Continuous Automated Analysis of Protection Scheme Communications Leads to Improved Reliability and Performance

Brian Waldron and Bryan Fazzari
Schweitzer Engineering Laboratories, Inc.

This paper was presented at the 71st Annual Conference for Protective Relay Engineers and can be accessed at: https://doi.org/10.1109/CPRE.2018.8349785.

Presented at the
Grid of the Future Symposium
Reston, Virginia
October 28–31, 2018

Previously presented at the
71st Annual Conference for Protective Relay Engineers, March 2018

Originally presented at the
Power and Energy Automation Conference, March 2018

Continuous Automated Analysis of Protection Scheme Communications Leads to Improved Reliability and Performance

Brian Waldron and Bryan Fazzari, *Schweitzer Engineering Laboratories, Inc.*

***Abstract*—As protection intelligent electronic devices (IEDs) have evolved, their communications and reporting capabilities have become more advanced. Protective relays can store, with high-accuracy time stamps, not only records of protection elements pertinent to the operation of the relay itself but also the arrival and departure of high-speed incoming and outgoing data that are used to coordinate other devices' protection algorithms. All the data can be sifted through and analyzed to locate maintenance indicators and to correct undesirable behaviors before a larger problem is created within the system.**

Some communications testing requirements are already outlined in NERC PRC-005-02 – Protection System Maintenance. This paper investigates how, by using the communications and reporting capabilities of these modern IEDs, a continuously running monitoring system can quickly identify and report signal transmission timing or delivery degradation in a protection system using high-speed peer-to-peer signals. This monitoring system functions regardless of protection scheme protocol selection or network design variations, and it provides immediately actionable data by delivering reports that indicate the exact contact or internal bit in a specific relay or set of relays involved in the problem.

I. Introduction

Device communications in the electric power industry have changed substantially since first making an appearance in remote terminal units (RTUs) in the 1970s. Communications between intelligent electronic devices (IEDs) started out as direct output-to-input contacts connected by copper wires, each transferring a single digital bit of information. Once processing power began to increase in substation devices, system operators began integrating digital communications protocols communicated over serial cables, primarily via EIA-232 or EIA-485 cable types.

Serial-based communications significantly increased the amount of information that could be passed between two devices. It offered data transfer at a known cyclic rate, but it was difficult to share the same information between multiple devices because serial cables primarily supported device-to-device communications. In the 1990s, Ethernet technology began to make a consistent appearance. With Ethernet, information could be shared between many devices with only a single cable connected to each IED. It also allowed significantly more information to be transferred than serial-based communications allowed. However, a fundamental difference between Ethernet and serial communications is that Ethernet-based protocols are primarily designed to prioritize the receipt of delivery rather than prioritize a consistent delivery time to the end device. This is a tradeoff that system integrators have to consider when choosing between serial and Ethernet communications.

In the past, the power system relied (and, in many cases, still does rely) on the consistent data transfer time (also referred to as determinism) that serial communications provided. This was to improve system protection and the reliability of communications-assisted protection schemes between IEDs. Monitoring communications between devices was as simple as watching for the Boolean indicators of online versus offline status provided by each device. Ethernet communications can be configured to provide consistent fast transmission times to participate in these communications-assisted protection schemes. Unlike serial, Ethernet communications monitoring can be more than observing a single Boolean online or offline value. Ethernet communications transfer times can and should be monitored between devices to ensure that information transfers between devices at a consistent time interval just like it does with serial communications.

However, many power system engineers are cautious about using Ethernet in their systems. Part of the problem is that little information about the communications infrastructure is integrated into their power system monitoring and recording systems. By monitoring and analyzing the times that application data take to travel between IEDs, power system operators can gain better confidence in their communications infrastructure.

This paper discusses how to use standard application data to detect a loss or degradation of high-reliability communications. In addition, it discusses the correlation of these alarms with Ethernet information. This paper only lightly touches on the monitoring information available from Ethernet networks.

II. Ethernet Monitoring Capabilities

While Ethernet requires more monitoring than serial communications, it also offers a wider range of statistics about its operation and performance. These statistics, coupled with the significant increase of available device processing power since the turn of the century, have made external monitoring of the communications system a more available and standard practice.

Ethernet technology provides information at several layers of the Open System Interconnection (OSI) model. Information about the first four layers (physical, data, network, and transport) are most likely be useful for applications. Monitoring this information can provide varying levels of data for analysis. The following are several examples of communications issues and how to identify them with standard information from an Ethernet communications network:

- *A device is no longer connected to the network*. This is determined by confirming the device link at the switch. The issue could be that the cable is unplugged from the switch.
- *A device is connected to the network but does not establish the expected communications link*. This is determined by confirming the links of both devices to the network and confirming that no network (Layer 3) or transport (Layer 4) connection has been established. The issue could be that the two devices have Internet Protocol (IP) addresses on separate subnets.
- *A device is connected to the network and able to establish a communications link with other devices, but the application does not receive the expected information*. This is determined by confirming the links of the devices to the network and confirming that a network (Layer 3) or transport (Layer 4) connection is established but the application does not receive the expected information. This situation implies a specific application problem or misconfiguration rather than a network problem. The issue could be that the devices are connected to the network and have the correct IP addresses, but the protocol for the application has been configured to use different Transmission Control Protocol (TCP) ports on each device.

These examples show an improvement over the information that serial (or hardwired) communication provides. Because communications data are not available in these types of applications, it is difficult to tell if the IED is not working, if there is a problem with the cable, or if there is a hardware compatibility or interface issue. Ethernet information about link status, packets sent and received, bandwidth, and TCP/User Datagram Protocol (UDP) connections allow for some troubleshooting to be automated and provide the operator more clarity for resolving detected issues. More than detecting communications issues, this same information can play a role in providing security and operator confidence in communications-assisted protection schemes. Communications allow for significant advances in protection schemes, but this also requires that communications provide consistent, reliable, and fast information transfer. Monitoring the available Ethernet information confirms that the communications channel retains this high level of reliability at all times. If, for some reason, it does not maintain a high level of reliability, then alarms are quickly generated so that the correct personnel can rectify the issue. In addition to initiating the alarm, this information can help guide personnel to the root cause of the problem, reducing the resolution time.

III. Monitoring During Commissioning

Commissioning a system that has communications-assisted protection schemes requires that substantial attention be paid to the time it takes IEDs to exchange information. Quite often, this is done through manual analysis. For example, commissioning personnel generate an event via test sets or switches and then the IEDs either provide Sequential Events Recorder (SER) reports or record event information in Common Format for Transient Data Exchange (COMTRADE) files. After the information is recorded, personnel go to individual devices, download the appropriate files, open the records on a monitor, line up the records to see the time stamps, and perform the calculations verifying that the communications network delivered the information in the time frame required for the application to perform as needed.

This type of test is often performed multiple times to establish a baseline. None of these actions are very complex; however, each step is time-consuming. This process does not easily allow for the creation of a historical record during testing unless an additional manual step is added to the process. Once the application timing requirements have been met, personnel often do not look at the application performance analysis again until a fault occurs and something does not work as intended. In that case, they follow the same process as before, downloading the files and manually examining them to determine what did not operate as expected. Creating an automated solution for monitoring the transfer time between IEDs provides several benefits for protection applications:

- A reduction in the time it takes to analyze communications-assisted protection scheme performance.
- Historical performance records.
- Continuous performance monitoring.

These benefits allow for considerable commissioning time savings and provide system operators with continued assurance after the commissioning is complete. Many systems today are commissioned by one group of personnel and maintained by a separate group of personnel. Automated record keeping provides additional assurance to the personnel who maintain and operate the system by allowing them to see the past performance of the system during commissioning and the continued performance of the system every time devices exchange information.

IV. Power System Applications Benefit From Communications Monitoring

Monitoring the health of communications networks with information such as link statuses, bandwidth, and counts of packets sent and received can determine if the communications links are working correctly over time, but these statistics do not show if the communications network continues to provide the performance that protection applications rely on. To determine this, operators must compare the time stamps of data sent and received between IEDs. This paper focuses on the aggregation of these time stamps into a single location, automating the process of calculating the time-stamp deltas, creating historical records, monitoring the deltas and finding anomalies in the transfer times, and creating easy-to-read reports for operators. The following subsections discuss application examples where creating a monitoring system as described benefits operators and system owners.

A. 87L Scheme Applications

Line current differential (87L) schemes use two or more protective relays that communicate over one or more high-speed communications paths. Because communications are part of this scheme type, an 87L scheme could be considered to be in the same category as other communications-assisted schemes such as transfer trip or trip blocking. But there is an important difference: these other schemes use the communications system to send one or more binary values from one relay to another. The receiving device acts on these binary values as soon as it receives them, and the values are often used directly in breaker trip or supervision logic and are easily simulated (for instance, via a pushbutton on the front of a relay).

An 87L scheme does not use binary trip or block signals. It must align high-speed remote current transformer (CT) measurements received over the communications channel at a continuous and consistent interval with its own local current measurements and determine whether the breaker should be operated. Because of this more complicated data exchange, 87L schemes can be more sensitive to communications disturbances and interruptions.

Testing these schemes can be challenging. CT measurements are difficult to simulate because of the need for a realistic and accurate relationship between the magnitudes, phases, and angles from each line terminal under test. Additionally, modern relays have countermeasures such as local disturbance detection in place to improve the security of the 87L scheme and prevent it from acting on bad or corrupted data [1]. These security measures make 87L schemes more challenging to test with simulation data.

Once an 87L scheme is commissioned and placed in service, it can be difficult to know if communications channel performance degradation is occurring over time. The relay Ethernet network interface is dedicated to the 87L scheme, making periodic channel testing difficult. Modern relays can perform continuous channel-monitoring functions, which can be used to generate both performance statistics and alarms when something goes wrong. Any maintenance personnel responsible for ensuring the health of the 87L scheme should analyze this information frequently. This is especially true if the scheme is communicating over an Ethernet network. Networks have a propensity to grow and accumulate new participants. Network events such as settings changes, new network paths, link status changes, and Rapid Spanning Tree Protocol (RSTP) topology changes can affect scheme performance. Because the evolution of a utility's network is continuous, the monitoring of all metrics relating to the health of an 87L communications channel should also be continuous [2]. Modern data concentrators and substation automation controllers can perform this task, relieving the system operators of the need to collect and collate large quantities of numerical data from multiple relays. These devices are typically already integrated into supervisory control and data acquisition (SCADA) systems, making them ideally suited to collect, format, and make sense of all the 87L channel status information available not only from the protective relays but also from the network switches that make the channel operate.

1) Important Channel Performance Metrics

Table I, as well as the following subsections, describes the most important channel performance metrics for an Ethernet-based 87L scheme [3].

TABLE I
87L SCHEME COMMUNICATIONS CHANNEL PERFORMANCE METRICS

Metric	Origin
Lost packet count	Relay and switch
Round-trip packet delay	Relay
Channel asymmetry	Relay
Channel failover	Relay
Topology changes	Switch
Link status changes	Switch

a) Lost Packet Count

Modern protective relays can employ error-checking methods to determine whether packet corruption has occurred. These relays can also determine if a packet from the remote device failed to arrive within the allotted time. This information drives lost packet count metrics. It is beneficial to collect these counts periodically and create a trend with them. This can reveal a slowly developing problem with the channel resulting in periodic corruption or delivery problems. It is also useful to correlate unusually high lost packet counts over a short period with logged events from network switches, if possible. The losses could be due to a power supply problem or a broken link in the system, or they could correspond to scheduled network maintenance or settings updates that may require further inspection. Automation controllers or front-end processors installed in the substation or at the SCADA system location can collect this information from the switches by using a protocol such as Simple Network Management Protocol (SNMP).

b) Round-Trip Packet Delay

The time a packet takes to travel from Relay A to Relay B and then back to Relay A is the round-trip packet delay. This value should remain as close to a single constant value as possible. A step change in the round-trip packet delay could indicate a network topology change, while an inconsistent variability could indicate a quality of service (QoS) settings issue, a failing component, or some other problem with one or more network devices.

c) Channel Asymmetry

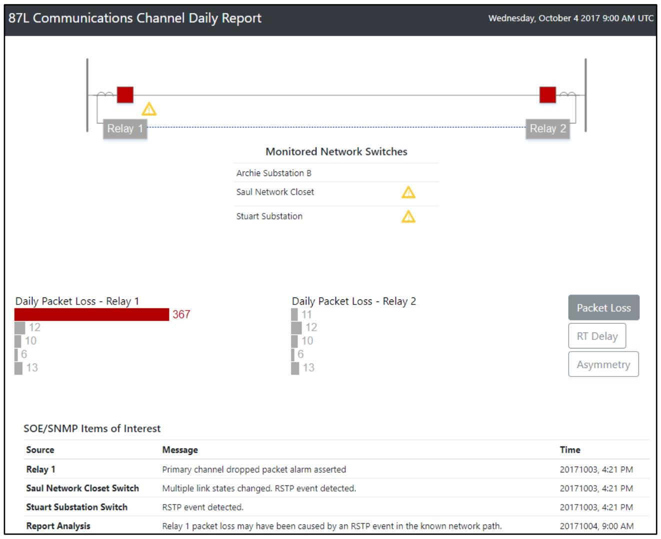
All communications channels take a finite amount of time to transport data from one location to another. This time is often referred to as latency, and it is often described as channel delay in 87L schemes. Some 87L schemes use an external high-accuracy time source to measure the channel delay. Others measure it by calculating the round-trip packet delay divided by two, which is an approximation of the one-way packet delay as long as the delay from local to remote and from remote to local are the same (i.e., symmetrical). If these delays become too different, then the relays participating in the scheme are unable to align CT measurement data properly.

d) Switch Status Information

Ethernet switches can provide a wealth of information about their statuses and configurations, as well as about events that have occurred on the network. Because networks grow and change over time, it is important to use the data that Ethernet switches can provide when monitoring 87L channel performances. An RSTP topology change resulting from a broken cable connection somewhere in the network can result in a new communications path using different hop counts and network devices. A settings update can change the QoS rules being applied to 87L scheme traffic. A change in the link status of one or more ports can indicate a failed device or a new device being added to the network. A spike in packet failures or a power supply alarm can indicate a potential problem in the future for the 87L channel. All of this information can be collected and placed into an 87L channel performance report created by a substation automation controller. Combining switch information with relay information makes it easy to identify network problems early and react to them before they cause more serious problems.

2) Automated Data Collection, Formatting, and Reporting

Today’s substation automation equipment is capable of communicating with a wide array of devices and of using numerous protocols to analyze, format, and deliver the data in a manner that is most beneficial to those responsible for maintaining the system. This section has focused on 87L schemes using an Ethernet communications infrastructure. To know the communications channel performance and whether a network change has negatively impacted these schemes, a large quantity of data needs to be collected from all the protective relays and network equipment involved. Fig. 1 provides an example of a formatted daily 87L channel performance report that can be provided by a capable substation automation controller.



87L Communications Channel Daily Report

Wednesday, October 4 2017 9:00 AM UTC

Monitored Network Switches

Archie Substation B

Saul Network Closet ⚠

Stuart Substation ⚠

Daily Packet Loss - Relay 1

Daily Packet Loss - Relay 2

SOE/SNMP Items of Interest

Source	Message	Time
Relay 1	Primary channel dropped packet alarm asserted	20171003, 4:21 PM
Saul Network Closet Switch	Multiple link states changed. RSTP event detected.	20171003, 4:21 PM
Stuart Substation Switch	RSTP event detected.	20171003, 4:21 PM
Report Analysis	Relay 1 packet loss may have been caused by an RSTP event in the known network path.	20171004, 9:00 AM

Fig. 1. Example 87L communications channel daily report (packet losses were forced to be non-zero for illustrative purposes)

These types of reports should contain actionable information and should be formatted to draw attention to the most critical items of interest. This type of report could be emailed to a user group or placed in a secure network location on a regular basis. Reports like this are possible because substation automation controllers can automatically collect and analyze information from all of the necessary devices, which obviates the need for manual data retrieval and organization.

B. Fast Bus Tripping Schemes and Restoration Schemes

Fast bus tripping schemes use communications to improve the reliability and protection of the system. These schemes are also sometimes used for bus protection since coordination is covered by a main feeder relay supplying the bus and by subsequent feeder relays communicating their overcurrent statuses to the main feeder relay. Communications in this scheme allow a main relay that sees a feeder fault to not trip immediately if feeder relays also see the fault, as shown in Fig. 2. This allows the feeder relay to trip the breaker and clear the fault on the system, allowing all the other feeders on the bus to remain energized [4]. At the least, the overcurrent (50/51) element status is transferred between IEDs, usually with additional information. In modern systems, these data are typically exchanged via a peer-to-peer high-speed digital communications protocol such as Generic Object-Oriented Substation Event (GOOSE). The information exchanged between relays in these schemes is commonly recorded with high-accuracy time stamps in Sequence of Events (SOE) logs. Since this information is recorded with high-accuracy time stamps, it is easy to collect and calculate the time delta between the relays and establish the time it took to transfer the information. In these protection schemes, additional protection pickup elements may also be exchanged between relays. These pickups may cause a breaker to trip on the system, but not necessarily every time. Since this information occurs more frequently than when a trip may actually occur, it provides more opportunities for the monitoring logic to calculate the average communications time between IEDs prior to a critical action being taken. This allows for normal data exchange to frequently show that the system is maintaining its performance expectations without additional manual testing.

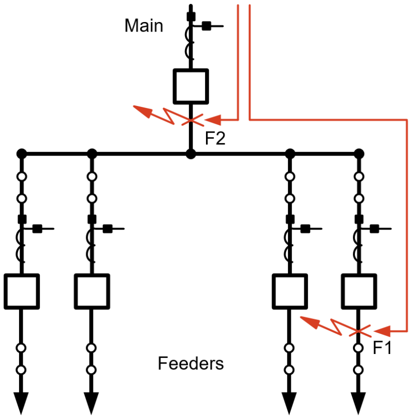


Fig. 2. Fast bus tripping scheme

Related to fast bus tripping schemes are the restoration schemes that come into play after a fault is isolated. In the past, restoration schemes were often performed manually. However, protective relaying advances and more capable logic controllers have automated many restoration schemes [5]. These automated restoration schemes rely heavily on communications to ensure that the correct and appropriate actions are taken across a variety of IEDs in the system (e.g., to avoid closing into an existing fault). This typically involves collecting information from reclosers that are not necessarily located in the substation. Reclosers located outside the substation are often connected over radio communications networks. While radios provide the great advantage of low infrastructure costs compared with other communications equipment, they can be susceptible to interference. Restoration schemes vary significantly in their speed requirements (from cycles to seconds), so these communications networks can have varying performance times.

Often, operators do not know about communications problems until there is an application-based issue. After manually investigating, they can determine that there was a communications issue in the radio network that would not have become so serious if they had detected and resolved the issue earlier. By monitoring the data exchange performance over these recloser radio networks, operators can consistently see the communications performance between IEDs. This monitoring helps identify restoration scheme performance and assures operators that restoration schemes will operate consistently in communications networks susceptible to interference, which is where restoration schemes are often needed. The output of this monitoring should be a simple report that shows the transfer times between IEDs in a format that is it easy to understand regardless of one's background in the power industry.

C. SCADA and HMI Applications

In SCADA and human-machine interface (HMI) applications, the time it takes to transfer data from the IED to a data concentrator, to the HMI, or to the SCADA system is typically in the single digits of seconds. In these applications, the accuracy of information time transfer between the IEDs is not as critical as it is in protection applications. However, most operators expect a reasonable data transfer rate that provides up-to-date and accurate data that they can view graphically or in log files. Sometimes during commissioning, while updating part of an in-service system, or over the course of time, operators may observe that information seems to update slower than it used to. In most systems today, this is a difficult claim to verify because of how data are time-stamped.

Data are typically time-stamped at the originating IED, and the data move through data collection without anything tracking the time it takes the data to traverse the system. The data may pass through multiple devices and may be translated between protocols. Fig. 3 shows an example of how data may flow through IEDs. To calculate the time difference between when information is initially processed from a relay and the time it is shown on a screen is not as easy to manually investigate as comparing SOE records from two relays. Modern substation HMI products typically have integrated logic engines. These logic engines can record the time that the HMI receives the data and calculate the difference between that time stamp and the one from when the IED sent the data. The logic engine can then use this difference to create a historical average transmission time between IEDs in the system and provide the operator this information. This monitoring provides the operators with verification that their SCADA system is maintaining consistent communications performance during its in-service life. Since this data collection and processing occurs at the HMI or data concentrator, there is little effort to automate this type of logic that provides continuous and historical information.

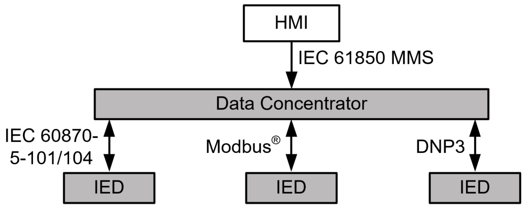


Fig. 3. Common HMI communications flow

V. COLLECTING TIME-STAMP INFORMATION

Several components compose the delta between two time stamps from two IEDs. These components are shown in Fig. 4.

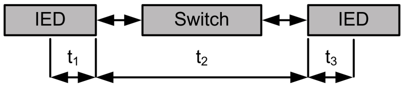


Fig. 4. Determining the delta between two time stamps

The first portion of the data exchange consists of the time that the sending device takes to put the information onto the communications network after its logic engine has time-stamped the data (t_1). The second portion is the time it takes for the information to travel through the communications network and be delivered to the receiving device (t_2). Finally, there is the time it takes for the receiving device to process the information (t_3). The portion that is most interesting to system operators is the time that the information spends traversing the communications network. However, this time is incredibly difficult to calculate without special logic in the relay or managed switch firmware. The application time-stamp records (which are often accurate to 1 ms) are easy to access and accumulate. The time that sending and receiving devices spend processing is likely to be consistent and cyclical. This allows operators to infer that any detectable difference between time deltas is a result of a changing time through the communications network. Identifying changing communications transmission times is the primary objective, so using time stamps that include a small amount of consistent processing time does not significantly affect the value of using high-accuracy IED time-stamped information to calculate changing transmission times. Note that it is important to use high-accuracy time-stamped data. Many relay products require that information that needs to be tracked with 1 ms accuracy be configured via a list in the relay settings. Otherwise, the data are likely to be accurate to hundreds or thousands of milliseconds.

Calculating the time-stamp differences in IEDs requires that high-accuracy information from the IEDs be acquired by a central processing device. Often, this is a data concentrator or

HMI with an IEC 61131 logic engine that can store and analyze these calculations and format the results. Protective relays typically offer some user-configurable logic. However, the user-configured logic engines are typically designed to coordinate tripping, blocking, and timing mechanisms involving protection elements. The protective relay logic engine does not usually offer the flexibility and recording capabilities that the time-stamp difference calculation logic requires. Protective relays typically offer their SER information through a report or in near real time through event data in a protocol. These mechanisms collect the information in a central location. The exact protocol used generally will not affect the information or results of this process. Multiple protocols may be used to transfer the high-accuracy time stamps from the protective relays to a data concentrator. Using different protocols likely affects the implementation of data parsing and of formatting it into a normalized format but does not affect the work of the delta calculations and analyzing algorithms.

This paper does not focus on the protocol used to collect this information. However, several protocols are better-suited for transferring time information than others. Typically, the protocols used to send information between the protective relays will not be used to collect this information for analysis. Peer-to-peer protocols focus on transfer speed between devices. As a part of that effort, less data to process results in faster operations. Therefore, time stamps are not typically included in that data transmission. A common example of this is IEC 61850 GOOSE. The data set that carries application data only contains the present status of the information, not time-stamp information. Protocols such as DNP3, manufacturing message specification (MMS), and IEC 60870-5-101 and IEC 60870-5-104 are typically the best for high-accuracy time-stamp information transfer as they all support data transmission with time stamps. These protocols make up the majority of the SCADA-related protocols used around the world.

One protocol missing from this list is Modbus. While Modbus does not offer a mechanism that allows a high-accuracy time stamp to accompany data, some Modbus server devices can send the time-stamp information as register values. In this way, the data concentrator can reconstruct the time stamps. Modbus is also sometimes used as a control protocol for set points where the timing is expected to be in the 50 to 500 ms range rather than the sub-10 ms range typically expected in protection applications. In this type of application, it is more difficult to collect the high-accuracy time stamps. However, some valuable information can still be extracted in implementations with Modbus like this. The time that it takes for a control signal to be sent and the time it takes for the feedback signal to change can still provide an indicator of network performance. However, there are additional timing factors included in this type of situation (poll periods and multidrop communications delays are prime examples). The same core algorithm can be used to measure this performance. The source of the data comes from the device issuing the control signal and reading the feedback signal, which is likely the local HMI or data concentrator.

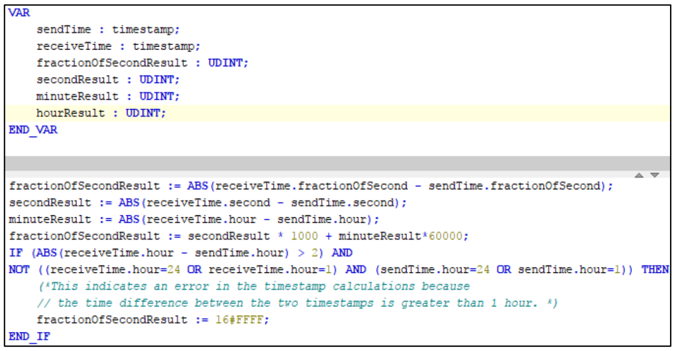
Because of how information systems in substations are typically designed, the configuration of this application is not likely to need additional system hardware. It is also very likely that most or all of the high-accuracy time information needed for these calculations is already collected for SCADA or HMI systems. This is largely why the data concentrator is an ideal location for comparing time stamps and keeping historical records: a large portion of the data collection work is already done for SCADA, minimizing the work required to implement this type of solution.

VI. Automating Time Difference Calculations

All the protocols discussed for transferring time stamps do transfer the time, but not every protocol transfers time in the same manner. The protocols transfer time stamps ranging from 12-bit to 56-bit values [6] [7] [8] [9]. When the delta calculation is performed, the time stamp is likely to only be accurate to the millisecond. Any time stamp that includes fractions of seconds with more detail than milliseconds is truncated for two primary reasons. First, it is unlikely the time stamp is truly accurate to anything more than the millisecond, and second, most protective relay logic engines operate in milliseconds.

There are two approaches to calculating the delta between these time stamps; which approach is easier to implement depends on the logic platform being used. The first method is to normalize the data into year, month, day, hour, minute, second, and fraction of second formats and then subtract the fraction of second values. The data transmission time is expected to be small, and it is likely that the other portions of the time stamps are identical; however, it is still important to verify this. If the other portions of the time stamps are not identical, then the final time delta calculation needs to be adjusted based on the time difference. The second method is to normalize the data into Unix time or epoch time (the time since January 1, 1970, not counting leap seconds) in milliseconds as a 64-bit integer. The second approach makes the time stamp delta simpler to calculate. However, depending on the implementation platform, it may be more work to normalize the data as milliseconds since the epoch.

The algorithm in Fig. 5 shows the time difference information calculated using an unsigned integer rather than a float value. Floats are an approximation of a number, with variable accuracy dependent on the number's magnitude. While quantities of six or fewer significant digits do not lose precision, depending on the method selected to normalize the data, precision loss may be a concern [10]. An easy way to eliminate this potential concern is to use unsigned integer types for the delta calculation.



```
VAR
    sendTime : timestamp;
    receiveTime : timestamp;
    fractionOfSecondResult : UDINT;
    secondResult : UDINT;
    minuteResult : UDINT;
    hourResult : UDINT;
END_VAR

fractionOfSecondResult := ABS(receiveTime.fractionOfSecond - sendTime.fractionOfSecond);
secondResult := ABS(receiveTime.second - sendTime.second);
minuteResult := ABS(receiveTime.hour - sendTime.hour);
fractionOfSecondResult := secondResult * 1000 + minuteResult*60000;
IF (ABS(receiveTime.hour - sendTime.hour) > 2) AND
NOT ((receiveTime.hour=24 OR receiveTime.hour=1) AND (sendTime.hour=24 OR sendTime.hour=1)) THEN
    (*This indicates an error in the timestamp calculations because
    // the time difference between the two timestamps is greater than 1 hour. *)
    fractionOfSecondResult := 16#FFFF;
END_IF
```

Fig. 5. Time delta calculation

Consider an example in an ideal situation where two devices are exchanging protection-related information via GOOSE messages. Each device shares the following information with the other IED in a data set: the 52A breaker status, the overcurrent (50/51) element status, the hot voltage value, and an auxiliary test bit. The auxiliary test bit in the data set (not associated with the test or simulation bit that is part of the GOOSE protocol) allows the communications channel performance to be tested independently of the protection information exchange. The test bit can be automated to change once an hour (or any time interval) to provide consistent performance data, which could highlight any potential degradation before the protection scheme needs to act. As shown in Fig. 6, each device provides high-accuracy time information to a data concentrator using the DNP3 protocol.

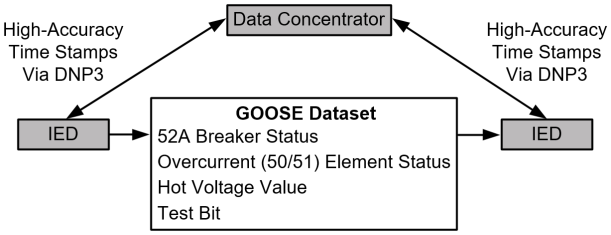
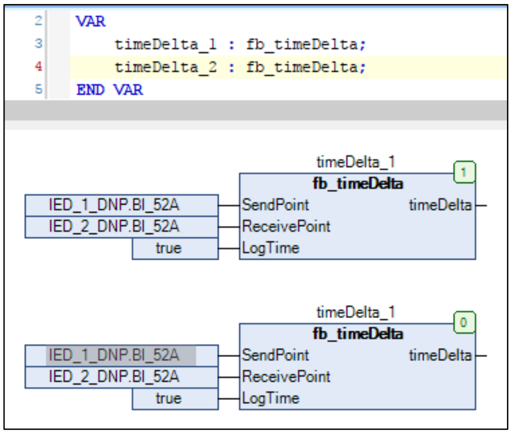


Fig. 6. Example system with two devices exchanging time information via DNP3

Once the data set and time stamp have been collected, the logic shown in Fig. 7 detects that the time stamps from both units have been updated by normalizing the data into an unsigned integer representing time since Unix epoch accurate to the millisecond. The logic then compares the current time stamp value against the last known value when a delta time was calculated. After detecting that both time stamps were updated, the logic confirms that the present data value from each unit matches and calculates the difference between the two IED time stamps. The integer result is the time difference in milliseconds. Because most applications are expected to exchange data in well under 1 second, the most appropriate time scale is milliseconds. Notice in this example that the logic for the calculation is written in IEC 61131-structured text, but the configuration is performed with a graphical approach using an IEC 61131 continuous function chart (CFC). Logic engines using IEC 61131 allow operators to implement this algorithm in several ways, which enables all operators to configure the logic regardless of their IEC 61131 format preferences [11].

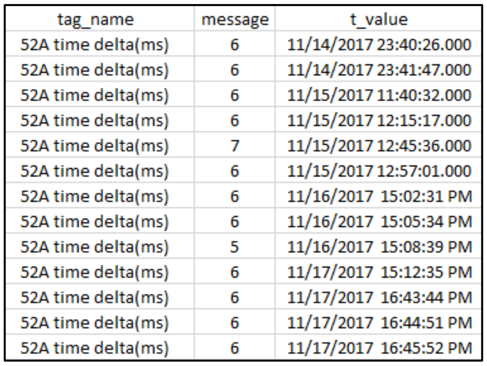
Another important factor here is that the logic that calculates the time delta is a generic algorithm that does not change between data points. Since it is generic, it is written as a function block that allows multiple function blocks to be declared, making it easy to add additional data points to be monitored.



```
VAR
    timeDelta_1 : fb_timeDelta;
    timeDelta_2 : fb_timeDelta;
END VAR
```

Fig. 7. Time delta calculation in CFC

This delta time is then stored in a log file. Most data concentrators have a historical log feature that many operators configure to keep track of the IED SER reports. The output of this logic can be stored in that same log file. These log files are typically available in a .csv file or in another spreadsheet-compatible format. Spreadsheet programs offer many filtering, sorting, and analyzing tools that are beneficial when viewing these kinds of data. Alternatively, some logic engines can create files themselves, which allows operators to generate specific reports that are custom-tailored to their preferences. Fig. 8 shows an example of a historical log file.



tag_name	message	t_value
52A time delta(ms)	6	11/14/2017 23:40:26.000
52A time delta(ms)	6	11/14/2017 23:41:47.000
52A time delta(ms)	6	11/15/2017 11:40:32.000
52A time delta(ms)	6	11/15/2017 12:15:17.000
52A time delta(ms)	7	11/15/2017 12:45:36.000
52A time delta(ms)	6	11/15/2017 12:57:01.000
52A time delta(ms)	6	11/16/2017 15:02:31 PM
52A time delta(ms)	6	11/16/2017 15:05:34 PM
52A time delta(ms)	5	11/16/2017 15:08:39 PM
52A time delta(ms)	6	11/17/2017 15:12:35 PM
52A time delta(ms)	6	11/17/2017 16:43:44 PM
52A time delta(ms)	6	11/17/2017 16:44:51 PM
52A time delta(ms)	6	11/17/2017 16:45:52 PM

Fig. 8. Example historical log file

Storing the information in a historical log file of previous transmission times allows for simple but effective analysis to identify, for example, any positive or negative trends in timing over the last week, month, or year. It can also easily calculate the minimum, maximum, and average transmission times and identify outlier measurements that warrant further investigation. From that analysis, it can also easily calculate a count of any transmission times that are within close proximity to the maximum, which could help reveal a possible issue that only periodically shows up under certain system conditions.

In this case, the historical log file is created as a .csv file, which allows the report to be opened in a spreadsheet program and enables the operator to easily generate graphs of the data, such as the example in Fig. 9.

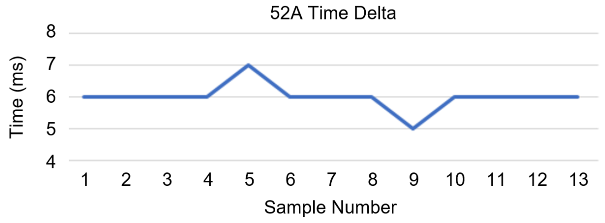


Fig. 9. Example graph of transmission time data

It is likely that the high-accuracy time stamps needed to use this time difference calculation algorithm are acquired via a protocol such as DNP3. However, some IEDs may not offer high-accuracy time stamps for some types of information that operators wish to monitor. High-accuracy time stamps are often stored in specific SOE reports or COMTRADE files. Another format that IEDs sometimes use is an ASCII-based report where a TCP connection is typically (but not always) created with a terminal program, and the operator enters an ASCII command to receive current and historical information from the IED. A capable data concentrator can access these reports, parse the time stamps out of the files and outputs, and normalize them into the same format that the protocol time stamps are stored in. While acquiring this information takes more logic, it is still beneficial for performance monitoring purposes.

VII. Good, Better, and Best Solutions

The example in the previous section discusses how to calculate the time delta for when data are sent from one IED to another. In the example, it is suggested that after the time is calculated, the output of the logic be stored in a log file. However, there are a variety of options for what to do with this information: it could be stored, compiled into a report, or formatted into graphical results and made easily filterable and interactive. These are all very attractive solutions. However, this paper primarily focuses on the value and benefits of collecting and calculating this information, so discussing the many possible data presentation options is beyond the scope of the paper. While a sophisticated programming solution can be implemented to automatically create elegant and fully analyzed reports, there are several implementation options that provide value for varying levels of effort. These levels are summarized in this section as “good,” “better,” and “best.” They represent the amount of implementation effort in increasing order, but they represent functionality and value in increasing order as well.

A. Good Solution

This solution is to collect the information, calculate the time delta, and store the information in an SOE or historical log file in the data concentrator. Most data concentrators offer a logic engine of varying capabilities, but subtracting integer values is within the capabilities of most of them. The result of this calculation is stored in the historical log file along with other station SOE data. This configuration is often already done as a part of normal SCADA or HMI data mapping. Most operators configuring SCADA mapping are comfortable with the logic necessary to implement this functionality. While the log file does not perform any automatic analysis, it is easy and less time-consuming to perform basic manual analysis in a spreadsheet program. This is a considerable improvement over manually comparing time stamps from individual relays or finding time stamps in a single log file and manually subtracting. Separate individuals often implement integration and protection schemes, and this solution does not require the integration implementers to know much about the timing or application requirements of the protection-related applications. However, it creates a report that is easily accessible for those responsible for the protection and coordination schemes.

B. Better Solution

This solution builds on the functionality implemented in the “good” solution. The data concentrator can create logic alarms by comparing the time delta against a known time window that the delta should not exceed. This comparison is used to drive an alarm that can appear in the local HMI or SCADA system or generate an email and/or text notification. IEC 61131 logic can also be used to detect an increasing trend in the value over time, even if it has not exceeded the alarm value. The logic could also calculate the minimum, maximum, and average transfer times, as well as determine if outliers are occurring at similar times of day (indicating possible resource or bandwidth competition with a periodic or network load profile event). All of these calculations are straightforward functionalities that are often performed on other data collected and monitored in the system. While these functionalities may take more time to implement and require some knowledge of the application, they reduce the amount of manual analysis required and allow the reviewer of the report to acquire significantly more information about the application’s performance over time.

C. Best Solution

This solution builds on the functionality implemented in the “better” solution. Instead of storing calculation information in the data concentrator’s SOE reports, the data concentrator creates individual reports that can contain HTML or SVG formatting so that the report file can easily be opened in a web browser and show graphical information about the last *x* number of operations. These reports can highlight any trends that may have occurred in the past hour, day, month, or year. They can integrate additional information from the managed switches or other network equipment in the system, and the data concentrator can use logic to coordinate an anomaly in performance data with another system event. For example, if the monitoring logic detects an extremely large delta in the transmission time, a report incorporating information from the managed switches involved in the event could include information indicating that a cable connecting two switches in a ring was lost around the same time. As a result, the network devices in this scenario restored communications between those two switches by creating a topology change using RSTP. This would show that the alarm condition corresponds with a known change in the communications connection. An operator reviewing the report could easily see this relationship and confirm that this is expected behavior due to system design or determine that there is an issue that needs to be resolved in the system.

It is also possible to automatically generate baselines for data transfer times between IEDs that, if exceeded by too much, could generate an alarm viewable in the SCADA system, in an HMI, or in a text message or email. This would eliminate the need to assign each data point an expected transmission time window since the data concentrator would determine the baseline transfer time over a specified number of iterations. This type of functionality would be exceedingly difficult in a CFC program—a structured text (ST) program would need to be used. It may also require more capabilities with regard to file system interface, email, and automatic report generation flexibility than some data concentrators offer. The required feature set would need to be verified prior to starting development of this type of solution.

VIII. Conclusion

Automating the performance monitoring of protection and integration data exchange provides great benefits to system commissioners and operators. As this paper highlights, there are several benefits to implementing automated solutions, all of which enable a great deal of information to be provided in a single location. These solutions can provide the latest performance results and analysis for each data transmission, which is helpful in determining the functionality of communications-assisted protection schemes. They can provide a historical view of the system performance, so if any issues are found and corrected during commissioning, there is a record of the resolution implementation in a numerical, direct-measurement form. They also provide the opportunity to monitor those same communications while other tests are performed during commissioning, providing operators with additional confidence in the system. Once the system is in service, a continuous monitoring system provides operators assurance that the system is performing as expected without them having to wait for a fault or significant event to occur. The communications network in a system has a propensity to grow and gain new participants as new functionality is added to the system. Automated monitoring provides an easy method of demonstrating how these new participants in the communications network affect existing data transfer.

The cost and effort of adding automated monitoring logic to a system are relatively low since the majority of information is likely already transferred through a data concentrator to a SCADA system. This monitoring method can also scale up as the amount of information being monitored by the data concentrator increases. In addition to monitoring communications data, this same type of logic is easily expandable to other types of data as well. Incorporating automatic monitoring and reporting of critical communications data points is something all power system owners and operators should consider.

IX. References

[1] Y. Xue, B. Kasztenny, D. Taylor, and Y. Xia, "Line Differential Protection Under Unusual System Conditions," proceedings of the 39th Annual Western Protective Relay Conference, Spokane, WA, October 2012.

[2] K. Zimmerman and D. Costello, "A Practical Approach to Line Current Differential Testing," proceedings of the 66th Annual Conference for Protective Relay Engineers, College Station, TX, April 2013.

[3] K. Lee, D. Finney, N. Fischer, and B. Kasztenny, "Testing Considerations for Line Current Differential Schemes," proceedings of the 65th Annual Conference for Protective Relay Engineers, College Station, TX, April 2012.

[4] C. Martin, S. Chase, T. Nguyen, D. J. Hawaz, J. Pope, and C. Labuschagne, "Bus Protection Considerations for Various Bus Types," proceedings of the 40th Annual Western Protective Relay Conference, Spokane, WA, October 2013.

[5] J. Roberts and K. Zimmerman, "Trip and Restore Distribution Circuits at Transmission Speeds," proceedings of the 25th Annual Western Protective Relay Conference, Spokane, WA, October 1998.

[6] IEEE Standard 1815-2012, IEEE Standard for Electric Power Systems Communications-Distributed Network Protocol (DNP3).

[7] ISO 9506-1, Industrial Automation Systems – Manufacturing Message Specification – Part 1: Service Definition, 2003.

[8] IEC 60870-5-101, Telecontrol Equipment and Systems – Part 5-101: Transmission Protocols – Companion Standard for Basic Telecontrol Tasks, 2003.

[9] IEC 60870-5-104, Telecontrol Equipment and Systems – Part 5-104: Transmission Protocols – Network Access for IEC 60870-5-101 Using Standard Transport Profiles, 2006.

[10] IEEE Standard 754, Standard for Binary Floating-Point Arithmetic, 2008.

[11] IEC 61131-3, Programmable Controllers – Part 3: Programming Languages, 2013.

X. Biographies

Brian Waldron is a lead automation engineer with Schweitzer Engineering Laboratories, Inc. (SEL) in Pullman, Washington. He has several years of experience in designing and troubleshooting automation systems and communications networks. He has authored several application guides focusing on integrating automation products. He has represented SEL at IEC 61850 interoperability demonstrations organized by Utility Communications Architecture (UCA) and frequently teaches engineering design and the application of IEC 61850 solutions. Brian graduated from Gonzaga University with a BS degree in electrical engineering.

Bryan Fazzari is a development lead engineer in the research and development division of Schweitzer Engineering Laboratories, Inc. (SEL). His primary focus is on the current and future application of automation products, protocols, and technology in the industrial and electric power sectors. He has extensive experience with the design, configuration, testing, and commissioning of a wide array of automation systems. These systems include distribution automation control solutions, industrial high-speed load shedding applications, data collection and concentration systems, HMIs, and custom simulations and training programs. Bryan joined SEL in 2007, and since that time he has worked in both research and development and engineering services in both technical and management roles.

Previously presented at the 2018 Texas A&M
Conference for Protective Relay Engineers.

20180209 • TP6850-01