2018 Georgia Tech Fault & Disturbance Analysis Conference

Abstract: “Strategic Data Sharing for Improved Fault Analysis”

Author: Fred Elmendorf

Address: 1206 Broad St, Chattanooga, TN 37402

Phone: 423-702-8136

Email: felmendorf@gridprotectionalliance.org

ABSTRACT

From an electrical perspective, the bulk electric system (BES) operates as a large interconnected grid. However, the physical infrastructure is owned and maintained by many distinctly separate companies. The arbitrary business boundaries make the process of fully analyzing faults and disturbances on the system impossible in some cases and difficult at best, when the event is near the edge of the company boundary, or even worse inside the neighbor’s boundary.

Many modeling techniques and lots of tribal knowledge have been used to minimize the impact of these business boundaries, but only so much can be done without actually having the data to describe the involvement of your neighbor’s equipment. A number of utilities have begun to think outside of the ‘company box’ to identify effective ways to address the issue.

This paper will describe an open source software approach that has the ability to eliminate the problem of data restriction by company boundary. Automated disturbance analytic tools that are already in place to analyze and report results on faults and disturbances have been modified to take advantage of standard business practices that facilitate automated data sharing. Simple data sharing for historical purposes can be accomplished in many ways, but if the desired scenario is to facilitate better operation and restoration of the power system, the software systems have to be designed to work together seamlessly. The paper will include results of a case study where multiple utilities are coming together to implement this strategy and contain details and references sufficient for anyone to replicate this solution. Automated strategic data sharing for improved fault analysis is now a practical solution for improving the way power systems are operated and maintained. The icing on the cake is that this is all made possible through presently available free open source software.

PROBLEM

When electric power system components fail to operate as designed, even small deviations from normal can create a significant financial burden, and customer dissatisfaction. The severity of the event is a big factor but if corrective action for this event, and preventative steps to reduce future events are taken quickly, the overall impact can be reduced. To facilitate those positive actions, some fundamental information is necessary; something happened, what happened, when did it happen, where did it happen? The need for this information is well understood, and the fault disturbance and analysis (FDA) conference is dedicated to improving knowledge and techniques for every aspect of power system disturbances. Regardless of the techniques used to distill available data into information, if your neighbor has some of the data you need your analysis grinds to a halt. Even with the most cooperative relationships, it simply takes time to request, receive, and analyze the data necessary to accurately and thoroughly evaluate the event, and valuable time is lost.

In many instances, the only resource available to get answers for events near the edge of the system is tribal knowledge. If you're lucky enough to have a good contact at the neighboring utility, they might find time to provide some helpful information or data. Unfortunately, neither of these scenarios is a good solution. Even if the tribal knowledge provides good answers, it is limited to one or a few engineers and is not a sustainable resource. If the neighboring utility's engineer is willing and available to provide helpful information you can eventually get the needed answers, but the process introduces a tremendous time lag.

Traditional tools used for disturbance and fault analysis are typically desktop applications on an engineer's desk. The idea of 'sharing data' has been to send an email with a file attached and hope that the person receiving the file can open it, analyze it, and get similar results, or to simply send a screenshot of what you have observed. And, while both methods have been helpful over the years, neither method provides any automation or timeliness in getting to actionable information.

STRATEGY

Leveraging technology is a better strategy than relying on emails and phone calls. Properly configured computer applications can securely and automatically share specific information with a targeted recipient. A computer-based solution provides consistent and timely information and can support near real time results for all involved parties. But historically, an automated strategy has been a non-starter due to isolated networks, different types of hardware and software, and often you don't know the right person to call on the other side.

Most companies suffer from some level of 'vendor lock-in' but over the past 5 to 10 years open source software (OSS) had significantly changed the landscape. Well developed and supported vendor and device independent software tools are now available to address every aspect of a fully automated disturbance analytic and fault location system.

The strategy presented here is to take advantage of modular offerings in the OSS community to overcome the issue of accessing data from external sources to solve internal problems. Using IT infrastructure that is standard practice and already in place at utilities can make this a reality.

SOLUTION

Several previous papers for the FDA conference have presented building blocks for the development of an automated disturbance and fault analysis system that includes one double-ended and multiple single-ended distance calculation methods. The building blocks as they have been presented solve the overall problem in steps that are broken down into getting the data, analyzing the data, storing the data and analytic results, providing proactive notification, and visualizing the information. These building blocks are the result of collaborative efforts between utilities, the Electric Power Research Institute, and developers. The importance of this paper is to add one more block, strategic data sharing. Each of the blocks will be summarized to set the context for strategic data sharing, and references for the OSS projects will be included at the end of the paper. A high-level architecture diagram that includes these building blocs is shown in Figure 1.

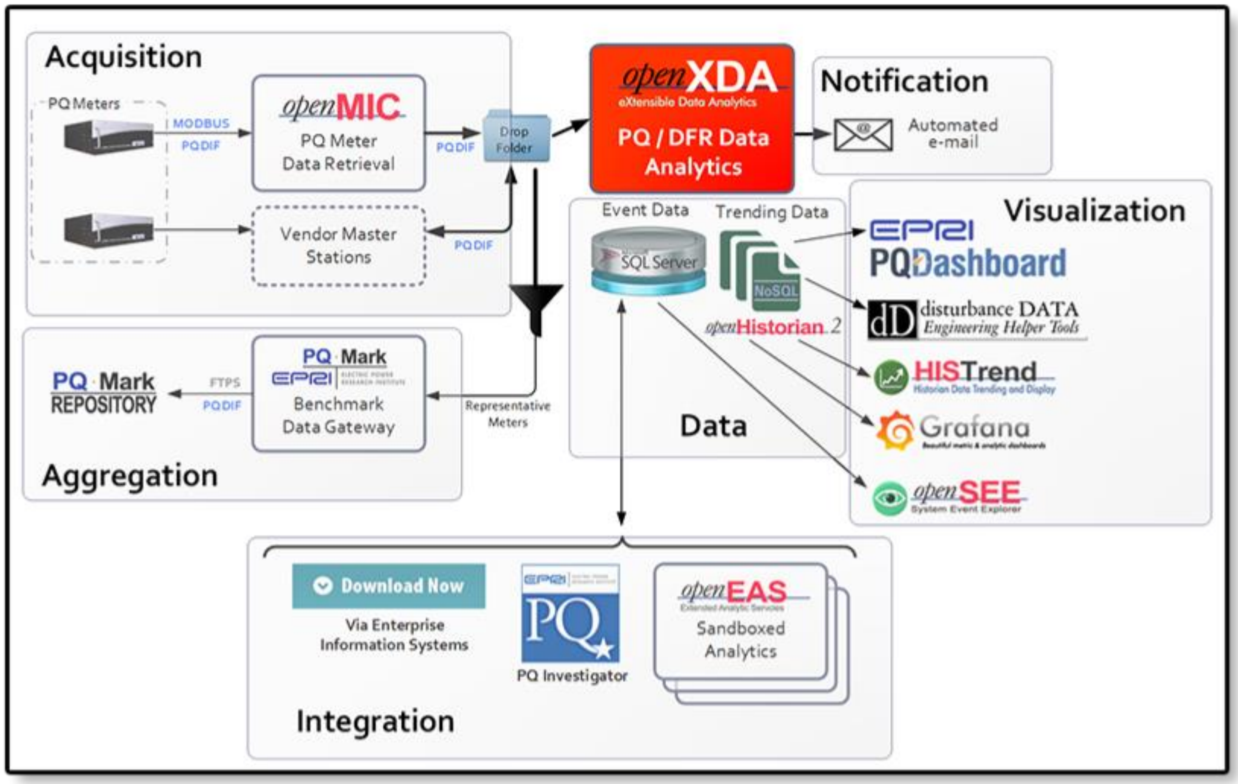

Figure 1. High-Level Architecture, Automated Disturbance Data System

Getting the data: The initial path of least resistance to gather data from remote devices is often to use the vendor supplied software that came with the device. Kinda seems like a no-brainer. However, there are a number of potential issues that may surface as a system grows. Vendor supplied software is typically and understandably device centric, it is often intended to be used as a desktop application, and seldom scales to a robust near real time enterprise solution. There are exceptions to each of these issues, but when multiple device types and multiple vendors are in the mix it gets really complicated. It is also time consuming from staff management perspective and often introduces additional latency in retrieving the data. A better approach is to use a device and file type agnostic OSS tool that presents a single user interface for all remote devices and manages a scalable number of simultaneous network and dial-up connections for file based and near real time Modbus or DNP3 data. This approach is being used in production systems and has shown significant productivity gains.

Analyzing the data: As mentioned in the context of getting the data, all hardware vendors supply some form of data visualization and analysis, with similar complexities and limitations that don't scale to enterprise solutions. Enterprise class OSS tools are available and deployed in production systems at some of the largest US utilities. A major advantage to the OSS solution is that additions or modifications to the analytics can be made at any time and are not restricted by any vendor's business cycle or driver.

Storing the data and analytic results: Some vendor's software simply retrieves event records and makes them available to open in a companion desktop software tool for on-demand analysis. Others analyze the retrieved data and build a database, but the database is locked up in a proprietary format and is only accessible using the specific vendor tools. The OSS solution is generalized to include waveform data from events, and trending data from Modbus enabled or power quality devices. To optimize this comprehensive solution, a relational database is used to store event data and analysis results, and a NoSQL repository is used for trending data. The data layer that manages these data stores includes robust interface options to facilitate external data use.

Proactive notification: Some devices and vendor software solutions provide automatic notifications using email or other messaging services, but a vendor specific solution is limited. The services are only available for the specific vendor’s devices and remote devices may have restricted access to notification services. The OSS solution provides automatic notification for results obtained by analyzing data from all related devices regardless of the manufacturer and sends notifications from a central location.

Visualizing information: The traditional desktop or vendor specific analysis tools described above also provide some level of visualization, but typically carry along some of the inherent proprietary issues. The OSS solution provides an enterprise wide view of all reporting devices from any manufacturer through a web-based dashboard user interface and includes aggregated information that is not available at a device centric level. An example of one dashboard view is shown in Figure 2.

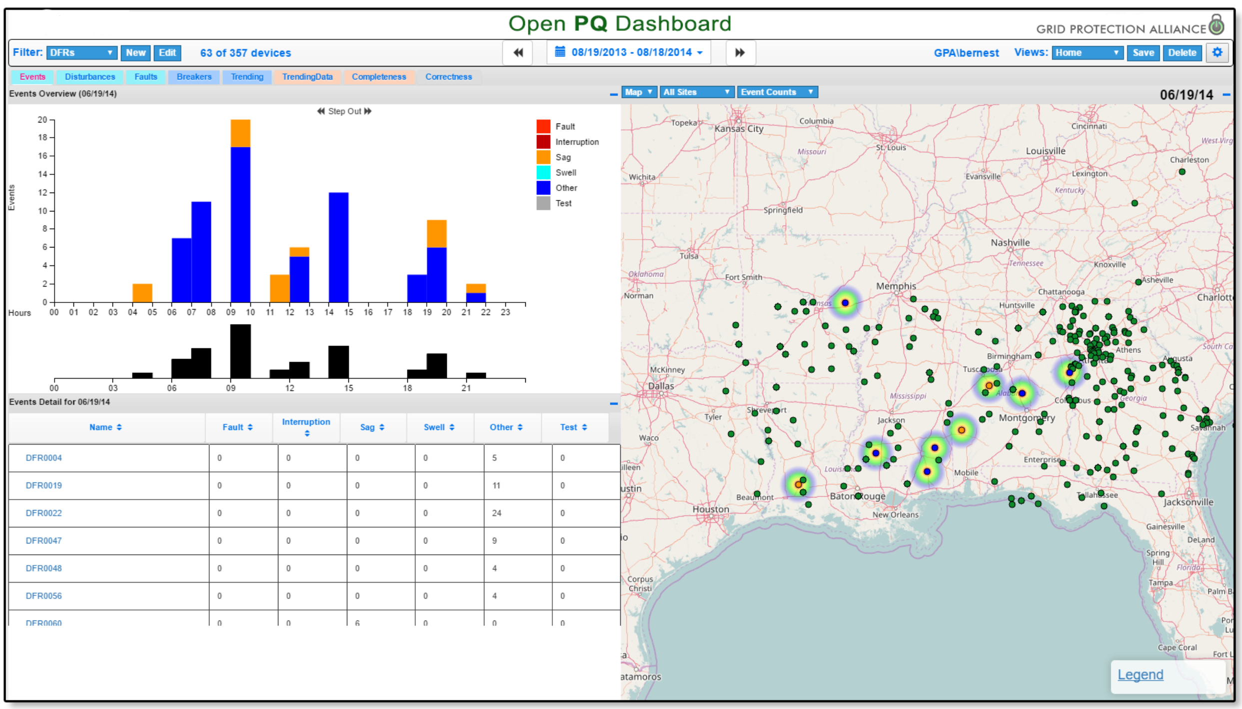

Figure 2. Example Dashboard View

Strategic Data Sharing: The final building block discussed here is the ability to automatically and securely share specific data with a targeted recipient, i.e. your neighboring utility. This data sharing function is built into the OSS data layer described in the previous blocks. The management tools allow the system administrator to identify data elements in their system to be shared with a specific data sharing partner, and to also accept selected data elements from that partner. A high-level architecture diagram of an example implementation of the strategic data sharing is included in Figure 3.

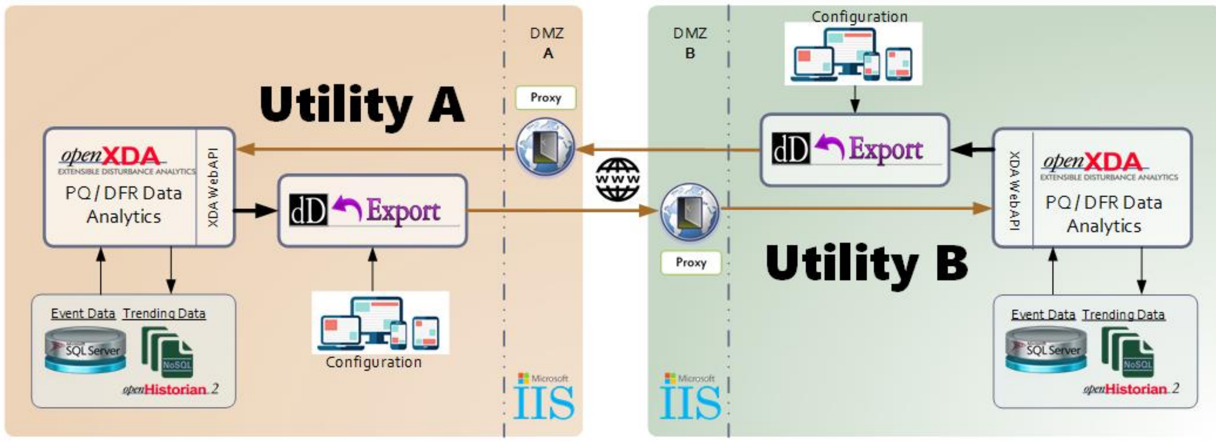

Figure 3. High-Level Architecture, Data Sharing

IMPROVED FAULT ANALYSIS

The end-goal for strategic data sharing is to facilitate more accurate fault distance calculations so that fault location results can be improved. Through the strategic data sharing described here, neighboring utilities that have monitored assets of mutual interest can now selectively and automatically share the data so that each utility's system can automatically produce double-ended fault distance calculations. Overall fault analysis is improved by having more comprehensive data and the distance calculations produced are significantly more accurate than previously available single-ended calculations where data from "the other end of the line" was not available.

OSS References

Each of the building blocks and functions described in this paper are available through OSS projects managed on Github. References to the projects are listed below.

Getting the data: **openMIC**
https://github.com/GridProtectionAlliance/openMIC

Analyzing, notifying, storing, and sharing the data: **openXDA**
https://github.com/GridProtectionAlliance/openXDA

Visualizing the data: **PQDashboard**
https://github.com/GridProtectionAlliance/PQDashboard