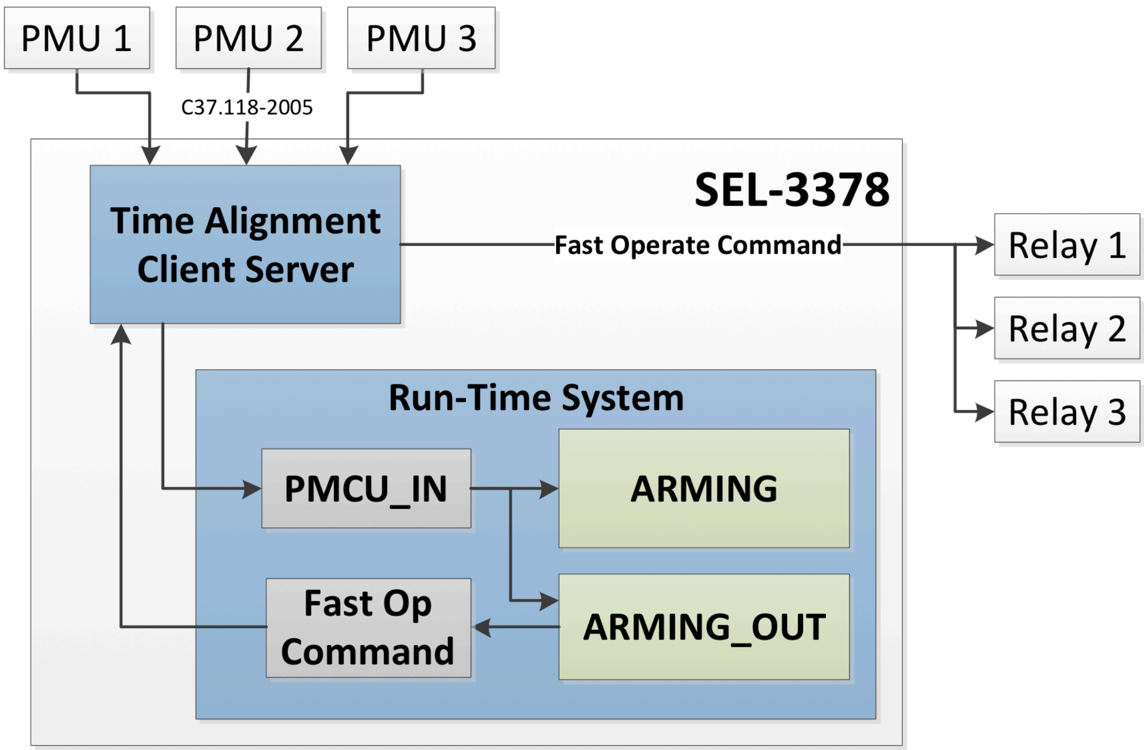
The Time-alignment Client Server (TCS) is responsible for aggregation and time alignment of the incoming synchrophasor data, and is the first module programmed for the SEL-3378. Figure 4.4 shows the Structured Text programmed for the Heavy Winter Model, consisting of three input PMUs and a single output stream to the PC. PMU2 (IDCODE 1) is also used as the "relay" in this situation for the arming signal acknowledgement. In the field implementation, this would be replaced with the relays monitoring the critical line where the voting scheme is to be deployed.

SEL-3378 Programming Logic

```
* 487E Station A – PMU*)
PMCU_INPUT[1].EN := TRUE;
PMCU_INPUT[1].IDCODE:= 2;
PMCU_INPUT[1].C37_118_CLIENT:=TRUE;
PMCU_INPUT[1].FASTOP:=1;
PMCU_INPUT[1].CONNECTION := ‘E’;
PMCU_INPUT[1].EPORT.SERVER_IP := ‘10.222.3.171’;
PMCU_INPUT[1].EPORT.SERVER_CMD_PORT := 4712;
PMCU_INPUT[1].EPORT.TRANSPORT_SCHEME := ‘TCP’;
PMCU_INPUT[1].EPORT.CLIENT_IP := ‘10.222.3.181’;
PMCU_INPUT[1].EPORT.CLIENT_DATA_PORT := 4712;

(* 487E #2 – Relay*)
PMCU_INPUT[2].EN := TRUE;
PMCU_INPUT[2].IDCODE:= 1;
PMCU_INPUT[2].C37_118_CLIENT:=TRUE;
PMCU_INPUT[2].FASTOP:=1;
PMCU_INPUT[2].CONNECTION := ‘E’;
PMCU_INPUT[2].EPORT.SERVER_IP := ‘10.222.3.178’;
PMCU_INPUT[2].EPORT.SERVER_CMD_PORT := 4728;
PMCU_INPUT[2].EPORT.TRANSPORT_SCHEME := ‘TCP’;
PMCU_INPUT[2].EPORT.CLIENT_IP := ‘10.222.3.181’;
PMCU_INPUT[2].EPORT.CLIENT_DATA_PORT := 4728;
PMCU_INPUT[2].EPORT.FASTOP_PORT := 23;
PMCU_INPUT[2].EPORT.FASTOP_PORT_TELNET_EN := TRUE;

(*Fiber PMU*)
PMCU_INPUT[3].EN := TRUE;
PMCU_INPUT[3].IDCODE:= 6;
PMCU_INPUT[3].C37_118_CLIENT:=TRUE;
PMCU_INPUT[3].FASTOP:=0;
PMCU_INPUT[3].CONNECTION := ‘E’;
PMCU_INPUT[3].EPORT.SERVER_IP := ‘10.222.3.176’;
PMCU_INPUT[3].EPORT.SERVER_CMD_PORT := 4713;
PMCU_INPUT[3].EPORT.TRANSPORT_SCHEME := ‘UDP’;
PMCU_INPUT[3].EPORT.CLIENT_IP := ‘10.222.3.181’;
PMCU_INPUT[3].EPORT.CLIENT_DATA_PORT := 4713;

(* OUTPUT01*)
PMCU_OUTPUT[1].EN := TRUE;
PMCU_OUTPUT[1].MRATE := 60;
PMCU_OUTPUT[1].CLIENT_IP := ‘10.222.3.185’;
PMCU_OUTPUT[1].CLIENT_DATA_PORT := 4750;
PMCU_OUTPUT[1].TRANSPORT_SCHEME := ‘UDP_S’;
PMCU_OUTPUT[1].SERVER_IP := ‘10.222.3.181’;


TCSconfigOK := TCS_CONFIG(EN := TRUE,
PDC_IDCODE := 1000,
pHID := ADR(HEADER_118),
NFREQ := 60,
MRATE := 60,
MISSING_MESSAGE_THRESHOLD := 6,
TIME_UNSYNC_BLOCK := TRUE,
pCMD_OUT_DATA_IN := ADR(PMCU_INPUT),
pCMD_IN_DATA_OUT := ADR(PMCU_OUTPUT),
pERROR := ADR(TCS_ERROR_OUT),
pSTATUS := ADR(TCS_STATUS_OUT));
```

Figure 15 elaborates on the SEL-3378 architecture and the internal interaction between processes following PMU data frame arrival.

Figure 15. SEL-3378 System Architecture



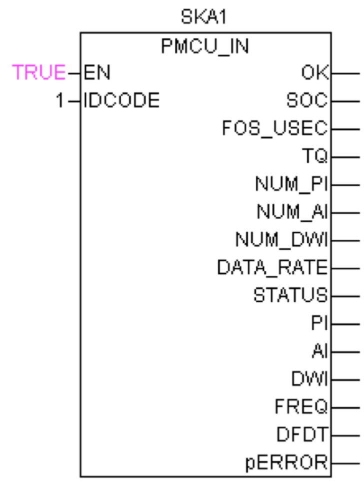
The TCS running internally in the SEL-3378 sends the time aligned C37.118-2005 data to the Run-Time System (RTS) and up to 6 external C37.118-2005 clients. Of interest for wide-area protection applications was the RTS functionality, which allows for processing of the time aligned data using IEC 61131-3 standard logic. The next function block programmed into the SEL-3378 was the Phasor Measurement and Control Unit block (PMCU_IN). A PMCU_IN block was generated for each incoming PMU or PDC. Figure 16shows the PMCU_IN block for PMU with TCS = 1. This PMU was currently enabled, so the PDC expected synchrophasor data from the device.

Each PMCU_IN variable (SKA1 in Figure 16) was set as a global variable, so that it could be accessed by other functions in the program. With the PMCU_IN and TCS functions created, the PDC was able to aggregate and time align PMU measurements and output a single C37.118-2005 data stream. The following sections describe the functions created for running the decision tree and sending a Fast Operate Message based on the state of the system.

Decision Tree and Arming Signal Implementation

Aside from the RTS performing the PDC tasks, an additional ARMING function was created to use the parsed PMUs packets for state determination using the decision trees. This function contained all the logic for the decision tree if-else statements as discussed in Chapter 2, and passed a STRESSED variable to the output function.

Figure 16. PMU Input Function Block



The Heavy Summer decision tree was programmed into the PDC using the testing threshold values. A digital bit, STRESSED, was used as the classification of the system. If STRESSED := TRUE, then the arming signal was be asserted; if STRESSED := FALSE, then the arming signal was restrained. The following structured text block was programmed into the SEL-3378 using SVP Configurator to model the decision tree output figure.

As discussed above, the ARMING function is where the STRESSED bit is set to TRUE or FALSE based on the values and their thresholds of phasor measurements. The STRESSED bit from the ARMING function was used to enable or disable function blocks in the ARMING_OUT function.

Heavy Winter Model Test Decision Tree

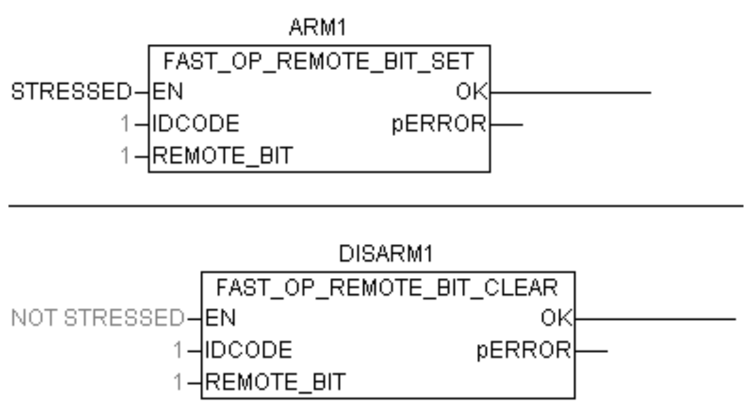
```
IF SKA1.PI[1].RE <= 44.2
THEN
  IF SKA2.PI[2].RE <= 4.21
  THEN
    IF SKA1.PI[1].RE <= 42.1
    THEN
      STRESSED := FALSE;
    ELSE
      IF SKA3.PI[2].IM <= 0.13
      THEN
        STRESSED := FALSE;
      ELSE
        STRESSED := TRUE;
      END_IF;
    END_IF;
  ELSE
     STRESSED := TRUE;
  END_IF;
ELSE
  IF SKA2.PI[2].RE <= 4.07
  THEN
     STRESSED := FALSE;
  ELSE
    STRESSED := TRUE;
  END_IF;
END_IF;
```

The first method explored for sending the arming signal to a relay was the SEL Fast Operate commands. This communications protocol uses a binary data stream over Telnet that an external device such as a relay uses to perform trip, close, set remote bit, or clear remote bit operations [15]. Since the adaptive protection scheme provides a signal to the relays for voting determination, the Remote Bit Set and Clear functions were used. The remote bit Fast Operate command has a total message length of 6 bytes, where the fourth byte into the message is the operate code for the external relay. This byte changes from 0x00-0x0F (hex) when the remote bit is cleared to 0x20-0x2F (hex) when the remote bit is set, for remote bits RB01-R16 respectively. Using the RB01, the operate byte in the stream was monitored for change from 0x00 to 0x20 for the transition from disable to enabled.

Figure 17 shows the function blocks used for setting and clearing the remote bit at the external device. The STRESSED bit and its binary inverse are sent to the ARM1 and DISARM1 blocks in the function. Therefore, when the system is classified as stressed based on the traversal of the decision tree, the FAST_OP_REMOTE_BIT_SET block will enable and set RB01 (REMOTE_BIT = 1) of the relay with TCS of 1 (IDCODE = 1). When this occurs, the disarm function is simultaneously disabled due to the enable bit of DISARM1 being false. Conversely, when the system is classified as normal operating conditions based on the tree results, the FAST_OP_REMOTE_BIT_CLEAR block is enabled and RB01 is cleared.

Fast Operate commands are a very simple method for controlling remote devices of SEL manufacture. The only problem encountered was integrating this control signal into a system of non-SEL devices. The solution to this problem was using the SEL-2515 Remote I/O Module in conjunction with the SEL-2812 Fiber Optic Transceiver/Modem to convert the Fast Operate messages to contact closure. The contact closure completes a DC circuit to close in a contact on a relay input contact terminal.

Figure 17. Fast Operate Control Function Blocks



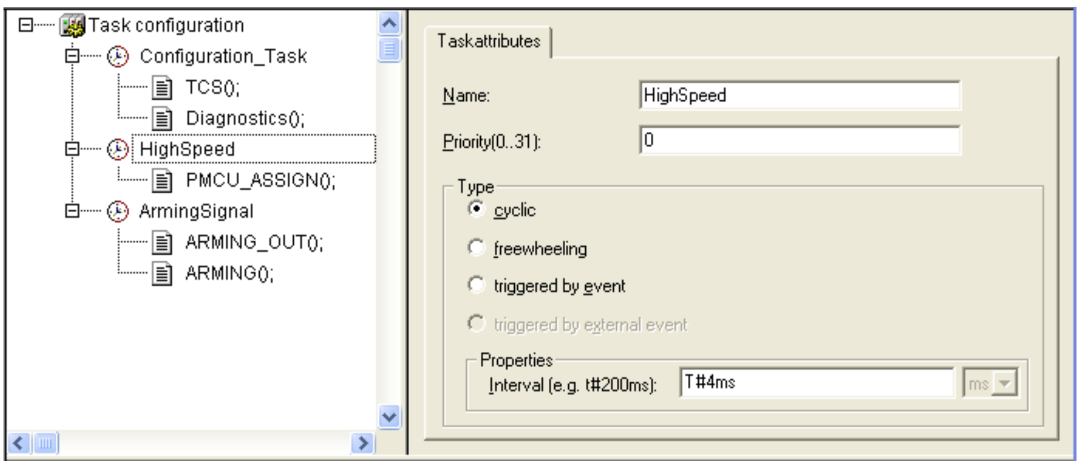
Other options also exist for communicating the arming bit to external devices. SEL also provides a patented Mirrored Bits communication technology that exchanges the status of eight internal logic points called Mirrored Bits, encoded in a digital message [http://www.selinc.com/]. The bits sent from one device (TMB) are “mirrored” by the received bits of the remote device (RMB). A newer technology currently being developed by SEL is the use of the C37.118-2005 command frames as a carrier of the Fast Operate control signal, rather than Telnet, to expedite transmission of the signal. The latest SEL 421, 451, 411L, and 487V versions will decode the fast operate commands using this methodology. Again, this is an SEL-based technology but may be expanded upon if proven effective.

The Fast Operate commands were implemented for this protection scheme because of the simple and effective means of transmitting a control signal to other devices, and the proved use with manufacturers of other devices.

Task Configuration

The priority of tasks was an essential component of setting up both the TCS as well as the decision tree program within the PDC. The various programs created were grouped into tasks as shown in Figure 18. Three task groups were created: Configuration_Task, HighSpeed, and ArmingSignal. The Configuration_Task was set to run every 2 seconds due to the low priority of the tasks. The HighSpeed task was the PMCU_Assign task, which performed the PDC alignment functionality for reading in PMU data. This had the highest priority and ran at an interval of 4 milliseconds. Lastly, the ArmingSignal task performed the decision tree functionality and control output. Based on advice from an SEL representative, this task was run at an interval of 16 ms.

Figure 18. SEL-3378 Task Configuration

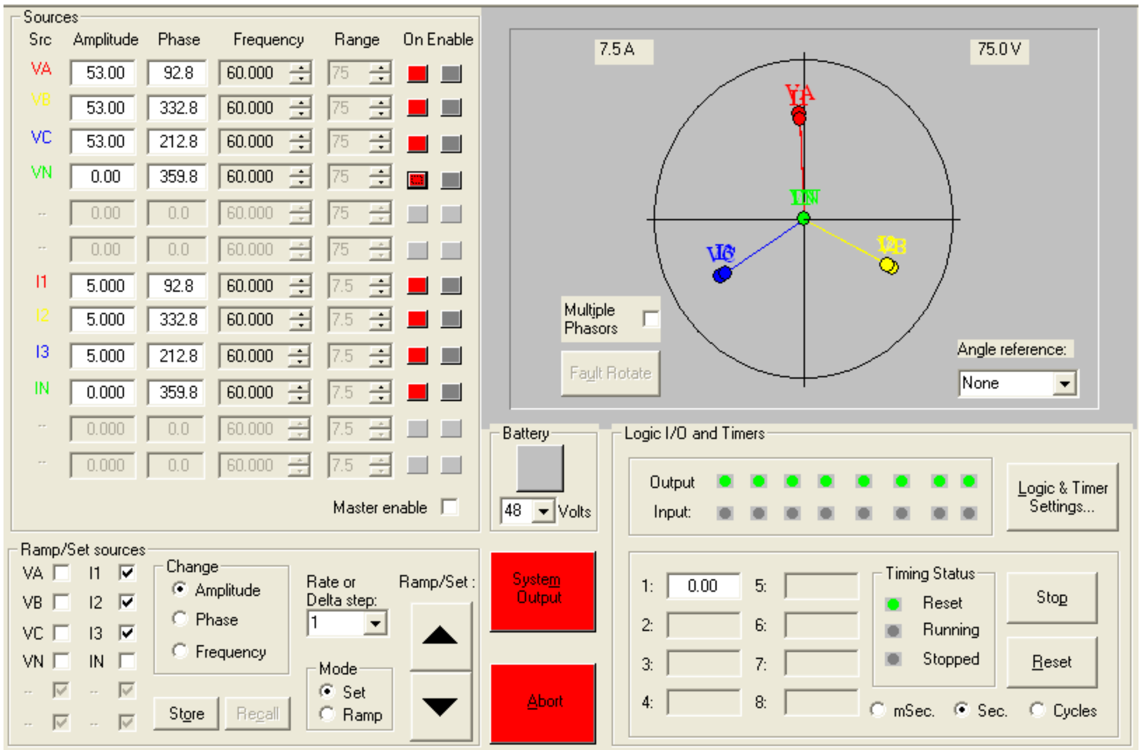


Hardware PDC Testing

A test procedure was devised to test all possible scenarios of the security/dependability adaptive protection scheme within the PDC. The focus of testing was to ensure that PMUs reporting the correctly monitored values to the PDC were parsed, the tree traversed, and the correct control action taken. The three PMU measurements of key locations on the system were used as the inputs and monitored using Wireshark. SVP Configurator was used to observe the STRESSED arming signal state in real-time, and the relay receiving the control signal over Telnet communications recorded SER Events to ensure that the signal was received by the device.

Six distinct states were created using combinations of the voltage and current available to test each classification for the Heavy Winter Model decision tree, starting from the bottom left classification of the tree moving right. The Doble F6150 was used to generate positive sequence phasors by setting the magnitude and phase angle of phase quantities. Values were chosen such that the real and imaginary components, as required by the decision tree, fell above or below the thresholds being tested. Important to note was that the Doble had a -92.8° angle offset on both the current and voltage channels that was accounted for during testing. The Doble Control Panel was used for setting the steady state conditions, as shown in Figure 19.

Figure 19. Doble Control Panel



The different voltages and currents applied to the PMUs are provided in Table 15. It was found that both the PDC Boolean STRESSED signal and the Remote Bit (RB01) in the relay receiving the control signal were set correctly for all expected output conditions. It was concluded based on this test that the PDC was functioning as expected for implementation of the decision tree for a set of steady state conditions.

Table 15. Hardware PDC Functionality Test Results

State		Inputs						Outputs		
	Technologies.	Doble Voltage			Doble Current					
#	Cond.	V	V_r	V_i	I	I_r	I_i	PDC Out	RB 01 Pickup	OUT1-6 Check
1	Safe	33/0°	33	0	3/0	3	0	FALSE		Y
2	Safe	43/0°	43	0	3/0	3	0	FALSE		Y
3	Stressed	43/0°	43	0	3/10°	2.95	0.52	TRUE	X	Y
4	Stressed	33/0°	33	0	4.4/0°	4	0	TRUE	X	Y
5	Safe	53/0°	53	0	3/0°	3	0	FALSE		Y
6	Stressed	53/0°	53	0	5/0°	5	0	TRUE	X	Y

Further work was performed to ensure that the STRESSED signal was being set accordingly by the correct classification in the tree, as an extra measure of functionality. Output variables, OUT1-6, were assigned to each classification terminal in the tree, and the test was repeated. As expected, the PDC functioned properly, only setting the correct output for a given set of system conditions.

These tests confirmed expectations of how the PDC would operate during normal conditions, i.e. steady state and no bad data. Transient analysis of the PDC is embedded in the later discussion on latency testing, which also requires correct functionality. The topic unexplored thus far was how the PDC would react to uncertain WAMS conditions such as a missing PMU measurements or a PMU with invalid IRIG-B.

The phasor measurements in a C37.118-2005 data frame for a given PMU are each preceded by a status word, STAT, that provides critical information about the quality and condition of the data as well as the measuring device. The STAT word originates at the PMU, but may be altered by other devices such as the PDC. Mainly of interest in the STAT word are the most significant bits, Bits 15-13. Bit 15 is the Data Valid bit, which is set to 1 when the PMU deems the data invalid. Bit 14 is the PMU Error bit which is set when the PMU experiences an internal error such as an A/D converter error [IEEE Std. C37.118-2005]. Bit 13 is the PMU Sync Error which is set to 1 when the PMU experiences a loss of synchronization; this can occur if either the time source fails or if the IRIG-B input fails.

The decision tree implementation was tested on the SEL-3378 with bad data to determine how it would react to these types of situations. In today's power system, with the increasingly complex and growing number of devices and networks, it is not uncommon to lose a single device in the field. Measures must be taken to allow for these single failures while repairs or replacements occur. Two tests were performed to "force" the bad data: remove the IRIG-B input to a PMU and entirely remove a PMU from the system.

A PMU was removed from the network by disconnecting the Ethernet connection to the hub. As expected, these packets did not arrive at the PDC, and the PDC responded by filling the phasor data with "0" in the PDC output packets. For the decision tree application, this posed a slight problem because the tree could misclassify the correct state of the system. For example, the voltage and current applied to the PMUs placed the operating condition into State 4 (OUT4 := TRUE). PMU2 was removed from the network, so the PMU2.IR measurement needed in the tree compared its phasor of "0" to the threshold value. This changed the operating state of the system to either State 2 (OUT2 := TRUE) or State 3 (OUT3 := TRUE) based on the measurement of PMU3.II. This example demonstrates how the reported operating condition of the system can be easily altered based on which PMUs are actually reporting phasor data. It was also noted that based on the structure of the Heavy Winter decision tree, there was no possible way of losing a PMU measurement and going into a STRESSED condition from a SAFE condition; only the opposite scenario held.

A PMU lacking a source of time synchronization reported its phasors as expected, but acknowledged this problem in the STAT word. The PDC picked up this error and filled the respective PMU's phasor measurements with "0". This caused significant problems with the decision tree functionality of the device. Voltages and currents were applied to the system using the Doble such that the decision tree fell into classification 6 (far right decision node). The IRIG-B was disconnected from PMU1 and the results were observed: no change in the STRESSED output signal. This is because the traversal of the tree moved from classification 6 to classification 4, still a stressed condition. The IRIG-B was replaced for PMU1, but then disconnected from PMU2: change in STRESSED signal. The tree traversal again changed, but from classification 6 to classification 5. The origin node moved the traversal to the right of the tree, but the next decision was using an erroneous “0” as its input. Therefore, the tree misclassified the operating condition of the system when bad data was present.

The fix to this problem involved adding simple logic to the decision tree to determine the state of the PMUs prior to traversing the tree. If any single measurement device reported a STAT bit that was not congruent with a functioning system, the decision tree was bypassed, and the STRESSED output was set to FALSE. Therefore, the voting scheme application would perform as normal with any one device or protection scheme allowed to send a tripping signal. The structured text below shows how this additional layer of security was implemented in the tree.

Heavy Winter Model Test Decision Tree

```
IF SKA1.STATUS.PMCU_OK AND SKA2.STATUS.PMCU_OK AND SKA3.STATUS.PMCU_OK
THEN
        [Decision Tree Implementation…]
ELSE
        STRESSED := FALSE;
END_IF;
```

The results obtained from hardware PDC testing using the SEL-3378 showed positive results. The device was able to fully implement the decision trees using synchrophasor measurements from PMUs. The state of the system was classified as stressed or safe and an output bit was set within the PDC. This bit was sent to the set of relays using Fast Operate Messages for SEL devices, and additional hardware was used to communicate between the SEL-3378 and non-SEL relays. Each operating condition was tested as well as injection of bad measurements in the form of lost PMU data or bad IRIG-B signal.

2.2.3.2 OpenPDC Implementation

The Tennessee Valley Authority (TVA) developed a phasor data concentrator in 2004, and made it available as open-source software free to download from http://OpenPDC.codeplex.com/. This software, OpenPDC, has been proven operable with over 120 PMUs on a single computer platform; this specification depends on the computer’s hardware limitations and capabilities. OpenPDC is compatible with most synchrophasor protocols and a number of databases for data storage. Database storage of synchrophasor data, statistics data, and other user-defined data allows for retrieval from other devices with the capability to communicate with a server database. TVA PMU Connection Tester [http://www.pmuconnectiontester.codeplex.com/] is used in conjunction with OpenPDC to provide a simple means of connecting and communicating with phasor measurement units or other phasor data concentrators on the network.

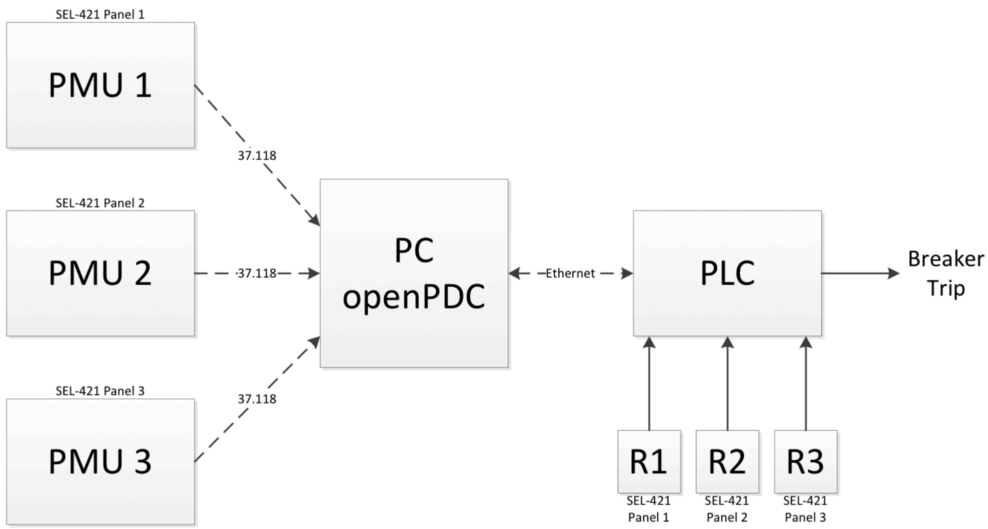
This section details the implementation of OpenPDC as a viable option for the adaptive voting scheme. It also describes how the full protection scheme was devised and applied.

<u>OpenPDC Implementation Network Architecture</u>

The configuration used for testing OpenPDC as a viable option for the adaptive voting scheme is shown in Figure 20. Three PMUs were aggregated and time aligned at the computer running

OpenPDC. C37.118-2005 protocol over TCP connection was used to send the data from the PMUs to the PC. The real-time determination of operation state of the system was performed in the PC using C# programming. A bit was stored in an SQL database reflecting this classification from the decision tree. A Rockwell Automation PLC was used to retrieve the arming bit in the database on the server and determine whether to vote or execute the relays as they normally would be deployed. Three SEL relays were hard-wired to the PLC inputs, communicating their trips signals based on instantaneous overcurrent faults. The PLC performed the actual 'breaker tripping' in the circuit regardless of the state of the Arming signal. A visual display was used to ensure that the scheme operated correctly as designed.

Figure 20. Software PDC Network Configuration



Due to available resources at the time of testing, the actual Heavy Winter and Heavy Summer trees were altered slightly. Only one synchronized source was available to a single PMU such that only one PMU could report phasors with exact real and imaginary current values. Other 3-phase sources were applied to the other PMUs but they were not synchronized to UTC and did not have a true 60 Hz frequency. Therefore, the phasors would rotate over a period of time, making it impractical to test the actual HW/HS decision trees. Figure 21 and Figure 22 show the Heavy Winter and Heavy Summer trees used for testing purposes, respectively. A classification node in the tree was selected to place the third PMU with exact real and imaginary components such that the functionality could still be tested. It was inferred that the rectangular form decision would function identically regardless of where it was placed in the decision tree.

<u>OpenPDC Setup and PMU Configuration</u>

Following installation, OpenPDC was configured for three separate PMU inputs. The PMU Connection file and Configuration files were extracted from PMU Connection Tester, and each PMU was configured to send at least positive sequence current and voltage phasors at 60 frames per second. A historian database was set up to archive PMU data following alignment in OpenPDC. Each individual measurement made by an input PMU was given a unique identification number within the database such that only the necessary measurements for the decision tree could be extracted and provided to the functions requiring the data.

Figure 21. Software PDC Heavy Winter Testing Tree

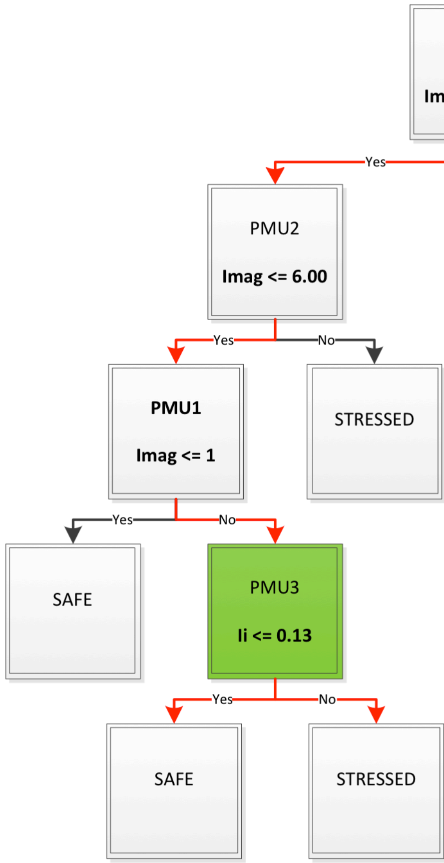
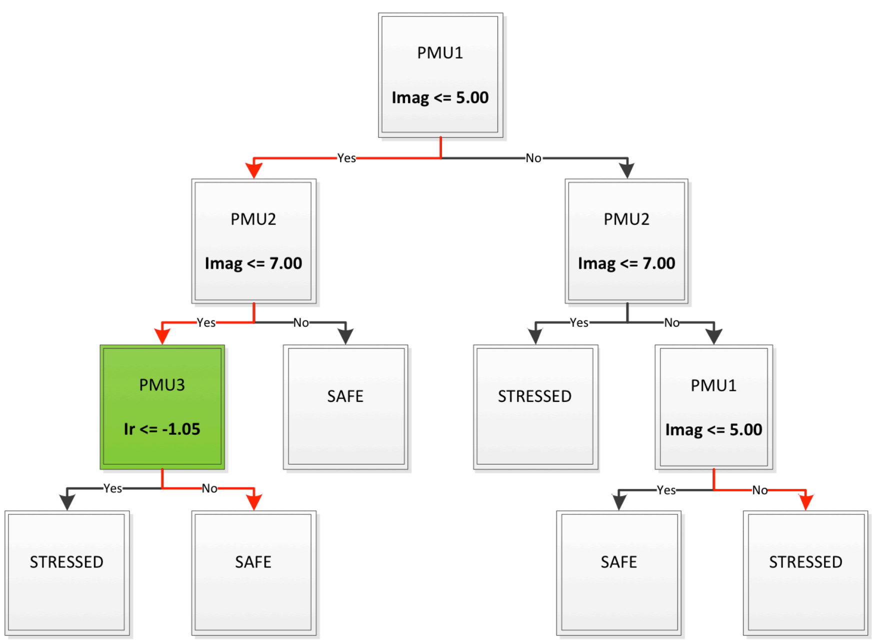
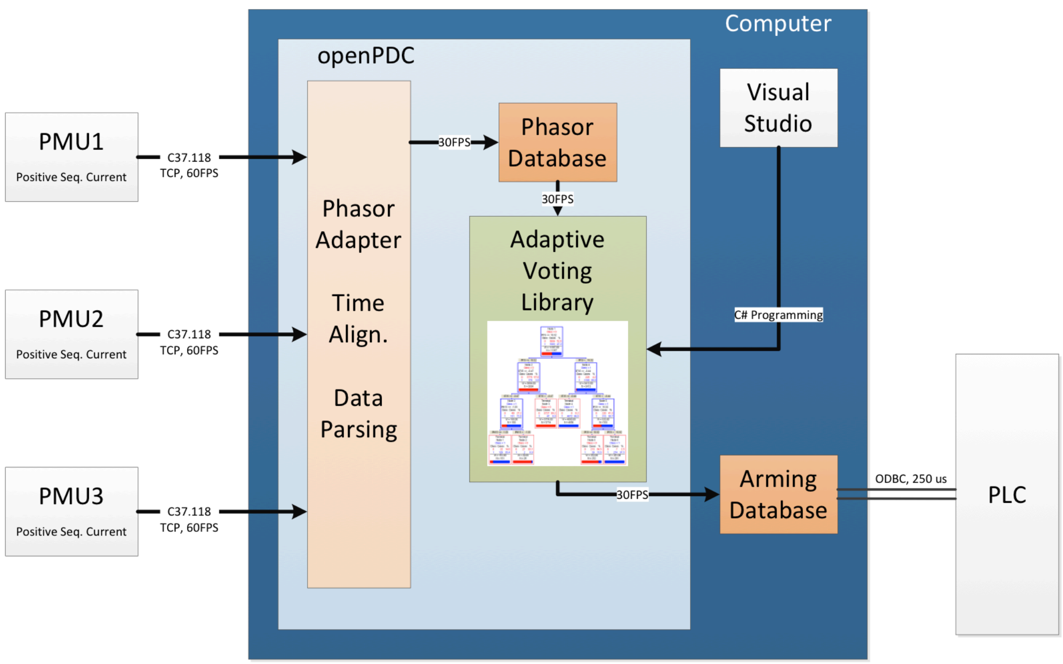


Figure 22. Software PDC Heavy Summer Testing Tree



OpenPDC is built on three types of adapters: input, action, and output adapters. The input and output adapters allow OpenPDC to communicate and remotely configure I/O such as subscribers/publishers, or historians and real-time data. The action adapters, on the other hand, perform functions or actions on the data once it has been time aligned. For implementation of the decision tree within , the action adapter layer was the only layer of importance. Figure 23 shows the OpenPDC architecture within the computer, and the connecting devices communicating with the computer. Upon time alignment, the phasor data was stored in a MySQL database, Primary Phasor Archive (PPA) [http://www.mysql.com/]. The measurements were passed to the PPA at 30 frames per second. Each frame was then sent to the adaptive protection code running the decision tree. The results of the decision tree were then pushed to a Microsoft SQL database [http://www.microsoft.com/sqlserver/en/us/default.aspx], storing the Arming signal.

Figure 23. OpenPDC Software Architecture



<u>Arming Decision Tree Programming</u>

Implementation of the decision tree running in OpenPDC was performed using C# programming in Visual Studio 2010. A Visual Studio library was created that consisted of code that the OpenPDC would execute each time a new measurement frame was received. Phasor measurements stored in frames were called using the PublishFrame() function from the TimeSeriesFramework library. Within the PublishFrame() function, a for loop was used to extract all measurements from the current frame. These measurements were stored in temporary variables. Since the decision tree requires real and imaginary components of the phasors, or rectangular form, the data was converted from polar to rectangular since OpenPDC reported the phasors in polar form:

```
I3imag = I3mag*Math.Sin(I3phase*pi/180);
I3real = I3mag*Math.Cos(I3phase*pi/180);
```

Both decision trees were implemented in the same function. Each tree was built using if-else programming logic, and the State was stored as either "Safe" or "Stressed". The Heavy Winter tree is shown below for demonstrative purposes. The trees were dynamically switched based on the current models being used for operation of the power system. A selector switch was implemented that provided the user of the program the ability to easily switch between trees. This was done by providing a 24V DC signal to the input of a PMU sending synchrophasor data. The input was programmed into the C37.118-2205 data frame as a digital word. OpenPDC was then able to parse the incoming frames to read the status of this word. The action adapter library that was implemented used this value to determine which decision tree to perform:

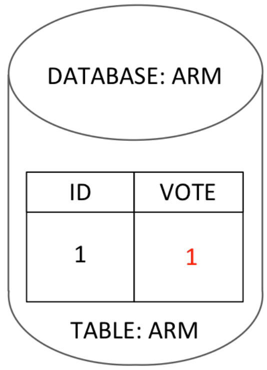
```
OnStatusMessage("Decision Tree: Heavy Winter");
        if (I1mag <= 5.00) {
          if (I2mag <= 6.00) {
            if (I1mag <= 1.00) {
              State = "Safe";
            }
            else {
              if (I3imag <= 0.13) {
                State = "Safe";
              }
              else {
                State = "Stressed";
              }
            }
          }
          else {
            State = "Stressed";
          }
        }
        else {
          if (I2mag <= 5.00) {
            State = "Safe";
          }
          else {
            State = "Stressed";
          }
        }
      OnStatusMessage("System Operating Condition: {0}", State);
      Update(State);
```

Following classification of the operating condition of the system, the data was stored in an SQL server for later use. A connection to the SQL server was established and the State variable was updated within the database table. The table consisted of a single important cell called 'vote' that was either set to 0 (meaning "Safe") or 1 (meaning "Stressed"). This bit was retrieved by the PLC and is discussed later in this chapter. Each Update() function within the library created would open a connection to the table, update the table based on the classification of the system from the decision tree, then close the connection to the server.

<u>SQL Database Implementation</u>

The SQL database used for the OpenPDC implementation, as shown in Figure 24, was created outside of the custom action adapter. It is therefore assumed that both the database and the table required for the application already exist when OpenPDC is executed.

Figure 24. OpenPDC SQL Arming Database



The database simply contains a single table where the system state is stored. The table was designed with only two fields named 'technologies' and 'vote'. The field 'technologies' is used as a reference field whenever a query is made for updating or retrieving the 'vote' field value, which is the arming bit retrieved by the PLC.

To enable OpenPDC to perform queries within the adapter and access the SQL database, the custom action adapter library was hard-coded with the necessary connection string, which identifies the server and the database. Every time the Update() method is called in the adapter, the connection is established and a query is executed to commit the running system state 'vote' value to the table:

```
public void Update(string status)
    {
      string query;
      if (status.Equals("Safe"))
      {
        query = "UPDATE Arm SET vote=0 WHERE technologies=1";
        OnStatusMessage("vote set to zero");
      }
      else
      {
        query = "UPDATE Arm SET vote=1 WHERE technologies=1";
        OnStatusMessage("vote set to one");
      }

      if (this.OpenConnection() == true)
      {

        SqlCommand cmd = new SqlCommand();

        cmd.CommandText = query;
        cmd.Connection = myConnection;
        cmd.ExecuteNonQuery();

        this.CloseConnection();
      }
    }
```

Future implementations can possibly include code for generating and verifying the arming database required by the adapter to enhance the portability of the OpenPDC version of the voting scheme.

Arming Signal Extraction from SQL Server

A Rockwell Automation ControlLogix 1756-L73 Controller was used as the PLC retrieving the vote signal from the database and performing the breaker trip logic based on the state of the vote signal. The ControlLogix controller is part of the Logix5000 family of Rockwell Automation controllers, and provides I/O modules linked by a common chassis or backplane [http://www.ab.com/en/epub/catalogs/12762/2181376/2416247/360807/360809/]. The controller used was equipped with both digital input and output cards, each with 16 points at 5V DC TTL (transistor-transistor logic) signal levels. Also included was an EtherNet/IP communications module for connecting to the network where the SQL Database Server resided. The 1756-L73 Controller is capable of 100 programs per task, and 32 tasks. It supports up to 500 connections to different points in the system, and also provides Ethernet communication [Rockwell, 2011]. Programming options include relay ladder logic, structured test, function blocks, and structured flowchart. All programming of the PLC was done in ladder logic.

The PLC served two critical purposes in this system: query the SQL database residing on the computer running OpenPDC, and perform the logic based on the operating condition results obtained. Based on the integer value of the 'vote' bit retrieved, the PLC would perform 2-out-of-3 voting if the bit was 1 and normal operation for relay tripping if the bit was 0.

Query of the SQL database was done using FactoryTalk Transaction Manager. Two connectors were established between the SQL database and the ControlLogix controller. A connection was established between each of the two ends of the tunnel for passing data, and the 'vote' bit was read out of the database. Based on all the testing performed, all transactions between both ends of the connection performed at 100% passing rate for successful transactions. Upon retrieval of the bit by Transaction Manager, the integer value obtained was stored in a Vote tag in the controller itself.

In essence, the database was continuously queried by Transaction Manager. Each query would update the Vote tag within the PLC itself, and the PLC would use this integer value obtained to determine whether the scheme should vote or perform normal operation.

Software PDC Testing

The implementation of the decision tree using a software-based PDC, OpenPDC, was performed similarly to testing of the hardware PDC. Prior to testing if the voting scheme functioned properly in the PLC, the OpenPDC decision tree was tested to ensure that the outputs were operating as expected. Table 16 shows the results for functionality testing of the Heavy Winter and Heavy Summer decision trees in OpenPDC. For each operating condition the PDC Console provided the correct output, the database contained the correct value for 'vote,' and the PLC arming signal was set to TRUE. This triplicated the confirmation that the tree was operating as expected.

The software-based PDC application of decision trees for the adaptive voting scheme focused on integration of the trees into OpenPDC as well as physical hardware configuration for the relay voting. The decision trees were both realizable within the OpenPDC platform itself using C# programming. Communication between the computer running OpenPDC and the PLC performing the trip logic was a critical component of the system. A Rockwell Automation ControlLogix controller was used to query the SQL database set up on the computer to retrieve the classification of the state of the system based on the PMU measurements. Regardless of a stressed or safe operating condition, the PLC always had responsibility of tripping the breaker whether voting or performing normal operation. Each relay was hard-wired to the PLC, and ladder logic was programmed into the device to coordinate relay operation and breaker tripping.

Table 16. Software PDC Functionality Testing Results

	State		Input Currents				Outputs	
	#	Technologies. Cond.	I1mag [A]	I2mag [A]	I3r [A]	I3i [A]	PDC Console	PLC Arming
Heavy Winter	1	Safe	0.5	3	3	0	vote=0	FALSE
	2	Safe	3	3	0	-3	vote=0	FALSE
	3	Stressed	3	3	0	3	vote=1	TRUE
	4	Stressed	3	6.5	3	0	vote=1	TRUE
	5	Safe	5.5	3	3	0	vote=0	FALSE
	6	Stressed	5.5	5.5	3	0	vote=1	TRUE
Heavy Summer	1	Stressed	3	3	-3	0	vote=1	TRUE
	2	Safe	3	3	3	0	vote=0	FALSE
	3	Safe	3	8	3	0	vote=0	FALSE
	4	Stressed	5.5	5	3	0	vote=1	TRUE
	5	Safe	5.5	8	3	0	vote=0	FALSE
	6	Stressed	5.5	8	6	0	vote=1	TRUE

2.2.4 Voting Scheme Configurations

This section covers the details of all the devices, settings, and wiring required for a voting scheme to function. This section begins the discussion with the fundamental tripping logic of the voting scheme.

The three voting scheme digital relays must have their settings updated to perform as an adaptive voting scheme. It is already known that the voting scheme, when armed, will only trip the circuit breaker if two or more relays have positive trip signals. Implementing a voting scheme requires the Master relay's logic equation to include the arming signal. With redundant slave relays, they will also receive the arming signal and therefore the settings of those relays must be updated accordingly [Thomas, 2011].

2.2.4.1 Master-Slave Tripping Logic

Master Relay Logic

The design of the voting scheme can be seen in the Master relay's truth table. With the voting scheme disarmed, the Master and slave relationship among the voting scheme relays no longer applies; the original protection scheme becomes active and will consist of three redundant relays. When the scheme is disarmed, the Master relay's trip functionality will not be influenced by the input trip signals from the slave relays. Therefore, if any of the three relays have a positive trip signal, it will send the trip signal directly to the circuit breaker telling it to open. This is shown in the first two rows of Table 17. The Master relay will trip the circuit breaker only if it has a positive trip signal itself; the Master is blinded from the actions of the slave relays.

Table 17. Truth Table for Master Relay with Arming Signal

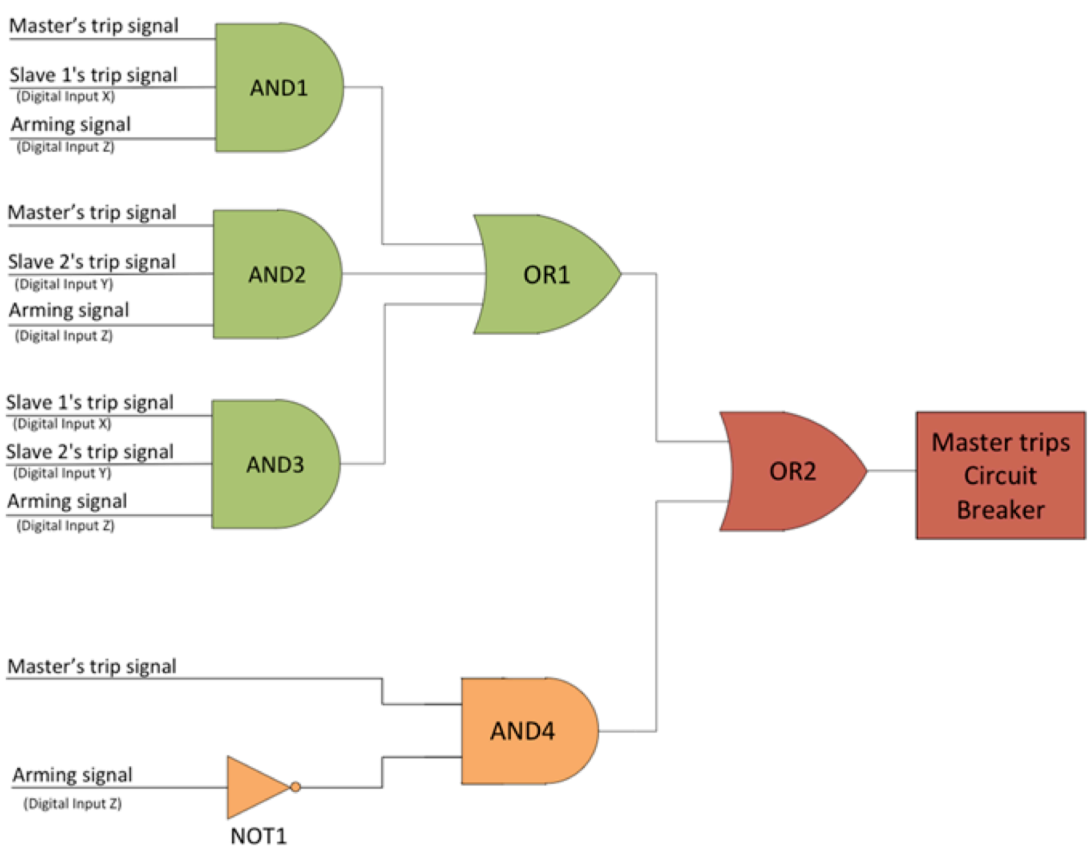
Voting Scheme Status	Arming Signal	Slave 1 Trip	Slave 2 Trip	Master relay Trip	Breaker Trip Signal*
Disarmed	0	Master not influenced	Master not influenced	0	0
	0	Master not influenced	Master not influenced	1	**1**
Armed	1	0	0	0	0
	1	0	0	1	0
	1	0	1	0	0
	1	1	0	0	0
	1	0	1	1	**1**
	1	1	0	1	**1**
	1	1	1	0	**1**
	1	1	1	1	**1**

***0 = No Trip**

***1 = Trip**

The truth table of Table 17 is used to construct the trip logic diagram and equation for the Master relay. Figure 25 shows the Master relay's logic diagram with the arming signal included. This diagram must be translated into an equation (Equation 4.1), which will then be the actual setting for the element of the Master relay that will tell the circuit breaker to open. This setting will be defined as "Master relay CB TRIP."

Figure 25. Logic Diagram for Master Relay Tripping Logic



$$\text{Master relay CB TRIP} = (\text{Master trip AND Slave 1 trip AND Arming signal})$$
$$\text{OR}$$
$$(\text{Master trip AND Slave 2 trip AND Arming signal})$$
$$\text{OR}$$
$$(\text{Slave 1 trip AND Slave 2 trip AND Arming signal})$$
$$\text{OR}$$
$$(\text{Master trip AND NOT(Arming signal)}) \quad (4.1)$$

Equation (4.1) can be simplified using standard logic equation notation for gates:

$$\text{Master relay CB TRIP} =$$
$$[(\text{Mt} * \text{S1t} * \text{ARM}) + (\text{Mt} * \text{S2t} * \text{ARM}) + (\text{S1t} * \text{S2t} * \text{ARM})] + [\text{Mt} * !\,\text{ARM}] \quad (4.2)$$

Where Mt = master trip, $S1t$ = Slave 1 trip, and $S2t$ = Slave 2 trip. Equation (4.2) is the simplified voting scheme trip equation.

Slave Relay Logic

The inclusion of the voting scheme may require alterations to the slave relays' original trip equations. The slave relays must be sent the arming signal if the slaves are to operate as redundant relays when the voting scheme is disarmed. Acting as redundant relays would mean the slave relays can trip the circuit breaker when the voting scheme is disarmed. However, if the Master relay should be the only relay to trip the circuit breaker, whether the voting scheme is armed or disarmed, then the slave relays will require no alterations to their original trip equations and will not need to receive the arming signal. In this case, the only requirement for the slave relays is that their digital trip signals be sent to the Master relay.

Table 18 is the truth table for the each slave relay with the arming signal included. This table is only for the slave relays, and represents all the possible events that would cause a slave relay to trip the circuit breaker of the line.

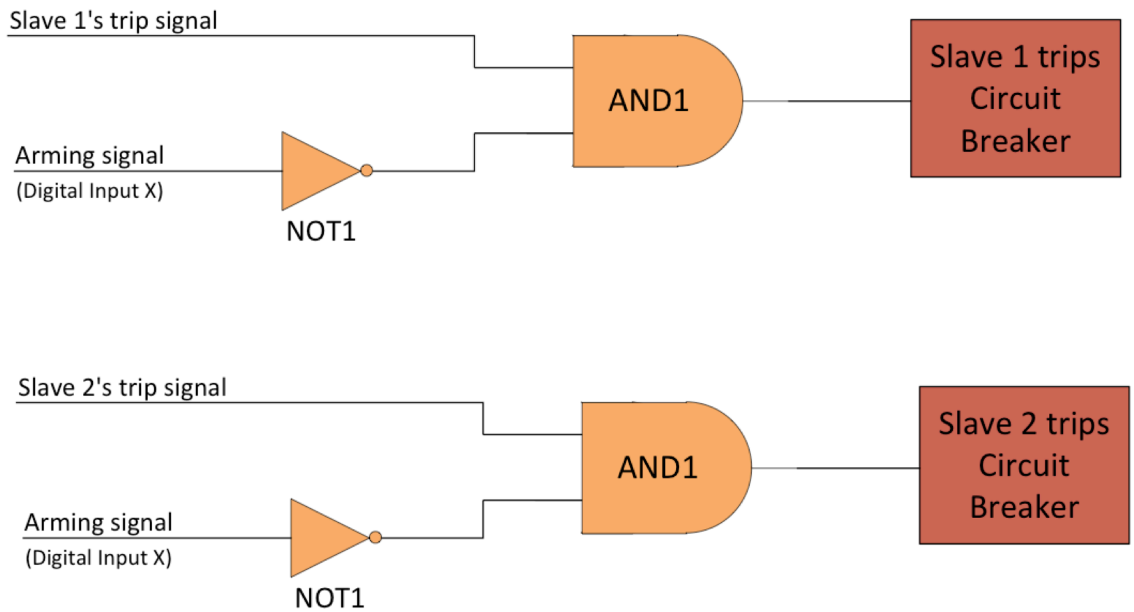
Table 18. Truth Table for Slave Relay with Arming Signal

Voting Scheme Status	Arming Signal	Slave 1 (or Slave 2) Trip	Will Slave relay trip circuit breaker?*
Disarmed	0	0	0
	0	1	**1**
Armed	1	0	0
	1	1	0

***0 = No   *1 = Yes**

The truth table of Table 18 is used to construct the trip logic diagram and equation for a slave relay. Figure 26 shows the logic diagram for Slave 1 and Slave 2 with the arming signal included for both.

Figure 26. Logic Diagram for Slave Relay Tripping Logic



The logic diagram for the slave relays is less complex than the Master relay's logic because the slaves do not have to implement the vote counting logic. Only the Master relay is responsible for implementing the vote counting logic.

The slave relay's logic diagram in Figure 26 must be translated into an equation, which will then be the actual setting for the element of each slave relay that will tell the circuit breaker to open. For the purposes of this thesis, these settings will be defined as "Slave 1 CB TRIP" and "Slave 2 CB TRIP".

$$\text{Slave 1 CB TRIP} = \text{Slave 1 trip AND NOT Arming signal} \quad (4.3)$$

$$\text{Slave 2 CB TRIP} = \text{Slave 2 trip AND NOT Arming signal} \quad (4.4)$$

Equations (4.3) and (4.4) can be simplified using standard logic equation notation for gates. Equation (4.5) and (4.6) are the simplified voting scheme trip equations for the slave relays.

$$\text{Slave 1 CB TRIP} = \text{S1t} * !\,\text{ARM} \quad (4.5)$$

$$\text{Slave 2 CB TRIP} = \text{S2t} * !\,\text{ARM} \quad (4.6)$$

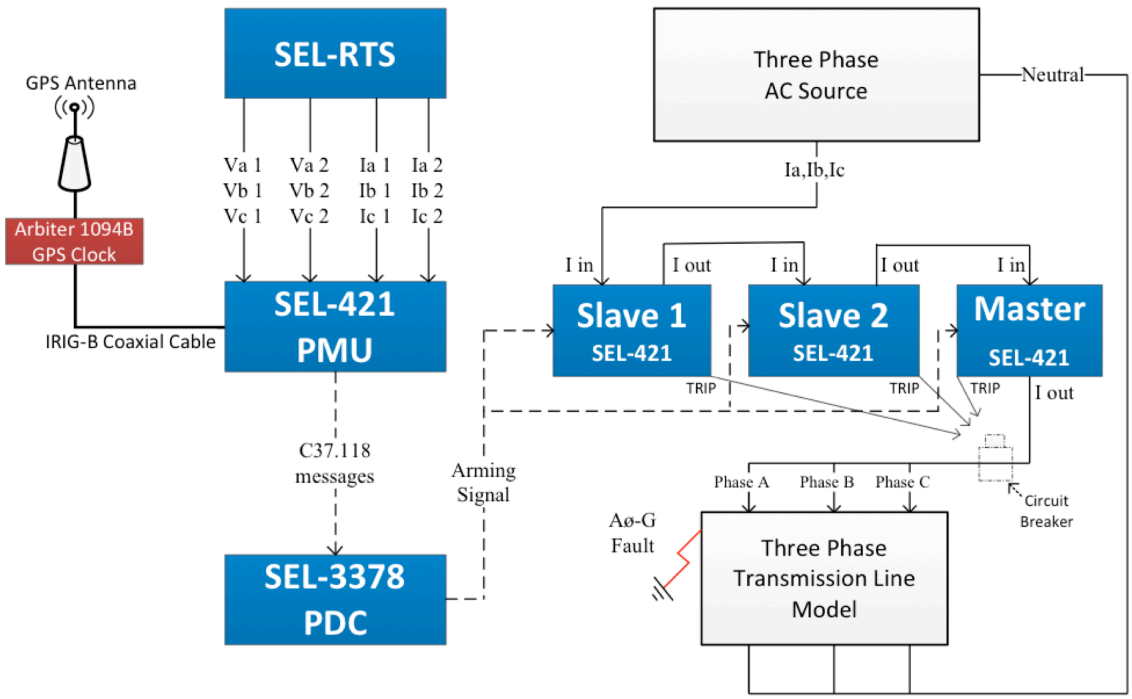
2.2.4.2 Master-Slave Voting Scheme Implementation

This section describes the different implementations of the Master-Slave configuration.

Single Manufacturer – SEL

The voting scheme implementation for single manufacturer configurations use three SEL-421 digital relays – two relays are slave relays and the third is the Master relay. The setup of this voting scheme implementation is shown in Figure 27.

Figure 27. Voting Scheme Implementation: All SEL-421 Relays



In this implementation, all three of the voting scheme relays are redundant relays when the voting scheme is disarmed. With a redundant relay voting scheme each digital relay must receive the arming signal to allow each relay to trip the circuit breaker if needed when the voting scheme is disarmed.

<u>SEL Mirrored Bits</u>

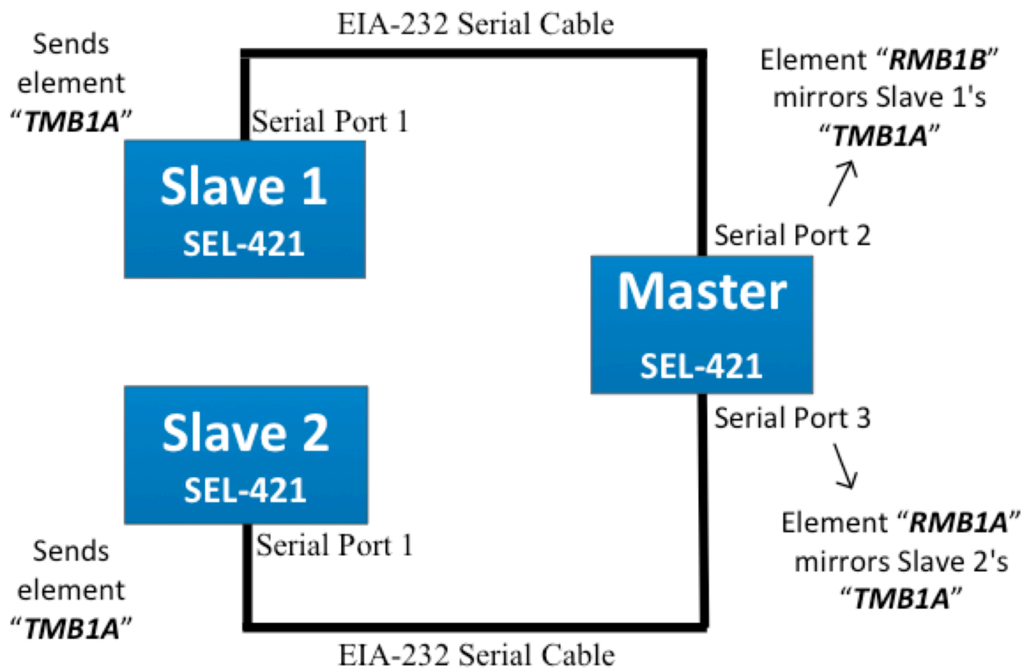
To implement SEL Mirrored Bits Communications for the trip signals, the configuration in Figure 28 was implemented.

For trip signals using this communications protocol, each SEL-421 slave relay is configured to send its trip signal through one Transmit Mirrored Bit, or TMB. With Mirrored Bits communication, the slave relays send all of their TMBs to the Master relay through an EIA-232 Serial Cable. If a slave relay trips with Mirrored Bits enabled, the Transmit bit used will assert to 1. This bit is sent through the serial cable to the Master relay, where the corresponding Receive Mirrored Bit, RMB, will receive the TMB and mirror the value of that TMB. The SEL-421 Master relay's voting scheme logic then uses the two internal RMBs corresponding to the two TMBs from the SEL-421 slave relays.

The results for this voting scheme implementation are shown in Table 19. The table shows that the all SEL, redundant, Mirrored Bits voting scheme implementation correctly operated for all tested events.

In each SEL-421 relay, the 50P1 element is the instantaneous overcurrent function. When the 50P1 element asserts, the SEL digital relay has measured a current input signal above the instantaneous overcurrent pickup setting in the relay. For all voting scheme implementations, the instantaneous overcurrent pickup for all relays was set to 1.0 Amp. For each event tested, the current input to the relays was forced above 1.0 Amp to make the 50P1 element assert for the relays that are tripped in an event.

Figure 28. Trip Signal Circuit Send Using SEL Mirrored Bits



The results in Table 19 were derived from the Sequential Event Records (SERs) from each SEL digital relay. In the two SEL-421 slave relays, the TMB1A element is the Transmit Mirrored Bit used in each slave to send the arming signal to the Master relay. The TMB1A element is assigned the trip signal of the slave relay. When the TMB1A element asserts, the Master relay's associated Receive Mirrored Bit element will assert. As shown in Figure 28, the Master relay is set to use the RMB1B element for Slave 1's trip signal and the RMB1A element for Slave 2's trip signal. When Slave 1 votes to trip, the RMB1B element in the Master relay will assert; when Slave 2 votes to trip, the RMB1A element in the Master relay will assert.

The last element in the SEL SERs is RB01. This element is the arming bit contained in the arming signal from the SEL-3378.

All of these elements are used in the voting scheme devices in this implementation, and most are in the logic equations of the relays. Equation (4.7), (4.8), and (4.9) show the logic equations for the Master, Slave 1, and Slave 2 digital relays, respectively.

$$\mathrm{SEL-421\ Master\ relay\ CB\ TRIP} = [(\mathrm{50P1} * \mathrm{RMB1A} * \mathrm{RB01}) + (\mathrm{50P1} * \mathrm{RMB1B} * \mathrm{RB01}) + (\mathrm{RMB1A} * \mathrm{RMB1B} * \mathrm{RB01})] + [\mathrm{50P1} * !\,\mathrm{RB01}] \qquad (4.7)$$

$$\mathrm{SEL-421\ Slave\ 1\ CB\ TRIP} = \mathrm{50P1} * !\,\mathrm{RB01} \qquad (4.8)$$

$$\text{SEL} - 421 \text{ Slave 2 CB TRIP} = 50\text{P1} * !\text{RB01} \quad (4.9)$$

Table 19. Results for All SEL, Redundant, SEL Mirrored Bits Scheme

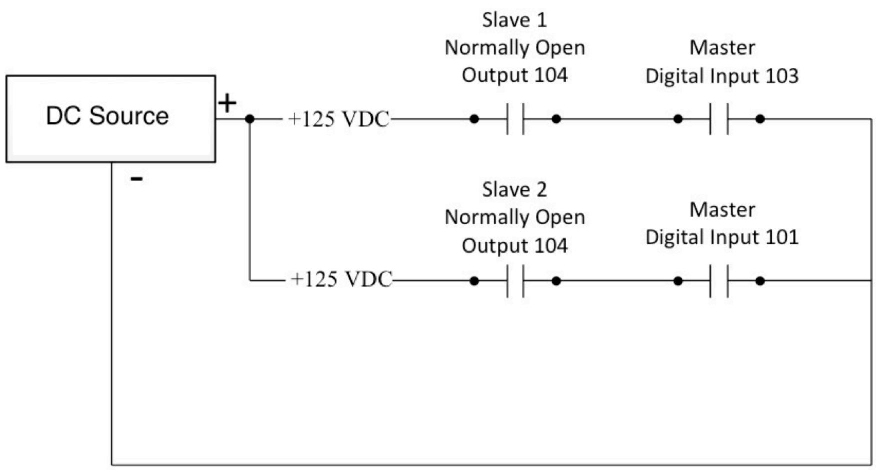
Voting Scheme Status	Arming Signal	Event	Circuit Breaker Status (Tripped = 1) (Not tripped = 0)	Which relays sent trip signals to the breaker?
Disarmed	0	**1**. Trip Slave 1, Slave 2, Master	**1**	Slave 1, Slave 2, Master
	0	**3**. Trip Slave 1, Slave 2	**1**	Slave 1, Slave 2
	0	**5**. Trip Slave 1, Master	**1**	Slave 1, Master
	0	**7**. Trip Slave 2, Master	**1**	Slave 2, Master
	0	**9**. Trip Slave 1	**1**	Slave 1
	0	**11**. Trip Slave 2	**1**	Slave 2
	0	**13**. Trip Master	**1**	Master
	0	**15**. No relays trip	0	None
Armed	1	**2**. Trip Slave 1, Slave 2, Master	**1**	Master
	1	**4**. Trip Slave 1, Slave 2	**1**	Master
	1	**6**. Trip Slave 1, Master	**1**	Master
	1	**8**. Trip Slave 2, Master	**1**	Master
	1	**10**. Trip Slave 1	0	None
	1	**12**. Trip Slave 2	0	None
	1	**14**. Trip Master	0	None
	1	**16**. No relays trip	0	None

Hard-wired: Single Manufacturer Configuration

This voting scheme implementation is the same as the previous implementation with one exception. For this voting scheme, trip signal communication was performed over the hard-wired path. To implement hard-wired trip signal communications, the circuit in Figure 29 was implemented.

A DC source was used to send the trip signals from the slave relays to the SEL-421 Master relay using hard-wired connections. For all hard-wired voting scheme implementations, each SEL-421 slave relay was configured to send its trip signal through the normally open output contact, OUT104. If a slave relay trips with this setup, its output contact OUT104 closes, providing DC voltage to a digital input on the SEL-421 Master relay. Slave 1 sends its trip signal to the Master's IN103 digital input, and Slave 2 sends its trip signal to the Master's IN101 digital input. When the DC voltage is applied to a digital input on the Master relay, the input contact will close and the digital bit corresponding to that input contact will assert to 1. The SEL-421 Master relay's voting scheme logic then uses the two digital bits corresponding to the two digital contact inputs.

Figure 29. Circuit for Hard-Wired Trip Signal Communications



The results for this voting scheme implementation are shown in Table 20. The table shows that the all SEL, redundant, hard-wired I/O voting scheme implementation correctly operated for all tested events.

The results in Table 20 were derived from the Sequential Event Records (SERs) from each SEL digital relay. After each of the 16 events in Table 20 were run, the SER was downloaded from each voting scheme relay and then analyzed to verify correct operation. It is important to note that in each SER, the oldest Relay Word bit element captured is numbered the highest, while the most recent Relay Word bit element captured is numbered 1.

In each SEL-421 relay, the TRIP element is the digital trip signal sent to the circuit breaker. When the TRIP element asserts, the SEL digital relay is directly sending the signal that will cause the circuit breaker to open.

In the SEL-421 slave relays, the OUT104 element is the trip signal of each slave relay sent to the Master relay via the hard-wired communications. When the OUT104 element asserts, the SEL digital relay sends the Master relay its trip signal decision. The OUT104 and the TRIP elements in the slave relay are not the same. An asserted TRIP element is when a slave relay is directly tripping the circuit breaker; however, an asserted OUT104 element is indication that a slave relay would trip the circuit breaker under normal operating conditions (i.e. the voting scheme is disarmed). When the voting scheme is armed the OUT104 element is a "vote" from the slave relay that is sent to the Master relay, as shown in Figure 29.

In the SEL-421 Master relay, the IN101 and IN103 elements are the digital trip signals from each slave relay input to the Master relay via hard-wired communications. The Master relay uses an asserted digital input as a tripping "vote" from the slave relay that sent the digital input. These digital inputs are used in the logic equation of the Master relay.

All of these elements are used in the voting scheme devices in this implementation, and most are in the logic equations of the relays. Equation (4.10), (4.11), and (4.12) show the logic equations for the Master, Slave 1, and Slave 2 digital relays, respectively.

Table 20. Results for All SEL, Redundant, Hard-Wired I/O Scheme

Voting Scheme Status	Arming Signal	Event	Circuit Breaker Status (Tripped = 1) (Not tripped = 0)	Which relays sent trip signals to the breaker?
Disarmed	0	**1**. Trip Slave 1, Slave 2, Master	1	Slave 1, Slave 2, Master
	0	**3**. Trip Slave 1, Slave 2	1	Slave 1, Slave 2
	0	**5**. Trip Slave 1, Master	1	Slave 1, Master
	0	**7**. Trip Slave 2, Master	1	Slave 2, Master
	0	**9**. Trip Slave 1	1	Slave 1
	0	**11**. Trip Slave 2	1	Slave 2
	0	**13**. Trip Master	1	Master
	0	**15**. No relays trip	0	None
Armed	1	**2**. Trip Slave 1, Slave 2, Master	1	Master
	1	**4**. Trip Slave 1, Slave 2	1	Master
	1	**6**. Trip Slave 1, Master	1	Master
	1	**8**. Trip Slave 2, Master	1	Master
	1	**10**. Trip Slave 1	0	None
	1	**12**. Trip Slave 2	0	None
	1	**14**. Trip Master	0	None
	1	**16**. No relays trip	0	None

$$\mathrm{SEL-421\ Master\ relay\ CB\ TRIP} = [(50\mathrm{P}1 * \mathrm{IN}101 * \mathrm{RB}01) + (50\mathrm{P}1 * \mathrm{IN}103 * \mathrm{RB}01) + (\mathrm{IN}101 * \mathrm{IN}103 * \mathrm{RB}01)] + [50\mathrm{P}1 * !\,\mathrm{RB}01] \quad (4.10)$$

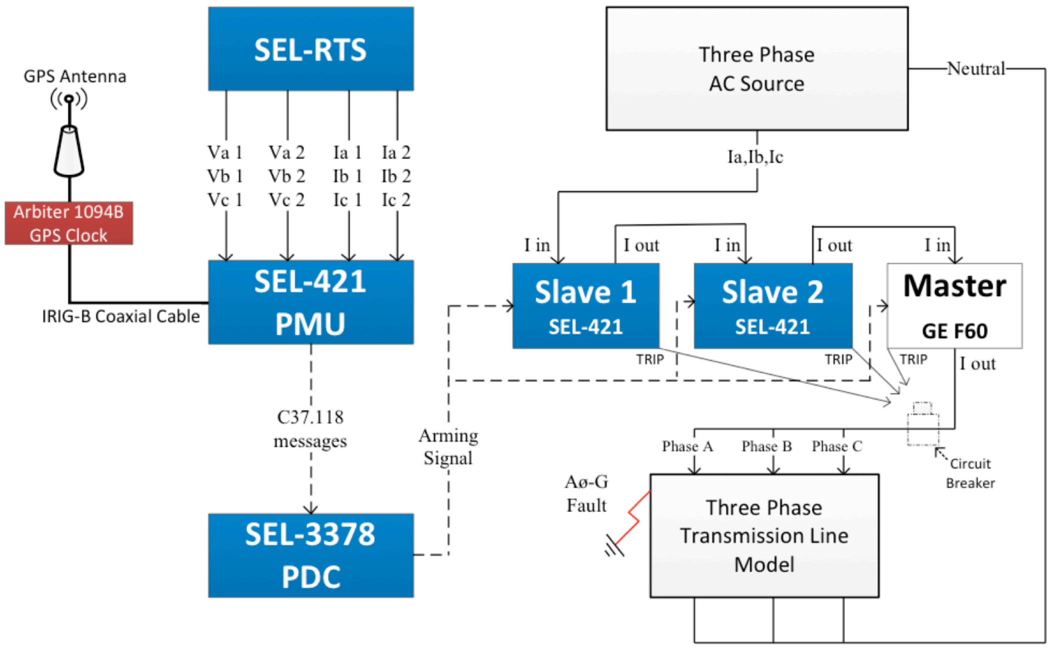
$$\mathrm{SEL-421\ Slave\ 1\ CB\ TRIP} = 50\mathrm{P}1 * !\,\mathrm{RB}01 \quad (4.11)$$

$$SEL - 421\ Slave\ 2\ CB\ TRIP = 50P1 * !\,RB01 \quad (4.12)$$

Hard-wired: Multiple Manufacturers Configuration

This voting scheme implementation uses two SEL-421 digital relays as the slave relays and a GE F60 digital relay as Master relay. The setup of this voting scheme implementation is shown in Figure 30.

Figure 30. Voting Scheme Implementation – GE Master, Redundant SEL-421 Slaves

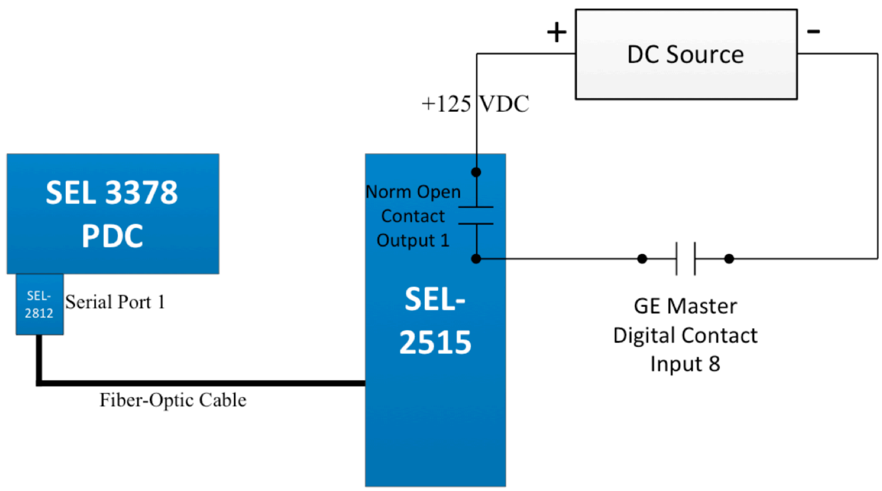


In this implementation, all three of the voting scheme relays are redundant relays when the voting scheme is disarmed. With a redundant relay voting scheme, each digital relay must receive the arming signal to allow each relay to trip the circuit breaker if needed when the voting scheme is disarmed. To send the arming signal from the SEL-3378 to the SEL-421 slave relays in this implementation, the same SEL Fast Operate Commands were sent over Ethernet through the local area network.

The SEL Fast Operate Commands only communicate with SEL digital relays; therefore, a new device was introduced into the voting scheme implementation that allows the GE F60 Master relay to receive the arming signal through SEL Fast Operate Commands. That device is the SEL-2515 Remote I/O Module. This SEL device accepts SEL Fast Operate Commands from the SEL-3378 over a fiber-optic cable and then translates those commands to hard-wired digital contact outputs. A digital contact input on the GE F60 Master relay can then be hard-wired to the appropriate SEL-2515 contact output to receive the arming signal. The circuit in Figure 31 shows this arming signal communication to the GE Master using the SEL-2515.

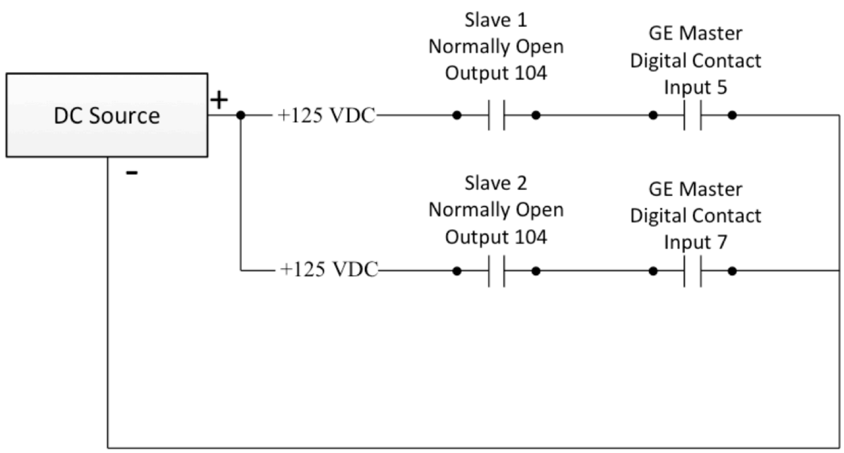
In Figure 31, a fiber-optic cable is used to send the SEL Fast Operate Commands from the SEL-3378 to the SEL-2515. Notice the addition of the SEL-2812 device on the serial port of the SEL-3378. This device is a fiber-optic transceiver that is connected to a serial port on one side and a fiber-optic cable on the other. Because the SEL-3378 does not have any fiber-optic ports, the SEL-2812 had to be used to communicate with the SEL-2515. When the SEL-3378 issues the SEL Fast Operate Commands to arm or disarm the voting scheme, the SEL-2515 receives the commands and will open or close its normally open Contact Output 1. Contact Output 1 is open when the voting scheme is disarmed and closed when the scheme is armed. When the contact is closed, DC voltage is applied to the GE Master's digital Contact Input 8 causing the internal digital bit corresponding to that input to assert. The GE Master relay's voting scheme logic uses that Contact Input 8 digital bit as the arming signal element.

Figure 31. Arming Signal Circuit for Communication Using SEL-2515



The last piece of this implementation is the trip signal communication. For all multiple manufacturer voting scheme implementations, trip signal communication was performed over the hard-wired path. To implement hard-wired trip signal communications, the circuit in Figure 32 was implemented.

Figure 32. Hard-Wired Trip Signal Circuit for Communications with GE Master



As discussed earlier, a DC source was used to send the trip signals from the slave relays to the GE F60 Master relay using hard-wired connections. The SEL-421 slave relays are the same as the previous implementations, and therefore send their trip signals through OUT104. Slave 1 sends its trip signal to the GE F60 Master relay's digital Contact Input 5, and Slave 2 sends its trip signal to the GE Master's digital Contact Input 7, as shown in Figure 25. When the DC voltage is applied to a digital input on the Master relay, the input contact will close and the digital bit corresponding to that input contact will assert to 1. The GE F60 Master relay's voting scheme logic then uses the two digital bits corresponding to the two digital contact inputs.

The results for this voting scheme implementation are shown in Table 21. The table shows that the multiple-manufacturer, redundant, hard-wired trip signal voting scheme implementation correctly operated for all tested events.

Table 21. Results for Multiple Manufacturer, Redundant Slave Scheme

Voting Scheme Status	Arming Signal	Event	Circuit Breaker Status (Tripped = 1) (Not tripped = 0)	Which relays sent trip signals to the breaker?
Disarmed	0	**1**. Trip Slave 1, Slave 2, Master	**1**	Slave 1, Slave 2, Master
	0	**3**. Trip Slave 1, Slave 2	**1**	Slave 1, Slave 2
	0	**5**. Trip Slave 1, Master	**1**	Slave 1, Master
	0	**7**. Trip Slave 2, Master	**1**	Slave 2, Master
	0	**9**. Trip Slave 1	**1**	Slave 1
	0	**11**. Trip Slave 2	**1**	Slave 2
	0	**13**. Trip Master	**1**	Master
	0	**15**. No relays trip	0	None
Armed	1	**2**. Trip Slave 1, Slave 2, Master	**1**	Master
	1	**4**. Trip Slave 1, Slave 2	**1**	Master
	1	**6**. Trip Slave 1, Master	**1**	Master
	1	**8**. Trip Slave 2, Master	**1**	Master
	1	**10**. Trip Slave 1	0	None
	1	**12**. Trip Slave 2	0	None
	1	**14**. Trip Master	0	None
	1	**16**. No relays trip	0	None

The results in Table 21 were derived from the Sequential Event Records (SERs) from each voting scheme digital relay. This voting scheme implementation is the identical to the previous hard-wired implementation except for the Master relay is now a GE F60 digital relay.

In the GE F60 Master relay, the PHASE IOC1 PKP A and PHASE IOC1 TCS A elements are the instantaneous overcurrent function elements. When the PHASE IOC element asserts, the GE digital relay has measured a current input signal above the instantaneous overcurrent pickup setting in the relay (set at 1.0 Amp).

The Virt TCS 1 element is the Master relay's trip element. When the Virt TCS 1 element asserts, the Master relay's voting scheme logic has decided to trip the circuit breaker. This element in GE Master relay represents the "Master trips Circuit breaker" box in Figure 18. To allow the GE Master relay to physically send the trip signal to the circuit breaker, the Cont TCS 1 element was used. This element provides the open or close command for the GE relay's digital Contact

Output 1. When Cont TCS 1 asserts, the Contact Output 1 closes and sends the trip signal to the circuit breaker. Whenever Cont TCS 1 is deasserted, the Contact Output 1 remains open and therefore no trip signal is sent to the circuit breaker. The Cont TCS 1 element was set equal to the Virt TCS 1 element. Therefore, the GE Master's tripping decision is ultimately sent to Contact Output 1 (Cont TCS 1) to directly trip the circuit breaker if the decision is made.

The Cont IP 5 and Cont IP 7 elements are the digital trip signals from each slave relay input to the Master relay via hard-wired communications. The Master relay uses an asserted digital input as a tripping "vote" from the slave relay that sent the digital input. As shown in Figure 24, the Cont IP 5 is the trip signal from Slave 1 and Cont IP 7 is the trip signal from Slave 2.

The last element in the GE relay's SERs is Cont IP 8. This element is the arming bit contained in the arming signal from the SEL-3378. Figure 24 shows how the GE F60 Master relays receive this Cont IP 8 arming bit.

All of these elements are used in the voting scheme devices in this implementation, and most are in the logic equations of the relays. Equation (4.13), (4.14), and (4.15) show the logic equations for the Master, Slave 1, and Slave 2 digital relays, respectively.

$$\begin{aligned}\text{GE F60 Master relay CB TRIP} = \text{Cont Op 1} = \text{Virt Op 1} = \\ [(\text{PHASE IOC1 PKP} * \text{Cont Ip 5 ON} * \text{Cont Ip 8 ON}) \\ +(\text{PHASE IOC1 PKP} * \text{Cont Ip 7 ON} * \text{Cont Ip 8 ON}) \\ +(\text{Cont Ip 5 ON} * \text{Cont Ip 7 ON} * \text{Cont Ip 8 ON})] \\ +[\text{PHASE IOC1 PKP} * !\,\text{Cont Ip 8 ON}] \qquad (4.13)\end{aligned}$$

$$\text{SEL} - \text{421 Slave 1 CB TRIP} = \text{50P1} * !\,\text{RB01} \qquad (4.14)$$

$$\text{SEL} - \text{421 Slave 2 CB TRIP} = \text{50P1} * !\,\text{RB01} \qquad (4.15)$$

2.2.4.3 Implementation for OpenPDC Configurations

This section details how the programmable logic controller (PLC) was configured to communicate with OpenPDC and how the logic was setup to perform voting or normal operation based on the arming signal. Since the voting scheme using an additional voting device does not affect the existing relay configuration, the overall configuration was simplified. Rather than the relays sending their trip signals to the breaker, the signals were fed to the PLC. The PLC then determined whether to vote and monitored for enabled inputs. If the logic was satisfied, then the PLC would trip the breaker. Figure 33 shows the interconnection between the relays, the PLC, and the tripping signal. The relays sent their trip signals to the PLC through output contacts. Based on the arming signal over Ethernet communications from OpenPDC, the PLC would determine whether to trip or restrain tripping.

<u>PLC Voting Logic</u>

Programming the PLC was performed using ladder logic in RSLogix5000; RSLogix5000 is the software that accompanies the ControlLogix controller. A set of ladder rungs were sequenced such that the PLC would determine what the operating condition was based on the decision tree output, and perform voting or normal operation for relay tripping. Figure 34 shows the ladder logic programmed into the PLC.

Figure 33. PLC Configuration

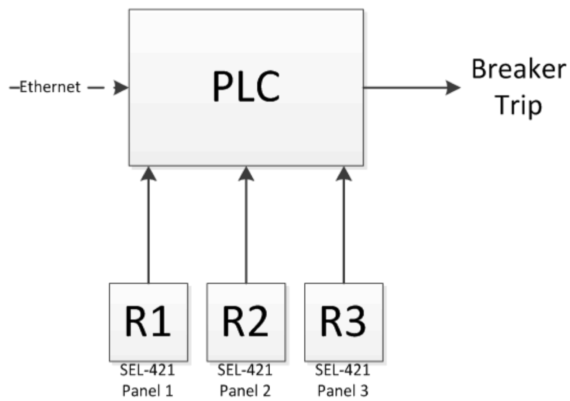
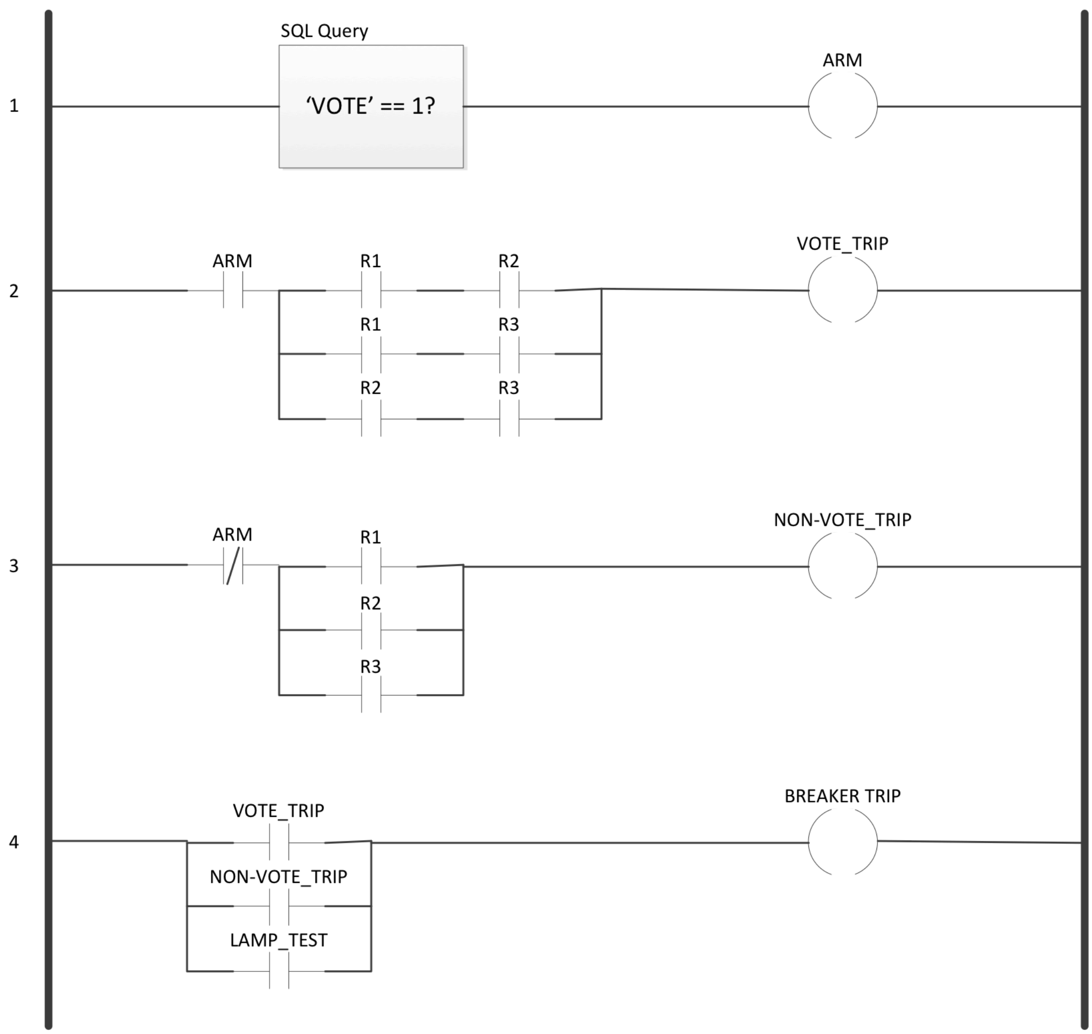


Figure 34. PLC Voting Ladder Logic



Rung 1 determines the state of the VOTE tag. If the VOTE tag is set to integer value 1, then the ARM bit goes true. Rungs 2 and 3 use this ARM bit to determine which rung to execute. If ARM = TRUE, then Rung 2 will execute and if ARM = FALSE, then Rung 3 will execute. When ARM = TRUE, the PLC performs voting of the relay trip signals using a parallel combination of 2 out of 3 relays trips in series. For any 2 out of 3 tripping scenario, the series elements will assert the input contacts and the VOTE_TRIP signal will go to TRUE. On the other hand, if ARM = FALSE and Rung 3 is executed, any individual relay input contact could complete the sequence and the NON-VOTE_TRIP element would go TRUE. Rung 4 shows that conditions in which the BREAKER TRIP output would assert. These include the VOTE_TRIP, NON-VOTE_TRIP, and LAMP_TEST contacts. The BREAKER TRIP output is the physical output contact that would apply a trip signal to the breaker itself. The LAMP_TEST input was used for testing to assure that the breaker would trip when required.

2.3 Field Implementation and Demonstration of Adaptive Relaying Algorithm

2.3.1 Technical Approach

2.3.1.1 Overview

The hardware and software implementation presented in the previous section were developed to give the Virginia Tech team the ability to support the implementation of the proposed schemes by a utility, but specifically the two utility partners for this project, SCE and PG&E. It was not expected that the laboratory implementation as developed by Virginia Tech could be adopted by the utilities without some additional adaptation to utility-grade systems; some differences in hardware and/or software implementations would likely be needed, based on the specific technologies and vendors products selected by the utilities and comprising their operational systems. Virginia Tech was able to obtain hardware similar to the hardware available to SCE for their implementation. This hardware similarity and the employment of one of Virginia Tech's students for one summer by SCE made their implementation similar to the laboratory implementation described. The PG&E implementation used a novel GE phasor data concentrator not available at Virginia Tech during the laboratory implementation of the algorithms. The support given to PG&E was limited to detailed explanation and mapping of the input signals to the GE PDC and detailed explanation of the algorithm logic and mapping of the processed signals inside the GE PDC. The section on utility implementation focuses on the SCE implementation since it was more influenced by the Virginia Tech team.

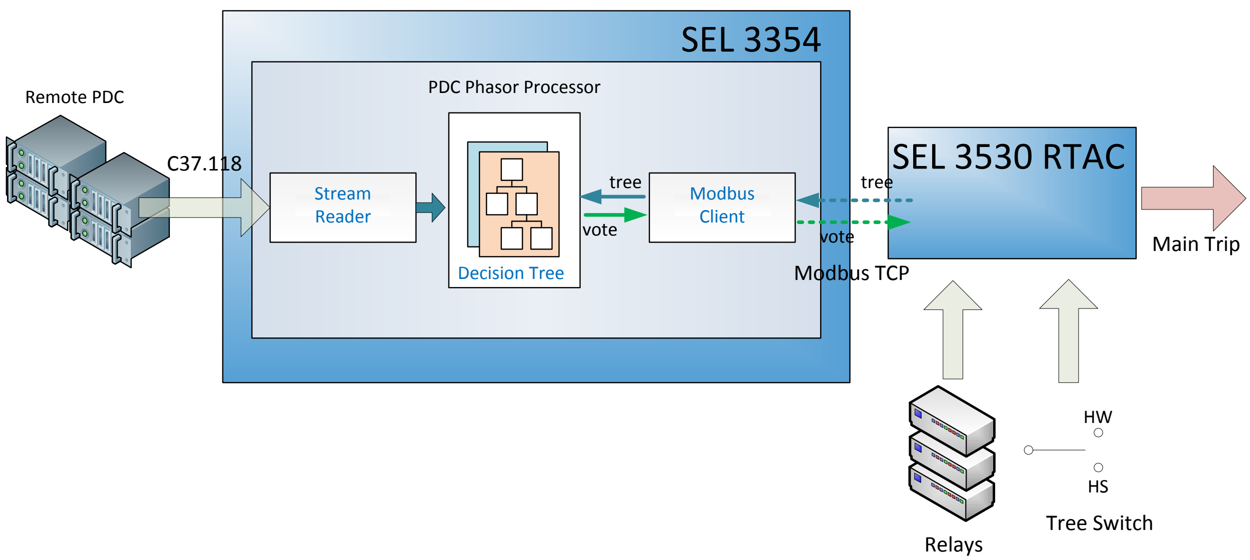
Although the previous voting scheme implementations in laboratory settings have worked as designed, they have not been without difficulties during implementation at the utilities. The SEL-3378 Synchrophasor Vector Processor tested has shown to be very versatile and powerful due to its programmable hardware. As of the last firmware update, the features of the unit, which opens multiple possibilities with phasor applications, falls short as the device can only process individual PMU streams, and never PDC streams. Data in the form of PDC streams guarantees that the information received is taken from aligned time steps, which is relevant to the decision tree application. Moreover, on two separate locations and two units of the same model, the phasor processor has been inconsistent in its operation, frequently experiencing intermittent interruptions of yet unknown causes.

The software-based PDC OpenPDC implemented at Virginia Tech, while promising in terms of its customizability, has yet to prove its reliability in a utility environment. The application's configurable adapters in itself provides the greatest advantage over the hardware phasor or PDC processors, but it appears that as the developers endeavored to cover greater versatility, the complexity of the program has resulted to compounded program glitches. On several occasions during extensive testing for possible utility implementation, OpenPDC stops retrieving the required phasor data. This has been observed on multiple installations of the software with at least two different versions. Thus an alternate implementation was sought to address the aforementioned issues for the SCE implementation.

2.3.1.2 Phasor Processor Implementation Architecture

Implementation of the adaptive voting scheme, as shown in Figure 35, was designed to contain the code for decision trees, then compiled together with the rest of the application. The required time-aligned synchrophasor data in this design is processed by a dedicated phasor data concentrator (PDC), and then streamed to the custom-made Phasor Processor application. This program parses the data for the required phasors and sends it to the appropriate decision tree. This decision tree will then determine, based on the phasor input, the operating condition of the system. The system state logic byte will be received by an SEL Real-Time Automation Controller (RTAC), which will apply the appropriate protection scheme.

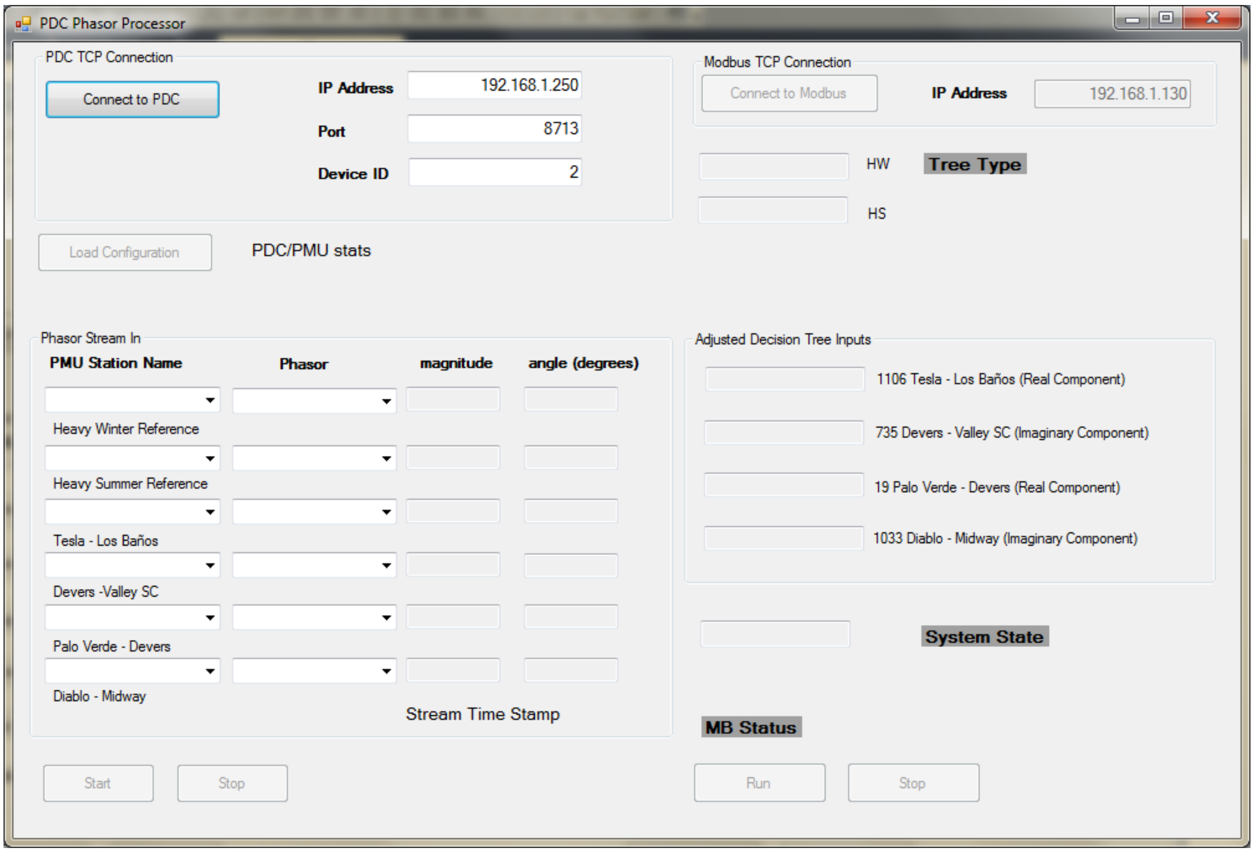
Figure 35. Phasor Processor Architecture



The PDC Phasor Processor software application, installed inside an embedded automation computing platform (SEL 3354), provides a seamless interface between input synchrophasor data, and system control. The C#-written application was designed to receive C37.118 data from a PDC, which contains an aggregate of time-aligned PMU data. The selected data from the stream is preprocessed then routed to the preprogrammed decision tree. The application then connects to the RTAC via Modbus TCP, which allows remote switching of decision trees from the RTAC; it also enables transmission of the resulting decision tree vote byte from the software program to the RTAC.

The Phasor Processor application was built to simultaneously connect to two data servers using two different protocols over the Ethernet. This enables the program to receive the phasor stream from a PDC while updating the automation controller to give the latest status of the system. All the required operations are residing inside one graphical user interface (GUI) as shown in Figure 36. Moreover, it does not require third-party applications to completely perform the task; the compiled program can be distributed as a standalone executable without external dependencies. There are three main building blocks to this single module: the PDC/PMU stream reader, the decision tree, and the Modbus TCP client.

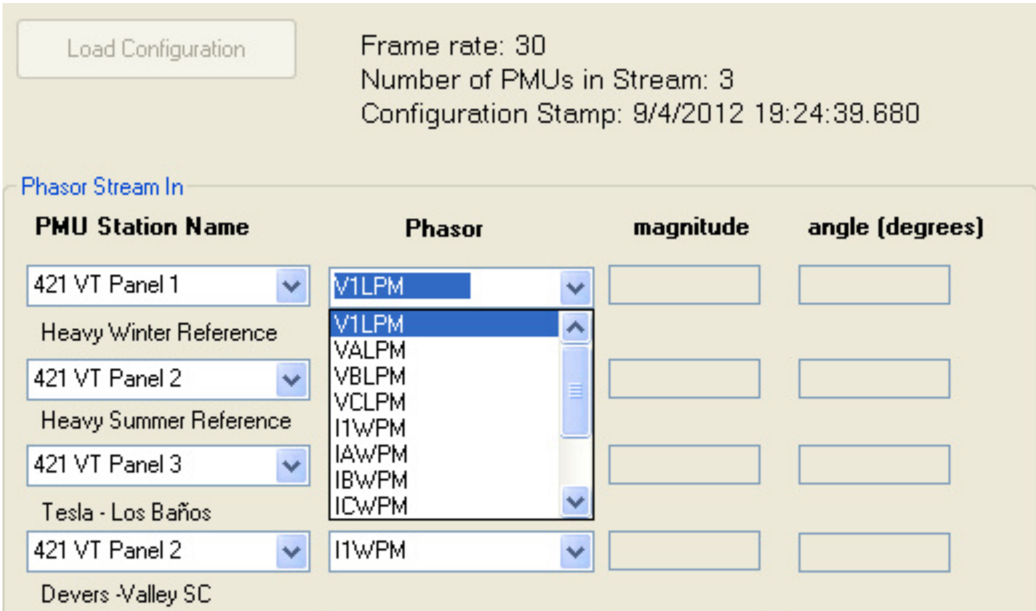
Figure 36: Phasor Processor GUI



2.3.1.3 PDC Stream Reader

The stream reader component of the PDC Phasor Processor is a C37.118 TCP client that can request a phasor data stream in real-time. It gives the user the option to select the appropriate phasor from the data stream for the decision tree subsection of the program. A connection to the server requires the server's IP address and the data port number. Upon connection, the server device number will be used to request for the data stream configuration frame. The configuration frame holds all the pertinent information from the PDC server and the contents of its outgoing data stream. The frame becomes the source of the stream reader for details such as PMU station names, phasor tags, data rate and others. Also, it enables the application to map the location of measurement data from the incoming data frames after the stream has been initiated. Since the adaptive protection scheme only requires phasor data, the GUI display only prompts users to select the proper PMU device and its corresponding phasor for a designated phasor entry in the decision tree (Figure 37). Once all the phasors have been selected, the user can begin the stream. The GUI displays the phasors in polar notation, and refreshed every second.

Figure 37. Phasor Selection from PDC Stream



2.3.1.4 Decision Tree

The phasors selected by the users from the PDC stream reader will accordingly be assigned to the decision tree, which is the second stage inside the PDC Phasor Processor application. Prior to the phasor value assignment, the program checks the value of the selected decision tree from the RTAC through the Modbus section of the program. Furthermore, the raw phasor inputs from the stream reader are preprocessed before the assignment to the decision tree. The phasors are initially set against a reference phasor angle to ensure a consistent value input to the decision tree algorithm. These reference phasors are also part of the user-selected phasors from the stream reader, where each decision tree type has an assigned reference phasor. The next stage of data preprocessing is the conversion of the phasor quantities into per unit values. Finally, the per-unit phasors are resolved into their rectangular form. The rectangular components will then be subsequently channeled as input values to the tree.

The result of the decision tree is retrieved by the Modbus sub-procedure and written into the registers of the RTAC. The output vote byte can assume three different values. A value ‘0’ for the safe or normal state, ‘1’ for the stressed state, and ‘2’ for a null vote. The null vote is a result of a tree selection error, and the vote forces the protection scheme to revert to the normal or dependability mode.

2.3.1.5 Modbus Interface

To facilitate a two-way communication between the PDC Phasor Processor program, which is installed in an SEL 3354 unit, and the RTAC, a Modbus software application was developed. The Modbus protocol employs a master-slave type of communication, where only the master can initiate queries in the form of function codes, contained in what is called a protocol data unit

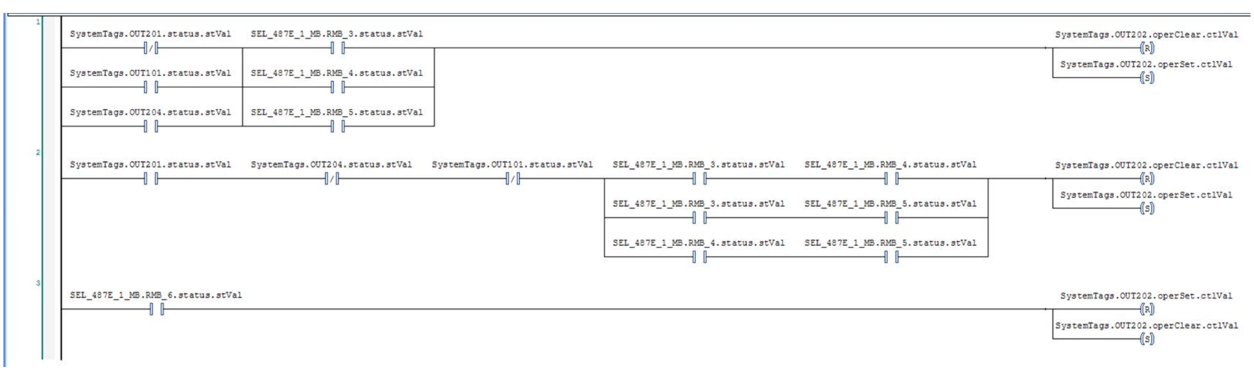
(PDU). The slaves in turn respond to these queries by reading or writing data requested by the master according to the function code received. The protocol is also accessible via TCP using a reserved system port 502. The Modbus data model consists of four primary tables that can be read-only, or read-write. These are: discrete inputs, coils, input registers and holding registers. For the purpose of the adaptive voting scheme application, input and holding registers were used.

The Modbus sub-procedure inside the PDC phasor processor was designed to both read and write registers inside the RTAC. The read-only registers, called input registers, are used for setting the tree values in the RTAC. The Heavy Winter and Heavy Summer decision tree types are each assigned a register. To enable a Heavy Winter selection for example, the Heavy Winter register byte should be a logic ‘1’ value and the Heavy Summer a register logic ‘0’ at the same time. If the two registers are simultaneously holding the same value, the phasor processor application throws a tree switch error and considers the output vote of the decision tree to be null. The decision tree output or vote is assigned to read-write registers, or holding registers. This allows the software application to modify the register value as needed, according to the output of the decision tree. The process of updating the decision tree type, and the vote value, is set to occur every second.

2.3.1.6 Real-Time Automation Controller (RTAC)

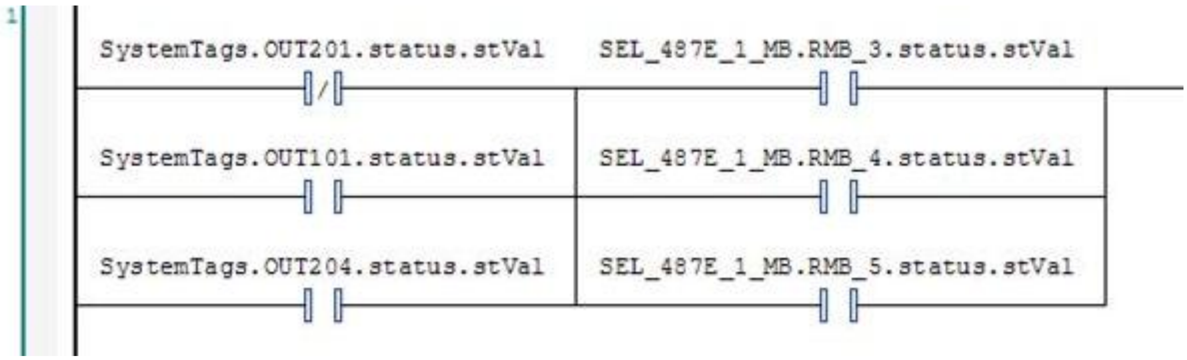
The real-time automation controller issues the main trip signal in the adaptive voting scheme implementation. It receives all relay trip logic inputs through the device’s control input panel and the vote logic via Modbus TCP. Also, the decision tree switch is assigned an input to the device’s panel and sends the user selection through Modbus TCP. The main trip of the RTAC is programmed using the ladder logic as shown in Figure 38, together with a structured text script, which is standard to programmable logic controllers (PLC) following IEC 61131-3.

Figure 38. RTAC Ladder Logic



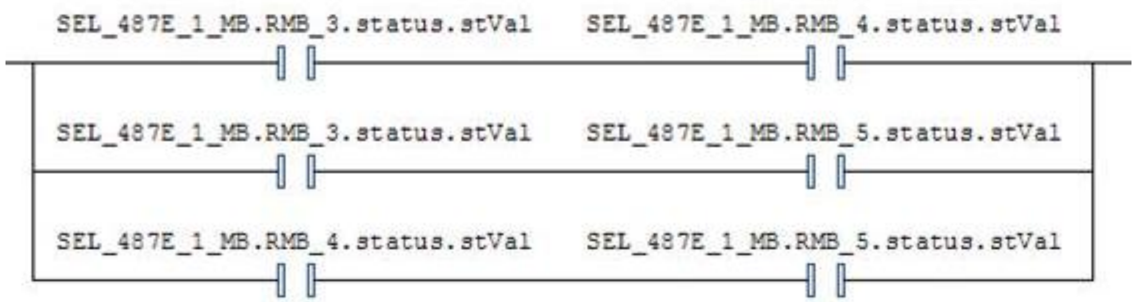
The adaptive scheme ladder logic consists of three steps or rungs (Figure 39). The first rung handles the dependability or normal mode of the protection scheme, where it allows any of the individual relays to activate the main trip of the controller. For testing purposes, the relay trip logic bits were issued using mirrored bits RMB_3, RMB_4 and RMB_5. The control output OUT201 in this rung, which is the vote contact, is assigned a normally closed contact; this allows the logic to flow through as long as the vote from the phasor processor is zero, or normal. At the far right end of the rung is the main trip, assigned to control output OUT202.

Figure 39. Normal/Dependability Mode Logic (Error contacts included)



The second rung enforces the security mode of the protection scheme, which is active when the vote byte input is high. This vote byte closes contact OUT201 in this ladder step. The main trip is issued only when at least 2 out of 3 of the relays send a trip signal. In addition to the relay and vote logic inputs, communication and switch error inputs are added to the first two rungs (Figure 40).

Figure 40. Stressed/Security Mode Logic



The control output OUT101 is assigned to communication disconnection between the RTAC and the 3354, while the control output OUT204 is assigned to the decision tree switch error. If any of these errors are active, it switches the protection scheme to dependability mode by default. The third rung in the ladder logic is set to reset the main trip. As with the relays, for testing purposes, the reset switch is enabled using mirrored bit logic, which in this case is RMB_6.

2.3.2 Adaptive Relaying Field Implementation

2.3.2.1 Introduction

In 2008, the California Energy Commission awarded a research project to Virginia Polytechnic Institute and State University on the Advanced Protection Systems Using Wide Area Measurements. This work developed advanced protection methodologies for three high-priority applications identified in the Phasor Business Case study using synchrophasor technology [Novosel et al., 2007]. Among these applications was the Security-Dependability Balance of Protection System on the Midway-Vincent No. 1 500 kV line (part of North-South Path 26 between SCE and PG&E).

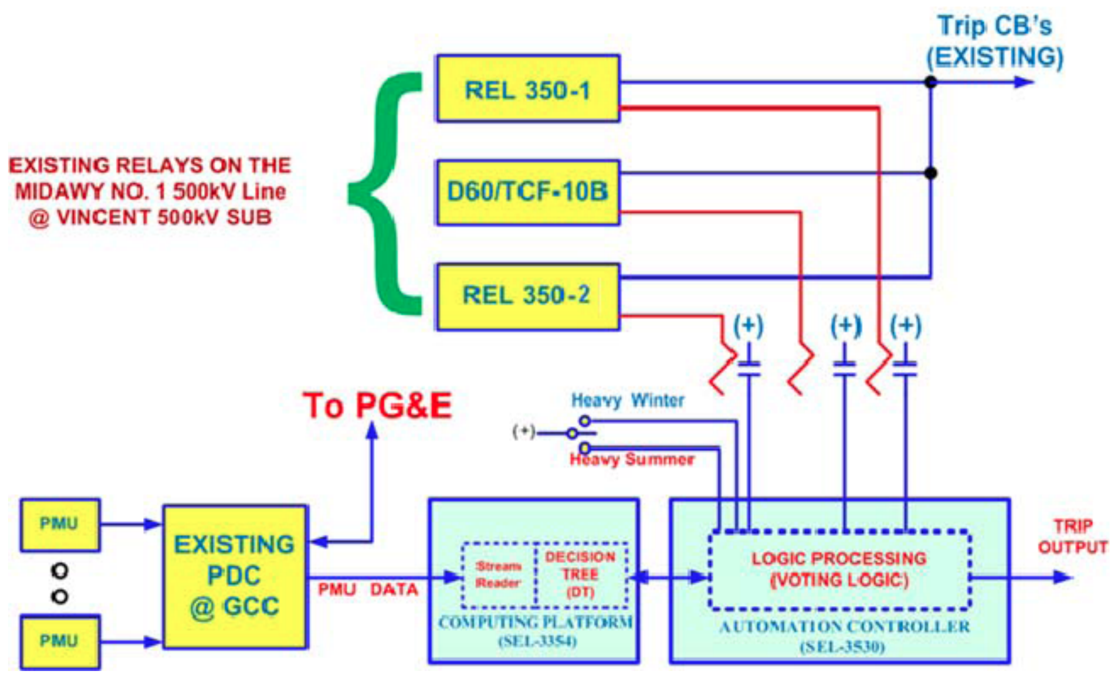
With three primary protection systems on the Midway-Vincent No.1 line, the Virginia Tech research team recommended using voting logic to implement the adaptive security/dependability philosophy. The dependable arrangement is when any of three relays sees a fault, it sends a trip signal to the breakers to clear the fault. A more secure decision would be made by requiring that two of the three relays see a fault prior to the trip signal being sent to the breakers. The advantage of the adaptive voting scheme is that the actual relays are not modified, but only the tripping logic responds to system conditions. The functional diagram of the proposed voting scheme was shown previously, in Figure 1.

The methodology used by Virginia Tech for assessing system stress from PMU data is based on Data Mining, specifically, on Decision Trees (DTs). Decision Trees can extract information from large data sets and intuitively represent the information through a series of if-else sentences [Bernabeu, 2009]. Decision Trees applied to power system stress assessment provide a classification of "stressed" or "safe" operating conditions based on a predetermined set of measurements across the network. This classification is used for arming a voting scheme that adapts its functioning based on system conditions.

2.3.2.2 Implementation of the Adaptive Protection System

The adaptive voting protection scheme conceptually developed by Virginia Tech University will be deployed at SCE Vincent Substation (also at PG&E Midway Substation). The adaptive voting scheme will be installed on Midway No. 1 500 kV line at SCE Vincent Substation (Midway-Vincent No. 1 500 kV line). The Midway No. 1 500kV line at Vincent Substation is equipped with three primary protective relays: Segregated Phase Comparison Relays (REL 350-1 & REL 350-2), and Hybrid Permissive Overreaching Transfer Trip (POTT) 3-Zone Distance/Directional Ground over Power Line Carrier (PLC) with Out-of-Step Blocking (D60/TCF-10B). The block diagram of the adaptive voting scheme design at SCE Vincent Substation is shown in Figure 41.

Figure 41. Block diagram of the Adaptive Voting Protection at SCE



Reference Bus

The previous Virginia Tech research project [Centeno et al., September 2009], used Pittsburg 500 kV bus as the reference bus. Since, at present, there is no PMU at this bus, Virginia Tech was advised by PG&E to repeat the development of the decision trees using another reference bus where there is a PMU installed on that bus.

The system models and decision tree methodology developed in the original study were used to develop new decision trees with reference bus chosen as a 500 kV bus with a PMU installed on that bus. The out-of-sample testing of the initial results revealed that the performance of the decision trees was reduced if a common reference was used for both the Heavy Summer and Heavy Winter system cases. The research team concluded that much more accurate results would be obtained if two different references are used for each of the two system cases. The Vaca Dixon (PG&E) 500 kV bus was selected as the reference bus for the Heavy Summer case, and the Tesla (PG&E) 500 kV bus was selected as the reference bus for the Heavy Winter case.

PMU Placement

To determine the critical PMU locations, an exhaustive analysis of hidden failures on the Path 15 (Midway-Vincent 500 kV line No.1, No. 2, and No. 3) was performed using a combination of a static index and a dynamic index. Table 22 displays the line current components that were used to develop the decision trees based on this analysis.

PMU placement to measure the current components indicated in the above table, and also PMUs that will serve as reference PMUs, are:

1. Devers Substation (SCE Area)
2. Tesla Substation (PG&E Area)
3. Diablo Canyon Substation (PG&E Area)
4. Vaca Dixon Substation (PG&E Area)

PMU placement locations are shown on the California map in Figure 42.

Table 22. Line Current Components for Decision Tree

Line Name	Line Current Component	Designation in Decision Tree
Devers – Valley (SC) 500 kV	Imaginary Part of the Current	II735
Palo Verde-Devers 500 kV	Real Part of the Current	IR19
Diablo Canyon – Midway 500 kV	Imaginary Part of the Current	II1033
Tesla – Los Banos 500 kV	Real Part of the Current	IR1106

Decision Trees

The work in the precursor CEC project [Centeno et al., September 2010] determined the optimal decision trees to be used for the adaptive voting scheme using both Heavy Summer (HS) and Heavy Winter (HW) models. The HS and HW decision trees are shown in Figure 43 and Figure 44, respectively.

Figure 42. Location of Synchrophasors

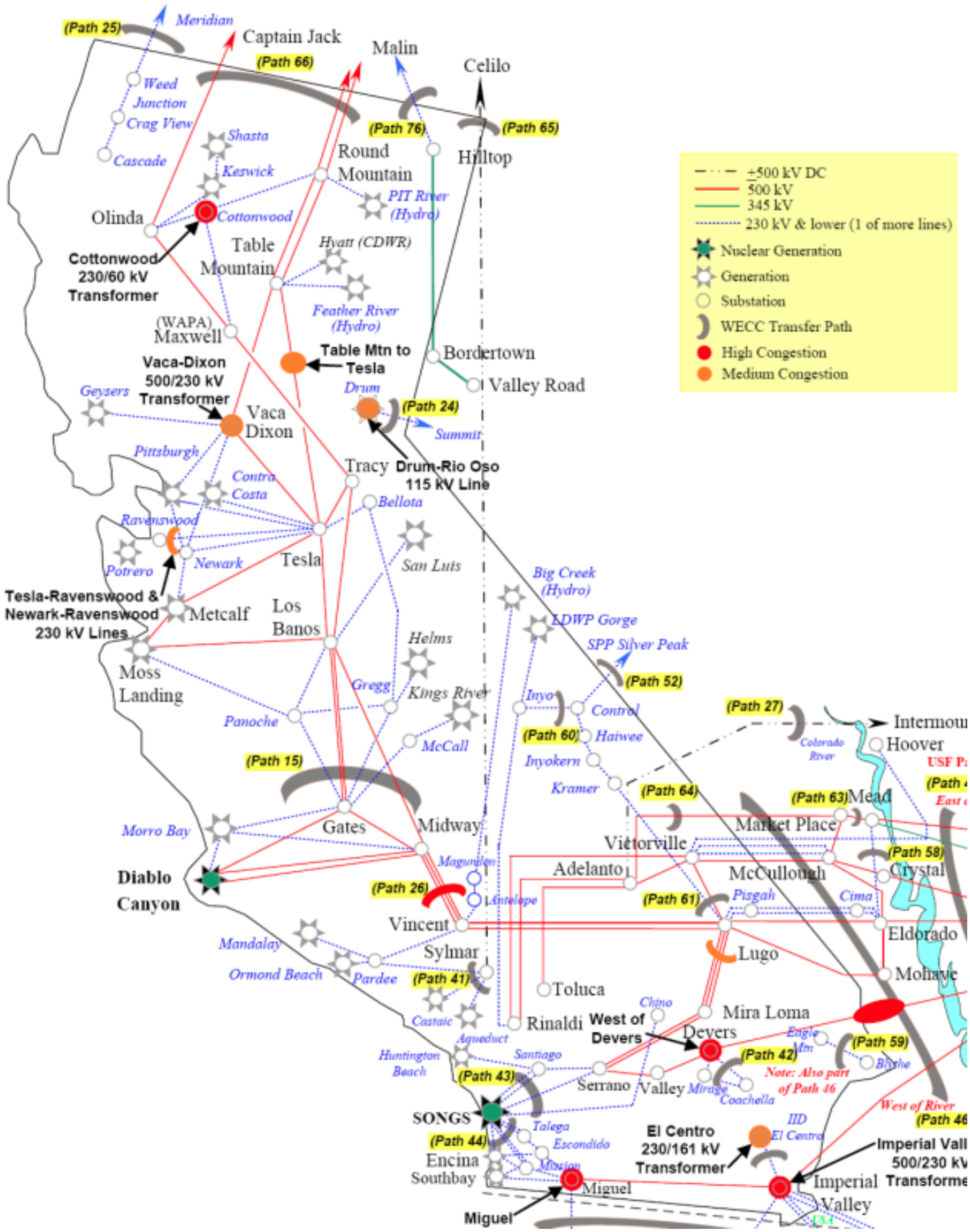
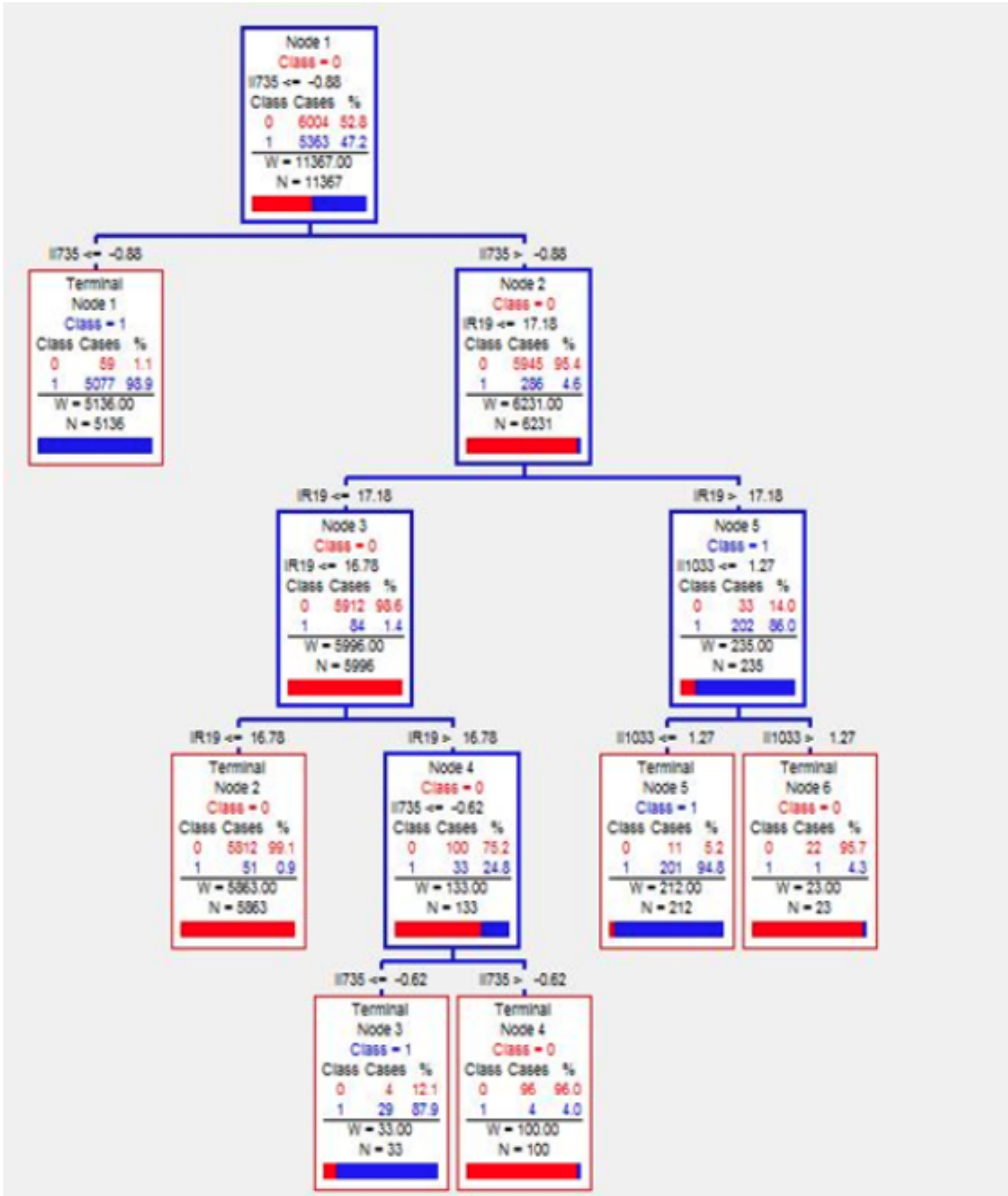


Figure 43. Heavy Summer Decision Tree



<u>New Data Link between PG&E and SCE</u>

In order to provide a secure connection for the communication of real-time PMU data between Midway and Vincent Substations, a new secure data link was built. Due to the substantial amount of logistics and cooperation necessary between SCE and PG&E technical crews to accomplish this work, the field implementation was significantly delayed. Thus, one of the key lessons learned from this project is that implementation of such a scheme involving two utility companies requires extensive planning and coordination if such delays are to avoided. Figure 45 shows the functional diagram for the required data connection.

Figure 44. Heavy Winter Decision Tree

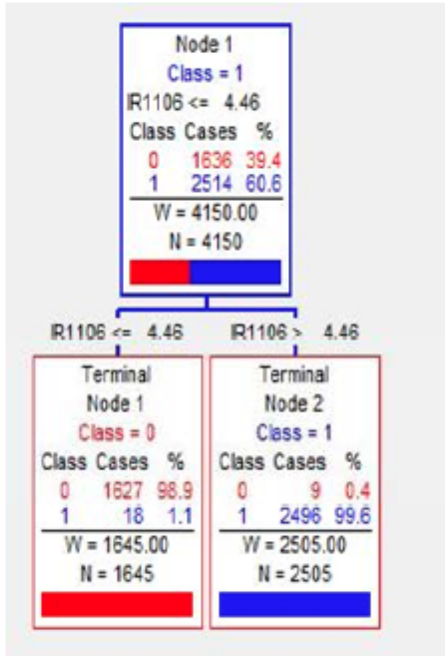
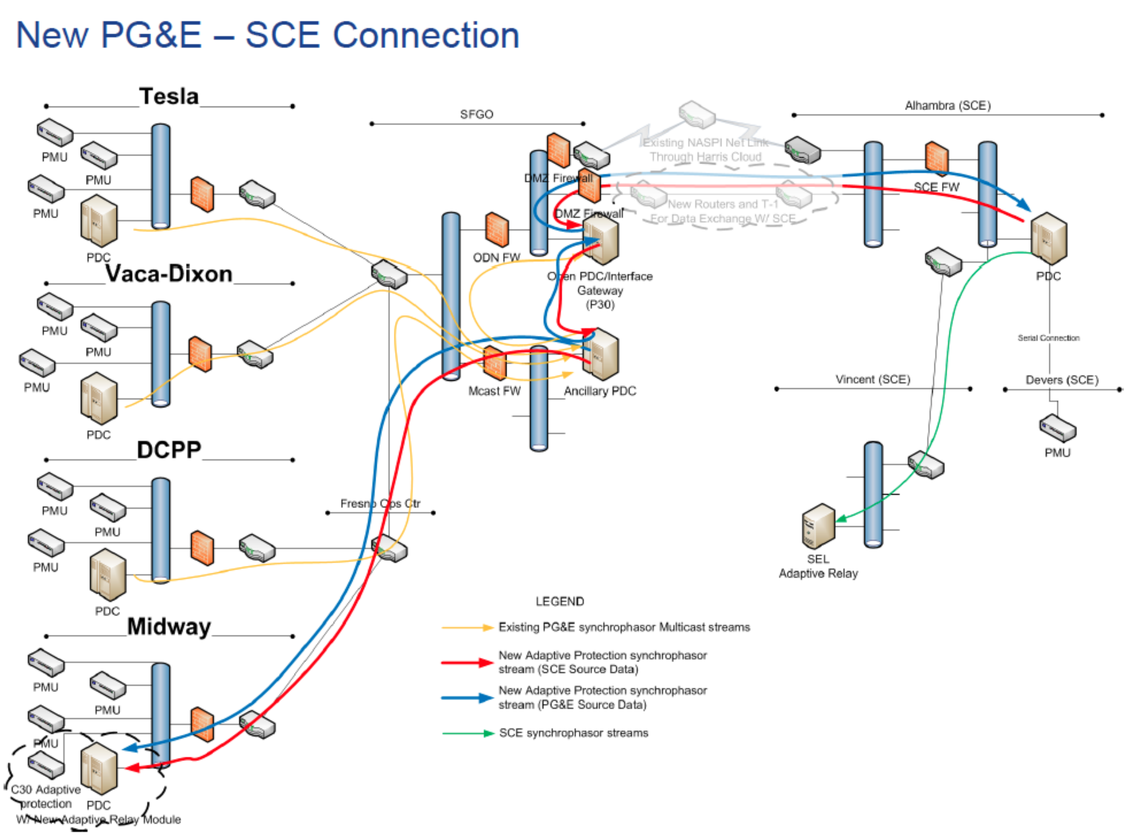


Figure 45. Real-Time Data Connection between PG&E and SCE



2.3.3 Implementation of Adaptive Relaying System at SCE

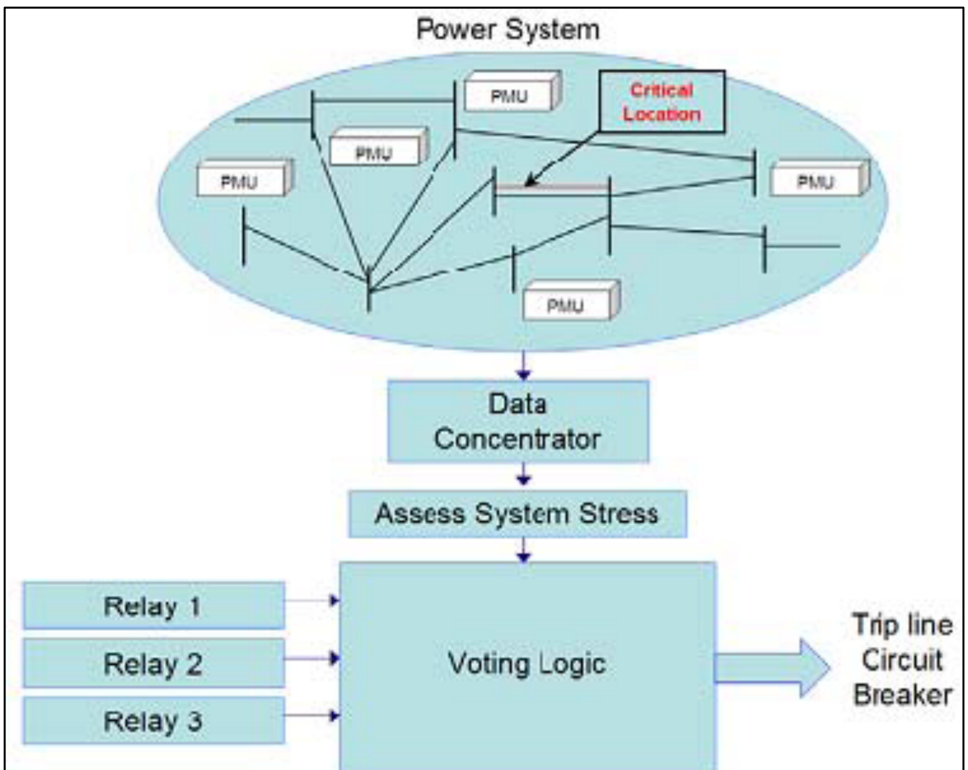
2.3.3.1 Deployment of the Adaptive Voting Scheme at SCE's Vincent Substation

The functional diagram of the SCE Adaptive Voting scheme is shown in Figure 46, and is basically the same as the generic scheme shown in Figure 1. The SCE Adaptive Voting Scheme design (Figure 47) uses two intelligent electronic devices, namely a computing platform (SEL-3354) and a real-time controller (SEL-3530). A 3-position selector switch is provided to facilitate choice of Heavy Summer and Heavy Winter seasonal cases with the center position is used to turn off the scheme (OFF position). The computing platform is used for reading PMU data and providing the appropriate data for decision tree (DT) application. The output which is a "safe" or "stressed" logic is transmitted to the real-time controller to disarm or arm the logic processor which is programmed to perform 2 out 3 voting function.

2.3.3.2 Adaptive Voting Scheme Implementation at Vincent Substation

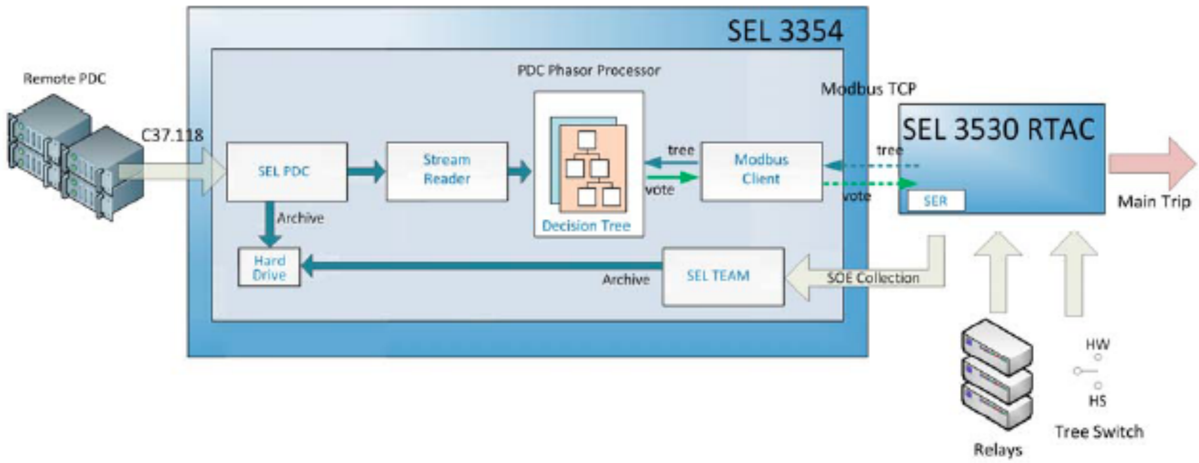
The determination of system operating condition for the voting scheme is done using decision trees. Decision trees are implemented by a custom-built phasor processor software, written in C# programming language, and installed in an embedded automation computing platform (SEL-3354). This application provides seamless interface between the synchrophasor data, and the logic processor device (Real-Time Automation Controller SEL-3530). The SEL-3530 device serves as a logic processor to implement relay trip decisions.

Figure 46. Functional Diagram of the Adaptive Protection System at SCE



Functional diagram of the adaptive protection system.

Figure 47. Architecture of the Adaptive Protection System at SCE



Within the computing platform a Phasor Processor application PDC is implemented to receive time-aligned synchrophasor data in C37.118 standard format from a remote Phasor Data Concentrator (PDC), process the data stream using the stream reader and extract real-time phasor data of specific PMUs of the transmission system. Decision trees are then fed with the proper phasors to assess the system condition (safe or stressed). The software then sends the system state to the logic processor, using Modbus protocol over Ethernet. The logic processor applies the system state to select appropriate tripping logic accordingly.

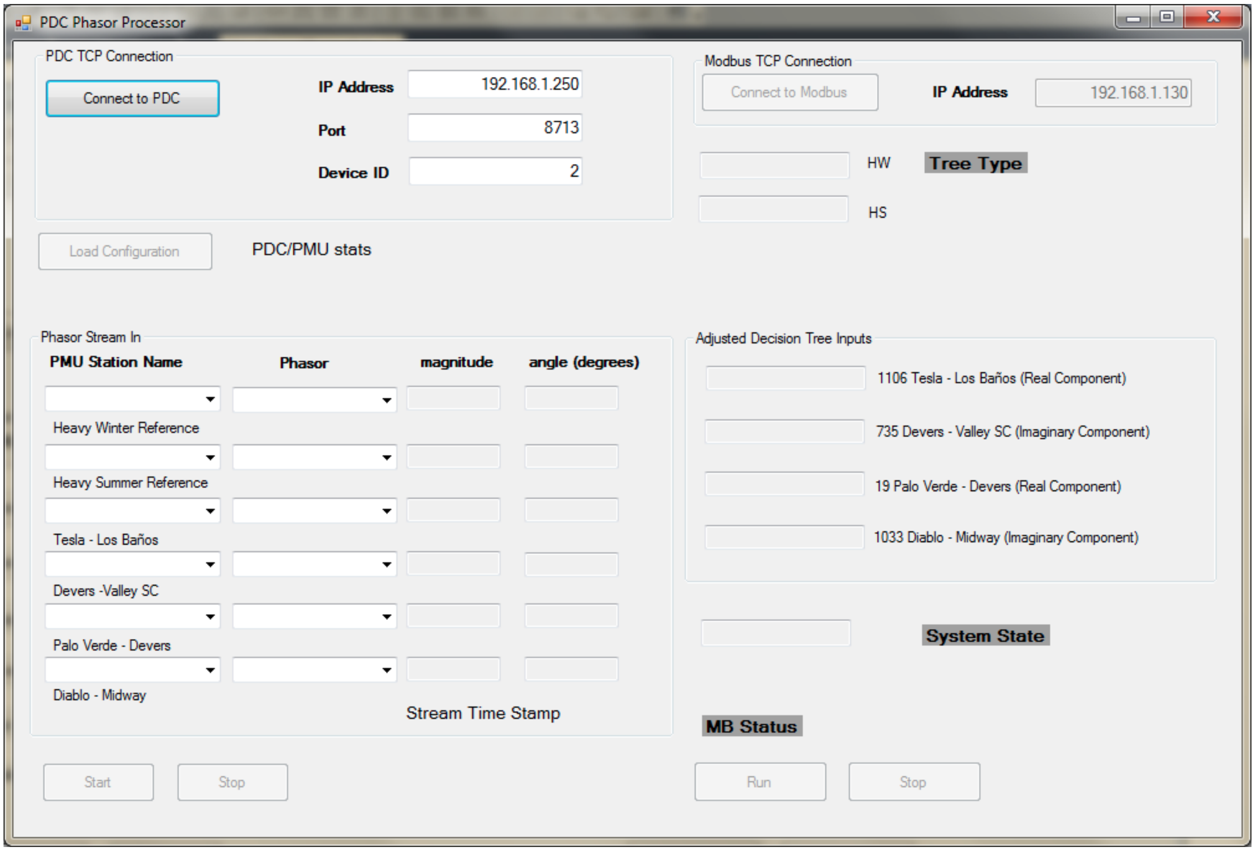
Phasor Processor Application PDC

The Phasor Processor Application PDC is developed to simultaneously connect the computing platform to two servers, the remote PDC and the logic processor, using two different protocols over the Ethernet. This enables the program to receive the phasor stream from a remote PDC while updating the logic processor about the state (safe or stressed) of the system. Figure 48 shows the Graphical User Interface (GUI) of the Phasor Processor Application PDC. The application can be compiled and distributed as a stand-alone executable without any external dependencies to any third-party applications. Three main building blocks are coded behind this user interface: the synchrophasor stream reader, decision trees and the Modbus TCP client.

Synchrophasor Stream Reader

The stream reader module of the PDC Phasor Processor is a C37.118 compliant TCP client procedure, implemented to read synchrophasor stream in real-time. The user interface enables the user to select the appropriate phasor from the data stream to be processed by decision trees. IP address of the PDC and a data port number are required by the interface in order for the software to connect to the PDC server successfully. Device TCS is also required, and is used to request the data stream configuration frame for the proper PDC output. This frame holds all the pertinent information from the PDC server and the contents of its outgoing data stream including PMU station names, phasor tags and data rate. Only PMU Station names and phasor tags are populated in the drop-down menus since the phasor processor requires phasor data only. Figure 49 shows the drop-down menus populated with PMU names and Phasor tags extracted from the configuration frame. The user selects the proper PMU device and its corresponding phasor to be processed by the decision trees before the streaming begins. The GUI displays the phasors read from the stream in polar form, and refreshes every second.

Figure 48. Phasor Processor User Interface



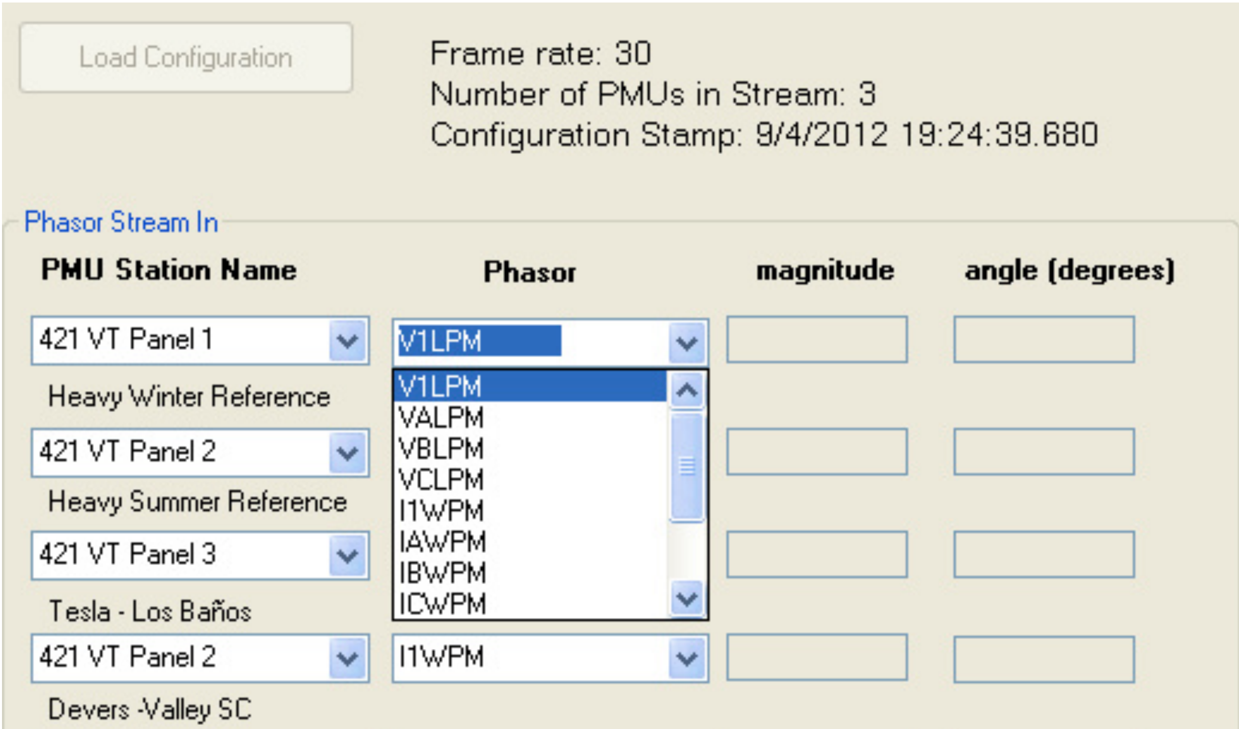
<u>Phasors Processing</u>

The phasors selected from the drop-down menus are processed before they are assigned to the proper decision tree. All phasors are referred to a common reference depending on the reference PMUs chosen from the first two drop down menus. These drop-down menus are populated with the reference PMU information. Phasor values are then converted to per-unit referred to a common base. Then, the per-unit phasors are resolved into their rectangular form before they are channeled as input parameters to the decision tree function. After processing phasors using the selected decision tree, an output representing "safe", "stressed" or "voting-off" is transmitted to the logic processor for security-dependability mode selection. Figure 50 shows the workflow of the system-state decision-making process done by the phasor processor software.

<u>Modbus Interface</u>

The communication between the computing platform that holds the phasor processor application and the logic processor is performed using Modbus protocol over a direct Ethernet communication link. The Modbus portion of the software is developed to handle this two-way communication, and to coordinate sending and receiving data across the same link. Master-slave

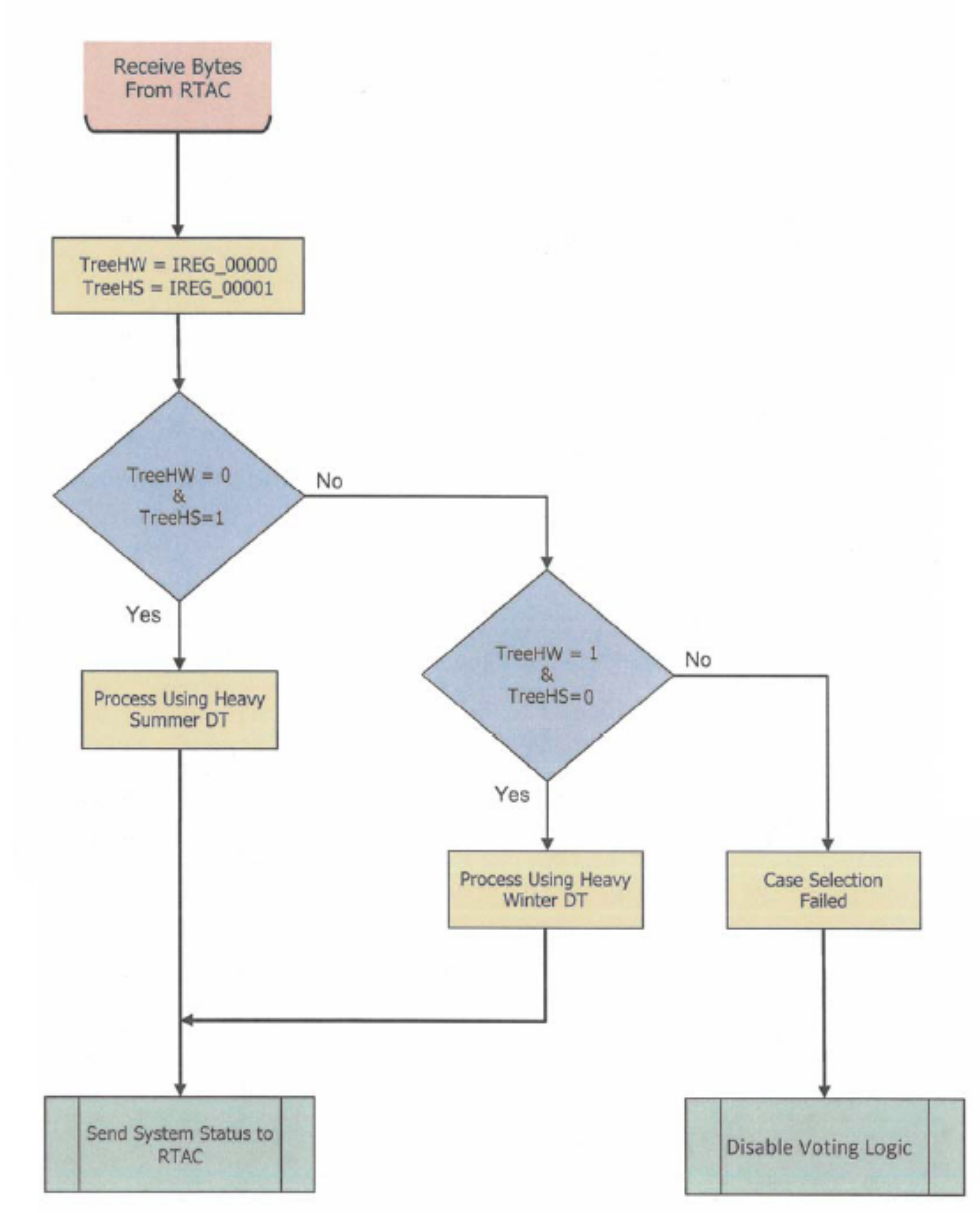
Figure 49. Phasor Selection from PDC Stream



architecture is implemented to manage the data transfer going both ways between the devices. In this implementation, only the master (logic processor) can initiate queries to send or request data, in the form of function codes, framed in a Protocol Data Unit (PDU). The computing platform, which is the slave, in turn responds to these requests depending to the function code sent by the server. The implemented protocol is accessible via TCP using the reserved system port 502. Modbus-enabled devices include data holding units called registers. Registers are used to hold data to be accessed by connected devices. For the purpose of the adaptive voting scheme application, only input and holding registers are used. Input registers are read-only registers used to hold data to be read by other devices. Holding registers are read/write registers which values can be read or altered by other connected devices.

Decision tree selection is made using a selector switch connected to the logic processor. Two registers are allocated for tree selection. Since the selection is based only on the switch position, input registers are used to hold these values. Unless the switch is turned to “off” position, one input register will hold “1” while the other will hold “0” depending on the switch position. Both registers are continuously polled by the computing platform to process phasors using the proper tree. Both input registers are updated every second to capture any change in the switch position. After processing selected phasors by the desired decision tree, the application writes a value representing the system operating state on the holding register. Using holding registers allows the software application to modify the register value as needed, according to the output of the decision tree. The holding register storing system state is updated every second.

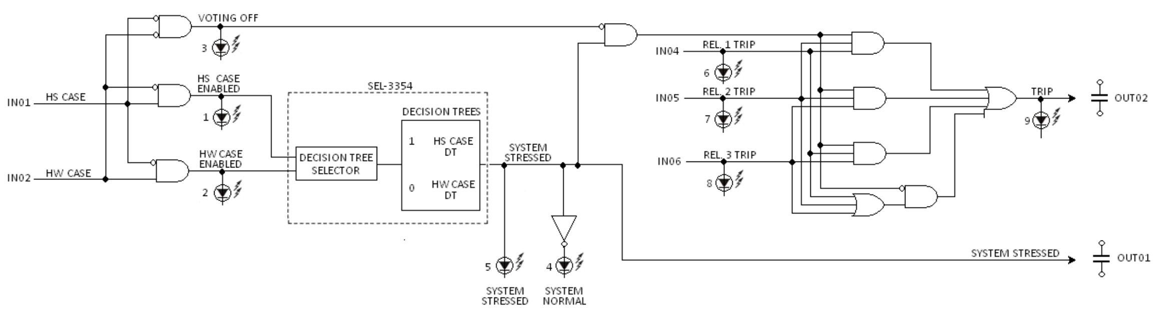
Figure 50. Phasor Data Processor Workflow



Logic Processor (SEL-3530)

When the system is operating under stressed condition, protection system should be biased towards security to ensure false trips do not happen. In that case, the trip decision is made using the logic processor with voting logic armed. Two out of the three relays must see the fault and trip in order for the logic processor to issue a signal to open the line. The Logic Processor is set to issue the main trip signal in the adaptive voting scheme implementation. This signal is programmed using a user-defined logic diagram, together with a structured text script following IEC 61131-3. Figure 51 shows the logic diagram of adaptive voting scheme.

Figure 51. Voting Scheme Logic Diagram



Relay trip outputs are fed into the logic processor through the digital contact inputs. The holding register storing the system condition is used to arm or disarm the voting scheme and is updated every second. Depending on the system condition, the logic processor either arms the voting scheme or passes any trip signal issued by any one of the relays to open the line.

As mentioned earlier, decision tree selection is made using a mechanical selector switch that provides digital input to the logic processor. Depending on the switch position, the logic processor sends a signal to the computing platform to select the desired decision tree. After selection is made, the Phasor Processor Application PDC processes system phasor data to classify the system operating condition as "stressed" or "safe." The system state is then sent back to the logic processor to arm or disarm the operation of the voting system. This process is done iteratively every second.

Two steps are involved in the logic diagram programmed inside the controller. The first step handles polling the selector switch position for decision tree selection, and then transmitting the desired selection to the Phasor Processor Application PDC using Modbus protocol. Figure 52 shows the first step in the user-defined logic used to program the processor.

The second step involves the logic processor trip decision-making. This operation depends on relays trip signals and the decision tree output representing system condition from step 1. The voting scheme is triggered in case of a "stressed" condition, and the logic processor will issue the trip signal if at least two out of the three relays sense the fault. The voting function will be disabled immediately when the system state changes to "safe" condition. Figure 53 shows the user-logic programmed in the logic processor to implement the voting scheme.

Figure 52. Decision Tree Selection User Logic

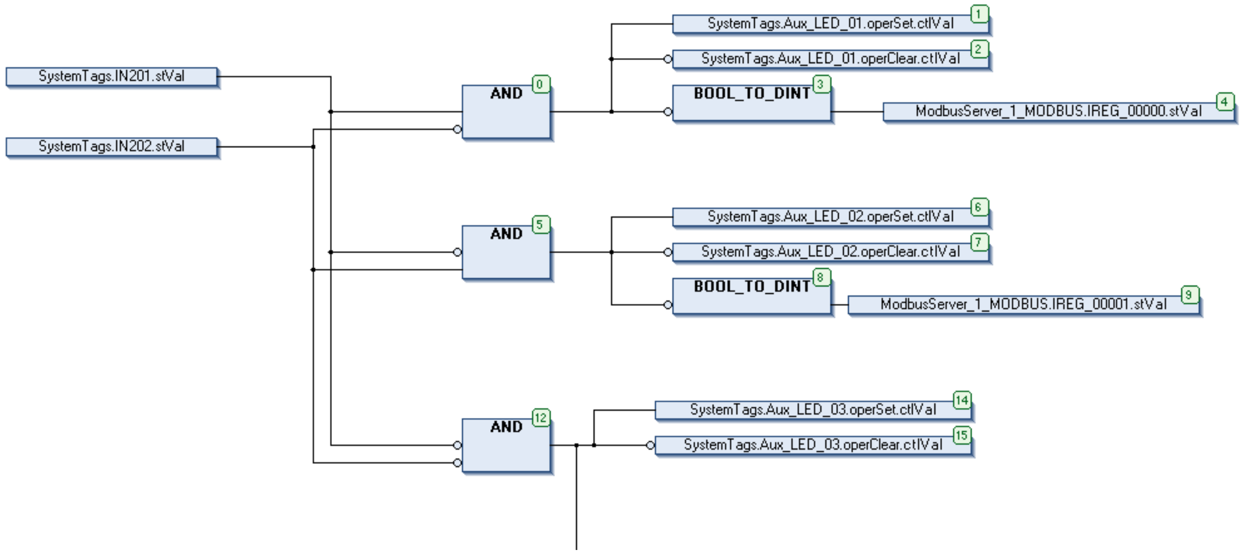
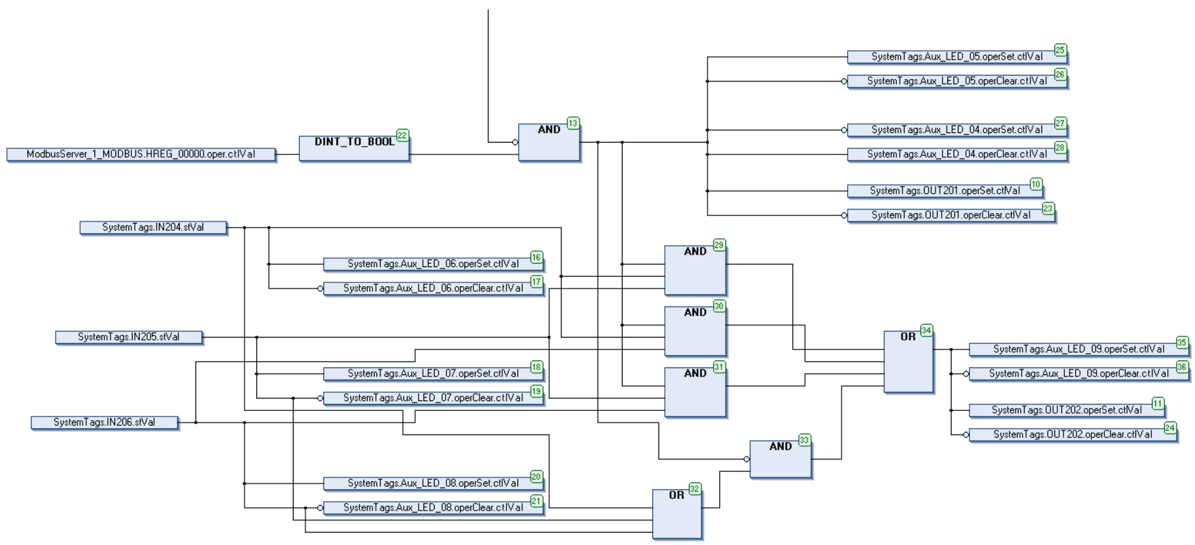


Figure 53. Security-Dependability Switching User Logic



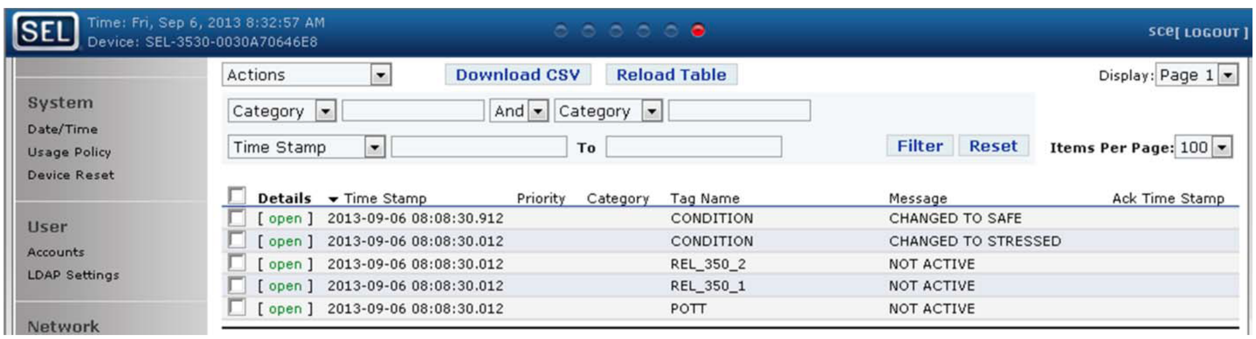
The 3-position selector switch, used to select the desired decision tree, has an “off” position which disables the voting function of the controller. When the switch is moved to the “off” position, the controller trip output will be activated if any relay sends a trip signal, regardless of the system operating state. Hence, the computing platform operation can be skipped and the controller can process the trip logic without receiving the system state. This is implemented in the logic processor user-logic and can be observed from the interconnection between the two steps. This connection bypasses the communication with Phasor Processor Application PDC software and allows the processor to perform the trip logic with the voting function disabled.

SOE (Sequence-of-Events) Recording and Collection

The logic processor (real-time automation controller RTAC) provides built-in Sequence of Events Recording (SER) feature. All inputs, variables, registers and outputs are monitored by the processor software and stored in variables called tags. Recording of RTAC tags can be enabled under SER tab for the desired tags. Time-stamped recorded data can be retrieved using the web HMI (Human-Machine-Interface) of the controller, or using the ODBC database.

RTAC SER function captures the change of any SOE monitored tag and records the present time-stamped value of it. For the purpose of this project, the RTAC SER code is rewritten using IEC code to alter the function of SER. The updated SER stores the present value of system condition tag and all relay trip tags when system condition changes to “stressed” or any trip signal is issued by relays. In addition, SER captures release of trip signals, system condition change to “safe,” and change in the selector switch position and records the present value of the changed tag only. Figure 54 shows the system tags recorded by RTAC after the system condition changed from “safe” to “stresses” and then back to “safe” again.

Figure 54. RTAC SOE Viewer Using Secured Web HMI Interface



Event reports are moved to a hard drive that is external to the logic processor to avoid records overwriting. SOE reports are collected by AcSELerator TEAM plug-in (SEL- 5045) and stored in the computing platform hard drive. AcSELerator TEAM polls the RTAC frequently for event reports and, if there are any, moves them to a preconfigured archiving path (Figure 55). SOE records can be accessed through the AcSELerator Quickset under TEAM tab. Event data can be exported and archived manually.

Synchrophasor Data Archiving

Synchrophasor data, sent by the remote PDC, are recorded continuously and can be manually archived on the compact flash drive of the computing platform. Phasor data collection is performed by software PDC (SEL-5073) installed on the same machine. The PDC continuously receives the pre-aligned synchrophasor stream sent by the remote PDC and collects phasor measurements for data recording. At the same time, the software PDC will be sending out a stream with C37.118 standard format to the Phasor Processor Application PDC.

This software PDC runs in the background as a Windows service, and can be configured using the PDC Assistant interface. Archives can be retrieved manually using this software, as well (Figure 56).

Figure 55. AcSELerator Quickset TEAM SOE Viewer

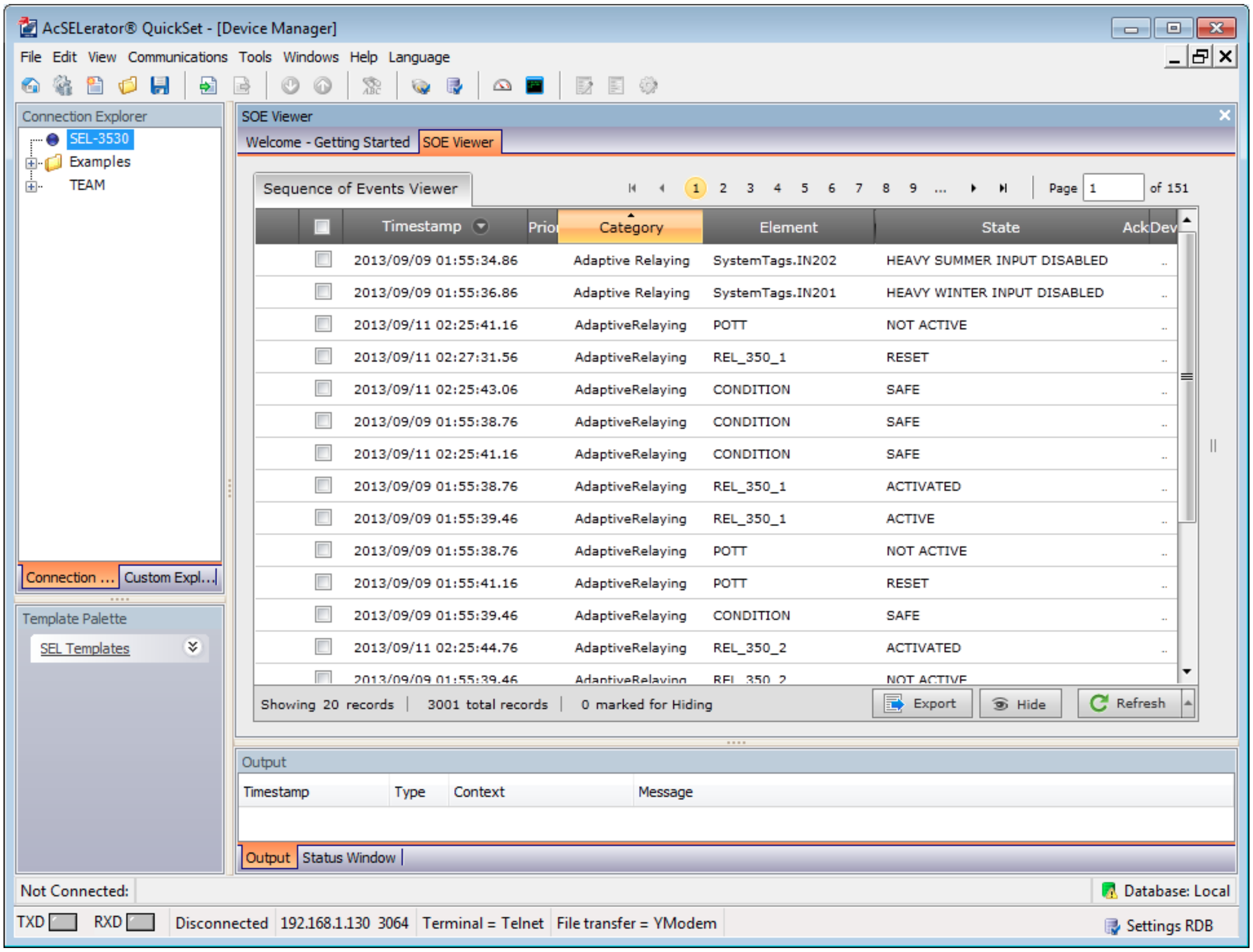
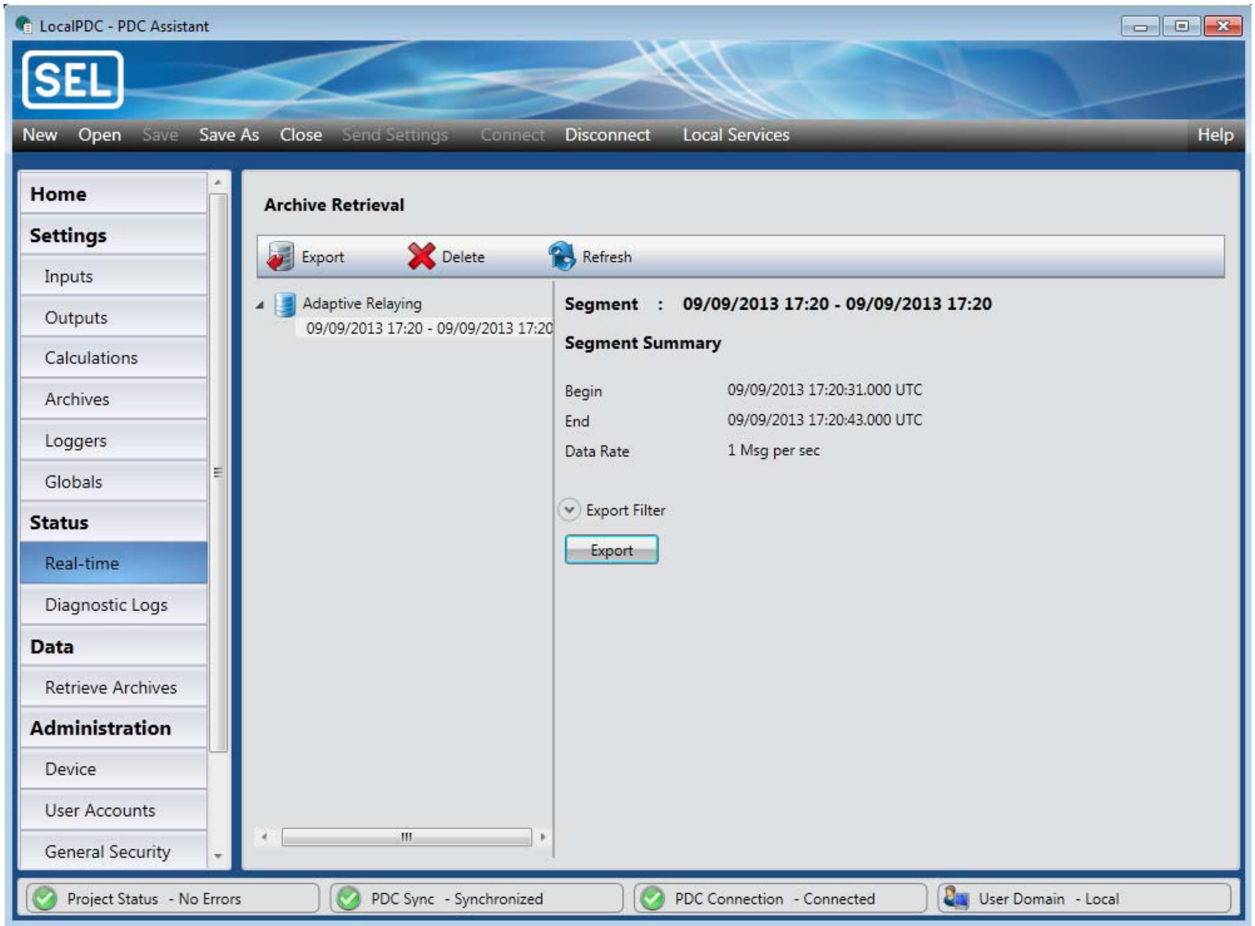


Figure 56. PDC Assistant Archive Retrieval Window



2.3.4 Implementation of the Adaptive Relaying System at PG&E

2.3.4.1 Overview

The PG&E design (see Figure 57 and Figure 58) uses two intelligent electronic devices, namely a Phasor Data Concentrator mounted with an Application Module (P30) and a Controller System (UR-C30). A 3-position selector switch is provided to facilitate choice of Heavy Summer and Heavy Winter seasonal cases with the center position is used to turn off the scheme (OFF position). The P30 platform is used for reading PMU data and providing the appropriate data for decision tree (DT) application. The output, which is a "safe" or "stressed" logic, is transmitted to the Controller to disarm or arm the logic processor which is programmed to perform the "2 out of 3" voting function.

Figure 57. Architecture of PG&E Adaptive Voting System

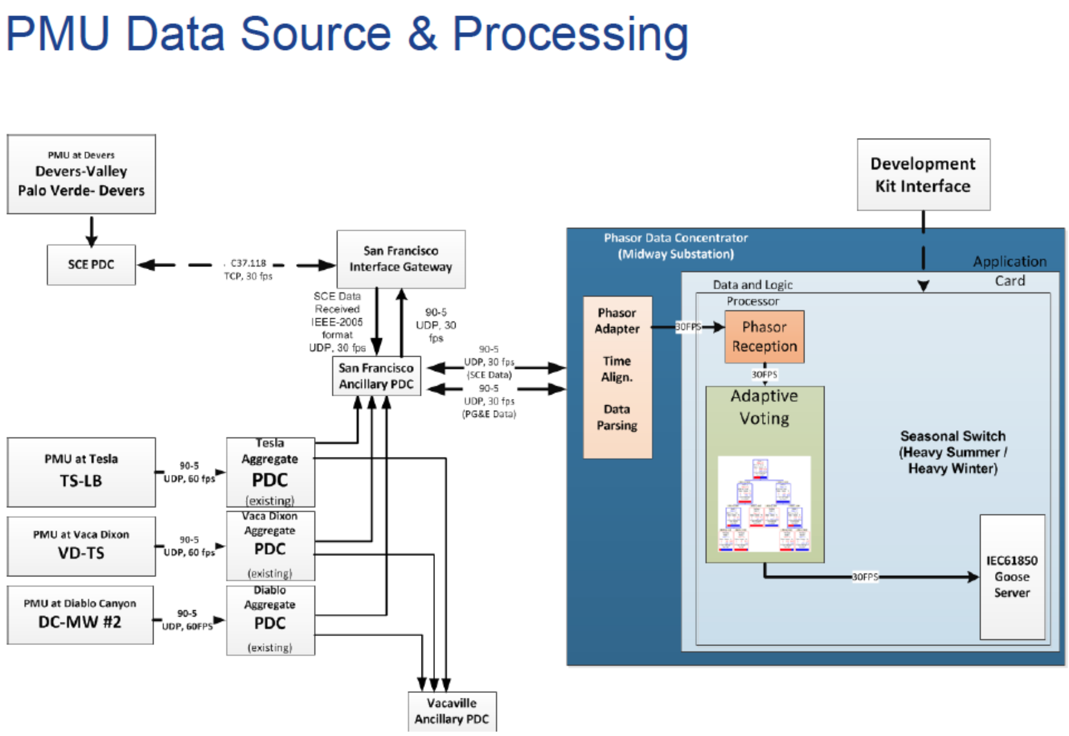
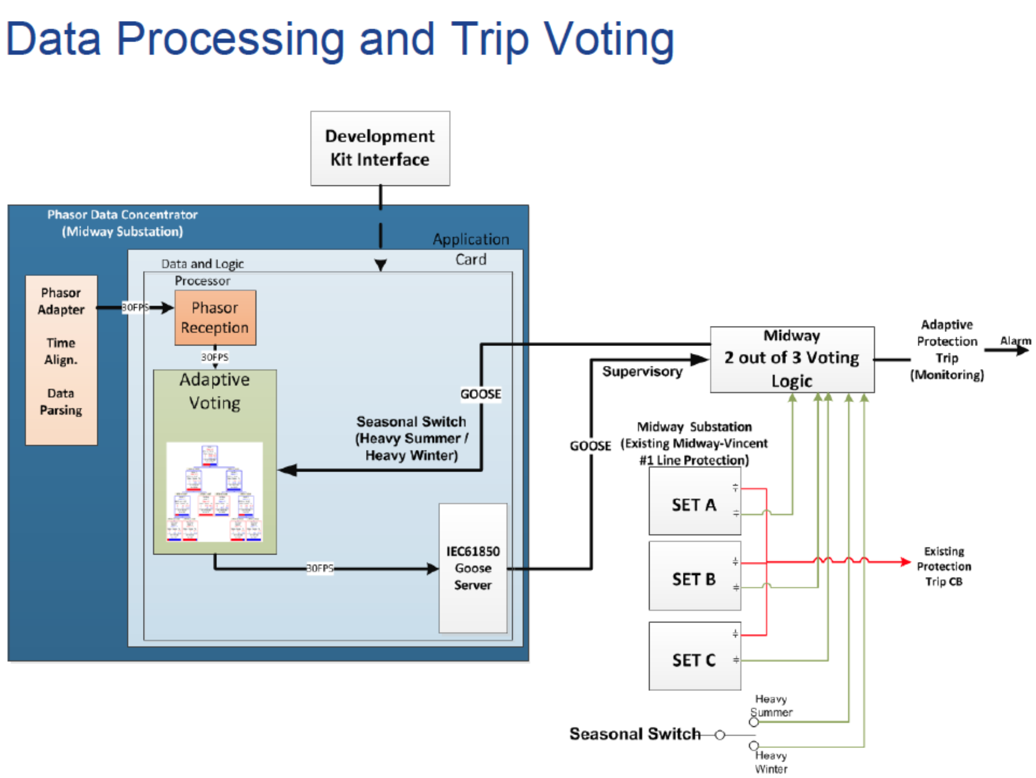


Figure 58. Implementation of Voting System



2.4.3.2 Adaptive Voting Scheme Implementation at PG&E Midway Substation

The determination of system operating condition for the voting scheme is done using decision trees. Decision trees are implemented by custom-built logic written in Straton Development language, and installed in the Application Module of the P30. This Application Module provides a seamless interface between the synchrophasor data and the Controller device (C30). The Application Module serves as a logic processor to implement the relay trip decisions.

Within the P30 device, the Synchrophasor Processor receives and time-aligns synchrophasor data in IEC 61850-90-5 format from a Control Center Phasor Data Concentrator (PDC), processes the data stream and concatenates the Decision Tree necessary input information into a single C37.118 Data Stream to feed the Application Module. Decision trees are then fed with the proper phasors to assess the system condition (safe or stressed). The Application Module then sends the system state to the logic processor, via IEC 61850 GOOSE message on status change. The Controller System (C30) applies the system state to select appropriate tripping logic accordingly.

<u>Phasor Processor Application PDC</u>

The Phasor Processor Application PDC is developed to simultaneously connect the computing platform to two servers, the remote PDC and the logic processor, using two different protocols over the Ethernet. This enables the program to receive the phasor stream from a remote PDC while updating the logic processor about the state (safe or stressed) of the system. Figure 7 below shows the Graphical User Interface (GUI) of the Phasor Processor Application PDC. The application can be compiled and distributed as a stand-alone executable without any external dependencies to any third-party applications. Three main building blocks are coded behind this user interface: the synchrophasor stream reader, decision trees and the Modbus TCP client.

<u>Synchrophasor Processor</u>

The Synchrophasor Processor module of the P30 is a dual IEC 61850-90-5 and C37.118 compliant TCP/UDP client, implemented to read synchrophasor stream in real□time. For Adaptive Voting Scheme, IEC 61850-90-5 has been selected as the interface on the input side while C37.118 has been privileged for interfacing with the Application Module. The user interface enables the user to select the appropriate phasor from the data stream to be processed by decision trees. IP address of the PDC and a data port number are required by the interface in order for the software to connect to the PDC server successfully.

Device ID is also required, and is used to request the data stream configuration frame for the proper PDC output. This frame holds all the pertinent information from the PDC server and the contents of its outgoing data stream including PMU station names, phasor tags and data rate. Figure 59 shows the drop-down menus populated with PMUs names and Phasors tags extracted from the configuration frame. Using these drop downs, the user selects the proper PMU device and its corresponding phasor to be processed by the decision trees before the streaming begins. The GUI displays the phasors read from the stream in polar form, and refreshes every second.

Phasor Processing

The phasors selected from the drop-down menus are processed before they are assigned to the proper decision tree. All phasors are referred to a common reference depending on the reference PMUs chosen from the first two drop down menus. These drop-down menus are populated with the designated reference PMU. Phasor values are then converted to per-unit referred to a

Figure 59. Phasor Selection from PDC Stream

SETTING	PARAMETER		
Function	Enabled		
Name	PGE_Data		
Description			
Protocol	IEC TR 61850-90-5		
Server IP Address	10.128.41.41		
CFG-2 TCP Port	4720		
ID Code	2001		
MsvID	PGE_Data-2001-P		
Data Timeout	2000 msec		
PMU Configuration Frame			
PMU Configuration	Read		
Input Data Rate	30 frames per sec		
	PMU 1	**PMU 2**	**PMU 3**
Station Name	VD_511_B3_2P_AC	TS_511_B2_2P_AC	DC_511_B3_2P_AC
ID Code	50532	51022	52332
Phasor Channel 1 Name	VD_TM_VA	TS_LB_VA	DC_MW3_VA
Phasor Channel 2 Name	VD_TM_VB	TS_LB_VB	DC_MW3_VB
Phasor Channel 3 Name	VD_TM_VC	TS_LB_VC	DC_MW3_VC
Phasor Channel 4 Name	VD_TM_IA	TS_LB_IA	DC_MW3_IA
Phasor Channel 5 Name	VD_TM_IB	TS_LB_IB	DC_MW3_IB
Phasor Channel 6 Name	VD_TM_IC	TS_LB_IC	DC_MW3_IC
Phasor Channel 7 Name	VD_TM_V1	TS_LB_V1	DC_MW3_V1
Phasor Channel 8 Name	VD_TM_V0	TS_LB_V0	DC_MW3_V0
Phasor Channel 9 Name	VD_TM_I1	TS_LB_I1	DC_MW3_I1
Phasor Channel 10 Name	VD_TM_I0	TS_LB_I0	DC_MW3_I0
Phasor Channel 11 Name	N/A	N/A	N/A

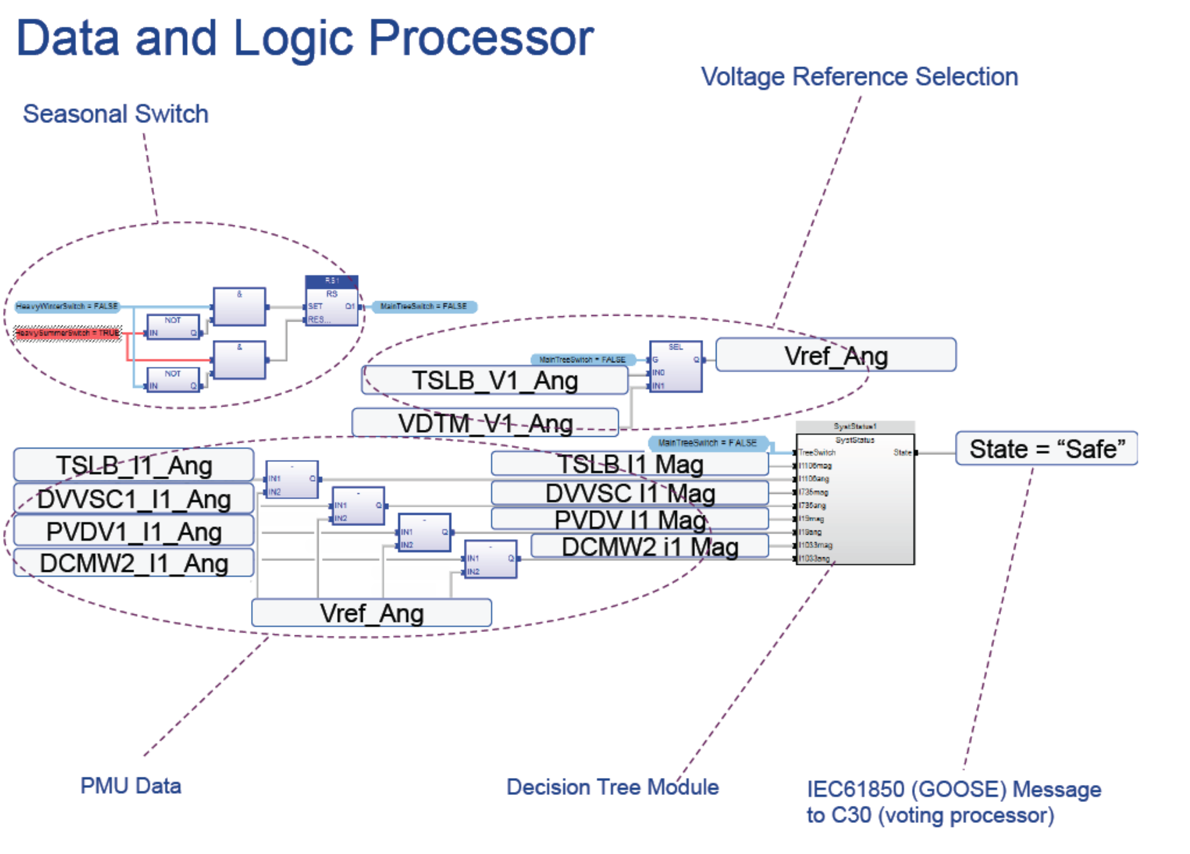
common base. Then, the per-unit phasors are resolved into their rectangular form before they are channeled as input parameters to the decision tree function. After processing phasors using the selected decision tree, an output representing “safe”, “stressed” or “voting-off” is transmitted to the Controller System for security-dependability mode selection. Figure 60 shows the workflow of system-state decision-making process performed by the phasor processor software.

GOOSE Interface

The communication between the Application Module and the Controller System is performed using IEC 61850 GOOSE (Generic Object-oriented Substation Event) messaging Modbus protocol over an Ethernet communication link. This interface has been selected to ensure prompt and secure delivery of the information between the devices. The GOOSE configuration is developed to handle this two-way communication, and to coordinate sending and receiving data across the same link.

Decision tree selection is made using a selector switch connected to the Controller System. Two input registers are allocated for tree selection. Since the selection is based only on the switch position, input registers are used to hold these values. Unless the switch is turned to “off” position, one input register will hold “1” while the other will hold “0” depending on the switch position. Both registers are continuously polled by the computing platform to process phasors using the proper tree. Both input registers are updated every second to capture any change in the switch position.

Figure 60. Phasor Data Processor Workflow



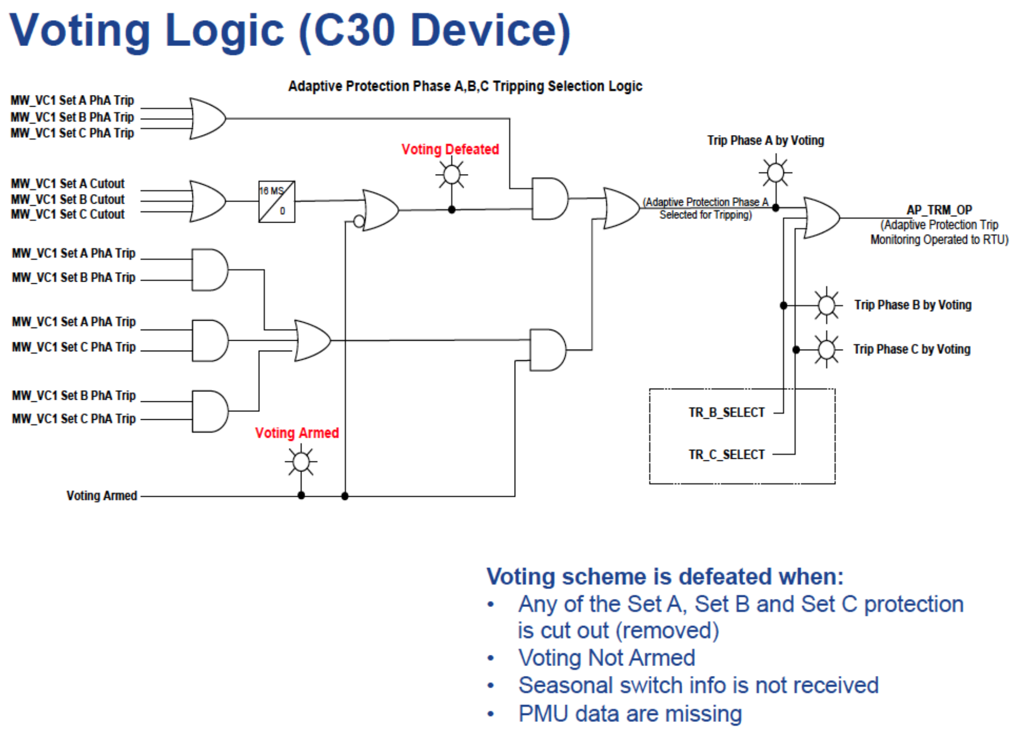
After processing selected phasors by the desired decision tree, the application sends a GOOSE message representing the system operating state. Using a different GOOSE message allows the software application to indicate the output of the decision tree.

<u>Controller System (UR C30)</u>

When the system is operating under stressed condition, protection system should be biased towards security to ensure false trips do not happen. In that case, the trip decision is made using the logic processor with voting logic armed. Two out of the three relays must see the fault and trip in order for the logic processor to issue a signal to open the line. The Logic Processor is set to issue the main trip signal in the adaptive voting scheme implementation. This signal is programmed using user-defined logic diagram, together with a structured text script following IEC 61131-3. Figure 61 shows the logic diagram of adaptive voting scheme.

Relays trip outputs are fed into the logic processor through the digital contact inputs. The input register storing the system condition is used to arm or disarm the voting scheme and is updated continuously and as soon as a change occurs. Depending on the system condition, the Controller System either arms the voting scheme or passes any trip signal issued by any one of the relays to open the line.

Figure 61. Voting Scheme Logic Diagram



As mentioned earlier, decision tree selection is made using a mechanical selector switch that provides digital input to the Controller System. Depending on the switch position, the Controller System sends a signal to Application Module to select the desired decision tree. After selection is made, the Phasor Processor Application PDC processes system phasor data to classify system operating condition as “stressed” or “safe”. The system state is then sent back to the logic processor to arm or disarm the operation of the voting system. This process is done iteratively every second.

Two steps are involved in the logic diagram programmed inside the controller. The first step handles polling the selector switch position for decision tree selection, and then transmitting the desired selection to the Phasor Processor Application PDC via GOOSE message.

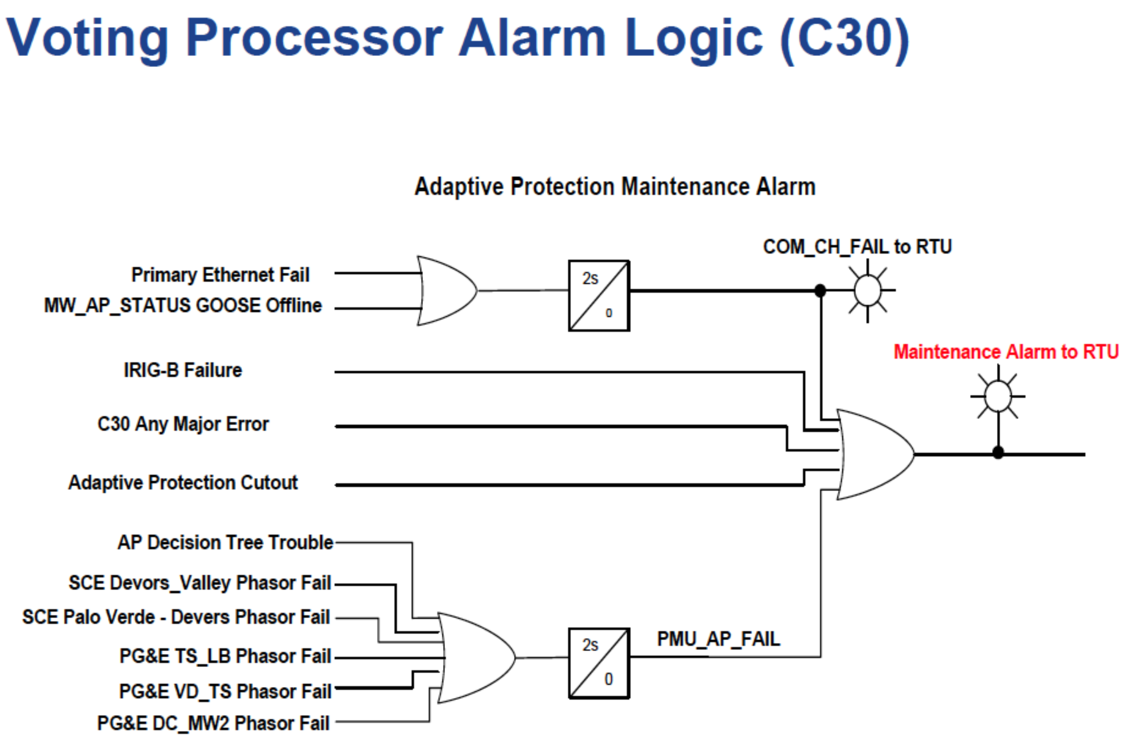
The second step involves the logic processor trip decision-making. This operation depends on relays trip signals and the decision tree output representing system condition from Step 1. The voting scheme is triggered in case of a “stressed” condition, and the logic processor will issue the trip signal if at least two out of the three relays sense the fault. The voting function will be disabled immediately when the system state changes to “safe” condition.

The 3-position selector switch, used to select the desired decision tree, has an “off” position which disables the voting function of the controller. When the switch is moved to the “off” position, the controller trip output will be activated if any relay sends a trip signal, regardless of the system operating state. Hence, the computing platform operation can be skipped and the controller can process the trip logic without receiving the system state. This is implemented in the logic processor user logic and can be observed from the interconnection between the two steps. This connection bypasses the communication with Phasor Processor Application PDC software and allows the processor to perform the trip logic with the voting function disabled.

Voting Processor Alarm Logic

In order to alert the System as well as the user of some unfavorable conditions, a logic is included in the Controller System. Figure 62 shows this logic. Figure 63 shows the LED assignment panel for the C30, allowing the user to check the status of the alarm logic.

Figure 62. Alarm Logic Diagram



SOE (Sequence-of-Events) Recording and Collection

The Synchrophasor Processor and the Controller System provides built-in Sequence of Events (SOE) feature. All inputs, variables, registers and outputs are monitored by the processor software and stored. Time-stamped recorded data can be retrieved using the software (EnerVista UR Setup or EnerVista P30 Setup) or by Modbus or FTP protocols.

Synchrophasor Data Archiving and Retrieval

The GE P30 Multilin™ Phasor Data Concentrator can process PMU data from up to eight C37-compliant client devices. It is equipped with a local Historian Processor, which continuously records the incoming streams of phasor data and provides for user retrieval of the data as needed (Figure 64), supporting an array of user applications such as dynamic disturbance recording (DDR), post-event analysis, wide area monitoring for situational awareness, and others.

Figure 63. LED Assignment Panel for the C30

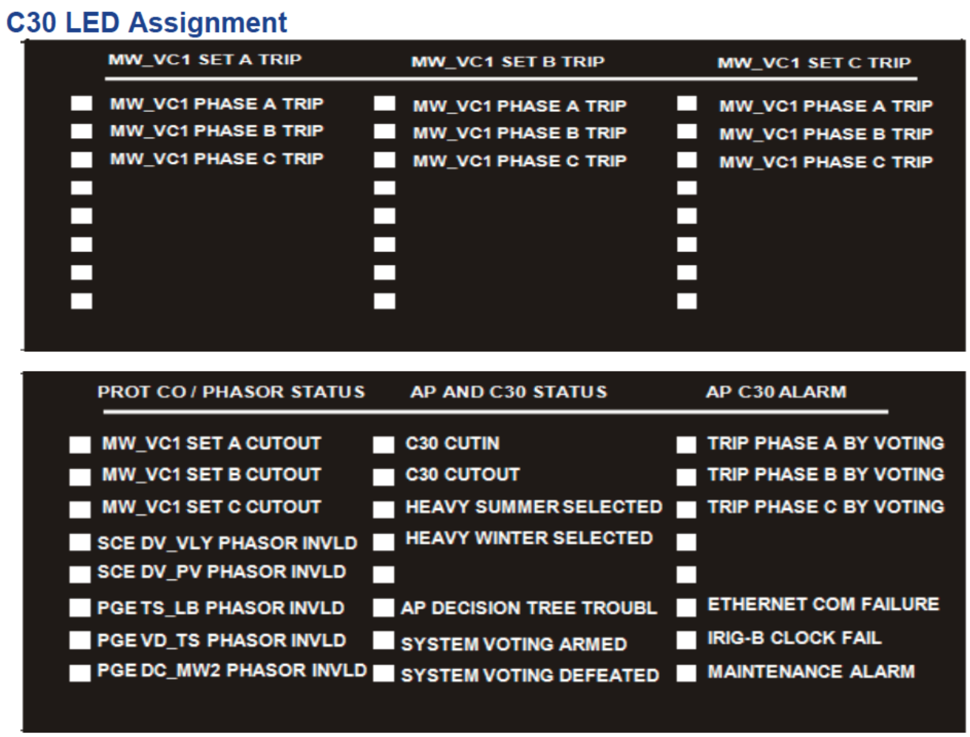
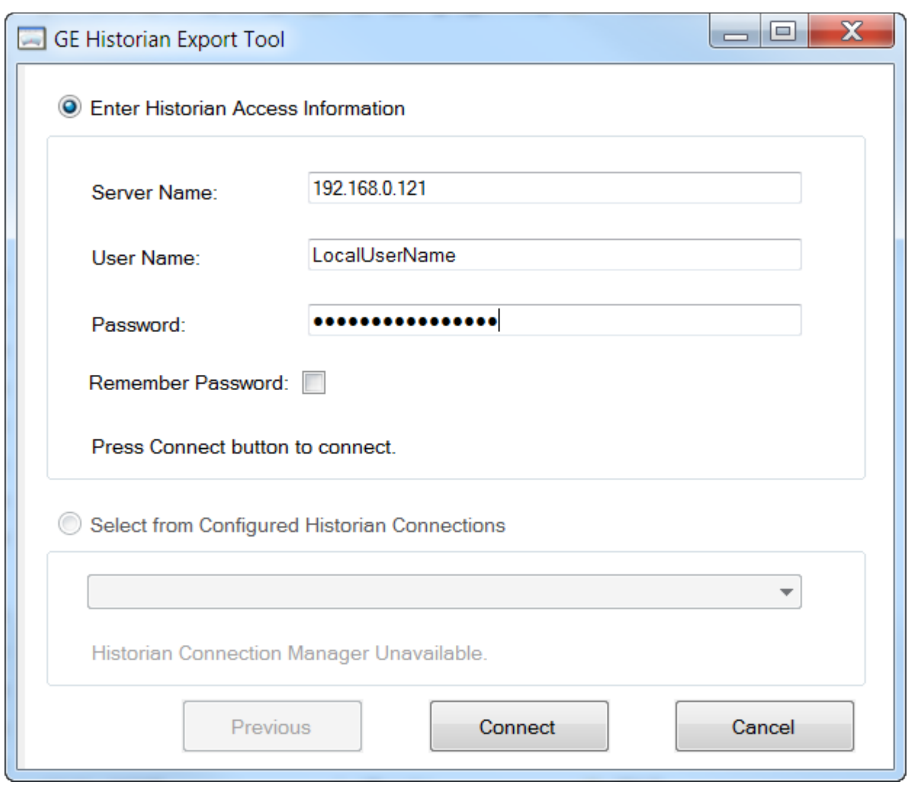


Figure 64. Synchrophasor Historical Data Retrieval



2.3.5 Data Collection and Analysis

For the utility implementation of the proposed algorithms to function, PMU data must be shared between utilities. Once data sharing is implemented the "real data" analysis of the project will be performed. A data analysis plan was developed to systematically analyze recorded events and determine the performance of the proposed algorithms. SCE and PG&E have agreed to provide all PMU data and any additional data from transient recorders or state estimated for all significant events. Initially all events in the transmission system of the WECC will be considered significant to help validate the models used for the developed decision trees. Once the models have been properly validated, the events will be limited to those that cause positive responses of the decision trees; those that do not generate positive responses will be disregarded.

2.3.5.1 Data Analysis Plan

The following Data Analysis Plan has been developed in consultation with PG&E and SCE to evaluate the performance of the proposed algorithms with real PMU data from the field installation.

I. Individual Event Analysis

A. Preliminary Analysis:
   Record date received and source
   Record event date and time
   List all collected records:
   - i. Recording device (if PMU provide PMU name)
   - ii. Device Location
   - iii. Recording Rate
   - iv. Recorded time
   - v. Recorded Data (Voltage, Current, frequency, df/dt, …
   - vi. For PMU data check data validity (GPS, comm. flags)

B. Model Validation/Qualification
   Based on event information:
   - i. Determine Initial Conditions and select seasonal Model
   - ii. Replicate event on Selected system model
   - iii. Classify Model for event as: Acceptable, Unacceptable or undetermined
   - iv. Document differences/limitations of unacceptable/undetermined model cases

C. Preliminary Data Type Qualification. Based on triggering/recording reason, qualify data case as:
   - i. Load Encroachment Event
   - ii. Security Dependability Event
   - iii. Unknown Event Type

D. Load Encroachment Event Analysis
   - i. Determine Triggering Event
   - ii. Verify Algorithm performance with recorded data
   - iii. Simulate Even on Seasonal Model
     - a. Perform simple model validation

b. For valid model: evaluate algorithm performance
c. For non-validated model: determine model or data limitation

iv. Prepare Event Report

E. Security/Dependability Event Analysis

i. Verify algorithm response with recorded data

ii. For verified responses:

a. Determine Event Security/Dependability Qualification based on algorithm Response
b. For Voting Disabled Cases
    1. Perform Simple model Validation
    2. For validated Model: Evaluate Algorithm Performance
    3. For Non-validated model: determine model or data limitation
c. For Voting Enabled Cases
    1. Perform Simple Model Validation
    2. For Valid Models: Evaluate algorithm Performance
        a) Time performance
        b) Surrogate Performance
        c) Simulation performance
    3. For Non-validated models: determine model or data limitations

iii. For non-verified Responses: Determine Algorithm Failure (bad data, insufficient data, programming errors, invalid model,…)

iv. Qualify Event and Prepare Event Report

a. Qualify Event as:
    1. Correct Operation
        a) Voting Enable
        b) Voting Disable
    2. Incorrect Operation
        a) Voting Enable
        b) Voting Disable
    3. Unknown Operation
        a) Data Errors
        b) Model Errors
        c) Mis-recorded Event
b. Prepare Event Report

II. Data Encroachment Events Analysis

Determine Algorithm Performance by:

1. Seasonal Models Verification
2. Collected Data

III. Security/Dependability Events Analysis

Determine Algorithm Performance by:

1. Seasonal Models verification
2. Collected Data
3. Surrogate Performance

IV. Prepare Data Evaluation Report

2.4 Conclusions and Recommendations

The work described in this Chapter details the development, testing and implementation of a Security/Dependability Adaptive Voting Scheme using decision trees [Bernabeu, 2009], for improving the reliability and security of the power system during times of system stress, using real-time wide-area measurements and decision tree logic. The decision trees categorize power system operating conditions into two states: Safe or Stressed. During Safe conditions, the operation of the protection system is not modified, and therefore operates towards Dependability; during Stressed conditions, a voting scheme supervises the protection system tripping logic to minimize the possibility of false trips, thereby ensuring increased Security during conditions characterized as Stressed. The conclusions and recommendations resulting from the research in this project are presented as follows, in order of project phases: Research and Development, Pilot Demonstration, and Field Demonstration.

2.4.1 Research and Development

This phase of the project reviewed the results from the precursor CEC project to determine what changes would be required in the Security/Dependability Balance algorithm. Based on the list of substations scheduled for PMU placement, and the specified requirements for the performance of the algorithm, it was determined that some revisions to the algorithm were required.

Two new decision trees were created for the Heavy Winter and Heavy Summer models of the California system. Analysis of the out-of-sample performance of the decision tree resulted in the implementation of an additional requirement, the specification of separate references for each of the decision tree models, in order to meet the specified performance requirements. The new decision trees required PMU measurements at Devers, Tesla, Diablo and Vaca-Dixon 500 kV substations, all of which were on the list of substations scheduled for PMU installation. Testing verified that the performance of the algorithm was satisfactory.

2.4.2 Pilot Demonstration

The revised Security/Dependability Balance algorithm was successfully implemented in a laboratory setting, using different configurations and equipment. Due to the ability to program user-defined functions along with performing time alignment and aggregation of phasor measurements, the SEL-3378 Synchrophasor Vector Processor and OpenPDC phasor data concentrators were used for implementing the decision trees. CART® output trees were converted to "if-else" logic statements to perform the classification of the operating state of the system based on key PMU measurements. An Ethernet network was set up in the lab, with the PMUs reporting their phasors at 60 data frames per second. The PDC executed both the time alignment and aggregation functionality along with the decision tree application. Based on the measurements from the incoming PMUs, the decision tree residing within the PDC would apply a control signal to the relaying scheme or PLC performing the breaker trip operation.

It was determined that both the hardware PDC (SEL-3378) and the software PDC (OpenPDC) successfully ran the decision tree following time alignment and aggregation of incoming PMU data in a laboratory environment. For the SEL-3378 solution, the relaying scheme was set up in a Master-Slave configuration. This was verified as a viable option for the voting scheme using devices from multiple manufacturers. During an armed state, the slave relay trip signals are restrained, while in a disarmed state normal relay operation is performed. Communications between relays in the form of SEL Mirrored Bits for an all-SEL configuration and hard-wired form for a combination of relay manufacturers was demonstrated. Modified logic for both the Master and Slave relays was developed and tested. For the OpenPDC solution, the voting scheme was simplified because the arming signal was retrieved from an SQL database by a programmable logic controller (PLC). This PLC also performed the voting logic and applied the breaker trip signal for both armed and disarmed states. Additional accomplishments during this phase, which would prove useful in the field demonstration phase, are as follows:

1. Dynamic switching of decision trees through C37.118-2005 communications from phasor measurement units was demonstrated.
2. Multiple voting scheme implementations, matching various protection practices existing today, were implemented and tested.
3. Two communication protocols for the hardware PDC with voting scheme implementations were tested, and both were determined to be acceptable for use in the adaptive voting scheme.
4. A procedure for testing voting scheme implementations was developed.

Two manufacturers of PDCs were used for implementing the adaptive protection scheme, but this leaves room for improvement. Another option for implementation that would be similar but possibly more effective is the SEL Real-Time Automation Controller (RTAC). It is a hardware-based controller that has the capability of performing PDC functions as well as running programmed user applications.

The system was demonstrated to PG&E and SCE engineers during a review meeting on September 22-23, 2011 at the Virginia Tech laboratory, at which time any apparent issues with transferring the system to the utilities' laboratory and operating systems in the field demonstration phase were discussed. The utility representatives expressed approval for the results of the R&D phase. It was the consensus of the team that the process of pilot (laboratory) testing at the utilities' respective facilities appeared to be straightforward, as was the prospective subsequent installations in the field demonstration on the Midway – Vincent 500 kV transmission line.

Soon after the Virginia Tech demonstration, PG&E installed and tested the system at its System Protection Laboratory in San Ramon, CA, and SCE installed and tested the system at its research laboratory in Westminster, CA. Virginia Tech personnel advised the utilities during the pilot demonstration process, and assisted the utilities in resolving the relatively minor installation issues that arose. The utilities subsequently approved the system for field installation in the final phase of the project.

2.4.3 Field Demonstration

PG&E and SCE installed the Security/Dependability Balance system at their field sites, Midway and Vincent 500 kV substations, respectively, albeit with some unanticipated scheduling delays, and the system was commissioned in September 2013. The specifics of the installations, in terms of the hardware and software implementations, differed in some respects between PG&E and SCE. However, the functional performance of the overall system was precisely as intended, leading to some interesting lessons learned in the flexibility that was available for utility implementation when using state-of-the-art synchrophasor and digital communications technologies.

The custom software solution implemented at SCE simplified their implementation, as the PDC input stream, decision tree and the vote output are all processed in a single application. Two-way communication between the software and the RTAC is enabled by using the Modbus protocol over TCP. This feature eliminates the need for a database as it allows the vote logic to be sent directly to the controller and the tree selection from the RTAC to be retrievable by the application. As with the OpenPDC solution, the RTAC was programmed with the tripping logic using ladder logic.

Both utilities utilized a Real-Time Dynamic Simulator to test the operation of the system, simulating lines tripping out of service and other typical contingencies, verifying that the operation of the decision tree logic was correct in identifying conditions of system stress and accordingly sending an arming signal to turn on the voting scheme at appropriate times. During such times of simulated stress, further simulation of a fault signal from a single relay verified that the voting scheme would inhibit tripping for that condition, but would allow it for trip signals from any two or more relays.

The final objective of the Field Demonstration phase of the project was to monitor the operation of the Adaptive Relaying scheme for one year, collecting data on its operation, and analyzing that data according to the Data Evaluation Plan, with the goal of evaluating the performance of the system using the field data.

Due to the considerable unanticipated delays in scheduling and logistics, the field test period was delayed by almost two years. Once the adaptive relaying system was fully functional, a number of transmission system reinforcements had been implemented in the Western grid, and other unforeseen factors came into play. These included:

- A new 500 kV substation (Whirlwind) was added in the Vincent-Midway No. 3 500 kV line; this substation is the collection point for Tehachapi Wind Farm development.
- Several substations and lines in the general area of the Midway-Vincent transmission line path were converted from 230 kV to 500 kV, significantly strengthening the system.
- On the eastern side of the SCE transmission system, typically a relatively weak part of the system, two new substations, Red Bluff and Colorado River, were inserted into the Devers-Palo Verde 500 kV line. A major solar PV plant is connected to Red Bluff Substation that changes the original dynamics of system as it was studied in 2009.
- Due to the economic downturn that started in late 2008, just before the project began, it was observed that system loads were generally somewhat lower than trend. This could have played some role in the very rare occurrence of high system loading events.
- The prevailing drought in California and the West had the effect of limiting the otherwise high transfers of hydro power typically available during the during the summer months on the Pacific Intertie, which was also a likely factor in fewer incidences of stress on the grid than has been the experience.

Therefore, the conclusion is that, for the Security/Dependability scheme to be accurate and reliable, it must be modeled as closely as possible on the actual system configuration. Significant changes in the system infrastructure would require that the decision trees be re-evaluated and the voting logic re-programmed to implement proper relay scheme supervision. In ongoing routine operation, this would be done on a recurring basis by the utility operation engineers. Unfortunately, for a research project such as this one, time and budget do not permit such additional effort.

In addition, disturbances and other dynamic events occurring on the transmission system are very important for checking and calibrating the underlying models of the Adaptive Relaying methodology. But, from midyear of 2013 (time of system commissioning) to the end of the project in September of 2014, system stress was much lower than expected, and no significant event relevant to the validation of model and relay performance has occurred, arguably due to the system reinforcements and other factors detailed above. Even if such events had occurred in that time span, due to significant alteration in system infrastructure and load, the data would not be useful in validating the model and assessment of relay performance. In order to perform such validation and assessment, it would be necessary to run new studies for the present system and train new data mining trees, which would then need to be reprogrammed into the relay algorithm.

2.4.4 Recommendations for Further Research

Some avenues for improvement were noticed during the laboratory and utility implementations. One of these is to expand upon the control signal technology available and deploy more efficient and widely used protocols or options. For example, if the PDC is located at the critical location where the voting takes place, hard-wired copper could be used as an arming signal rather than Ethernet communications. This would most likely reduce the time required to act on the signal provided to the relays. Similarly, the GE N60 Universal Relay using DNP V3.00 has a Binary Input scanning period of 8 scans per power system cycle, which would be about 2 ms on a 60 Hz system [Symmetricom, 2009]. If the PDC could output binary statuses over DNP V3.00, the control signal latency would be reduced drastically. Along with reducing the time requirements, these options would allow interoperability of devices between manufacturers (an ancillary objective of standards).

Another avenue for improvement would be to expand upon the control signal technology available and deploy more efficient and widely used protocols or options. For example, if the PDC is located at the critical location where the voting takes place, hard-wired copper could be used as an arming signal rather than Ethernet communications. This would most likely reduce the time required to act on the signal provided to the relays. Similarly, the GE N60 Universal Relay using DNP V3.00 has a Binary Input scanning period of 8 scans per power system cycle, which would be about 2 ms on a 60Hz system [Symmetricom, 2009]. If the PDC could output binary statuses over DNP V3.00, the control signal latency would be reduced drastically. Along with reducing the time requirements, these options would allow interoperability of devices between manufacturers (an ancillary objective of standards).

The voting scheme implemented using OpenPDC and the Rockwell Automation PLC has room for expansion. A user could benefit from a human-machine interface (HMI) that could display the current state of the system. This could be in the form of a dynamically updated decision tree, with the current state highlighted to indicate to the user what the current attribute values are and where the system lies relative to Stressed or Safe operating condition. Furthermore, the PLC could also be equipped with a database such that the incoming relay trip signals, breaker trip output, and arming state changes from OpenPDC could be time stamped and stored for offline analysis.

SCE and PG&E have each developed their own specific implementation schemes using different hardware and software configurations but using the same real PMU data. The proposed schemes were developed using a California System Model developed with utility support from the full WECC system model. This model was as accurate as it could be at the start of this project but due to the lack of sufficient PMU installations at the time it has not gone through a proper model validation. The evaluation of the performance of the proposed algorithms was intended to determine: a) the reliability of the proposed algorithms, b) the accuracy of the models used for algorithm development, and c) the effect of model inaccuracies on the proposed algorithms.

The research team, which includes both university researchers and utility engineers, concludes that further monitoring of the Adaptive Relaying system would be valuable, as it is very likely that in the near future new stresses will materialize in the Western grid and event data will become available for model validation and assessment of relay system performance. With some additional assistance from the researchers, the utility engineers should be able to re-calibrate the system and further validate its effectiveness. It is strongly recommended that the evaluation and analysis of data obtained from the field implementation of the Security/Dependability Balance system be continued for an additional 12 months past the end of the research project. For this reason, the system should be kept in place, and analysis can continue by the utilities for the prescribed period.

2.4.6 Research Publications

Publications resulting from, or influenced by, the development and field demonstration of the Security/Dependability Balance in this project, include:

1. Adhikari, U.; Morris, T.H.; Dahal, N.; Pan, S.; King, R.L.; Younan, N.H.; Madani, V.: *Development of Power System Test Bed for Data Mining of Synchrophasors Data, Cyber-attack and Relay Testing in RTDS,* 2012 IEEE Power and Energy Society General Meeting, pp. 1-7, July 2012.

2. Bernabeu, E.; Thorp, J.; Centeno, V: *Methodology for a Security-dependability Adaptive Protection Scheme Based on Data Mining,* IEEE Transactions on Power Delivery, vol. 27, no.1, January 2012.

3. Habibi-Ashrafi, F.; Johnson, A.; Vo, S.; Catanese, D.: *Phasors Point the Way,* Transmission & Distribution World, Vol. 63, No. 1, pp. 26-32, January 2011.

4. Madani, V.; Parashar, M.; Giri, J.; Durbha, S.; Rahmatian, F.; Day, D.; Adamiak, M.; Sheble, G.: *PMU Placement Considerations – A Roadmap for Optimal PMU Placement,* 2011 IEEE/PES Power Systems Conference and Exposition (PSCE), pp. 1-7, 2011.

5. Mazur, D.C.; Quint, R.D.; Centeno, V.A.: *Time Synchronization of Automation Controllers for Power Applications,* IEEE Transactions on Industry Applications, Vol. 50, Issue 1, pp. 25 -32, 2014.

6. Pal, A.; Sanchez-Ayala, G.A.; Centeno, V.A.; Thorp, J.S.: *A PMU Placement Scheme Ensuring Real-Time Monitoring of Critical Buses of the Network,* IEEE Transactions on Power Delivery, vol. 29, no. 2, pp. 510-517, April 2014.

7. Quint, R.; Mazur, D.; Badayos, N.: *A Protective Relay Voting Scheme Utilizing Automation Controllers,* 2012 IEEE Industry Applications Society Annual Meeting (IAS), pp. 1-8, 2012.

CHAPTER 3: IMPEDANCE RELAY ZONE ENCROACHMENT SYSTEM

3.1 ZONE ENCROACHMENT ALGORITHM DEVELOPMENT

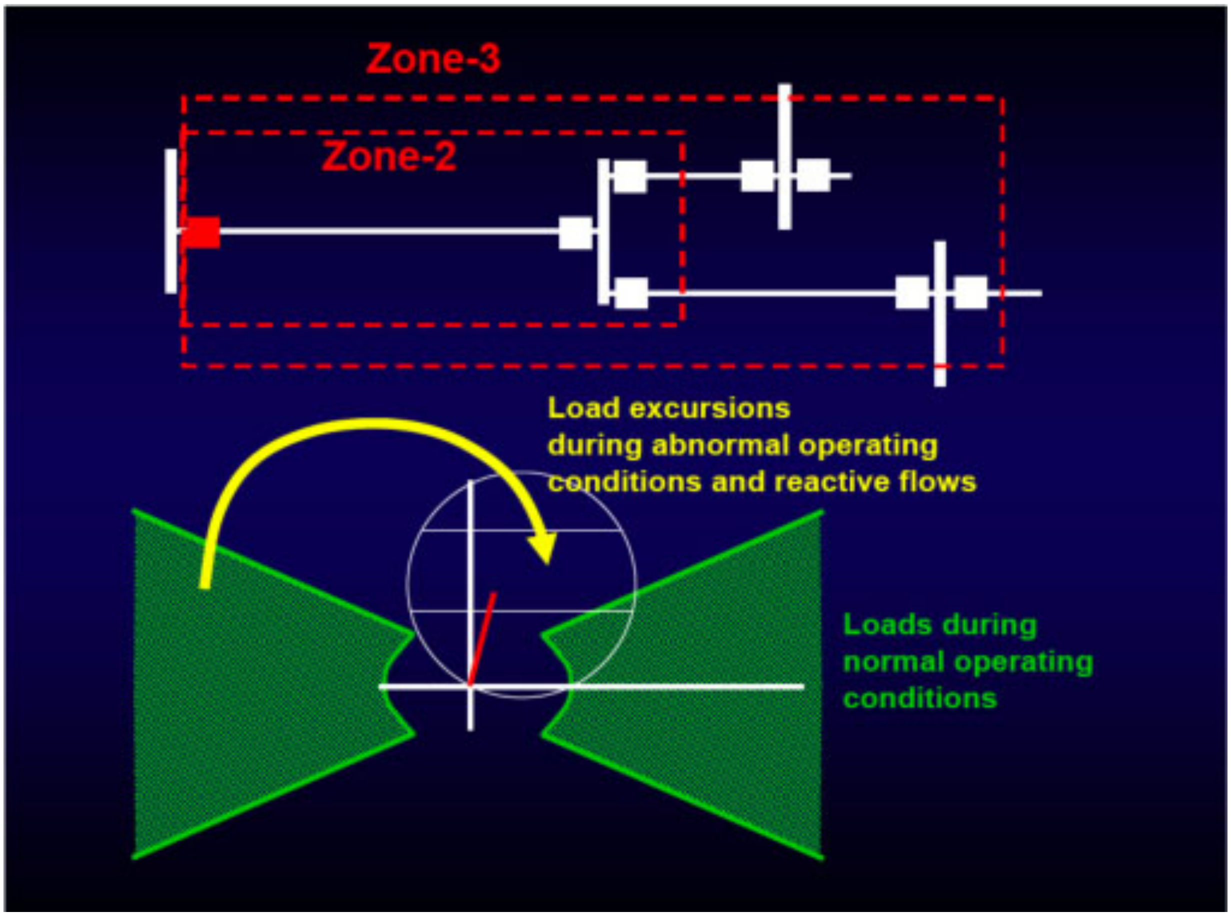
3.1.1 Background

After major disturbances there have been calls to remove back-up protection systems, such as zone-3 relays, over concerns that misoperation of these relays may make matters worse during extreme events like cascading outages. But relay engineers have countered that zone-3 relays are required in certain situations, and should not be eliminated entirely. If zone-1 protection was not adequate to keep the system stable, then zone-3 relays must operate as back-up to keep the system stable. A solution may be to monitor other relays in the vicinity to supervise zone-3 systems. That is, if an appropriate combination of zone-1 relays does not see a fault, the zone-3 relaying system could be blocked, as this situation is likely a false-trip scenario. And evidence from post-mortem analysis of major outages has identified just such false trips as contributing to system problems prior to system collapse.

However, implementing a system similar to the "voting" scheme as described in the previous chapter may be overkill in terms of dealing with the zone-3 relay issue. Experience with zone-3 impedance relays has indicated that the conditions that lead to possible zone-3 false trips evolve gradually over time. That is, when the relays are first calibrated, system studies are used to set the impedance relay zone such that stable disturbances will produce impedance trajectories that do not encroach on the relay trip characteristics, while unstable swings will enter the relay's zone and produce a line trip. Even with multiple contingencies (N-2, etc.) the impedance relays will continue to operate correctly. But, as the transmission system undergoes evolutionary changes in load, generation, and line configurations, the characteristics of the actual disturbance trajectories will change in terms of starting points, end points, and the paths between them. It is entirely possible that eventually, if the relay is not re-calibrated according to the current system conditions, a stable impedance point (steady-state operating condition) or a stable impedance trajectory (dynamic swing from a system disturbance) may impinge on the relay zone, and undesirable tripping will occur. Some type of warning system, in which the system operator or the relay engineer can be given notice when stable impedance points start to approach the relay zone, may be sufficient. If a buffer zone of sufficient magnitude is established around the relay zone, then when system impedances start to encroach on the buffer zone, there should be adequate time for engineers to perform updated analyses and the relays re-calibrated before false trips actually start to occur.

Figure 65 illustrates the rationale behind the approach to be investigated. The circle is the tripping zone of an impedance relay drawn on an R-X (resistance-inductance) plot, developed from system studies. Any time the relay "sees" an impedance inside the circle it is presumed to be an unstable situation and a trip signal is generated; presumably the zone-1 relays have already operated correctly but the system disturbance is so severe that additional line tripping is needed, or the zone-1 relays have failed for some reason and zone-3 is providing the needed back-up. All points outside the circle are presumed to be stable, and the relay will not operate. The green areas represent normal, steady-state operating conditions, and should all lie outside the relay zone. Figure 65 actually depicts some overlap of normal conditions with the zone, such as would be determined from updated studies with relay settings that had remained unchanged for a period of time. The arrow shows how a system swing can start from a stable point but travel into the relay zone; this could actually be a stable swing, ending outside the zone. But because it “encroached” on the zone, the line will be tripped when it was not necessary or desirable for it to do so.

Figure 65. Impedance Relay Zone Encroachment



For this project, the supervisory boundary of the relay zone is chosen to be a circle 50% larger than the relay trip zone. This is somewhat arbitrary, in that engineering experience and judgment is needed to determine whether a smaller or larger boundary is better in actual practice; much depends upon the relay design philosophy and the overall rate of change of the system in a given utility; the size of the boundary can be easily specified according to the judgment of the user. The bottom line is that, once an appropriate boundary is established, the algorithm should continually monitor the system impedance and provide warnings and impedance information when the supervisory boundary is encroached upon. The more frequent the warnings, and the closer the encroachments get to the actual relay boundary, the sooner that relay engineers should perform updated system studies and re-calibrate the relays.

3.1.2 Determination of Test Location

The purpose of the steady-state analysis is to determine bulk transmission system capacity with the failure of multiple lines in the system. The “California heavy summer” power flow base case that was developed by Virginia Tech was used for these studies. System modifications were considered at the 500kV, 230 kV and 115kV voltage levels. This study identifies locations in the system where thermal overloads were caused by multiple line outage contingencies.

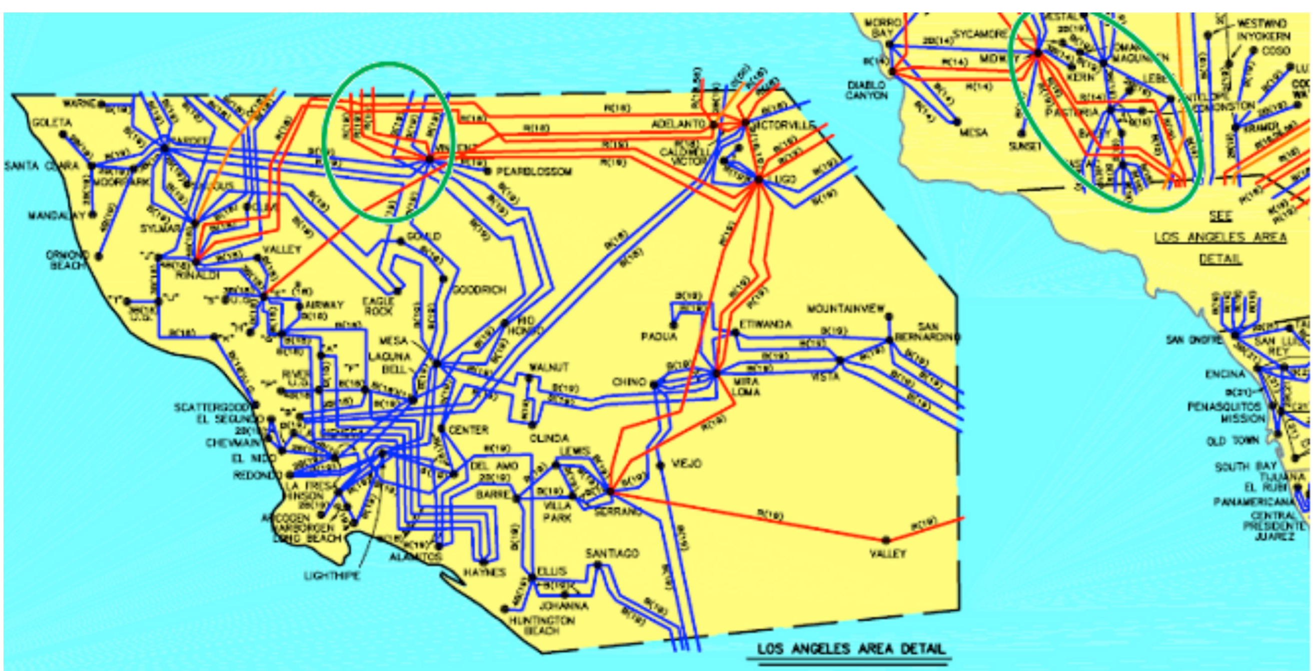
The power flow case used for the study was created from the “California heavy summer” power flow base case that was developed by Virginia Tech. Case studies are verified using the WECC full-loop model. Transmission line overloads are considered based on the “Rate A,” which is the normal operation rating.

Single and multiple contingency analyses (including double circuit lines) were run on the California transmission system model and thermal overloads in transmission lines are identified for power flows on any element that exceeded Rate A. The apparent impedances seen by the relays at these monitored lines are plotted to compare the pre- and post-contingency operation conditions. For the overloading conditions, the apparent impedances seen by the relays on the monitored lines are plotted on the R-X plane to observe any encroachments to the distance relay protection characteristics (zone 1, zone 2 and zone 3).

First, the overloads in the 500 kV system were detected and relay characteristics were monitored for these overloaded lines. There were no zone 3 encroachments found for multiple line outage contingencies, but some post-contingency operating points approached the supervisory boundary. The study cases are the pre- and post-contingency steady-state analyses of the bulk transmission system. In some cases transient analyses are also performed to understand the stability of the system. In a few contingency cases load encroachments are possible but the system reaches instability. In the 500 kV system the following contingency splits the system in two parts:

> Midway to Vincent lines (all three circuit lines) out of service; Captain Jack to Olinda (single line) and Round Mountain to Table Mountain (double circuit lines) out of service.

Figure 66: Midway – Vincent 500kV Transmission Corridor



In the 230kV system lines are monitored after creating contingencies in the 500kV lines so that power is forced to flow through the lower voltage network to supply loads. In the northern area of San Francisco, if the 500kV corridor is taken out-of-service due to maintenance and fault, thus power needs to flow through the 230 kV lines to supply loads in San Francisco. For the following 500 kV line outage contingencies, the Cottonwood 230 kV buses are monitored (Figure 67):

Contingency #1:

Captain Jack to Olinda (single line) and Round Mountain to Table Mountain (double circuit lines) out-of-service (Figure 68).

Result:

Post-contingency, the 230 kV transmission systems fails to supply the required power flow through the network, thus the bulk transmission system reaches instability.

Figure 67. Monitoring Cottonwood 230 kV Bus

Based on this analysis, the closest scenarios to zone 3 encroachments were found in the northern San Francisco region for the following contingencies (Figure 69).

Contingency #2:

500 kV line out-of- service from Round Mt. to Table Mt. circuit 2 for maintenance.

500 kV line out-of-service from Captain Jack to Olinda due to fault.

Result:

The contingency created in the system causes an overload in the monitored line. The post-contingency operating point moves closer to the relay supervisory zone and could potentially encroach upon the relay settings (Figure 69).

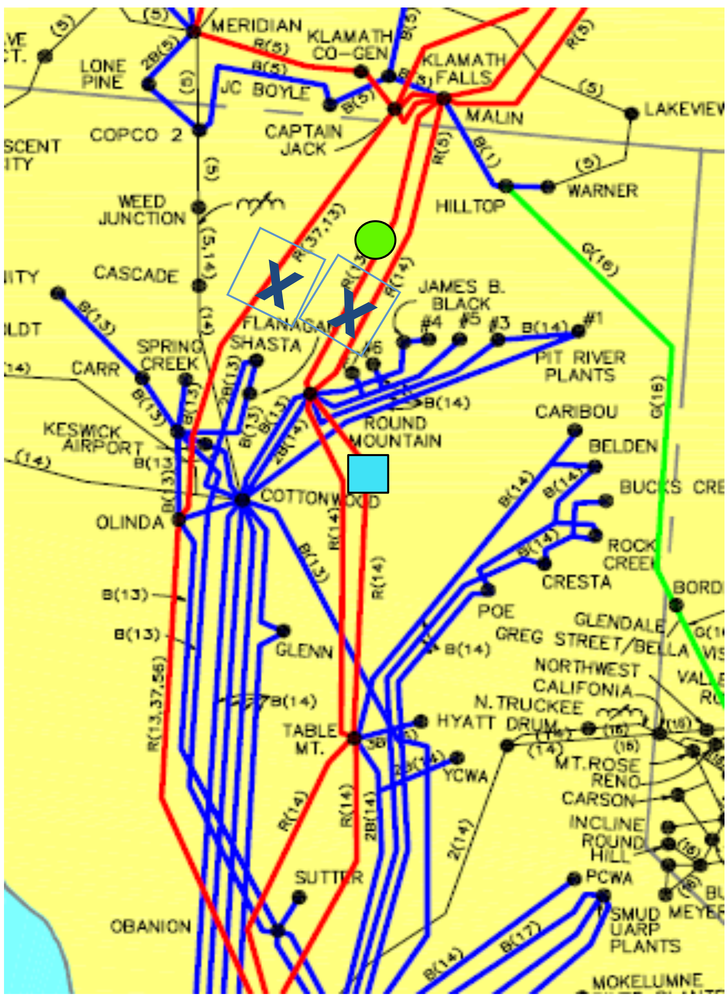
The following case involves a scenario of close load encroachment of the apparent impedance seen by the relay at Olinda (500kV) for the following contingencies:

Contingency #3:

500 kv Line out-of-service from Round Mt to Table Mt. Circuit 1, for maintenance.

500 kv Line out-of-service from Round Mt to Table Mt.Circuit 2 due to fault.

Figure 68. Relay at Malin, Monitoring Malin – Round Mt. #1 Line

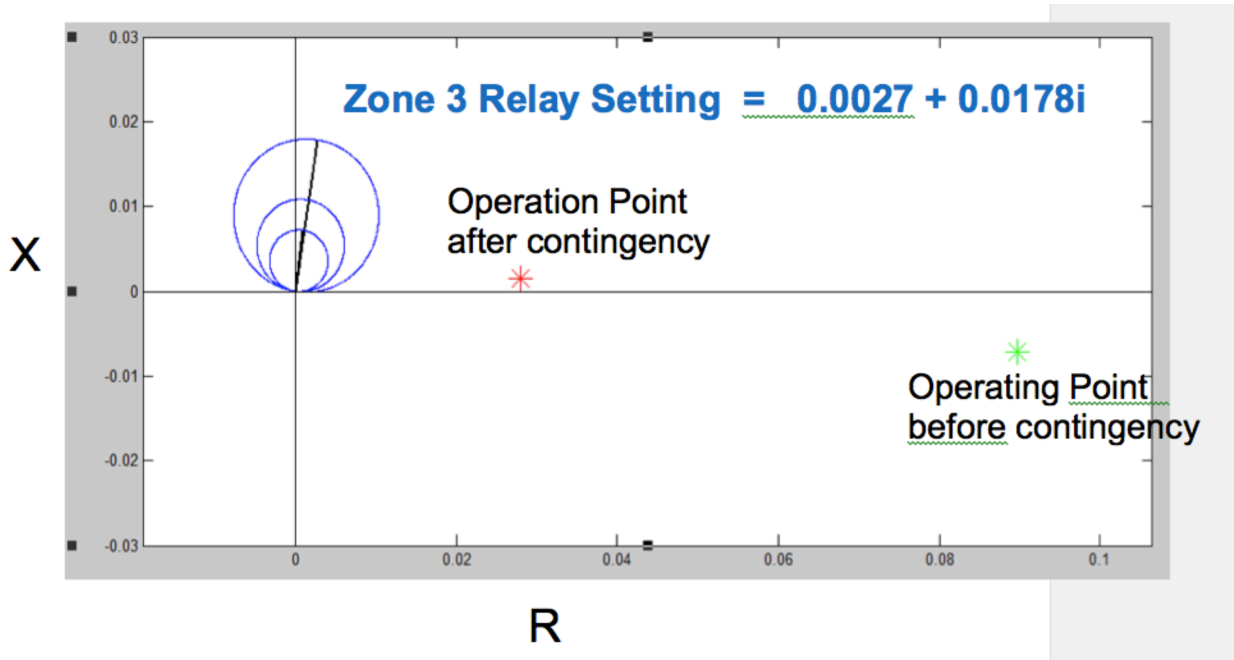


Monitored Line
X Disconnected Line
Phasor Measurement Unit

Result:

In the previous case, we have our monitoring device at the Round Mt. 500 kV bus; if we have a monitoring device at Olinda 500 kV to simulate a relay at Captain Jack, we see an operating point which is inside the supervisory zone. The post-contingency operating point moves within the relay supervisory zone and almost encroaches upon the relay settings (Figure 70).

Figure 69. Apparent Impedance Seen by Relay at Malin for Contingency #2



Based on the steady-state analysis of multiple line outage contingencies, we therefore concluded that the Captain Jack to Olinda 500kV line is the location where we see the closest encroachment to the supervisory zone of protection, and is the best candidate for demonstration of the Impedance Relay Zone Encroachment algorithm.

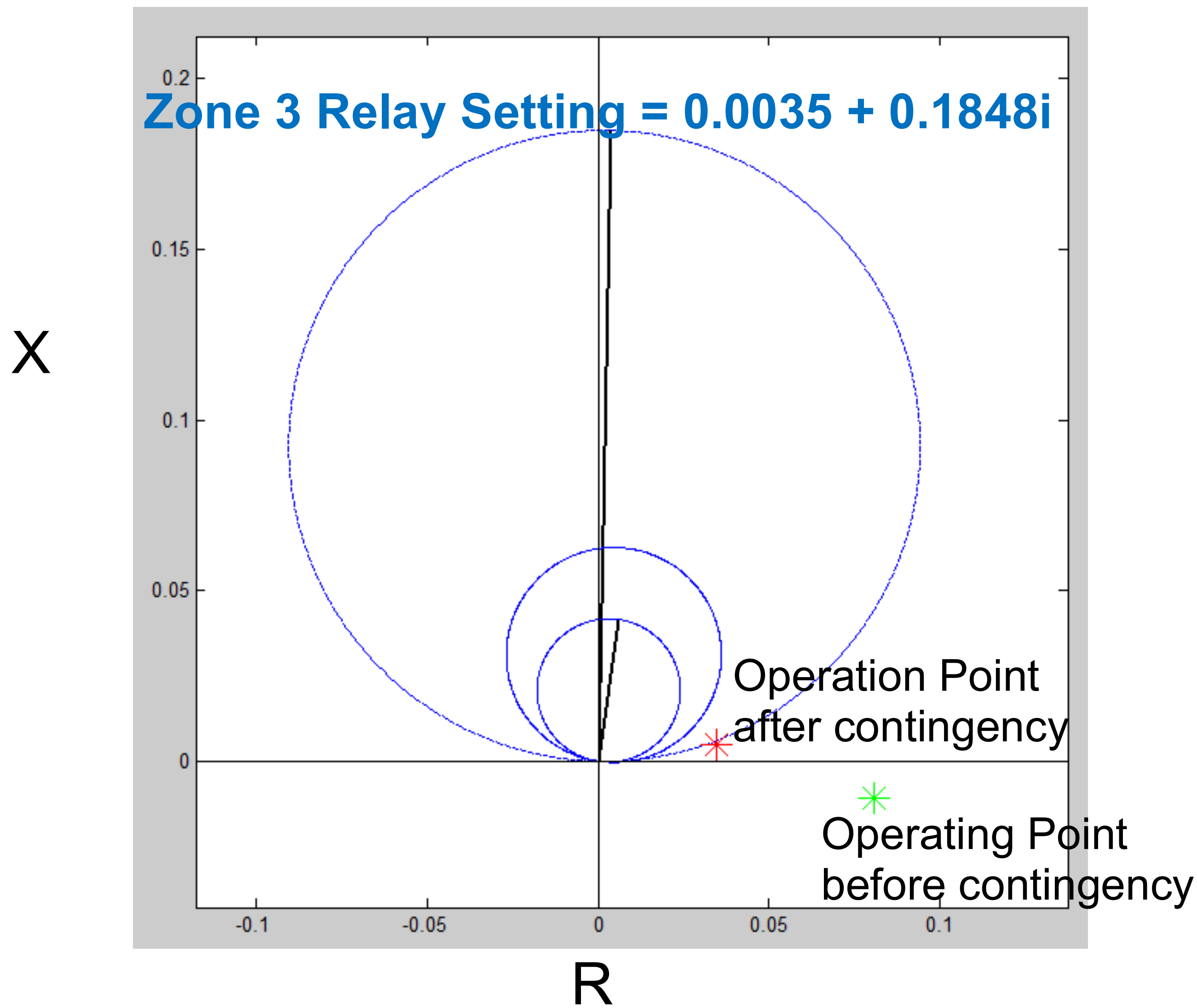
3.2 Zone Encroachment Algorithm Laboratory Implementation

The analysis in Section 3.1 described the process used to determine the best field location for testing the Impedance Zone Encroachment algorithm, based on the likelihood of system disturbance swings causing the apparent system impedance to encroach upon impedance relay trip zones. The line for which this was most likely was determined to be the Captain Jack – Olinda 500 kV line. However, using this line for the field demonstration would not be feasible because of the lack of PMUs in the necessary locations. For the purposes of this project, it was deemed sufficient to implement the impedance zone encroachment algorithm in the hardware and software of the laboratory test system, using the Midway – Vincent transmission line 500 kV line model. This was done, and the algorithm was tested with simulated PMU data.

While there is little danger of zone encroachment on this line during the field demonstration phase of the project, it is expected that the field implementation will show that the algorithm can be implemented on any line for which the utility has the requisite PMU data. For the Midway – Vincent 500kV line, the PMUs used for the Security/Dependability Balance algorithm, along with other PMUs available in the vicinity of the line, would be adequate to demonstrate the capability of the algorithm.

The prototype encroachment algorithm was demonstrated to host utilities PG&E and SCE at a review meeting on September 22-23, 2011 at the Virginia Tech lab. The implementation of the algorithm in the utility testing environment was considered to be straightforward, as it would be a simple matter of programming the algorithm into a research workstation at PG&E's Proof of Concept Synchrophasor Laboratory in San Ramon, CA. PG&E agreed to be the host for the demonstration of this algorithm, and SCE would participate in an advisory role.

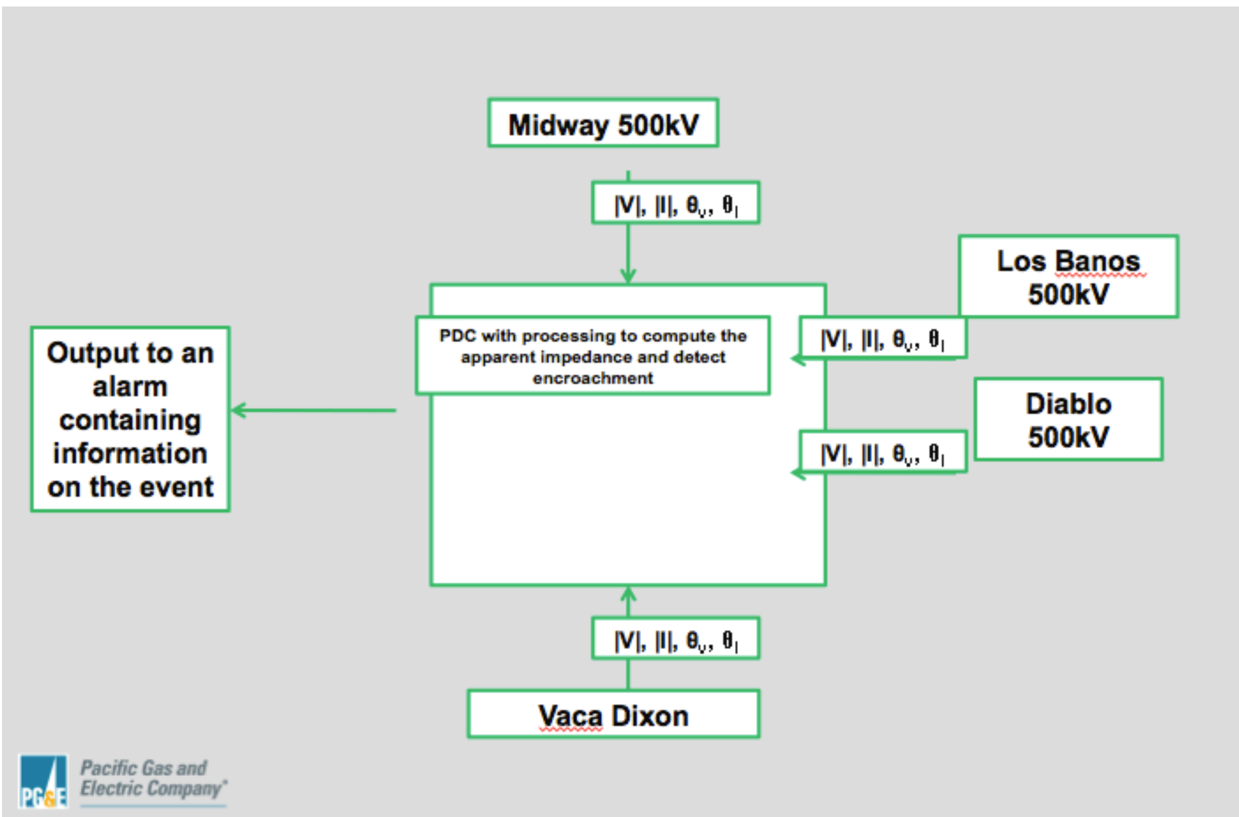
Figure 70. Apparent Impedance Seen by Relay at Malin for Contingency #3



3.3 Zone Encroachment Implementation at PG&E Control Center

PG&E first installed the Impedance Relay Zone Encroachment algorithm at its System Protection Laboratory, located at PG&E's Advanced Technology Solutions (ATS) facility in San Ramon, CA. This Laboratory, which PG&E refers to as a "Proof of Concept" facility, was expressly designed for the thorough testing and evaluation of synchrophasor-based technologies in an environment as close as possible to actual operations, prior to implementation in actual system operations. Figure 71 shows the basic scheme. The algorithm monitors the available PMU data and simulates the zone of the Midway impedance relay with a 50% supervisory zone surrounding the normal zone. System conditions that result in impedance values that fall within the supervisory zone will generate an alarm with accompanying information on the triggering event.

Figure 71. Impedance Relay Zone Encroachment Implementation



Once an alarm occurs, the operator can pull up a screen on the workstation, displaying graphical and quantitative information about the event (Figures 73 and 74). PG&E system operators demonstrated the system to DOE sponsors and others at a meeting at the PG&E Control Center, 77 Beale St., San Francisco, CA, on September 25, 2013.

Figure 72. Operator Screen Shot with Relay Encroachment Information

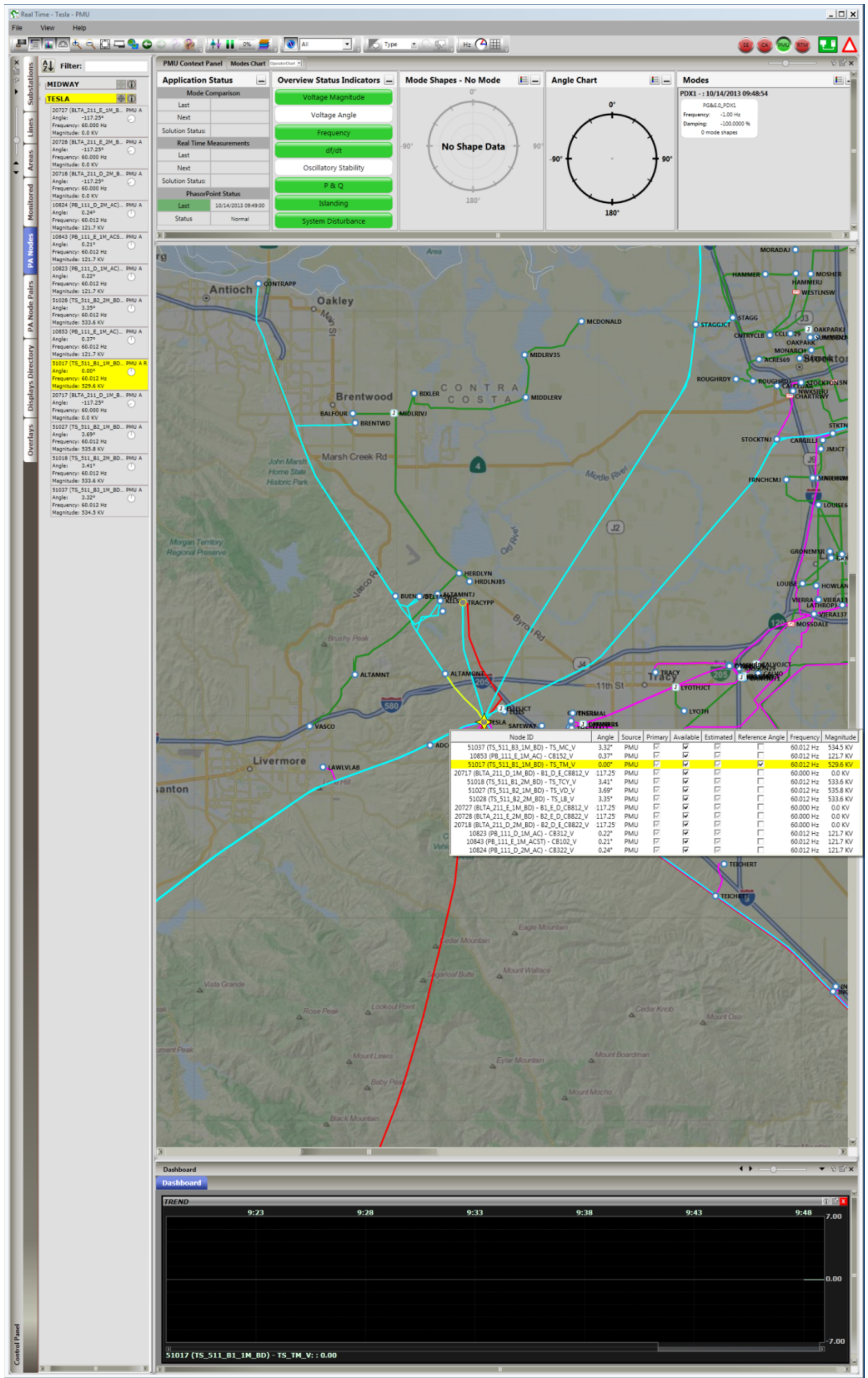
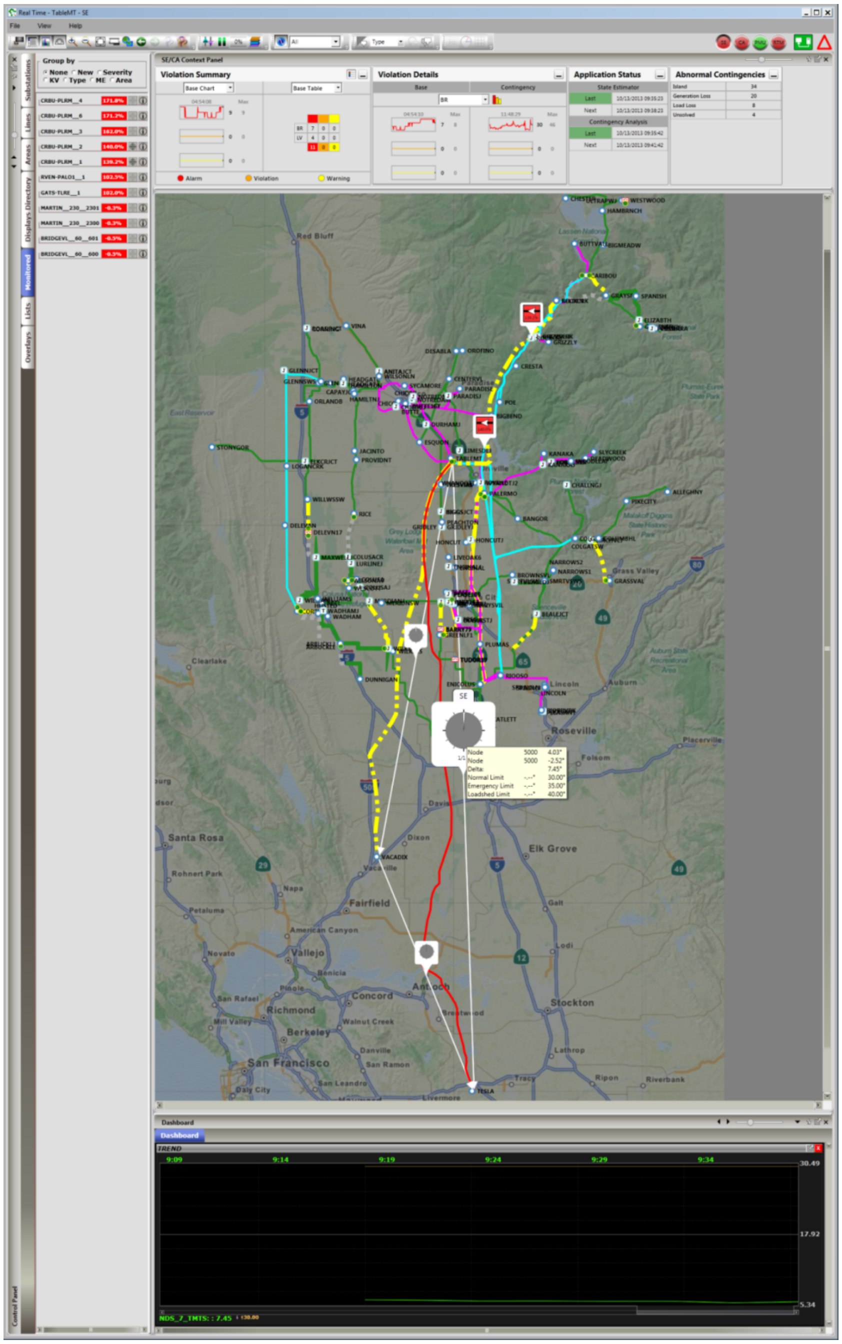


Figure 73. Zooming in on Line with Relay Encroachment



3.4 Conclusions and Recommendations

3.4.1 Research and Development

This first phase of the project required a review of the results from the precursor CEC project [Centeno et al., September 2010] by the research team and the Technical Advisory Group (TAG) to determine what changes were required in the algorithm developed for the Impedance Zone Encroachment system. Based on the steady-state analysis of multiple line outage contingencies and the list of 500 kV substations with PMUs, the research team observed that the Captain Jack – Olinda 500kV line was the location where the closest encroachment to the supervisory zone of protection has been observed, and would likely be observed in the near future, and was therefore proposed as the most suitable line for encroachment analysis. The original algorithms were modified for this location.

3.4.2 Pilot Demonstration

The Captain Jack – Olinda 500 kV line, chosen as the most likely major transmission line to encounter impedance relay zone encroachment, was deemed to be inappropriate for this project due to the lack of PMU data from the necessary locations. The project team decided that, for the purposes of demonstrating how to implement the system, the Midway – Vincent 500 kV transmission line would be an appropriate candidate. The PMUs used for the Security/Dependability Balance algorithm, along with other PMUs available in the vicinity of the line, were used as the appropriate sources of synchrophasor data.

The prototype encroachment algorithm was demonstrated to host utilities PG&E and SCE at a review meeting on September 22-23, 2011 at the Virginia Tech lab. PG&E agreed to be the host for the demonstration of this algorithm, and an implementation strategy was developed.

3.4.3 Field Demonstration

PG&E successfully installed and tested the encroachment algorithm at its Protection System Laboratory, and subsequently installed it in its Control Center in San Francisco. System operators demonstrated the system to DOE sponsors and other interested parties at a review meeting on September 21, 2013, and declared it to be a very promising tool for keeping their protection systems updated effectively. As more PMUs are added to the PG&E system, more lines can be monitored at very little additional effort and expense.

CHAPTER 4:
PROTECTION INFORMATION TOOL

4.1 PROTECTION INFORMATION TOOL DEVELOPMENT

4.1.1 Background

Current data visualization strategies for PMUs and other data systems are not consistent across systems, nor are they necessarily designed to support information synthesis and decision support for system operators. Effective presentation of data to systems operators can reduce the mental workload imposed on operators allowing them to focus on decision-making and response execution. Further, most data streams are presented independently of other information, though the interrelationships of these data streams likely provide an enhanced view of the system. By taking account of known human limitations in information processing in a control room setting in the design of data visualizations, system and operator performance can be improved.

Further, the lack of standardization of the data exchange mechanism to encapsulate, store, exchange, and configure the power system data sets and the information that goes into monitoring and forecasting is an impediment for effective decision making. This project adapts the Open Geospatial Consortium's (OGC) Sensor Web enablement framework-based protection information tool using information from PMUs and other related sensors. It is adaptable to the needs of developing metadata systems that can evolve as new innovations are implemented in making measurements. Also, it enables the development of descriptions for sensor network state-of-health, data quality assurance, and quality control.

The protection information tool (PIT) consists of two components. A visualization module that incorporates strategies for effective visualization of the PMU data is developed through the synthesis of cognitive task analysis (CTA) conducted with various system operators and protection engineers. The second component deals with the standardization, interoperability and situational awareness aspects and is focused on building a sensor web that enables a uniform and standardized methodology for information exchange.

4.1.2 PIT Development Tasks and Accomplishments

4.1.2.1 Overview

Currently there is a mix of legacy systems and emerging modules in distributed environments; hence, the use of standards is a key component for successful integration and interoperability. The National Institute of Standards and Technology (NIST), in its roadmap for the Smart Grid, stressed the need for common meteorological and geospatial models, and recommended OGC information models integration. The OGC provides tools and information models necessary to make complex spatial information and services accessible to several geospatial applications. These standards from OGC can be implemented in the form of sensor web enablement (SWE) for the Smart Grid sensors.

This first, research and development phase of the project included the following specific goals:

- Identify various components of the protection information tool (PIT) and develop the architecture of the PIT.
- Develop the Sensor Observation Service (SOS) for PMU data streams and geospatial database development of the PMU data.

- Describe the PMU and related sensors with general models and XML encodings through SensorML and Event Stream processing via Sensor Event Service (SES) to facilitate situational awareness.
- Develop the Task Diagram based on semi-structured interviews with operators and engineers in system operations and protection engineering departments.
- Conduct a knowledge audit using semi-structured interviews with operators at control room operations desks identified during the task diagram development.

4.1.2.2 Accomplishments

Sensor Web Enablement of PMUs and Related Sensors

A sensor web refers to web accessible sensor networks and archived sensor data that can be discovered and accessed using standard protocols and interfaces. It facilitates the development of metadata input systems that can scale by orders of magnitude along with increasing numbers of sensors. The OGC standardized sensor metadata (SensorML [http://vast.uah.edu/SensorML/]) can be updated as new instruments are connected to the sensor network. Also, it is adaptable to the needs of developing metadata systems that can evolve as new innovations are implemented in making measurements. Further, it enables the development of descriptions for sensor network state-of-health, data quality assurance, and quality control. The standards-based sensor web is agile and can respond to the demands of accurate tracking of sensor and network changes in close to real time. This sensor web for the electric power grid is further enhanced by creation of actionable intelligence in a timely manner. In other words, it provides the operator with a set of timely, prioritized actions as a result of machines processing the data near the human conceptual level.

Architecture Extending OpenPDC and Development of a Simulated Environment

An open source implementation of the Phasor Data Concentrator (PDC) from TVA, i.e., OpenPDC (http://103penPDC.codeplex.com/) has been tested. The PMU sensors integration aspects were studied and an Open Geospatial Consortium (OGC) standardized Sensor Web architecture was developed, and designed to utilize the data from OpenPDC. Sample PMU data sets from OpenPDC were tested for their utility as data sources for the Sensor Observation Service (SOS), which is an implementation based on the services oriented architecture (SOA) (Figure 74). The SOS will enable users to query the phasor data at several levels of spatio-temporal granularity. The OpenPDC has been interfaced with the PMUs that were installed by GE in the Power Lab at MSU. Protocols such as TCP and UDP have been tested with the OpenPDC and were successful in their deployment.

The project set up a simulated environment that mimics the enhanced interoperability and seamless access of data from PMUs and information exchanges between utilities (e.g.. SCE and PG&E) in emergency scenarios. This is done via web accessible sensors in the sensor web enablement (SWE) framework (Figure 75).

An example of an emergency scenario, incorporating the exigencies in a hurricane disaster event, was simulated. The key benefit is the simultaneous presence within a single information system of a geographically and temporally wide array of sensor data.

Figure 74. Extending OpenPDC with Sensor Web Components

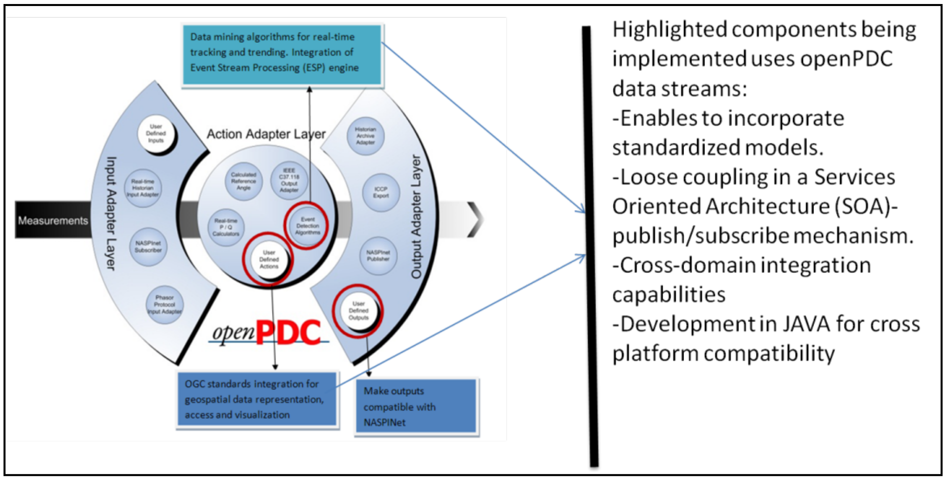
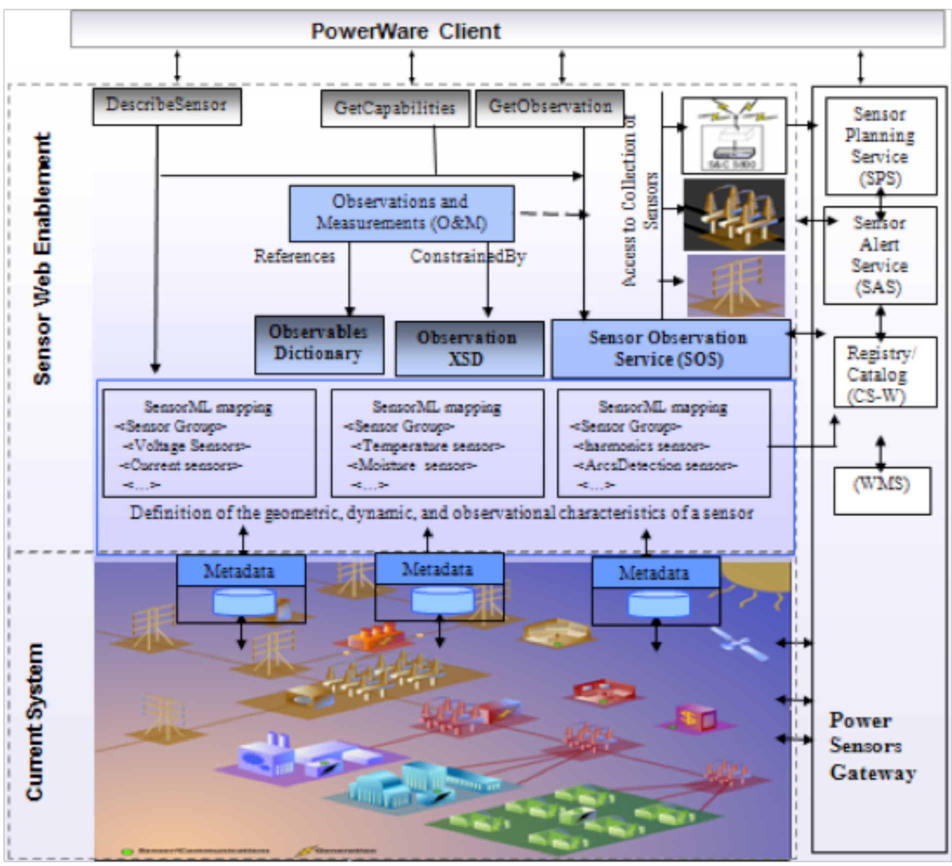


Figure 75. Architecture of the Sensor Web



Sensor Model Language (SENSORML)

SensorML (http://vast.uah.edu/SensorML/) is a foundational part of OGC's Sensor Web Enablement (SWE). It provides a functional model of the sensor system, rather than a detailed description of its hardware. The root for all SensorML documents is Sensor, or an extension of

Sensor such as SensorGroup. Sensor has a unique technology (of type xs: technologies) as a required attribute. Two optional attributes, documentDate (xs: dateTime) and documentVersion (xs:string), provide the ability to quickly check version information. The sensor description is divided into nine main informational components, each of which has sub-parts. Several of the components have "plug-and-play" capabilities, such that they can accept models that are appropriate for a given class of sensors. SensorML provides a standard schema for metadata that describes sensor and sensor system capabilities. SensorML treats sensor systems and a system's components (e.g., sensors, actuators, platforms, etc.) as processes. Thus, each component can be included as parts of one or more process chains that can either describe the lineage of observations or provide a process for geolocating and processing the observations of higher-level information. In the context of the power application, the Phasor Measurement Units (PMUs) are being modeled in SensorML in order to obtain the characteristics of the sensors (e.g., accuracy), to determine the reliability of sensor data and to make crucial decisions to avoid contingencies. Modeling of PMUs based on Sensor Model Language (SensorML) was performed and validated.

Sensor Observation Service (SOS)

A service-oriented architecture (SOA) is an application framework that breaks applications down into individual business functions and processes, called services, which are the building blocks of an SOA. Web service specifications such as Sensor Observation Service and Sensor Event Service define how data collection requests are expressed, observations retrieved and alert or alarm conditions defined. The OGC Sensor Observation Service specification [http://www.opengeospatial.org/standards/requests/32] defines an Application Programming Interface (API) for managing deployed sensors and retrieving sensor data and, specifically, "observation" data. The SOS implementation specification defines the interfaces and operations that enable the implementation of interoperable sensor observation services and clients.

In the architecture that we have developed for PMU data integration from various utilities, the SOS forms the backbone that will enable the interoperability of various data sources and provide a standardized web-accessible method for retrieving the PMU measurements. The SOS is the intermediary between a client and an observation repository or near-real-time sensor channel such as a PMU (Figure 76). Clients implementing SOS can also obtain information that describes the associated sensors and platforms. The SOS GetObservation operation includes an ad-hoc query capability that allows a client to filter observations by time, space, sensor, and phenomena, as shown in Figure 75 of the Sensor Web architecture. The different requests such as GetCapabilities, GetObservation, and DescribeSensor are handled by the sensor observation service.

The SOS, which forms the platform for post-event analysis, was developed and installed, and augmented with a geospatial database developed from PMU data from SCE and PG&E. Simulated PMU measurements from OpenPDC were integrated into the SOS, and several spatio-temporal querying operations were tested.

Geospatial Database Development

The geospatial database which forms the backend of the Sensor Observation Service (SOS) is based on the Postgresql RDBMS combined with the POST GIS spatial database engine. This allows querying at various spatio-temporal scales.

Figure 76. Sensor Observation Service (SOS) Data Consumer Sequence

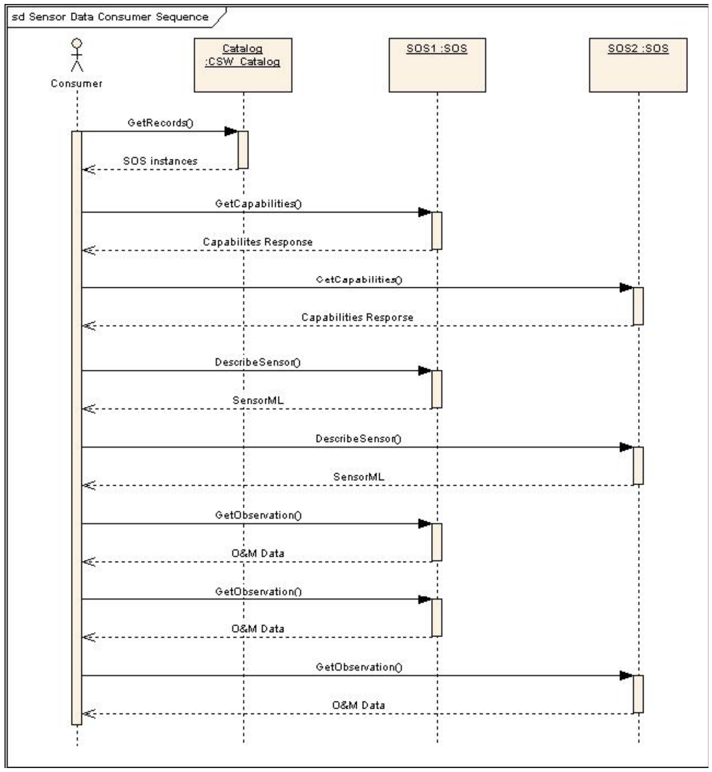


Figure 77. Integrating Various External Information Resources with the PMU Sensor Data Through the Sensor Observation Service (SOS)

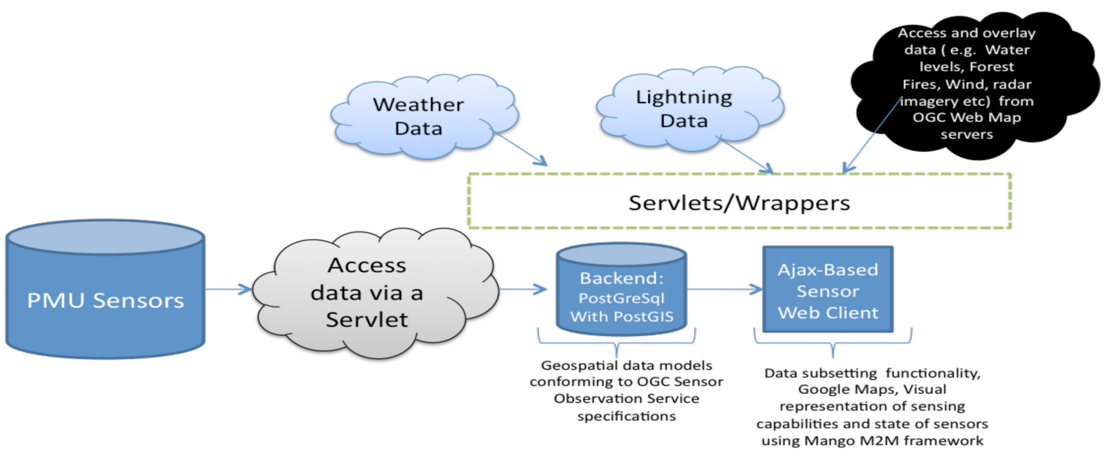
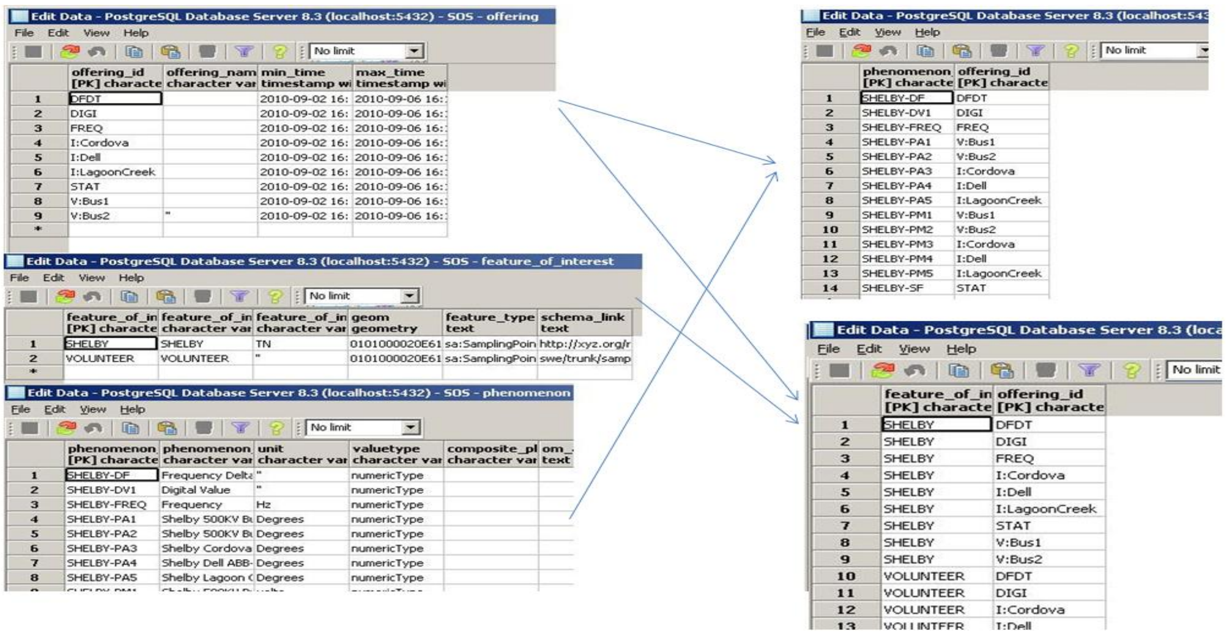


Figure 79 shows some snippets of the underlying database of SOS that was developed for the PMU data. An example table in the SOS database such as "Observation Offerings" is characterized by the following parameters:

- Specific sensor systems that report the observations.

- Time period(s) for which observations may be requested (supports historical data).
- Phenomena that are being sensed.
- Geographical region that contains the sensors.
- Geographical region that contains the features that are the subject of the sensor observations (may differ from the sensor region for remote sensors).

Figure 78. Snippets of the Spatio-temporal Database for SOS



OGC Sensor Event Services (SES)

A taxonomy of phasor applications has been developed by the North American Synchrophasor Initiative (NASPI) (www.naspi.org/resources/dnmtt/phasor_applications_taxonomy2a.xls). Table 23 lists the latency, data resolution, time window, and data requirements for various applications, such as Situational Awareness, Monitoring/Alarming, Analysis/Assessment, etc. In a situational awareness scenario the latency for real-time compliance monitoring with reliability standards (angle of separation, voltage and angle profiles, MW, MVAR flows, load-resource imbalance, etc.) is about 1-5 seconds. Similarly, for Monitoring/Alarming applications such as anomaly characterization and alarming, the latency is 1-5 seconds with a data resolution of 30 samples/sec and a time window of ~1 hour. In all the above applications there are dynamically changing data streams, each of which needs a different kind of mechanism to adapt quickly to changing scenarios to make decisions in real time. Also, the ability to rapidly process the incoming streams of data in real time and send alerts and warnings is necessary. This is possible only if the sensor networks are able to do complex event streams processing. There could be several events happening over extended periods of time during the day, and these events have moving time windows in which the coming data stream has to be processed.

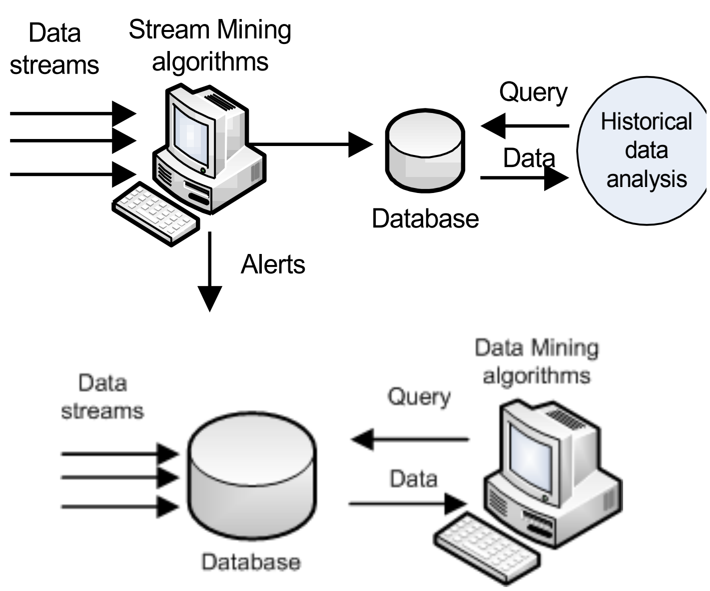
Table 23. Latency Requirements for Various Applications

Application	Latency	Data Resolution	Time Window
Situational Awareness Dashboard	1-5 Seconds	1 sample/sec.	Snapshot
Angle of Separation, Voltage & Angle Profiles, MW, MVAR flows, Load-Resource Imbalance)	1-5 Seconds	1 sample/sec.	30 Minutes
Frequency Stability/Islanding	1-5 Seconds	30 samples/sec.	Few Minutes
Voltage Stability Monitoring/Assessment	Few Seconds	30 samples/sec.	~ 1 hour
Anomaly Characterization and Alarming (Real time alarming on hard limits & "out of normal" conditions, suggest preventive action)	1-5 Seconds	30 samples/sec.	~ 1 hour

Event Stream Processing

Traditional data storage, access and querying systems, such as a Relational Data Base Management System (RDBMS) (Figure 79, top) which is the most common form of repositories for these sensor networks, do not have the ability to process event streams in real time. Normally the data are stored and then queries are executed on the data, whereas event stream processing (ESP) (http://esper.codehaus.org/) works the other way around. In fact, a common way to summarize these contrasting characteristics is by saying that while traditional RDBMS deal with persistent data and transient queries, Data Streams processing-based systems deal with transient data and persistent queries (Figure 79, bottom).

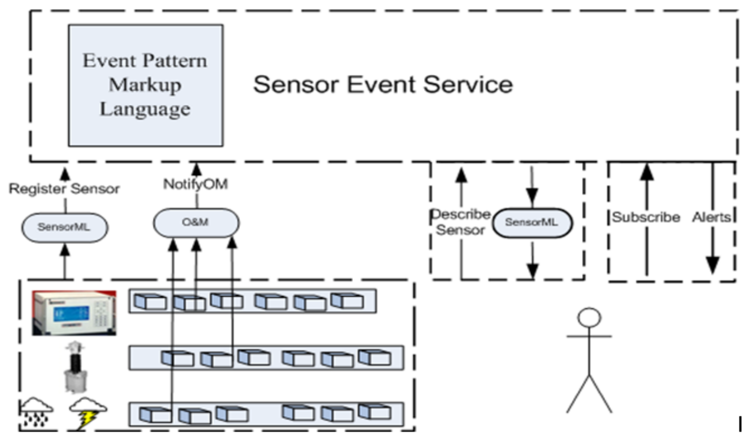
Figure 79. Data Mining Systems



Top: Traditional approach for mining databases.
Bottom: Setup for mining continuous data streams via event stream processing

An Event Stream Processing (ESP) engine has been integrated with the OpenPDC and Sensor web (Figure 80). The OpenPDC and Sensor applications process large volumes of incoming messages or events, analyze those messages or events in various ways, and respond to conditions of interest in real time. We have developed an action adapter for OpenPDC that outputs the data from OpenPDC via Simple Object Access protocol (SOAP) messages to enable its use in the Sensor Event Service (SES). This will facilitate real-time detection of events and classification, and is also useful for situational awareness applications. The SES was successfully executed on a test PMU data set.

Figure 80. Architecture of Sensor Event Service (SES) for Event Analysis from PMU Data



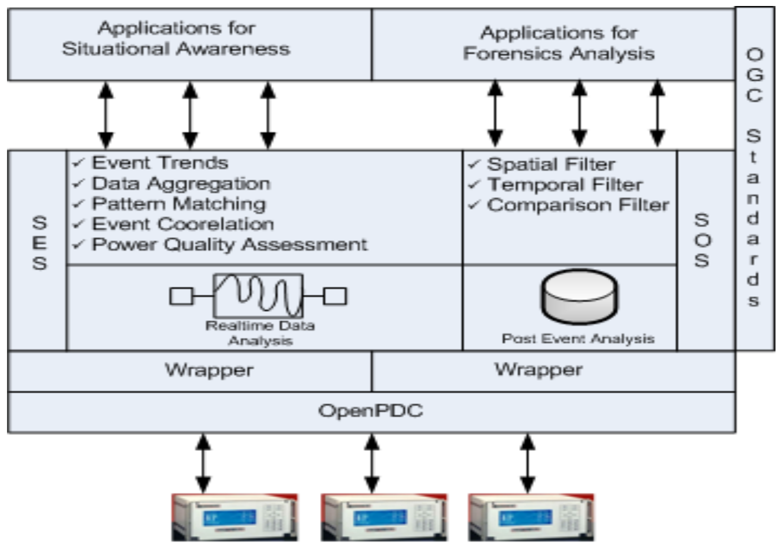
Event Processing Language (EPL) will be used to encode rich event conditions and correlations spanning several time windows. The execution model of the ESP engine is continuous rather than responding only when a query is submitted, so real-time response can be obtained as certain specified conditions occur.

Through the retrieval of real-time or time-series observations in OGC-based standard encodings, the project will enable operators to seamlessly incorporate data obtained by more advanced queries (e.g., filtering, tasking, etc.) from spatially heterogeneous web services or registries from other utilities, thereby overcoming data interchange problems.

Data Stream Mining

Most stream data are multi-dimensional in nature and need multi-level and multi-dimensional processing. The core assumption of data stream processing is that the training examples can be briefly inspected a single time only; they arrive in a high-speed stream, and then must be discarded to make room for subsequent examples. The algorithms that process the stream have no control over the order of the examples seen, and must update their models incrementally as each example is inspected. Also, the algorithms should have the potential to process an infinite amount of streaming data. In data stream processing, concept drift occurs when the underlying concept defining the target being learned begins to shift over time. We have identified the algorithms (e.g., Hoeffding trees) for clustering and classification of the PMU data streams to detect and classify events based on the phase angle. Figure 81 depicts the combined architecture for situational awareness based on dynamic events processing via the Sensor Event Service and the use of the SOS, which serves as a geospatial standards-based historian of PMU measurements for various post-event analyses.

Figure 81. Standards-Driven Hybrid Architecture for Situational Awareness and Post-event Analysis



AJAX-Based Rich Client for Web Services

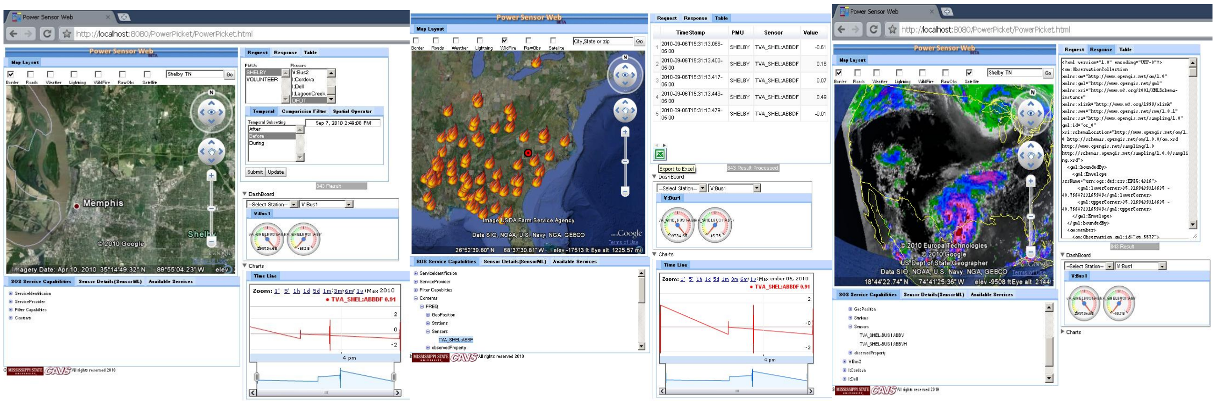
The web-based client that was developed in this project is based on the Google Web Toolkit (GWT), which is a Java implementation of Asynchronous JavaScript and XML (AJAX). This client will allow the SOS and SES to be queried via standardized Service Oriented Architecture (SOA) protocols, and the results could be customized for particular scenarios. In this phase of the project we have begun the development of the Sensor web client interface (Figure 82) to interact with the above-mentioned web services Integration of the Google Earth component (already completed). Integration of diverse geospatial data sets from weather, traffic, fire, earthquake, and other sources is being done. Figure 82 shows the ability to subset the displays at various spatio-temporal scales and to integrate various ancillary information (weather, earthquakes, forest fires, etc.). Several data sets for this purpose have been identified, and integrated with the client (Figure 81).

Cognitive Task Analysis and Visualization

The objective of the cognitive needs and task analysis was to capture the important steps and decision points for system dispatchers and protection engineers. The ever-increasing demand for energy has resulted in the power grid approaching its limit. Power utilities collect thousands of pieces of data, which provide critical, as well as not so critical, information on the status of the power grid. Increased power demands put the power grid in a vulnerable position, subject to both internal/external and natural/intentional threats.

Power system monitors (or operators/dispatchers) are responsible for viewing and synthesizing data collected from all aspects of the power grid's infrastructure, taking appropriate action based on incoming data, and ensuring that power is available to all entities within their regions.

Figure 82. Snapshots of the AJAX-based Client for the Sensor Web



Operators have the ability to organize their workstations and monitors according to their preferences (e.g., viewing data in graphical or numerical format, changing location of various screens, etc.), hampering the ability of multiple workers to work from a single station at a specific point in time. Protection engineers serve a different role in that they are viewing system limits, understanding the ever-changing system and those impacts on system limits, and providing protective schemes to prevent system damage or outages.

Cognitive task analysis (CTA) is a technique used to identify system aspects that impose large cognitive (mental) demands on the operator. Typically, this implies that the task requires significant use of memory, attention, and decision-making. As the focus of CTA is from the user's perspective, the results of a CTA can be used to improve the task or system according to those aspects that are most likely to result in error.

Several methods exist for conducting CTAs, including, but not limited to, task diagrams and knowledge audits. A task diagram is used to direct the rest of the CTA; it provides an overview of the task, identifying the complex cognitive aspects. The steps in conducting a task diagram include obtaining task steps from the operators, using arrows to show relationships and flows between steps, and determining which steps are most cognitively challenging (operators identify these after prompting). Knowledge audits are used to obtain details on the complex steps. To conduct this step, interviewers will typically ask probes to elicit a specific scenario. Additional probes are used to identify what information and/or cues were used to identify the situation, possible solutions, etc.

Interviews were scheduled with system dispatchers at Pacific Gas and Electric Co. (PG&E) and Southern California Edison Co. (SCE). Additionally, group discussion interviews were conducted at PG&E with various engineering groups, including protection engineering. As the use of PMU data across engineering groups within power companies has been somewhat limited to date, one goal of these interviews was to quantify current PMU knowledge and visualizations. Interviews were conducted in July, August, and September 2010. Table 24 provides a general description of individuals who participated in these interviews.

<u>Preliminary Visualizations</u>

From the semi-structured interviews, preliminary needs and requirements documents for dispatchers and engineers for viewing PMU data were developed. Table 25 provides a brief

Table 24. Individuals Interviewed for the CTA and Preliminary Visualization

July 2010		August 2010		September 2010	
Group Discussion	**Dispatcher**	**Group Discussion**	**Dispatcher**	**Group Discussion**	**Dispatcher**
Operations Engineering—5 individuals	Sr. Dispatcher—8 individuals	Operations Engineering—6 individuals	Dispatchers—10 individuals (SCE-2)	Protection Engineering—1 individual	Dispatchers—5 individuals
Protection Engineering—1 individual	Dispatcher—3 individuals	Protection Engineering—2 individuals (SCE-1)	Shift Supervisor—1 individual		Shift Supervisor—1 individual
Operations Engineering—5 individuals	Trainee—1 individual	ISTS—4 individuals			
ISTS—4 individuals	Shift Supervisor—1 individual	Engineering and Technology—2 individuals			
Engineering and Technology—2 individuals		Dispatchers—3 individual			
EMS—1 individual		GE Representative—1 individual			
Dispatchers—1 individual		Unknown—3 individuals			
RAS—2 individuals Transmission Coordinators—2 individuals Unknown—1 individual					

description of the draft need/requirements documents. In general, dispatchers need intuitive data that provided immediate indications of system status. Engineers needed access to raw data to further analyze data to make informed decisions and develop appropriate remedial actions for future events.

Table 25. Preliminary Needs/Requirements for Dispatchers and Engineers

Dispatchers	
• Know the reference (are angles between adjacent points or from a reference point)	• Integrate or incorporate other displays
• Do not add more displays without taking away old	• Easy to get a quick overall view of what is going on in the system
• Ability to choose which PMU data is displayed based on how and where the system is being stressed at that particular time	• Zoom in capabilities for the map as well as the trending data
• Modify timeline on trending data	• Able to simulate what would happen to PMU data with prospective changes to the system
• Access historical data displays (i.e. data from the past week)	• Being able to calculate different points throughout the system to assess system health
• Ability to calculate the angular values at the end of the lines	• Interconnection data
• Maintenance/updating critical	
Engineers	
• Ability to zoom in on trending data at a very granular level	• Simulation capabilities (i.e. what happens to the angular values if a specific line goes down)
• Calculation of different angles at the end of lines and ability to do calculations of any two points on the system	• Easy access to the raw data
• Data displayed in tabular form as well as graphical	• Rate of change and direction of change for angle data

From this point, preliminary visualizations were developed using a paper-based format and presented at the August and September meetings for feedback, in conjunction with continuing the CTAs. Figures 84-87 provide illustrations of these visualizations. The data used in these depictions is fictitious and for illustrative purposes only. Initially, efforts were made to focus on dispatcher visualization needs; subsequently, more engineering-directed visualizations were developed.

Figure 83 depicts a screen that tells the user that there is a problem. A "speedometer" graphic is used to provide an indication of the current angular difference between two points on the line. When angles enter unacceptable ranges, a warning will notify the dispatcher of the problem. The speedometers have been increased to represent a 180-degree plane to allow for an indication of the direction of the power flow.

Figure 84 provides a representative drill-down screen from the visualization presented in Figure 83. When an abnormality is detected, users can bring up a historical trace of the PMU data, complete with tolerance bars. Additionally, the visualization will allow for the selection of specific data series for further inspection and analysis.

Figure 83. Sample Visualization for the 500kV Line

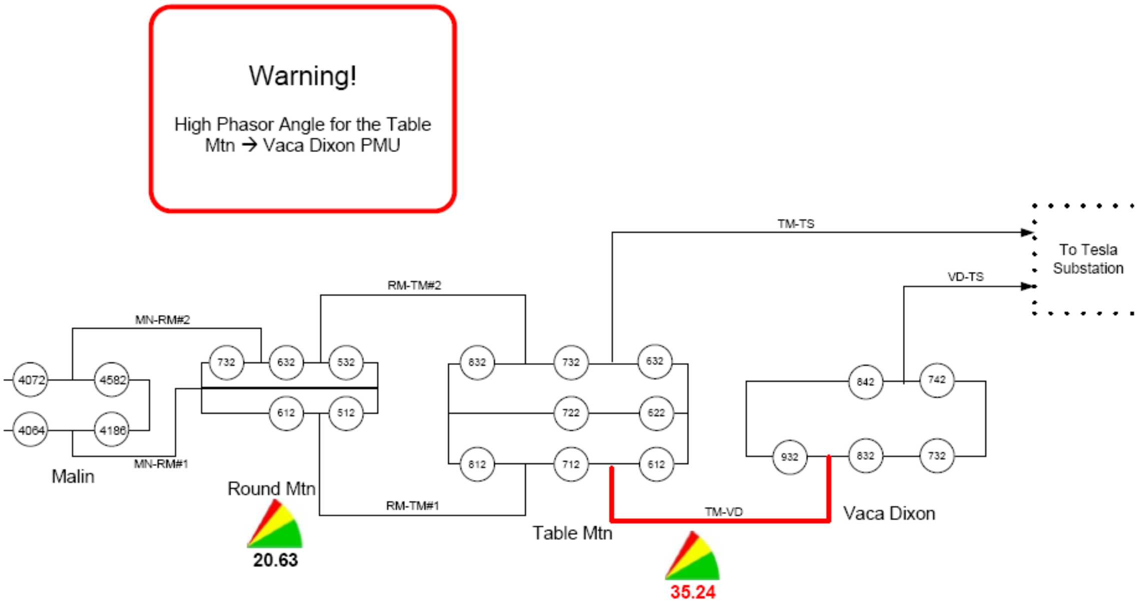
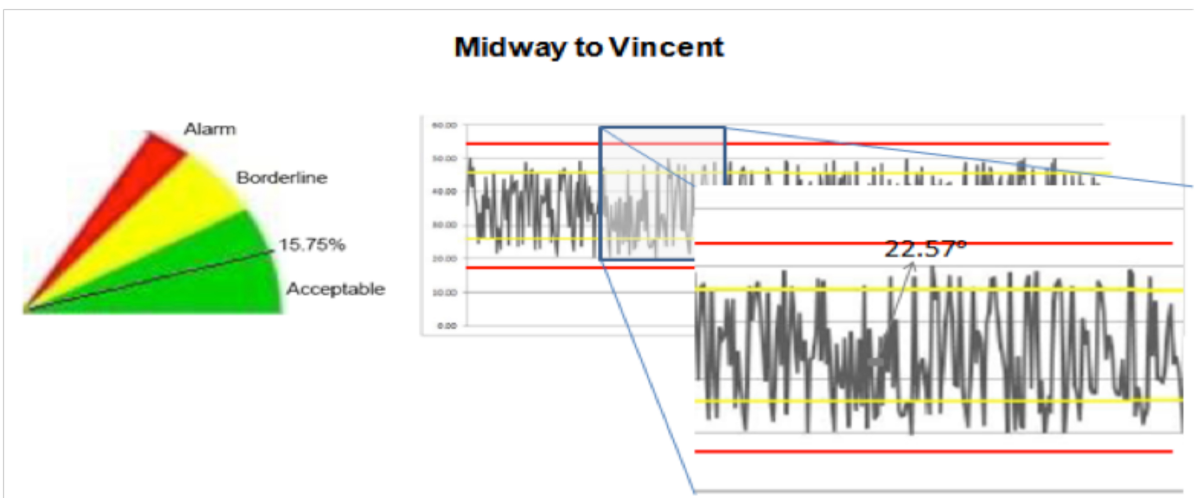


Figure 84. Drill-down Screen for Investigating Trace Information



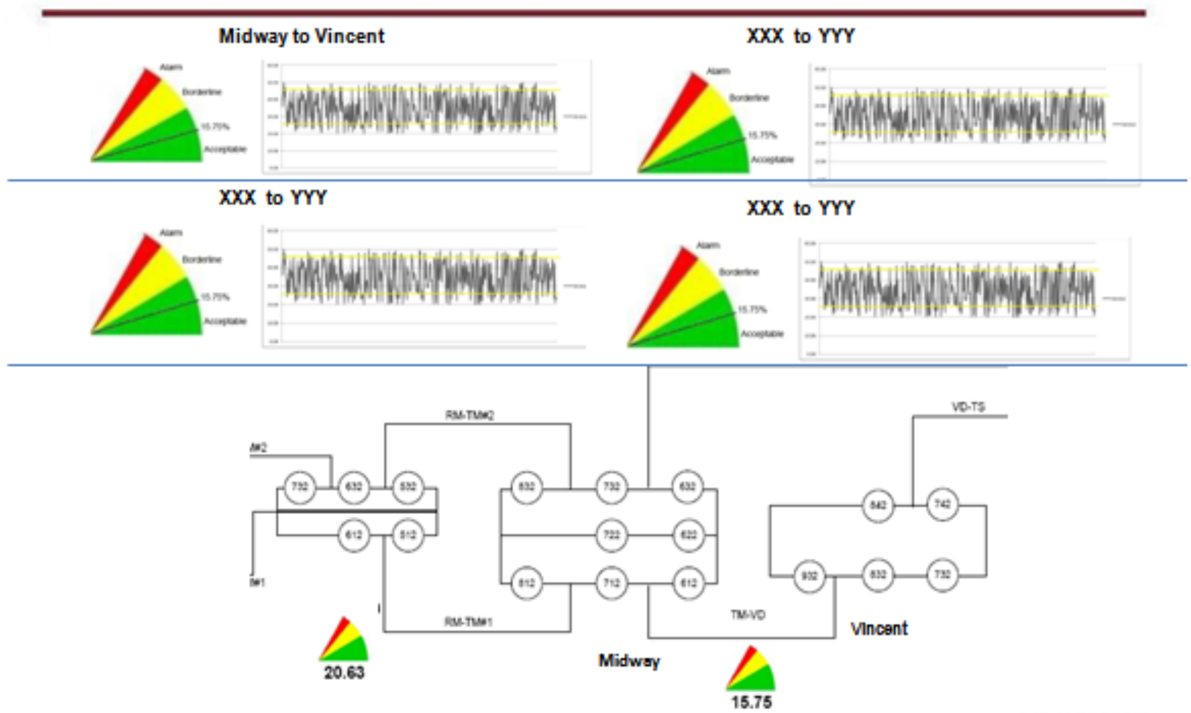


Figure 85 represents the ability of operators to customize screens based on personal preference, current situations, or historical problem areas. Operators will be able to select a number of traces to view continuously while still maintaining the entire overview screen. Figure 86 is a more tabular format of the initial screens. The end-points used to calculate the angular differences are provided along with the specific angle and the direction the angle is moving. The two traces represent the actual and the predicted angular states.

Figure 85. Customization Screen to Allow Users to Preselect Specific Areas to Monitor in More Depth

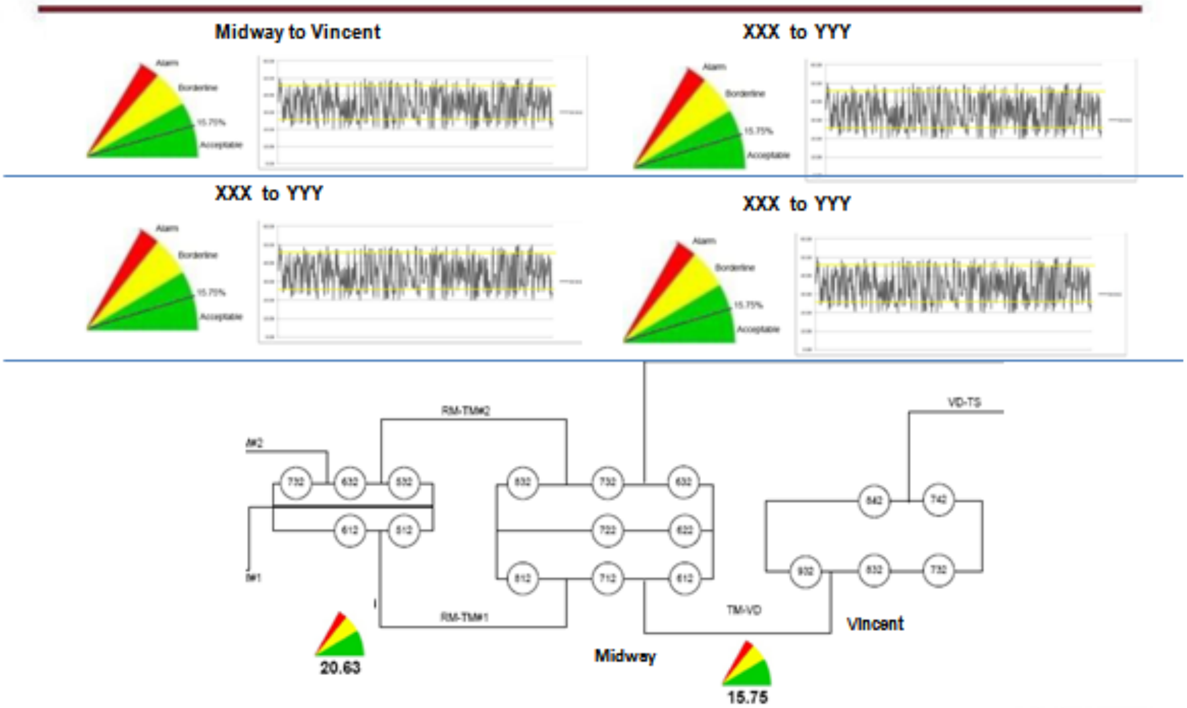
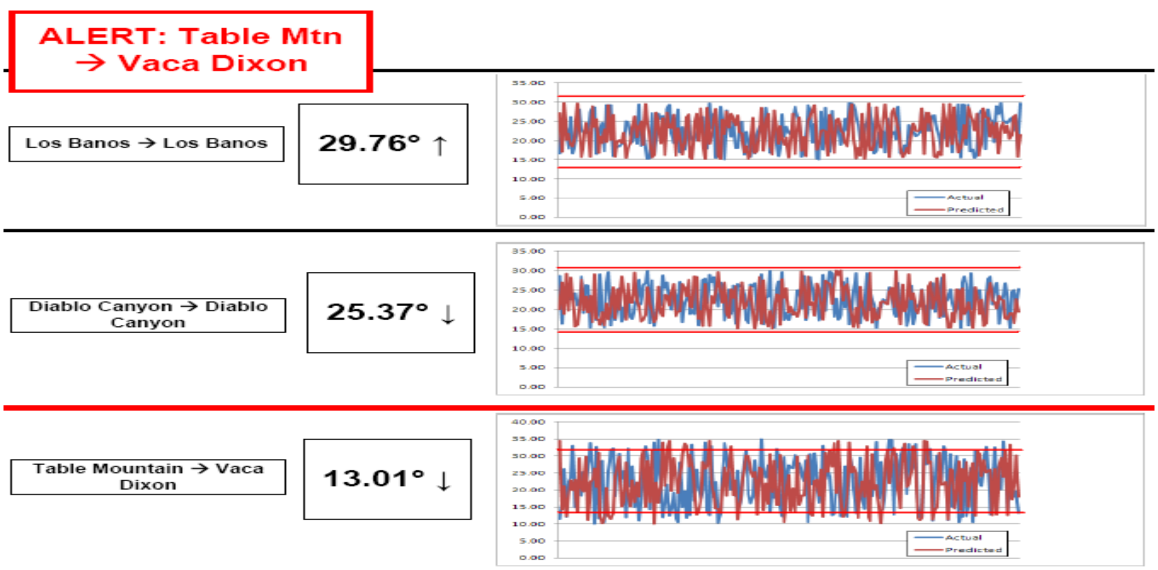


Figure 86. Tabular Representation with Actual and Predicted Angular States



4.2 PROTECTION INFORMATION TOOL PILOT DEMONSTRATION

The objective of this phase of the project was to evaluate and modify, as necessary, the method of presentation of protection information, based on the feedback from the previously described focus groups, comprising control area operators, protection engineers, regulators, etc., in cooperation with the host utilities. Further, the Sensor Observation Service (SOS) for PMU data streams, the geospatial database of PMU measurements, and a description of PMUs and related sensors, with general models and XML encodings through sensor technologies, were developed. The use of real-time data mining algorithms for fault classification was completed and utility prototype testing was conducted in the form of moderate fidelity prototypes using actual field data.

4.2.1 Sensor Web Enablement of PMUs and Related Sensors

A sensor web refers to web-accessible sensor networks and archived sensor data that can be discovered and accessed using standard protocols and interfaces. It facilitates the development of metadata input systems that can scale by orders of magnitude along with increasing numbers of sensors. The OGC standardized sensor metadata (e.g., SensorML [http://vast.uah.edu/SensorML/]) can be updated as new instruments are connected to the sensor network. Also, it is adaptable to the needs of developing metadata systems that can evolve as new innovations are implemented in making measurements. Further, it enables the development of descriptions for sensor network state of health, data quality assurance, and quality control. The standards-based sensor web is agile and can respond to the demands of accurate tracking of sensor and network changes in close to real-time. This sensor web for the electric power grid is further enhanced by creation of actionable intelligence in a timely manner. In other words, it provides the operator with a set of timely, prioritized actions as a result of machines processing the data near the human conceptual level.

4.2.2 System Architecture

In this project OpenPDC, an open-source superPDC developed by the Tennessee Valley Authority (TVA), was chosen as the platform for concentrating synchrophasor data. The source code of OpenPDC is publicly available, so it is easier to customize it rather than use proprietary software. Several Open Geospatial Consortium (OGC) standards, such as SensorML, Sensor Observation Service (SOS), and Observation and Measurement (O&M) were chosen as querying platforms for this framework. All of the OGC standards are based on Geographic Markup Language (GML), which comply with standards used in this project. The National Institute of Science and Technology (NIST) has adopted GML as the standard for exchange of location-based information addressing geographic data requirements for many smart grid applications.

Figure 87 illustrates the architectural overview of the proposed framework of data exchange. Synchrophasor data from Phasor Measurement Units (PMUs) are concentrated on OpenPDC. PostGreSQL database support has been added to OpenPDC, so that database triggers could be used to update Sensor Observation Service (SOS) database as new measurements arrive in OpenPDC as shown in Figure 88. Database triggers update work at database level. Database changes infrequently in stable applications, so this spatial extension of OpenPDC to SOS requires less maintenance than if it was done in application level. The SOS database is updated with the latest synchrophasor data each time OpenPDC receives new data. SOS enables querying of synchrophasor data via internet in XML format which can be accessed through a variety of devices. The client can be a thin client or a thick client.

With the help of Sensor Web, the user can get sensor data in standard OGC format. In addition to power sensors, other sensors such as meteorological sensors, temperature sensors, etc., can be helpful for monitoring the stability of a power system. All kinds of sensor systems can easily be blended in the monitoring system with the help of the sensor web. This versatility of sensor web would enable monitoring of electrical parameters and external parameters through the same application using the same standard, sparing users from the complexities behind the acquisition of data.

Figure 87. Proposed Data Exchange Architecture

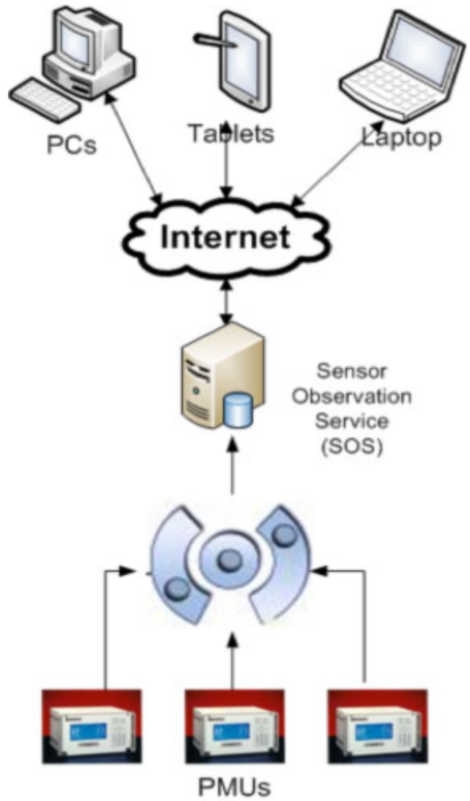
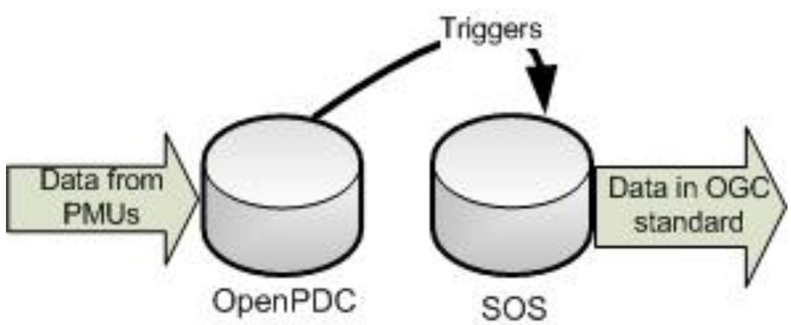


Figure 88. Database Triggers for Up-to-date SOS



4.2.3 Plug-and-Play Web Client

In order to illustrate usability of data exchange framework developed, an AJAX-based client for synchrophasor data exchange was developed.

The client can configure with any sensor web automatically. It gets metadata of sensor web using GetCapabilities request. The metadata consists of sensor technologies, identification of measured parameters, filtering capabilities, formats supported, etc., which are required when sending other requests. The client has also capability of configuring to multiple SOS at the same time making it a probable tool to be used by reliability coordinators, who need to access data from multiple utilities concurrently.

The client populates properties of sensors via SensorML in a tree format as shown in Figure 89, which can be navigated with mouse pointer, so that desired properties of sensors can be accessed. This feature will enable users to access sensor characters such as accuracy, manufacturer contact, model of sensor, etc. A snippet of SensorML (XML document) is shown in Figure 90.

Figure 89. Tree Widget for SensorML Visualization

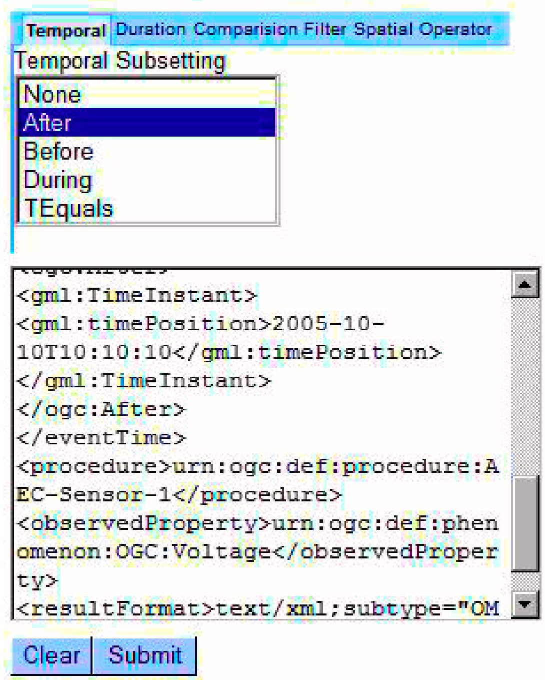
Figure 90. Snippet of SensorML

```
<System id=" Sensor ID ">
  <identification>
   <IdentifierList>
    <identifier name="longName">
     <Term> Sensor Type </Term>
    </identifier>
    <identifier name="shortName">
     <Term>PMU</Term>
    </identifier>
    <identifier name="modelNumber">
     <Term qualifier="urn:ogc:identifier:modelNumber">
    Model Number </Term>
    </identifier>
    <identifier name="manufacturer">
     <Term qualifier="urn:ogc:identifier:manufacturer">
Manufacturer Name </Term>
    </identifier>
   </IdentifierList>
<inputs>
   <InputList>
    <input name=" Parameter Measured ">
     <swe:Quantity
definition="urn:ogc:def:phenomenon:OGC:Voltage"/>
    </input>
   </InputList>
```

Once a client is configured properly with a new SOS, users can query for actual synchrophasor data using User Interfaces (UI) as shown in Figure 91. The data can be queried from SOS based on spatial, temporal and comparative operator individually or combinations of these operators.

For example: Using a combination of the operators, a user can send a request such as “Send me Voltages value less than 132KV from PMUs within 100 mile of Starkville substation between June 10 2011 and July 10 2011.”

Figure 91. User Interface for Querying SOS

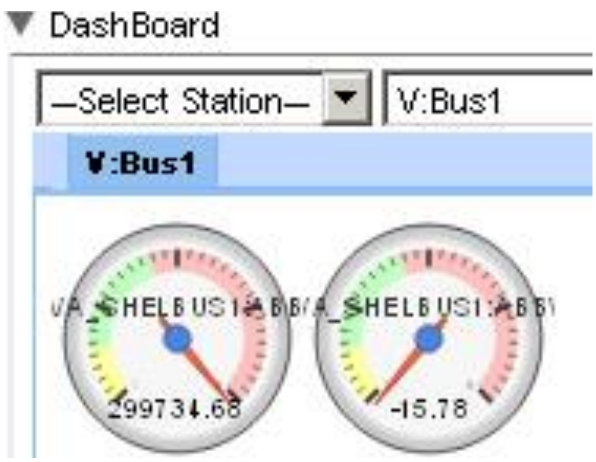


The prototype has multiple areas for visualization of synchrophasor data. The shaded area under the curve in Figure 92 displays trends in data for a specified temporal subset.

Figure 92. Temporal Trends of Synchrophasor Data

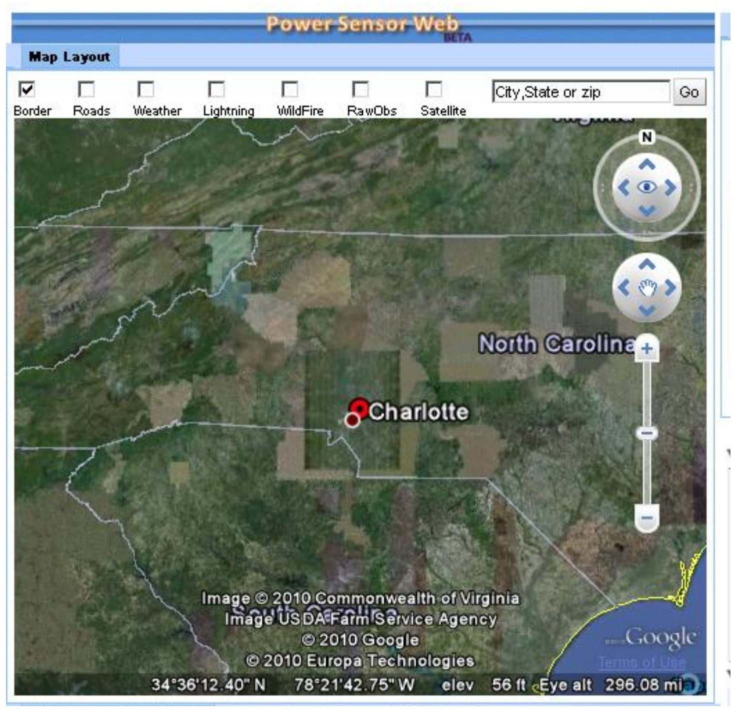
There is a mechanism of polling SOS at a regular interval so that the latest data from a set of PMUs could be displayed as a dashboard view, as shown in Figure 93.

Figure 93. Dashboard View for Latest Data



GoogleEarth™ has been embedded using GoogleEarth plug-in for geospatial visualization of synchrophasor data. All PMU locations in a query result from SOS are marked as shown in Figure 94.

Figure 94. Geo-spatial Visualization



Different KML layers can be added to the map layout to show parameters such as weather, wildfires, street view, etc. Figure 95 shows MODIS satellite data for wildfires in the continental

United States layered on map layout of prototype application. Any user defined KML layer can be added without any modification.

Figure 95. MODIS Wildfire Data Layered on Map

4.2.4 Event Stream Processing of Synchrophasor Data

With the deployment of PMUs, the electric industry now has the capability of monitoring grid health parameters in real time. However, if the underlying information in high-speed data streams cannot be extracted, then it is not possible for operators to make informed decisions. Typically, mathematical calculations such as power flows are used to analyze power systems, but the computation time required for mathematical calculations makes it infeasible to use such tools for real-time situational awareness.

Machine-learning algorithms, such as Artificial Neural Networks (ANN) and decision trees (DT), are being extensively studied for online prediction of power system stability based on phasor data in an actionable period of time. Conventional machine-learning techniques such as ANN and DT are designed to work with a limited amount of sample data. They make multiple scans of data to build a model before making predictions.

Decision trees look promising for modeling of power systems based on phasor data. Decision trees can work equally well with continuous data as with discrete data, and the results of decision trees can be interpreted by humans, which make them an ideal choice for power systems. However, the number of samples from a PMU increases exponentially as the number of parameters being considered or the number of PMUs being deployed increases. For example, in a 24 hour period a single PMU produces 24x60x60x30 = 2,592,000 samples for a single parameter. With the limited computational time and memory available in computer resources, this can limit the size of decision trees built using traditional machine learning algorithms. Therefore, it may be hard to accommodate a huge decision tree in limited computer memory without losing information.

A new method known as Data Stream Mining can extract information from high-speed data streams, facilitating decision-making within constraints of resources and time. In this project, a decision tree was built using a Hoeffding bound (see next section) for a data stream with limited available memory, to guarantee that the result obtained is as good as that of a conventional decision tree. Data stream mining is a good approach for the extraction of information from PMU data stream.

4.2.5 Hoeffding Trees

Domingos and Hulten introduced Hoeffding trees in one of the pioneer works in the area of massive data stream mining [Domingos & Hulten, 2000]. The Hoeffding tree induction algorithm builds a decision tree by scanning the incoming data stream only once. There is no need of storing the data as in traditional decision trees. The tree itself holds sufficient statistics in its leaves to grow the tree and also to make classification decisions of incoming data.

Instead of using a large number of samples, which leads to a huge tree to accommodate in memory, number of samples that are needed to split at each node is determined using Hoeffding bound. The use of Hoeffding bound keeps size of a decision tree within bounds while also maintaining accuracies statistically competitive to traditional batch processing decision trees.

The single most important feature of decision trees is to split a node. The effectiveness of attribute selection to split node determines the accuracy of the decision tree. Criteria such as Gini index and information gain are used for selecting attributes and in determining the "Goodness" of a resulting tree. The calculation of information gain is slightly more complicated in data stream mining than in traditional data stream mining because of the unavailability of simultaneous training data to the algorithms. Domingos and Hulten proposed a criterion known as the Hoeffding bound, which guarantees statistically the same decision for stream mining as that with traditional batch processing algorithms.

The Hoeffding bound states that with probability 1-∂, the true mean of a random variable of range R will not differ from the estimated mean after n independent observations by more than:

$$\epsilon = \sqrt{\frac{R^2 \ln\left(\frac{1}{\partial}\right)}{2n}}$$

This bound is useful because it holds true regardless of the distribution generating the values, and depends only on the range of values, number of observations, and desired confidence. A disadvantage of this approach being so general is that it is more conservative than distribution-dependent bounds.

4.2.6 Result Evaluation Methods

The results of a learning process have to be evaluated on some basis to compare effectiveness of algorithms. The batch learning algorithms use the following evaluation processes.

Holdout Method

In this method of evaluation, a set of random samples are held out from training process as an independent evaluation set. An independent set is used to test the effectiveness of the algorithm on unseen samples. It is generally used when there are abundant samples in training examples.

Cross-Fold Method

In this method of evaluation, training set is divided into K folds. The training is repeated K times using each set as an evaluative "independent" set. The final result is average performance of the algorithm for each train/test set. It is useful when training samples are limited.

Interleaved Test-Then-Train

In this method of evaluation, a sample is used for testing before it is used for training the model. The accuracy is incrementally updated. Also, the algorithm is tested on samples it has never seen before. It makes very effective use of training samples for testing. The downside of this approach is that there is no distinction between training and testing time.

4.2.7 Result Evaluation Measures

Several parameters can be defined to measure the performance of algorithm. Basically there are three areas of performance that are of interest in processing synchrophasor data: how accurate is the classification, how fast the algorithm runs (latency), and how efficiently memory resources are utilized by the algorithm. The following points give some insight into the details for performance measures that have been considered for synchrophasor data processing.

Accuracy Measure

In power systems, normal data are more common than events. Events (such as single line to ground faults) get cleared in a very short time (milliseconds). One of the most common measures of performance of a learning algorithm is accuracy. But the accuracy measure is only useful when the classes to be detected are in the same ratio, which is not always the case. If 98% of instances are normal and 2% percent are faults, then any "dumb" classifier can achieve 98% accuracy by just labeling each incoming instance as normal. A different evaluation measure has to be used that can evaluate the algorithm regardless of the imbalance in classes.

Kappa Statistics, introduced by Cohen in 1960, is a more appropriate measure to represent the performance of stream classifiers. It normalizes the accuracy by that of the chance predictors, which is more credible. The kappa statistic is defined as follows:

$$\kappa = \frac{\rho 0 - \rho C}{1 - \rho C}$$

where $\rho 0$ and ρC are prequential accuracy and chance accuracy respectively. If a classifier is always correct, then $\kappa = 1$. If the accuracy coincided with chance, then the classifier $\kappa = 0$.

Memory Requirement

RAM-hours will used as an evaluation measure of the algorithms used for synchrophasor data mining. Every GB of RAM deployed for 1 hour equals one RAM-Hour. Commercial cloud services such as GoGrid which handle huge amounts of data charge their customers for memory usage based on RAM-hours.

Evaluation Time

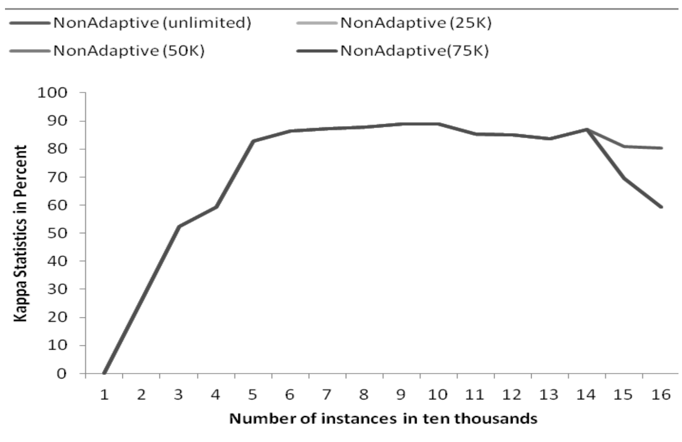
Evaluation time is the time in seconds required for algorithm to run. The interleaved test-then-train method of model evaluation does not have a clear separation between the training and testing phases of an algorithm. A new sample is tested first, then the model is trained on it, so the total evaluation time consists of both testing time and training time.

4.2.8 Results

Experiment 1

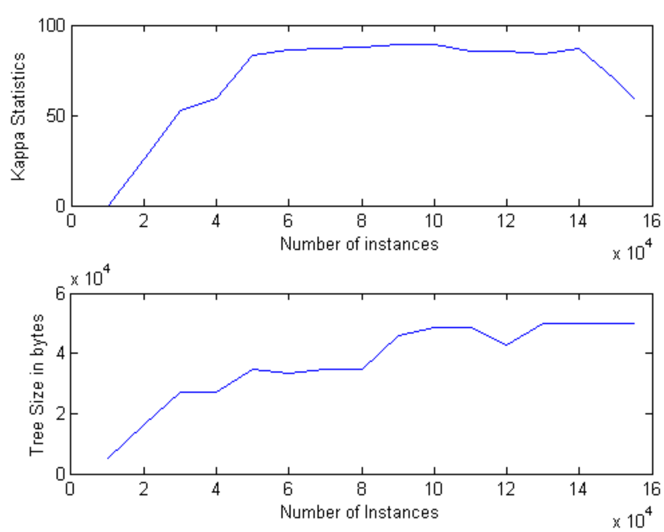
In this experiment, the size of the Hoeffding tree (in bytes) is fixed to see the effect on accuracy of classification. For the purpose of illustration, non-adaptive Hoeffding tree has been chosen for this experiment. Four cases have been chosen in which memory is fixed to unbounded memory (memory of host computer), 25K bytes, 50K bytes and 75K bytes. The performance of the algorithm for each memory limitation are exactly same, while the unbounded memory performance is better after 140K samples as shown in Figure 96.

Figure 96. Kappa Statistics Plot for Algorithm with Fixed Memory



The performance successively deteriorated for 25K, 50K, and 75K bytes when the size of the tree hit the maximum allocated memory, as shown in Figure 97.

Figure 97. Number of Instances vs. Tree Size for Tree Size of 50K Bytes



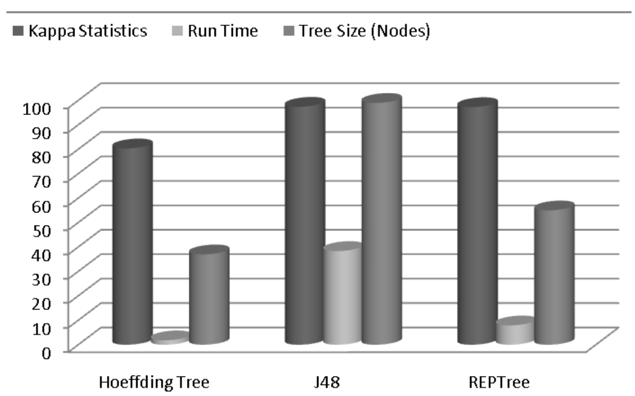
This experiment supported the hypothesis that data stream mining algorithms can adapt to lower memory bounds without deteriorating much in accuracy.

Experiment 2

In this experiment, the performance of a non-adaptive Hoeffding tree with traditional decision tree algorithms (such as J48 and REPTree, available in WEKA) are compared. Figure 98 illustrates the performance comparison based on runtime, size of tree and accuracy. The Hoeffding tree algorithm significantly outperformed others even when the runtime comprises both testing and training phases, while the runtime of J48 and REPTree algorithms includes only model building.

The Hoeffding tree algorithm was also found to be better in terms of efficiency in memory. Tree size is used as a measure of memory resource used by the algorithm because it was the only parameter available for all algorithms under study. The accuracy measure of a Hoeffding tree is found to be slightly lower than that of J48 and REPTree. It may be because of the fact that Hoeffding tree is an “over-pruned” version of a tree. If the number of samples is increased then the Hoeffding tree may even catch up with the accuracy of other decision tree algorithms. Nevertheless, the performance of Hoeffding trees is found to support our proposed method of handling the huge amount of synchrophasor data within the limited memory resources and latency requirements of situational awareness applications.

Figure 98. Performance Comparison of Three Algorithms Based on Runtime, Accuracy and Memory Requirements



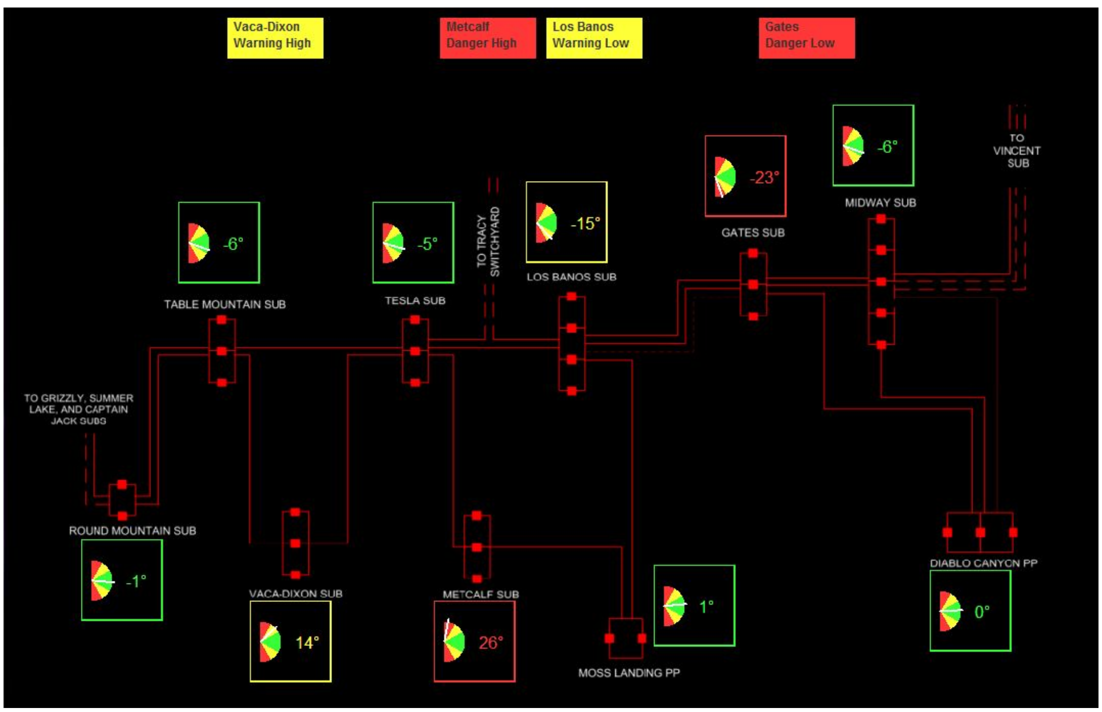
4.2.9 Static Visualizations

Using data from Phase I, three primary and two secondary visualizations were developed to assist various user groups in using PMU data to monitor system stability. Non-functional, static visualizations were developed and presented to control room operators, supervisors, and protection engineers for evaluation and refinement. Typically focus groups of 3-5 users were completed simultaneously and feedback recorded through audio-recorders. Some one-on-one evaluations were also conducted with supervisors to understand implementation and overall usage issues. Feedback was used to refine and modify the static visualizations. Figures 14 – 16 below illustrate the static visualizations for the primary visualizations.

Single-line Diagram

A single line diagram was developed for use in control room operations. Three forms of the single line diagram were developed: filled speedometer (Figure 99), hollow speedometer (Figure 100), and arrows (Figure 101). The speedometers provide information pertaining to current system status with respect to boundaries. Hollow speedometers were developed to reduce operator information processing needs; that is, by removing the areas in which the current angle is not residing, operators would not need to determine current status; rather, the color present illustrates status. Directional arrows and arrow sizes were used to indicate power flows and rates of change. User feedback revealed preferences for the speedometer views, with approximately a 50/50 split on preference for filled and hollowed speedometers. One reason the hollowed speedometer was not preferred by some reviewers was the difficulty in locating the needle of the visualization. Changes in color, size, and length may address this deficiency.

Figure 99. Single Line Diagram – Filled Speedometer View



A drill-down screen for each substation within a system allows operators to compare redundant angular data or to view angular data between discrete points adjacent to a particular substation (Figure 102). Operators can use point and click technology to remove or add PMU trace visualizations to view a history of the angular data (Figure 103). Further, a zoom bar is provided to modify the time course of data viewed (ranging from 30 seconds of data to 5 minutes) (Figure 104). Finally, a tabular format button was provided to allow for data excursions, typically for use by protection engineers (Figure 105). Further, the historian button would provide operators and engineers alike the ability to view 24-hour increments of data, or a user-specified time series of data. Work is continuing on these functionalities.

Figure 100. Single Line Diagram – Hollow Speedometer View

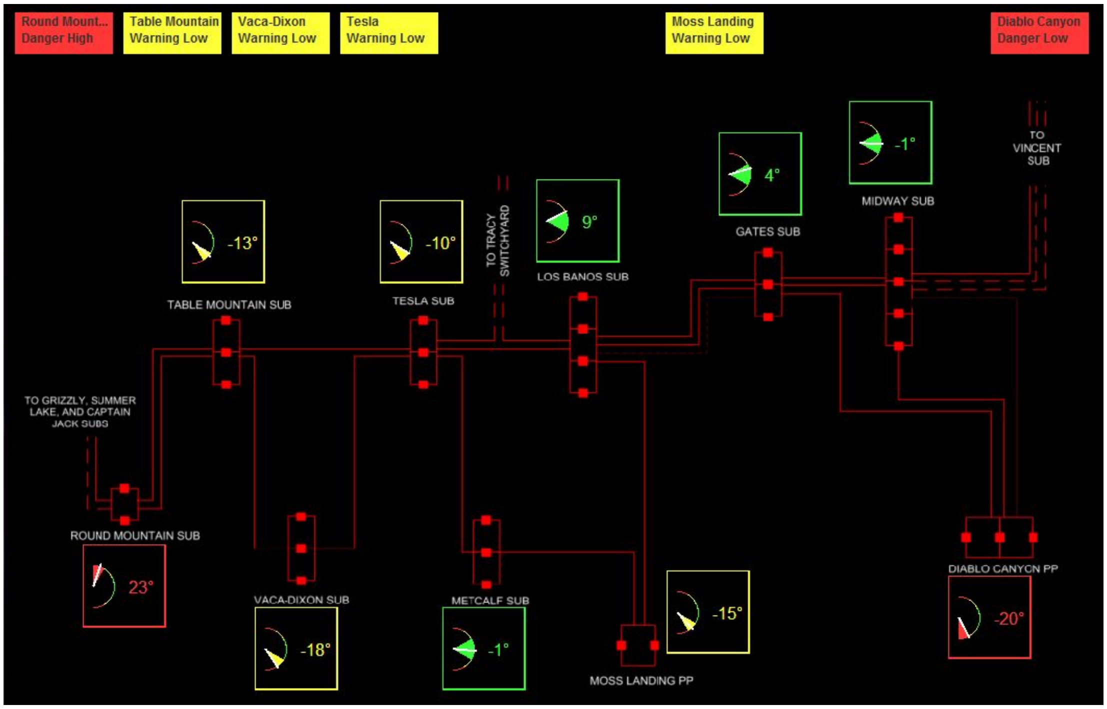


Figure 101. Single Line Diagram – Arrow View

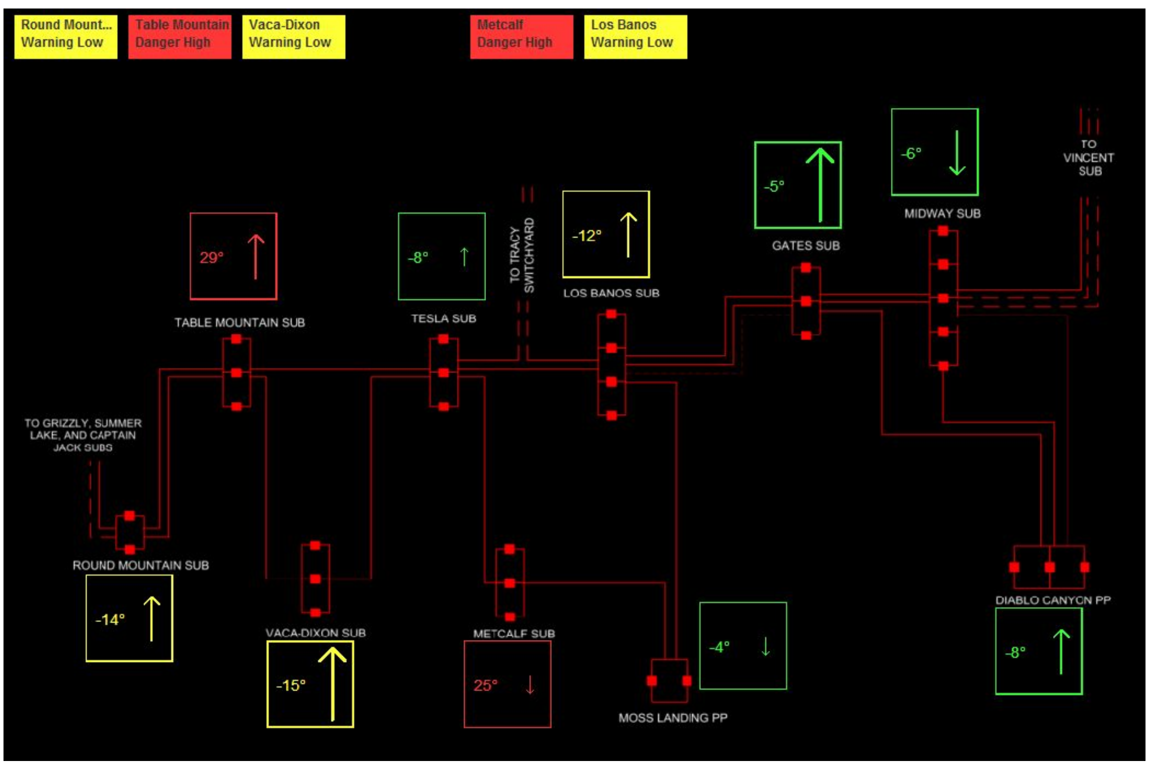


Figure 102. Single Line Diagram – Substation View

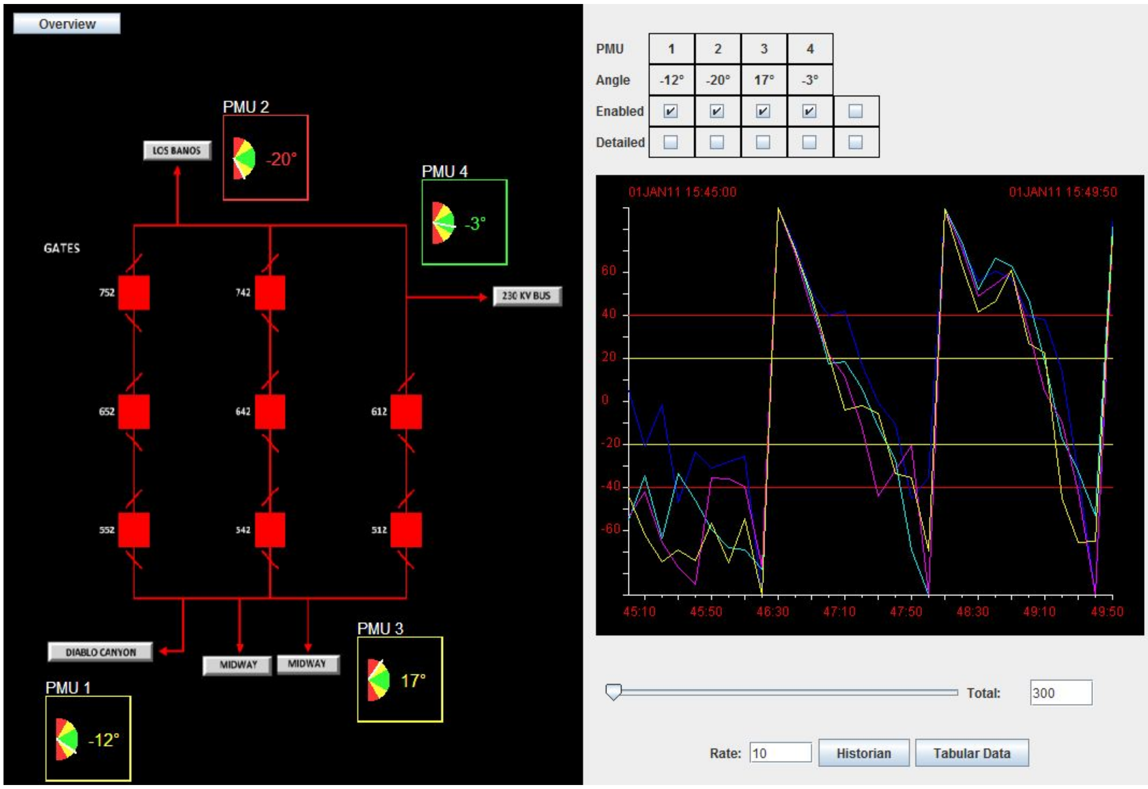


Figure 103. Single Line Diagram – Deletion of PMU Trace Data

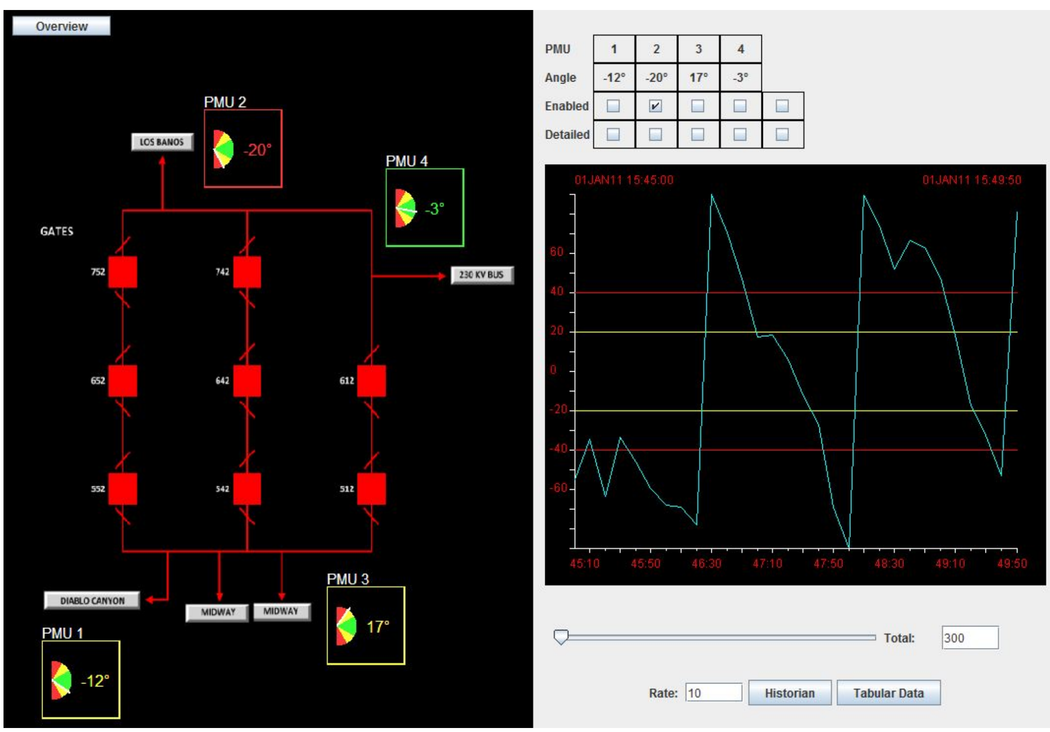


Figure 104. Single Line Diagram – Zooming of Historian Trace

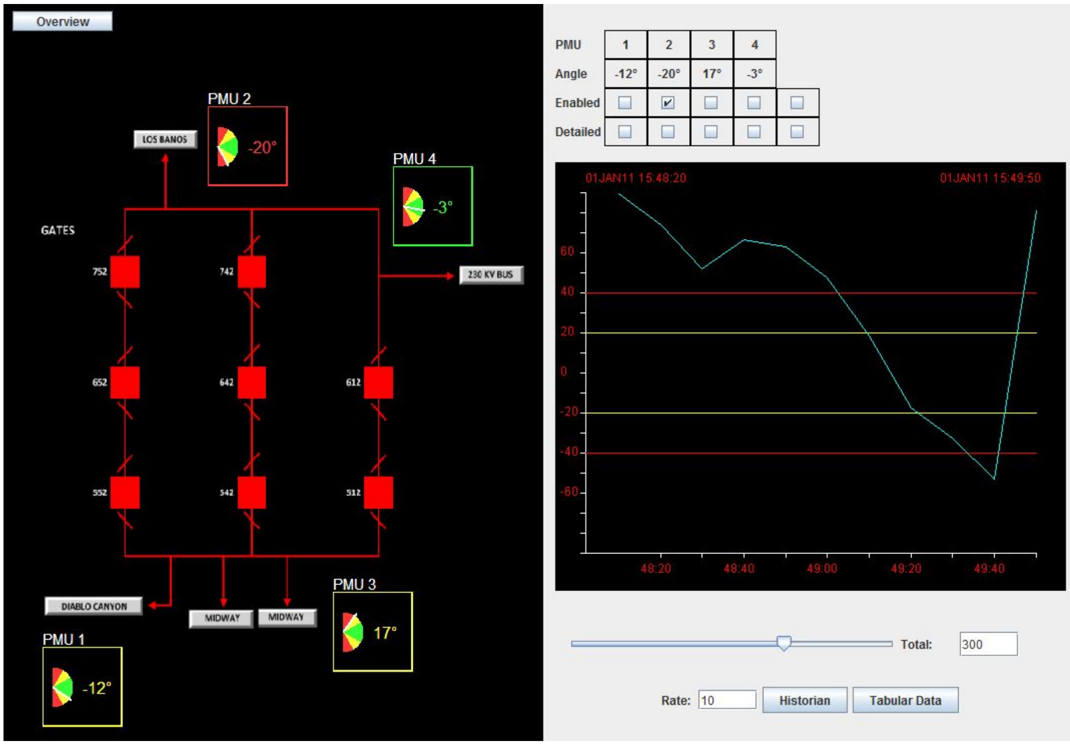
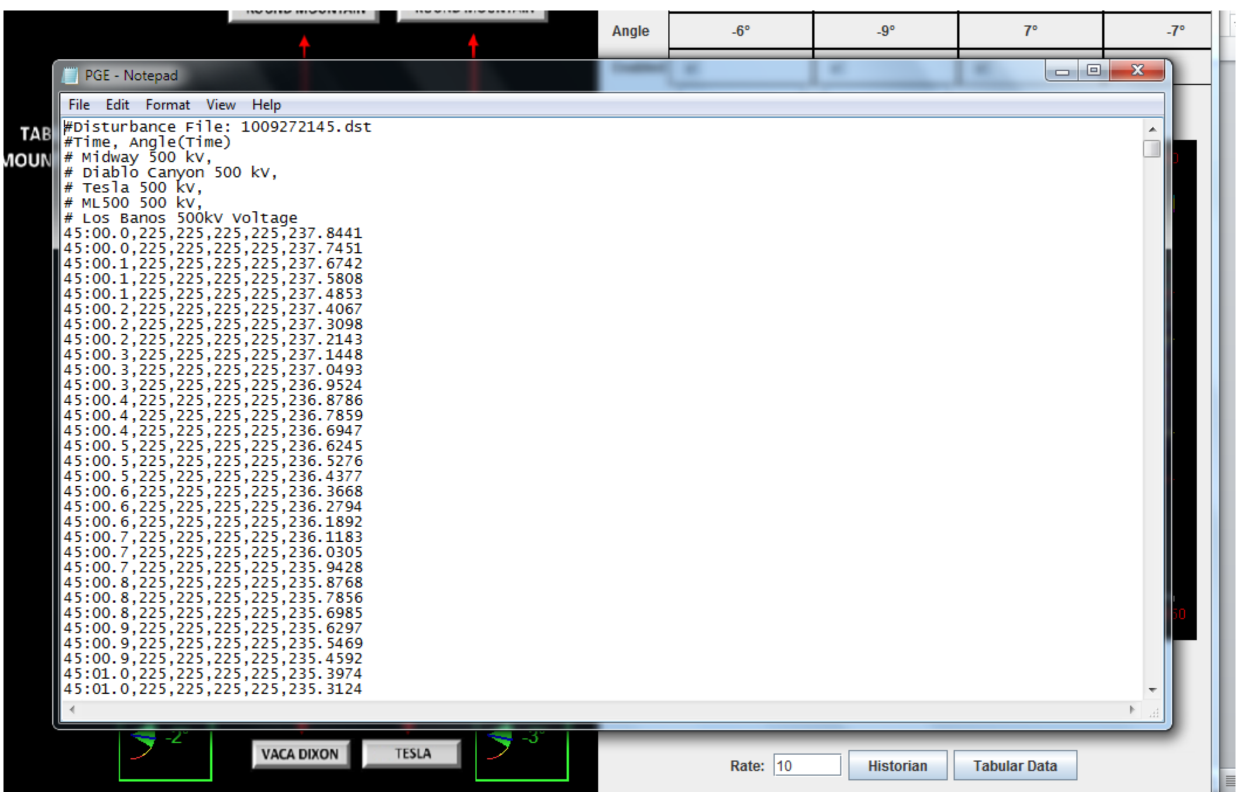


Figure 105. Single Line Diagram – Tabular View

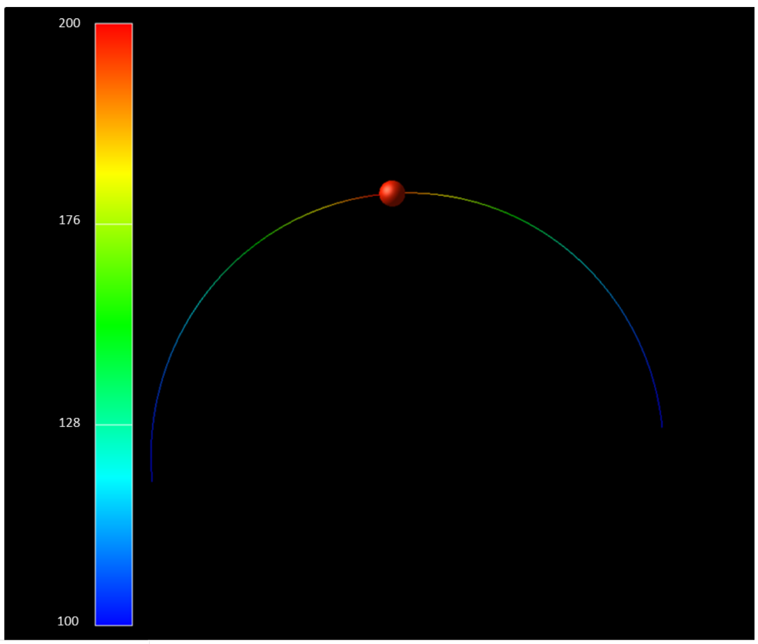



```
#Disturbance File: 1009272145.dst
#Time, Angle(Time)
# Midway 500 kV,
# Diablo Canyon 500 kV,
# Tesla 500 kV,
# ML500 500 kV,
# Los Banos 500kV Voltage
45:00.0,225,225,225,225,237.8441
45:00.0,225,225,225,225,237.7451
45:00.1,225,225,225,225,237.6742
45:00.1,225,225,225,225,237.5808
45:00.1,225,225,225,225,237.4853
45:00.2,225,225,225,225,237.4067
45:00.2,225,225,225,225,237.3098
45:00.2,225,225,225,225,237.2143
45:00.3,225,225,225,225,237.1448
45:00.3,225,225,225,225,237.0493
45:00.3,225,225,225,225,236.9524
45:00.4,225,225,225,225,236.8786
45:00.4,225,225,225,225,236.7859
45:00.4,225,225,225,225,236.6947
45:00.5,225,225,225,225,236.6245
45:00.5,225,225,225,225,236.5276
45:00.5,225,225,225,225,236.4377
45:00.6,225,225,225,225,236.3668
45:00.6,225,225,225,225,236.2794
45:00.6,225,225,225,225,236.1892
45:00.7,225,225,225,225,236.1183
45:00.7,225,225,225,225,236.0305
45:00.7,225,225,225,225,235.9428
45:00.8,225,225,225,225,235.8768
45:00.8,225,225,225,225,235.7856
45:00.8,225,225,225,225,235.6985
45:00.9,225,225,225,225,235.6297
45:00.9,225,225,225,225,235.5469
45:00.9,225,225,225,225,235.4592
45:01.0,225,225,225,225,235.3974
45:01.0,225,225,225,225,235.3124
```

P-V and P-Q Curves

Two visualizations were developed primarily for protection engineers: PV (Figure 106) and PVQ curves (Figure 107). These curves provide a representation of system stability. Color-coding was used to provide information to users about the current state of the system (green/blue for normal or stable status and red for danger status). Users indicated the need to rotate the PV image 90 degrees to match current mental models of how a PV curve is illustrated. Color-coding should be modified to be red from the approximate mid-point of the curve down as a status indicator located in this region would illustrate a crash of the system. Further, the need for (X, Y) and (X, Y, Z) coordinates are needed to provide additional information on the location of the status indicator. These same comments held true for the PVQ curves, with the exception of rotating the images.

Figure 106. P-V Curve (Static)



Dynamic Visualizations

After modifying the static visualizations per the user group evaluations, dynamic versions of these images were developed based on simulated data obtained from a single utility (Figures 109 and 110). A variety of scenarios for to PMUs were visualized ranging from a single line outage to three phase oscillations. Again, evaluations with user groups were conducted and modification and refinements to the visualizations recorded. Modifications for the dynamic visualizations were minor and centered on adding detail (e.g., line connector names for the single line diagram to scaling changes to the (X, Y) PV and (X, Y, Z) coordinates to correspond with user expectations. These have been implemented and are currently ready for final user testing. The utility of these visualizations was demonstrated and comments from protection engineers specifically indicated the potential for using these visualizations to understand the potential impacts of various remedial action schemes on system stability, prior to implementation.

Figure 107. P-V-Q Curve

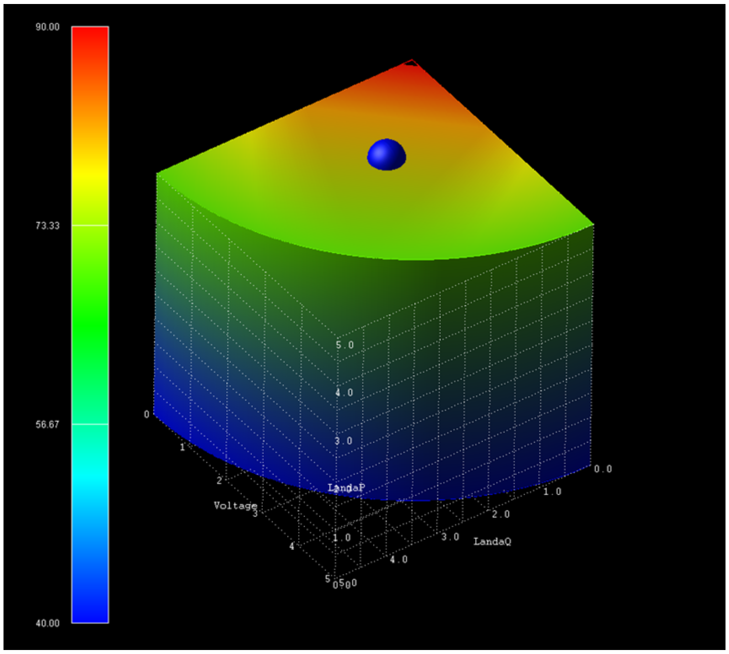


Figure 108. Screen Shot of a Dynamic PV Curve Test Case

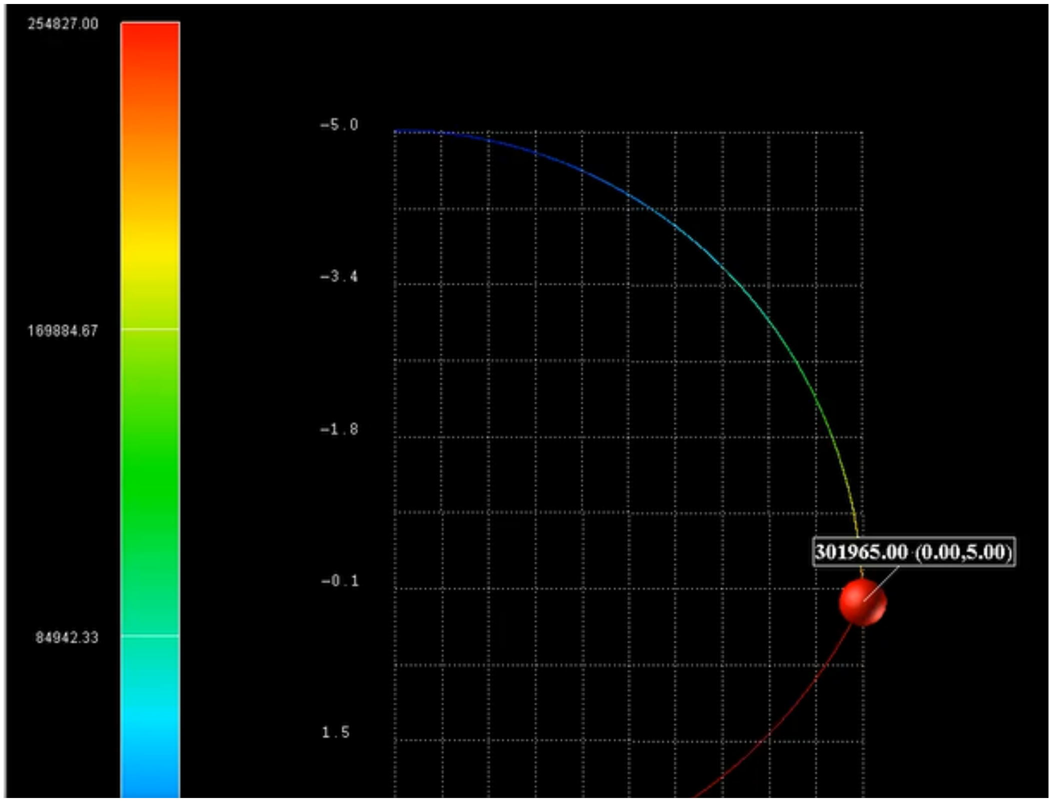
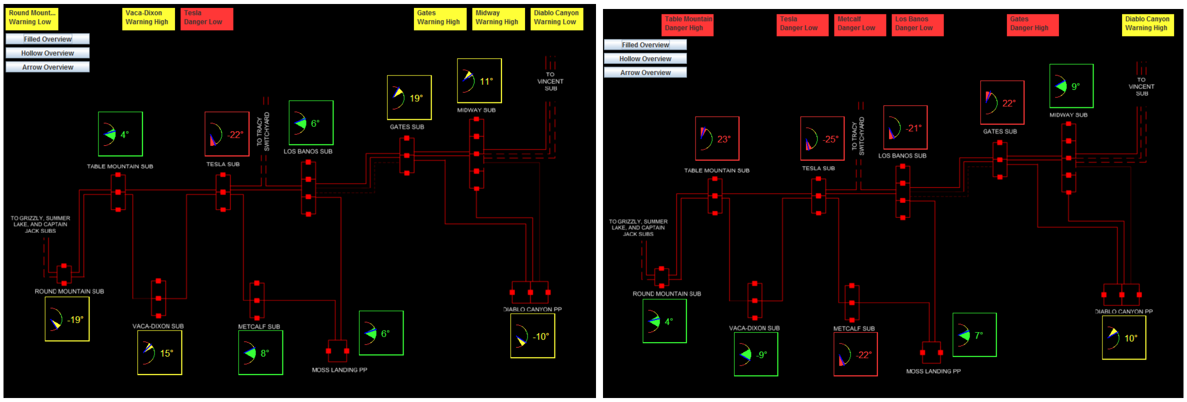


Figure 109. Screen Shot of a Dynamic Single Line Diagram Visualization



Secondary Visualizations

The results of user evaluations indicated the potential need for two additional visualizations. The first visualization is a repeat of a current tool in use at a single utility. This visualization includes all, or a select subset of all, PMUs to assist in gauging the relative angles at a single glance (Figure 110). While the utility of this visualization was rated low during Phase I, the desire for this particular visualization was noted multiple times by users, particularly control room operators. The second visualization provides a relative footprint overview with a directional arrow indicated the angular relation of a region to an adjacent region (Figure 111). Protection engineer evaluations revealed a potential value for this visualization. Due to the labor-intensiveness of developing dynamic primary visualizations, these secondary visualizations have not been pursued, though future efforts may be well served by revising them.

Figure 110. Visualization of Relative Phase Angles

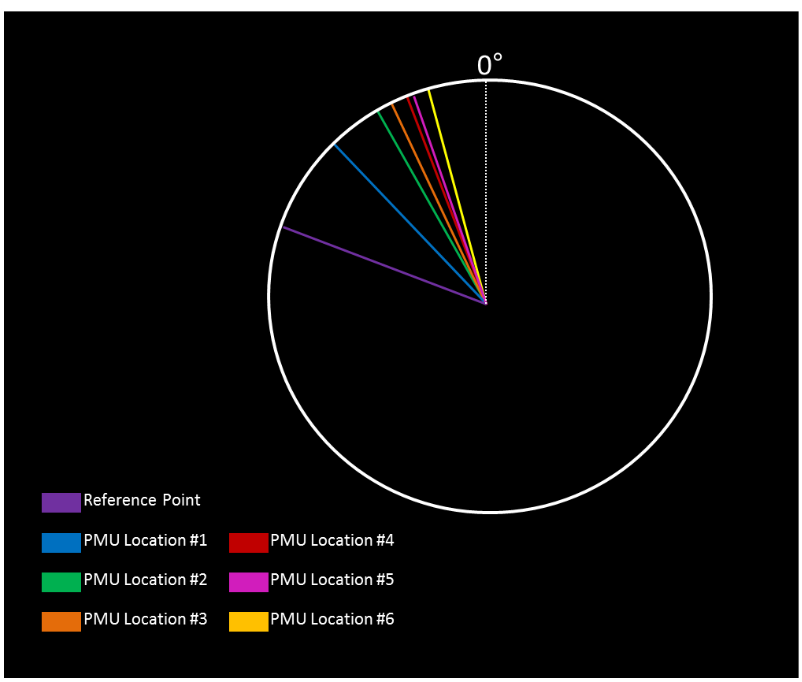
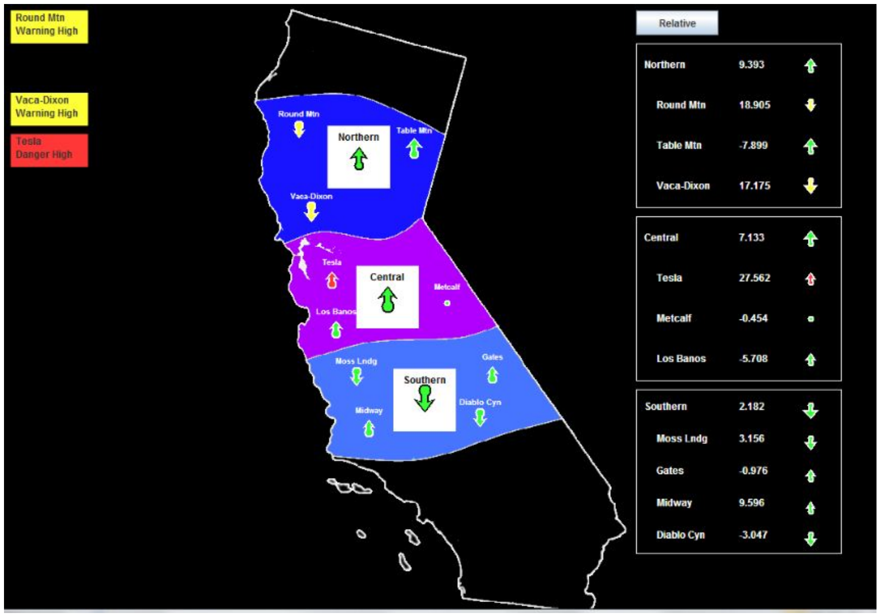


Figure 111. Regional Cluster Visualization



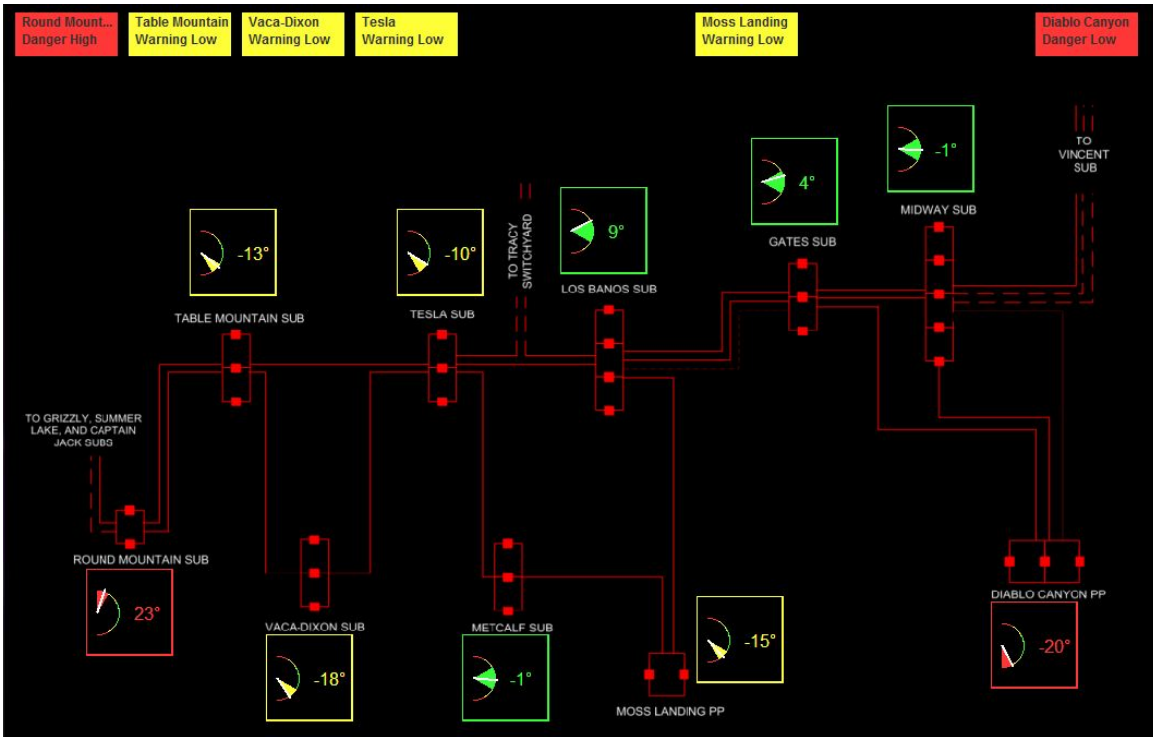
4.3 PROTECTION INFORMATION TOOL FIELD DEMONSTRATION

4.3.1 User Feedback on Visualizations

In this final phase of the project, follow-up visits to each of the host utilities were used to illustrate and discuss implementation of the developed visualizations. The single line diagram overlay was demonstrated with event data provided from a single utility. Implementation of the visualization was to be undertaken by the utilities themselves, as it required programming specific to the utilities' single line diagram visualization. Based on focus group discussions with control room operators, engineers, and protection engineers, the hollow speedometer was the most preferred design (Figure 112). Users indicated a preference for this design due to the limited information present, though the information presented was intuitive in its meaning. The hollow speedometer provided a simple recognition strategy to understand current system status. The drill-down screens were also well received, though changing the speedometers to hollow images was requested. The imaging of the redundancy of the system in these drill-down views was seen as a bonus. Of particular interest was the ability to remove disparate information traces from the graphs.

For the PV and PVQ curves, all requested changes from previous discussions were implemented (e.g., scaling changes, rotation, added detailed information for context and location). To our knowledge, these visualizations have not been implemented at a utility to allow for testing during normal operations. The use of PMU data in this manner (real-time analysis) is currently not standard practice and therefore, there are concerns pertaining to normal operating conditions, contingency analysis, and interpretation. Despite these concerns, engineering knowledge of what these visualizations are displaying does present an opportunity for a change in procedure as information relating to these concerns begins to be better understood.

Figure 112. Hollow Speedometer View

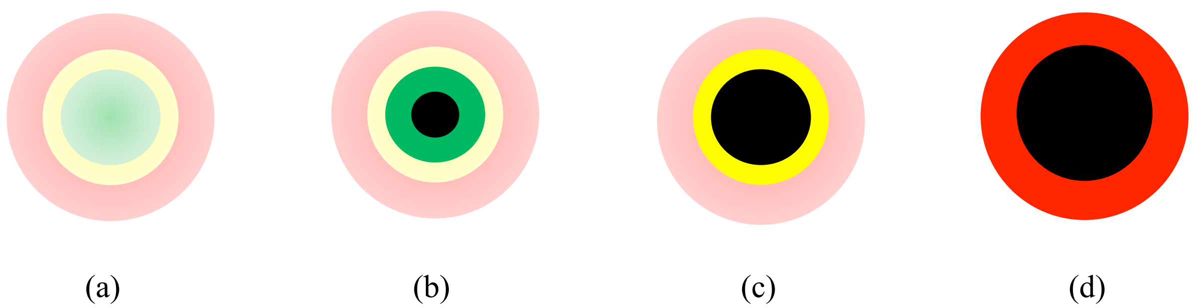


4.3.2 Additional Visualization Development

Though not part of the original project, the researchers continued to receive feedback on visualization design and developed additional, alternative designs. A concern of potential users of the visualizations was that the speedometer view may not represent the actual phasor angle, which in itself may mislead operators in their understanding of current system status. In this manner, a target design using the same color coding was developed (Figure 113). The green range of the target indicates "normal" or "good" system status, yellow indicates "warning", and red indicates "danger" or "critical" systems status. The black center represents the relative size of the phasor angle. As the angle increases, the black center will increase traveling through the various zones. As system status moves through the zones, that area of the display will brighten or highlight. For example, in Figure 113(b), only the green area is highlighted indicating that the system status is good; however, in Figure 113(d), the red area is highlighted indicating the system is in a danger zone.

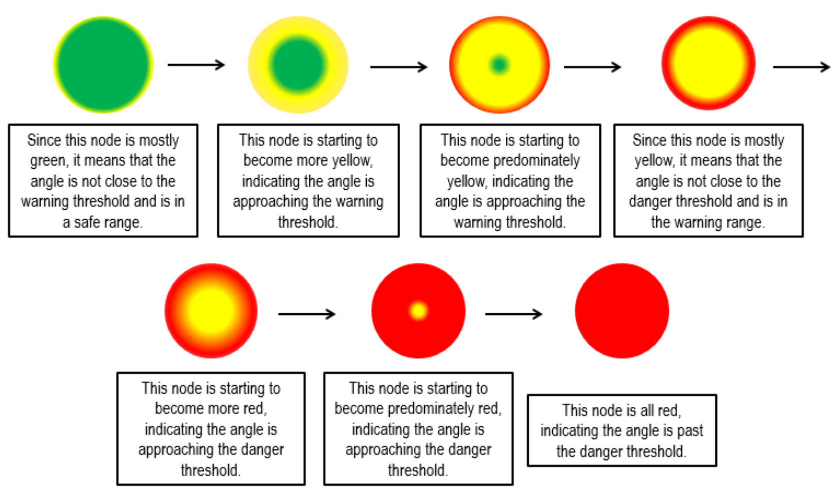
There are some inherent advantages to this alternative design. First of all, this in easily implemented design. It also can be implemented as an overlay to an existing single line diagram or geographical map, or coded into other existing displays. Second, the color scheme is easily recognizable, and has already been deemed acceptable by potential users through our previous design efforts. Third, the display can be easily animated. The black center can pulse at a rate equivalent to that of the rate of change in the phasor angle. As it was repeatedly indicated in our interview that both the magnitude and rate of change of the angle were important pieces of information, the size of the circle would be representative of the magnitude of the phasor angle, while the pulse rate would be representative of the rate of change. Other animation opportunities exist as well. For example, as the systems changes from one status to another, the entire display would briefly flash. Using different rates to represent increases and decreases, as well as changes between normal, warning, and danger, or as the system nears a category boundary, would also provide information to the operator without overloading their already burdened visual system. We would recommend of course that auditory tones be implemented for critical changes.

Figure 113. Possible Designs for Targets



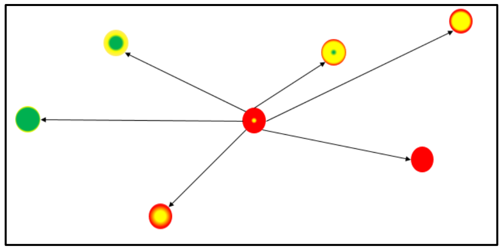
A node network was also considered (Figure 114). Again, a target design is used to represent each node of the network (e.g., PMU, substation, line). As phasor angles increase in magnitude, the lower level colors will be overcome with higher-level colors. Using network nodes implies that the system has a predetermined comparison point. This is consistent with current PMU visualization practice.

Figure 114. Network Node Design



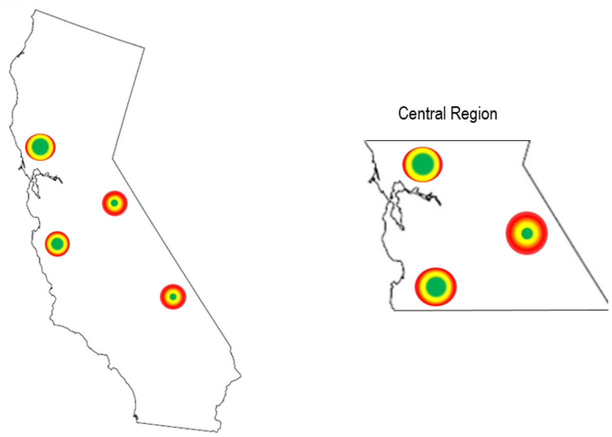
In this system the distance between nodes and the color of the nodes are both indicators of system status (Figure 115). This display takes advantage of geography without the space requirements. Longer lines between nodes can be in direct proportion with geographic distance.

Figure 115. Network Node Design Example



If desired, a geographic design can also be used and is illustrated in Figure 116. With this design we capitalized on the drill down design used in the hollow speedometers. The high level view will present only one PMU at any location; however, when operators drill down to view substation specific data, each PMU will have its own node. In this manner, operators can determine of data from a specific data source is corrupted.

Figure 116. Representative Geographic Node Network



4.4 Conclusions and Recommendations

4.4.1 PIT Research & Development

Phase I of this project required an understanding of current OpenPDC architecture and adaptive relay operator job tasks. A review of relevant literature and the development of a simulation environment to replicate the interoperability of power systems was conducted. Directions for architecture adaptation and extendibility were identified. Additional data sets for testing the simulation space were identified.

Interviews with protection engineers were conducted to understand how users view and use information from various data sources. One common theme of these interviews was that protection engineers at different utilities can perform similar tasks in very different manners, and depending on the utility, described their job functions very differently. Therefore, it was difficult to begin to identify a common visualization scheme that would benefit all users. Also, it was of interest that any visualization developed should have potential utility for other users, such as control room operators who need improved real-time evaluations of system status.

The initial visualizations developed were based on initial discussions with potential users on what aspects of a display design would benefit them the most in terms of their job functions. Additionally, current standard displays in use at select utilities were reviewed and improvements made to those designs, or those designs were eliminated as potential candidates, though the meaning and/or intent of those designs was captured in future versions. Standard color-coding and the use of existing displays was the target of the initial display designs.

4.4.2 PIT Pilot Demonstration

The intent of the second phase of the project was to evaluate various methods for accessing real-time data for effective system status evaluation, and to further test and refine the initial visualizations. Experiments showed that the Hoeffding tree algorithm was found superior to other methods tested (J48 and REPTree) in terms of efficiency, although the accuracy was slightly lower. It is our hypothesis that this decrease in accuracy is the result of over-pruning of the Hoeffding tree, which will likely be overcome as more data flow through the algorithm. Therefore, the performance of the Hoeffding tree algorithm was found to support our proposed method of handing a huge amount of synchrophasor data within the limitations of the memory resources and latency of situational awareness applications.

In general, user feedback on the visualizations was extremely positive, particularly for real-time use. The visualizations developed were intuitive in their use, did not require the use of additional screens, and capitalized on existing displays for real-time analysis of system status. Engineers had a higher affinity to the PV and PVQ curves as they were directly related to post event analysis and again were tied to power and voltage. The main criticism of all of the visualizations is that insufficient data currently exist in the power industry to identify the boundaries for the phasor data. Studies are currently ongoing to identify that information, but establishment of acceptable values for these boundaries is likely not going to be available prior to the field demonstration or the completion of the project. As such, other visualization strategies were suggested to provide users with additional options for completion of this task.

4.4.3 PIT Field Demonstration

The final steps in this project were to develop the simulation environment and run experiments though the environment to identify any needed changes in the visualizations, and to identify or develop measurement metrics of performance. Additionally, initial visualizations will be developed in a basic computing language (Java) to illustrate functionality based on user feedback.

PMU data visualizations for real-time evaluation of system status were, in general, well received in the final demonstrations with utility engineers. However, actual implementation of the visualizations in operations systems was not accomplished. There are number of potential reasons for this. First, and likely the most pressing, is the lack of empirical data for what constitutes the boundaries for the various color codes (normal—green, warning—yellow, danger—red). As the primary visualization was developed as an overlay to the existing single line diagram, it can be implemented by a utility at any time, with simple coding by an information technology specialist. It is our understanding that studies continue to be conducted to generate data for these boundaries, but as yet such data are not available. A second reason for the lack of implementation was the focus on real-time analysis of the system. Protection engineers are primarily post-event evaluators, according to our cognitive task analysis. Therefore, some of these visualizations may seem less relevant. However, further real-time and post-event analyses of these visualizations are recommended before arriving at any firm conclusions regarding their utility for protection engineers.

4.4.4 Research Publications

Research papers published as a result of, or influenced by, this work in visualization of synchrophasor data include:

1. Dahal, N.; Durbha, S.; King, R.; Younan, N.: *Geo-Enabled Synchrophasor Data Exchange Framework Based on Sensor Web*, 2012 IEEE PES General Meeting, San Diego, CA. July 2012.

2. Dahal, N.; King, R.; Durbha, S.; Adhikari, U.; Madani, V.: *Event Stream Processing for Improved Situational Awareness in the Smart Grid,* draft paper submitted to IEEE Transactions on Power Delivery, April 2012.

3. Dahal, N.; King, R.; Madani, V.: *Online Dimension Reduction of Synchrophasor Data,* 2012 IEEE PES Transmission and Distribution Conference, pp. 1-7, Orlando, FL, May 2012.

4. Dahal, N.; King, R.; Younan, N.; Madani, V. *Dimension Reduction of Synchrophasor Data by Optimization of Mutual Information*, 2012 IEEE International Conference on Data Mining (ICDM 2012), Brussels, Belgium, December 10-13, 2012.

CHAPTER 5:
CONCLUSIONS AND RECOMMENDATIONS

5.1 SECURITY/DEPENDABILITY BALANCE

This project has demonstrated a feasible approach for implementing a Security/Dependability Balance algorithm in a real-time operations environment, with the objective of avoiding false trips during times of system stress, thereby reducing the likelihood of cascading outages. The system was thoroughly tested and validated in a university laboratory environment, using industry-standard software and hardware and simulated PMU data, replicating actual utility system performance to the extent possible. The system was then successfully transferred to PG&E's and SCE's respective protection system laboratories for proof-of-concept testing before actual field implementation. In fact, two somewhat different approaches were used by PG&E and SCE in terms of the specific hardware employed and where the algorithm software was installed, but achieved the same result: an adaptive protection system using synchrophasor data to enhance the reliability and security of a major transmission path.

The Field Demonstration phase of the Security/Dependability Balance implementation, unfortunately, ran into unexpected issues and delays. The chief issue encountered was one of logistics: because of limited personnel and contracting resources available at the utilities, it took much longer than planned to schedule the resources necessary for the installations. Given that this was a research project, utility operations and infrastructure projects necessary for scheduled system enhancements took precedence in the allocation of personnel time. The adaptive relaying systems also needed to utilize the existing utility communications infrastructure, and additional delays were caused by the scheduling difficulties resulting from coordinating with routine operations and system outages, where necessary to install and calibrate equipment. In retrospect, it is clear that installation of such a system into existing operations is not a trivial task, and would best be accomplished where the application is considered and planned along with other system enhancements. It would also have been much more straightforward if the demonstration had been done on a transmission line owned by a single utility, thereby simplifying the amount of logistical coordination that was required.

From a technical standpoint, the field implementation of the Security/Dependability Balance system was, eventually, a success: both PG&E and SCE were able to set up and commission their respective systems, even though there were some differences in the specifics of the hardware and software used by the two utilities in their respective implementations of the system.

But, as a result of the scheduling difficulties, an additional two years' time extension to the project's original three-year plan was required. By the time the field data collection period began, significant grid enhancements had been made to the Western transmission network. In fact, many of these system enhancements were most likely responsible for the lack of resources to perform the research implementation, as capital projects had first priority on the utilities' budgets and personnel resources, and the research project had secondary priority. The net effect was that by the time the Security/Dependability Balance system was implemented, the Western grid was stronger and more stable than before, and periods of system stress were very few. Other contributing factors were relatively mild West Coast weather with line outages being less common, transient and dynamic events were rare and relatively mild, and overall system loading and power transfers were lower than in recent years (a combination of low hydro in the Pacific Northwest and lower loads in Southern California). As a result, very little useful event data was collected for validating the field performance of the voting scheme, although the limited data that were collected were positive in that regard.

In terms of lessons learned, it should be noted that the Security/Dependability Balance algorithm needs to be "trained" with system studies based on the expected conditions in real-life operation. In this case, Heavy Summer and Heavy Winter load flow cases were provided by SCE and PG&E at the start of the project (October 2009), and the algorithm developed from those cases. (Heavy Summer and Heavy Winter conditions are defined by specific time periods during the year to which they apply; the algorithm is not operational outside those specified times.) By the time the system was implemented in the field in late 2013, both base cases were in need of updating, as system configuration had changed due to additions and retirements of various transmission lines, generators, etc., and generation and load patterns had changed as well. Time and budget did not allow for another round of algorithm development in this research project, but in actual utility practice these base cases and the corresponding algorithm revisions would be routinely updated by the utility's planning engineers. It is the judgment of the project team, which included pragmatic utility engineers, that the cost and effort to implement the adaptive relaying algorithms is reasonable, and commensurate with other, similar types of applications in the operating environment.

5.2 Impedance Relay Zone Encroachment

The project also demonstrated an economical and feasible approach to implementing an Impedance Relay Zone Encroachment system to inform protection engineers via on-screen alarms and notifications when impedance relay settings should be re-evaluated and/or updated for increased system security. The algorithm was adapted from previous research results (funded by the California Energy Commission's Public Interest Energy Research (PIER) program), and further developed to use real-time PMU data. Initial studies had pinpointed a 230 kV transmission line in PG&E's northern area as the best candidate for testing the system, in terms of the possibility of seeing potential zone encroachment. However, synchrophasor data for the system would be needed at locations where no PMUs were currently installed, and surrogate PMUs at other locations were not deemed to be suitable replacements. The project team concluded that the Midway – Vincent 500 kV line would provide an adequate substitute; while there was little hope for seeing actual zone encroachment on this line, the implementation of the algorithm and the basic performance of the system could be demonstrated. The system was first implemented and tested in a university laboratory environment, then successfully transferred to PG&E's and SCE's respective protection system laboratories for proof-of-concept testing prior to actual field implementation. In practice, the system was shown to be accurate, relatively straightforward in implementation, and could be used on any impedance relay for which PMU data were available.

5.3 Data Visualizations

Mississippi State developed preliminary data visualizations based on state-of-the-art Cognitive Task Analysis (CTA) methodologies, along with interviews and workshops with utility operators and engineers to present the initial data visualizations and collect a substantial amount of comments and feedback for further improvements. Advanced visualizations were developed, using simulated PMU data to evaluate the revised designs against the updated performance criteria. A second round of interviews and workshops was conducted with the utilities to validate the revised visualizations and solicit further feedback. The resultant Protection Information Tool (PIT) was enthusiastically received by the utilities, whose personnel contributed substantially to its usefulness. Important knowledge was gained from practicing utility engineers, operators and technicians regarding methods of visualizing synchrophasor-based protection system data. The prototype visualization tools were developed and vetted to meet the cognitive demands of utility personnel for clear and concise representations of complex data that can be quickly absorbed, allowing problems to be efficiently analyzed, and decisions formulated and carried out expeditiously.

5.4 Recommendations for Further Research

Both utilities that participated in this project, PG&E and SCE, recommend that the Security/Dependability system undergo further field validation for an additional year, so that more data can be gathered for analysis. This can be easily done by leaving the system in place and operating, continuing to monitor the system, and performing data evaluation as the data become available. Both utilities have engineering personnel that have the expertise to perform this work, it does not require significant effort, and they are willing to perform this work without any additional outside funding, as it falls within their line operations. Plus, both utilities are leaders in the field of synchrophasor technology implementation, and have a stake in definitively validating its performance and furthering the technology. Using the additional data from the extended field monitoring period, the utilities can re-assess the system models and associated decision tress and updated where required, with the results being instructive to utility engineers regarding how often this process should done in order to maintain sufficient relaying system accuracy.

The Impedance Relay Zone Encroachment system can be implemented on any transmission lines of interest with a modest amount of additional programming. Very little, if any, additional research is considered necessary by PG&E or SCE in order to use the system as need dictates. However, more operator experience with the system, especially on transmission lines where zone encroachment occurs frequently, will be valuable for further validation purposes, as well as determining any additional improvements that may be desirable.

Visualizations of synchrophasor data are critical to many real-time applications, such as the Impedance Relay Zone Encroachment system, and should be a critical part of their development and demonstration. The next step in research and development is for utilities to install these visualizations in their control room environment for hands-on experience; actual in-service use is necessary for optimization of the technology. Additional programming and adaptation of the PIT needs to be done by the utilities themselves, because of the custom engineering necessary to address the different hardware, software and data communications systems implemented by the different utilities.

Finally, this project represents an effort to reduce to practice just a few of the many potentially beneficial synchrophasor-based applications identified in the Kema "Phasor Business Case" Study [Novosel et al., 2007], and more such applications are likely to emerge in the future. Some, like the aforementioned Generator Out-of-step Relaying, are applications in the field of Adaptive Protection, which can benefit greatly from the accurate, high-resolution and time-synchronized data provided by PMUs. But there are a number of other applications in other areas of grid operations and planning that would also be enabled or greatly enhanced by synchrophasor data, and research efforts to reduce these promising applications to utility practice are urgently needed and strongly recommended.

GLOSSARY

A	Ampere
AC	Alternating Current
AJAX	Asynchronous JavaScript and XML
API	Application Programming Interface
CART	Classification and Regression Tree
CB	Circuit Breaker
CTA	Cognitive Task Analysis
DC	Direct Current
DNP	Distributed Network Protocol
DT	Decision Tree
EMS	Energy Management System
EPL	Event Processing Language
ESP	Event Stream Processing
GML	Geography Markup Language
GOOSE	Generic Object-Oriented Substation Event
GPS	Global Positioning System
GUI	Graphical User Interface
GWT	Google Web Toolkit
HMI	Human-Machine Interface
HS	Heavy Summer
HW	Heavy Winter
IEC	International Electrotechnical Commission
I/O	Input/Output
IP	Internet Protocol
kV	kiloVolt
ms	millisecond
MVAR	MegaVolt-Ampere Reactive
MW	MegaWatt
O&M	Observation and Measurement
OGC	Open Geospatial Consortium
PC	Personal Computer

PDC	Phasor Data Concentrator
PDU	Protocol Data Unit
PIT	Protection Information Tool
PLC	Programmable Logic Controller
PMU	Phasor Measurement Unit (aka “synchrophasor”)
PPA	Primary Phasor Archive
PV	Power-Voltage
PVQ	Power-Voltage-reactive power
R	Resistance (ohms)
RAS	Remedial Action Scheme
RDBMS	Relational Database Management System
RMB	Receive Mirrored Bit
RTAC	Real-Time Automation Controller
RTS	Run-Time System
SER	Serial Event Record
SES	Sensor Event Service
SOA	Service-oriented Architecture
SOAP	Simple Object Access protocol
SOS	Sensor Observation Service
SQL	Structured Query Language
SWE	Sensor Web Enablement
TCP	Transmission Control Protocol
TCS	Time-alignment Client Server
TMB	Transmit Mirrored Bit
TTL	Transistor-Transistor Logic
UDP	User Datagram Protocol
UTC	Coordinated Universal Time (French: “universel temps coordonné”)
WAM	Wide-Area Measurement
WAMS	Wide-Area Measurement System
WECC	Western Electricity Coordinating Council
X	Inductance (ohms)
XML	Extensible Markup Language
Z	Impedance (ohms)

REFERENCES

Begovic, M.: *Wide Area Protection and Monitoring,* Protection, Automation and Control World, Autumn 2009.

Benmouyal, G.; Schweitzer, E.; Guzmán, A.: *Synchronized Phasor Measurement in Protective Relays for Protection, Control, and Analysis of Electric Power Systems,* 29th Annual Western Protective Relay Conference, Spokane, Washington, October 22-24, 2002.

Bernabeu, E.: *Methodology for a Security-dependability Adaptive Protection Scheme Based on Data Mining,* Ph.D. dissertation, Virginia Polytechnic Institute and State University, 2009. [http://scholar.lib.vt.edu/theses/available/etd-12172009-201325/]

Centeno, V.; Thorp, J.; Durbha, S.; Cibulka, L.: *Application of Advanced Wide Area Early Warning Systems with Adaptive Protection – Phase 1 Technical Report,* interim project report to the US DOE, December 2010.

Centeno, V.; Thorp, J.; Babski-Reeves, K.; Cibulka, L.: *Application of Advanced Wide Area Early Warning Systems with Adaptive Protection – Phase 2 Technical Report,* interim project report to the US DOE, December 2011.

Centeno, V.; Thorp, J.; Phadke, A.: *Advanced Protection Systems Using Wide Area Measurements,* project report #CEC-500-2013-078, California Energy Commission, September 2010.
[http://www.energy.ca.gov/2013publications/CEC-500-2013-078/CEC-500-2013-078.pdf]

Dahal, N.; Durbha, S.; King, R.; Younan, N.: *Sensor Web for Interoperability in Power Systems,* 2009 North American Power Symposium (NAPS), pp. 1-4, October 2009.

Dahal, N.; Mohan, V.M.; Durbha, S.S.; Srivastava, A.K.; King, R.L.; Younan, N.H.; Schulz, N.N.: *Wide Area Monitoring Using Common Information Model and Sensor Web,* IEEE/PES 2009 Power Systems Conference and Exposition (PSCE '09), pp. 1-7, March 2009.

Domingos, P.; Hulten, G.: *Mining High-speed Data Streams,* Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 71-80, 2000.

Elizondo, D.: *Hidden Failures in Protection Systems and Its Impact on Power System Wide Area Disturbances,* M.S. dissertation, Virginia Polytechnic Institute and State University, 2000.

Guoyin, W.; Jun, H.; Qinghua, Z.; Xianquan, L.; Jiaqing, Z.: *Granular Computing Based Data Mining in the Views of Rough Set and Fuzzy Set,* 2008 IEEE International Conference on Granular Computing, 2008.

Horowitz, S.; and Phadke, A.: *Power System Relaying (3rd ed.),* Elsevier, 2008. [http://www.knovel.com/knovel2/Toc.jsp?BookID=3343]

Institute of Electrical and Electronics Engineers (IEEE): *IEEE Standard for Synchrophasors for Power Systems,* C37.118-2005 (Revision of IEEE Std. 1344-1995), 2006.

Institute of Electrical and Electronics Engineers (IEEE): *Wide Area Protection and Emergency Control System,* IEEE Power Engineering Society, Power System Relaying Committee, Protection Subcommittee Working Group C-6, 2002. [http://www.pes-psrc.org/Reports/Wide%20Area%20Protection%20and%20Emergency%20Control_2002-.pdf]

Kai, S.; Likhate, S.; Vittal, V.; Kolluri, V.S.; Mandal, S.: *An Online Dynamic Security Assessment Scheme Using Phasor Measurements and Decision Trees,* IEEE Transactions on Power Systems, vol. 22, pp. 1935-1943, 2007.

Mori, H.: *State-of-the-Art Overview on Data Mining in Power Systems,* 2006 IEEE PES Power Systems Conference and Exposition, pp. 33-34, 2006.

North American Electric Reliability Corporation (NERC): *Real-Time Application of Synchrophasors for Improving Reliability,* October 2010. [http://www.nerc.com/docs/oc/rapirtf/RAPIR%20final%20101710.pdf]]

Novosel, D.; Snyder, W.; Vu, K.; Cole, J.: *A Business Case Study on Applying Synchrophasor Measurement Technology and Applications in the California and the Western Electricity Coordinating Council Grid,* CEC PIER Report #CEC-500-2013-045, June 2007. [http://www.energy.ca.gov/2013publications/CEC-500-2013-045/CEC-500-2013-045.pdf]

Phadke, A.; Thorp, J.; Adamiak, M.: *A New Measurement Technique for Tracking Voltage Phasors, Local System Frequency, and Rate of Change of Frequency,* IEEE Power Engineering Review, vol. PER-3, pp. 23-23, 1983.

Rockwell Automation: *ControlLogix 1756 System.* [http://www.ab.com/en/epub/catalogs/12762/2181376/2416247/360807/360809/]

Schweitzer, E.; Whitehead, D.; Guzmán, A.; Gong, Y.; Donolo, M.; Moxley, R.: *Applied Synchrophasor Solutions and Advanced Possibilities,* 2010 IEEE/PES Transmission and Distribution Conference and Exposition, pp. 1-8, April 2010.

Schweitzer Engineering Laboratories (SEL): *Fast Operate Command Processing,* SEL Application Guide II, 2011.

Schweitzer Engineering Laboratories (SEL): *Remote I/O Module Instruction Manual,* SEL-2515, 2008.

Schweitzer Engineering Laboratories (SEL): *Synchrophasor Vector Processor Instruction Manual,* SEL-3378, 2010.

Symmetricom, Inc.: *PCI/PCIe Time and Frequency Processor User's Guide,* 2009.

Thomas, M.K.: *Implementation of the Security-dependability Adaptive Voting Scheme,* Virginia Polytechnic Institute and State University, 2011. [http://scholar.lib.vt.edu/theses/available/etd-05102011-225623/]

United States Department of Energy (US DOE): *The Electric Power Outages in the Western United States, July 2-3, 1996,* Report to the President, August 1996.

U.S.-Canada Power System Outage Task Force: *Final Report on the August 14, 2003 Blackout in the United States and Canada: Causes and Recommendations,* April 2004.

Western Electricity Coordinating Council (WECC): *Western Interconnect Synchrophasor Program (WISP).* [http://www.wecc.biz/awareness/Pages/WISP.aspx]

Western Systems Coordinating Council (WSCC): *Western Systems Coordinating Council Disturbance Report for the Power System Outage That Occurred on the Western Interconnection, August 10, 1996,* October 1996.

PUBLICATIONS

Adhikari, U.; Morris, T.H.; Dahal, N.; Pan, S.; King, R.L.; Younan, N.H.; Madani, V.: *Development of Power System Test Bed for Data Mining of Synchrophasors Data, Cyber-attack and Relay Testing in RTDS,* 2012 IEEE Power and Energy Society General Meeting, pp. 1-7, July 2012.

Bernabeu, E.; Thorp, J.; Centeno, V: *Methodology for a Security-dependability Adaptive Protection Scheme Based on Data Mining,* IEEE Transactions on Power Delivery, vol. 27, no.1, January 2012.

Dahal, N.; Durbha, S.; King, R.; Younan, N.: *Geo-enabled Synchrophasor Data Exchange Framework Based on Sensor Web,* IEEE Power and Energy Society General Meeting, pp. 1-6, July 2012.

Dahal, N.; King, R.; Durbha, S.; Adhikari, U.; Madani, V.: *Event Stream Processing for Improved Situational Awareness in the Smart Grid,* draft paper submitted to IEEE Transactions on Power Delivery, April 2012.

Dahal, N.; King, R.L.; Madani, V.: *Online Dimension Reduction of Synchrophasor Data,* 2012 IEEE PES Transmission and Distribution Conference, pp. 1-7, May 2012.

Dahal, N.; King, R.; Younan, N.; Madani, V. *Dimension Reduction of Synchrophasor Data by Optimization of Mutual Information*, 2012 IEEE International Conference on Data Mining (ICDM 2012), Brussels, Belgium, December 10-13, 2012.

Habibi-Ashrafi, F.; Johnson, A.; Vo, S.; Catanese, D.: *Phasors Point the Way,* Transmission & Distribution World, Vol. 63, No. 1, pp. 26-32, January 2011.

Madani, V.; Parashar, M.; Giri, J.; Durbha, S.; Rahmatian, F.; Day, D.; Adamiak, M.; Sheble, G.: *PMU Placement Considerations – A Roadmap for Optimal PMU Placement,* 2011 IEEE/PES Power Systems Conference and Exposition (PSCE), pp. 1-7, 2011.

Mazur, D.C.; Quint, R.D.; Centeno, V.A.: *Time Synchronization of Automation Controllers for Power Applications,* IEEE Transactions on Industry Applications, Vol. 50, Issue 1, pp. 25-32, 2014.

Pal, A.; Sanchez-Ayala, G.A.; Centeno, V.A.; Thorp, J.S.: *A PMU Placement Scheme Ensuring Real-Time Monitoring of Critical Buses of the Network,* IEEE Transactions on Power Delivery, vol. 29, no. 2, pp. 510-517, April 2014.

Quint, R.; Mazur, D.; Badayos, N.: *A Protective Relay Voting Scheme Utilizing Automation Controllers,* 2012 IEEE Industry Applications Society Annual Meeting (IAS), pp. 1-8, 2012.

APPENDIX A:
MATLAB IF-ELSE LOGIC GENERATOR

```
%This function reads in the CART file manually created, and generates an
%if/else representation of the decision tree.
%
%The output of this function is a .m file that will be called continuously
%by the Security Assessment function for determinig the system state.
function [] = buildTree(CARTFile)

fid = fopen(CARTFile); % CARTFile must be formatted to a specific standard
C = textscan(fid, '%s %f %s %f');
fclose(fid);

LTCount = 0; RTCount = 0; IfCount = 0;
TotLTCount = 0; TotRTCount = 0; TotIfCount = 0;

outFile = fopen('treeOutput.m','w');

% Determine counts for IFs RTs and LTs for the whole tree.
for x = 1:length(C{1})
  if strcmpi(C{1}(x),'O')||strcmpi(C{1}(x),'L')||strcmpi(C{1}(x),'R')||strcmpi(C{1}(x),'RI')
    TotIfCount = TotIfCount + 1;
  elseif strcmpi(C{1}(x),'LT')
    TotLTCount = TotLTCount + 1;
  elseif strcmpi(C{1}(x),'RT')
    TotRTCount = TotRTCount + 1;
  end
end

endC = TotIfCount - TotRTCount;

s = sprintf('function [ FLAG ] = treeOutput(PDCstream)\n');
fprintf(outFile,s);

% Build tree from input file.
for x = 1:length(C{1})
  % Options for node types are:
  % O: Origin Node
  % L: Left Child Node
  % LT: Left Terminal Node
  % R: Right Child Node
  % RT: Right Terminal Node
  % RI: Initial Right Child Node (used for counter)

  if strcmpi(C{1}(x),'O')
    s = sprintf('if( %s <= %f )\n',char(C{3}(x)),C{4}(x));
    IfCount = IfCount + 1;

  elseif strcmpi(C{1}(x),'L')
    s = sprintf('if( %s <= %f )\n',char(C{3}(x)),C{4}(x));
    IfCount = IfCount + 1;

  elseif strcmpi(C{1}(x),'LT')
    s = sprintf('%s\n',char(C{3}(x)));
    LTCount = LTCount + 1;

  elseif strcmpi(C{1}(x),'R')
    s = sprintf('else\nif( %s <= %f )\n',char(C{3}[5](x)),C{4}(x));
    IfCount = IfCount + 1;

  elseif strcmpi(C{1}(x),'RT')
    for y = 1:abs(C(x)-C(x-1))-1
      s = sprintf('end\n');
      fprintf(outFile,s);
      endC = endC - 1;
    end

    s = sprintf('else\n%s\nend\n',char(C{3}(x)));
```

```
    RTCount = RTCount + 1;

  elseif strcmpi(C{1}(x),'RI')

    if TotIfCount - TotRTCount > 0 && LTCount > RTCount
      s = sprintf('end\n');
      fprintf(outFile,s);
      if endC > 0
        endC = endC - 1;
      end
    end

    s = sprintf('else\nif( %s <= %f )\n',char(C{3}(x)),C{4}(x));
    IfCount = IfCount + 1;

  else
    error('Incorrect node classification in input txt file.')
  end

  fprintf(outFile,s);

end

while endC >= 1
  for x = 1:endC
    s = sprintf('end\n');
    fprintf(outFile,s);
    endC = endC - 1;
  end
end

if x == length(C{1})
  s = sprintf('Tree built')
end

end
```