A Cloud-Hosted Synchrophasor Data Sharing Platform

Eugene Litvinov, Xiaochuan Luo, Qiang Zhang – ISO New England Inc.

Ken Birman, Theodoros Gkountouvas – Cornell University

Dave Anderson, Carl Hauser, Anjan Bose – Washington State University

Abstract: The deployment of Phasor Measurement Units (PMUs) could support a new generation of wide area monitoring and situational awareness systems, but this has not yet occurred. Instead, PMU data exchange occurs through bi-lateral agreements, each reflecting substantial human involvement. Our work proposes a new model for PMU data sharing, based upon today's *cloud computing* technology base. Cloud-based data capture, archiving, analysis and sharing represents a new paradigm, and provides access to flexible resources well-suited to intermittent bursts of heavy computational work. In addition, collaboration among entities could be greatly facilitated by hosting common applications in the cloud, with the further assurance when different operators examine the same data, they will see consistent information. Accordingly, we created GridCloud, a new cloud-hosted synchrophasor data sharing platform. GridCloud, which overcomes some apparent limitations of commercial cloud offerings, was developed and tested here to demonstrate the security, scalability, low latency and cost effectiveness. The system scales well enough to support nationwide deployment.

Key words: Phasor Measurement Unit, Cloud Computing, Wide-Area Monitoring and Situational Awareness

1. Introduction

During the August 14 2003 blackout, automatic protection systems worked well at the interface between New York and New England, limiting sustained outages in New England to approximately 2,500 MW of load in Connecticut, Massachusetts and Vermont. Nonetheless, the outage left 50 million people without power in nine states and one Canadian province. A major factor contributing to the blackout was inadequate situational awareness: transmission operators were uncertain of the state of the regional network and lacked ways to obtain trustworthy "ground truth" information or to coordinate their remedial actions. As a result, the US-Canada Power System Outage Task Force recommended development, evaluation and adoption of a new generation real-time tools for operations, using time-synchronized data for wide-area situational awareness [1][2].

One of the post-2003 blackout recommendations was that synchrophasor technology be deployed within the power transmission network, and an effort to achieve this goal has been underway for some time. The pace of rollout accelerated with national investment through the U.S. Department of Energy's Smart Grid Investment Grant (SGIG) program, funded by the 2009 stimulus bill [3]. As a result, synchrophasor state tracking is now a viable option, with the potential to provide substantial value for power system operations. This chapter discusses the ISO New England (ISO-NE) synchrophasor deployment and presents work on capturing data into a cost-effective data storage and analysis platform, which we call GridCloud. GridCloud is a free, open-source platform that includes a PMU-based state estimation application, and can easily be customized to run additional applications. We envision that over time, the system will scale to capture more and more data, and to host an ever-growing collection of analytic tools for system operations.

1.1 Synchrophasor System at ISO New England

We begin by reviewing the regional synchrophasor investment program. With the support from the SGIG, ISO-NE and the region's seven major transmission owners (TOs) have invested $14.9 million on a Synchrophasor Infrastructure and Data Utilization (SIDU) project. The project started on July 1st, 2010 and its implementation phase was completed on June 30th, 2013. This was followed by a two-year observation phase, which completed on June 30th, 2015. The core objective of the SIDU effort was to create a synchrophasor infrastructure and technology platform, upon which advanced analysis and wide area visualization tools could be developed and operated to enhance situational awareness.

At the time of this writing, the major components of the New England synchrophasor system include:

- PMU installations at 44 substations in New England, with adequate coverage to achieve full visibility of the 345kV network, and redundancy to maintain visibility under various PMU failure scenarios.
- Eight Phasor Data Concentrators (PDCs) at the ISO and seven transmission owners.
- Communication infrastructure to support streaming PMU data from substations to the transmission owners and then to the ISO.
- GE's PhasorPoint application for enhanced wide-area monitoring and situational awareness.
- V&R Energy System Research's Region of Stability Existence (ROSE) application to compute the operational stability boundary and margin.
- In-house developed on-line PMU Data Quality Monitoring System (DQMS) to monitor and alert on PMU data validity and health.

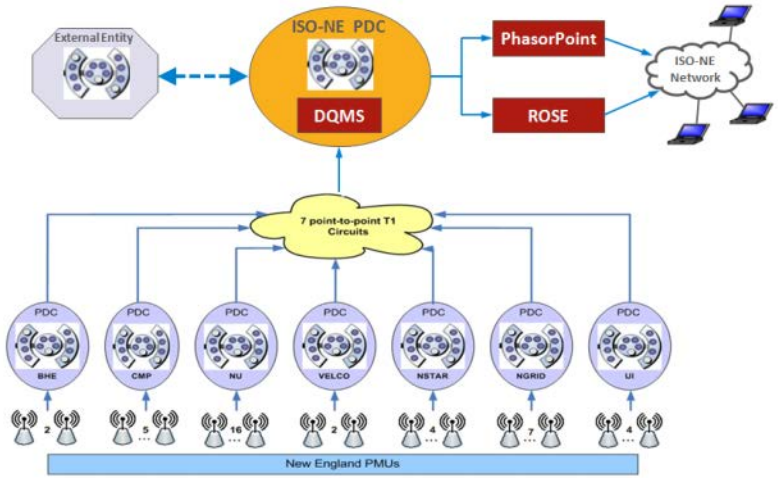

Figure 1. New England SIDU Project Implementation in June 2015

Fig. 1 shows the overall architecture of the New England synchrophasor system completed through the SIDU project in June 2015. The synchrophasor data network is shown in what is referred to as a "bottom-up tree convention." Starting at the bottom of the figure, each PMU sends data to Transmission Owner's Phasor Data Concentrator (PDC) through the corporate WAN network. The PDC receives synchrophasor data from different PMUs, time-aligns the records and then relays the resulting processed data "up" through point-to-point T1 circuits to the ISO-NE's PDC. ISO-NE's PDC performs a second stage of time-alignment as it receives data from different transmission owners, and then feeds advanced applications with one concentrated real time data stream.

1.2 PMU Data Exchange with External Regions

Although the SGIG program makes it possible for each of the major regional operators to instrument its own network, data sharing is a prerequisite for wide-area regional situational awareness. The New

England SIDU project initially lacked an infrastructure to acquire external regions' PMU data. Accordingly, in 2016 ISO-NE initiated an effort aimed at an exchange of PMU data with NYISO, PJM and MISO.

Each external region's PMU data is acquired over the existing Eastern Interconnect Data Sharing Network (EIDSN), thereby avoiding any additional costs for communication infrastructure, routers and firewalls. The EIDSN is designed to allow authorized operating entities to securely share operational reliability data, including both SCADA and synchrophasor data, and is intended to promote reliable and efficient operation of the Eastern and Quebec Interconnections, a goal consistent with the new effort.

New PDC servers were installed at the ISO-NE to receive and concentrate external PMU data, stream the resulting data to the existing PDCs, and then send the full external dataset to the PhasorPoint application together with internal ISO-NE PMU data for wide area monitoring. In order to support the added volume of data in the PhasorPoint production environment, additional disk capacity was added too, allowing retention of three years of raw synchrophasor data for both internal and external PMUs.

By sharing synchrophasor data among different regions, the ISO/RTOs enable the creation of a shared platform that could achieve real-time visibility of the dynamics and operating states covering both the ISO/RTO's own systems, and that of neighboring systems. Such a capability could be used to prevent cascading failure or blackout at an early stage. For example, it was noted in the study of the Northeastern blackout on Aug. 14, 2003 that at the outset, the operators lacked wide area situational awareness tools. As a result, when a localized fault occurred, it was observed and interpreted inconsistently by different operators. Lacking agreement on the network state, confusion arose, and the different operators reacted through actions that actually exacerbated the disruptive impact of the fault. This is ironic because the original fault was a relatively routine problem, and certainly could have been managed safely and without widespread disruption had the network state been recognized in time.

When neighboring ISOs/RTOs share synchrophasor data, consistent wide-area situational awareness is feasible: even distinct grid operators working in different control centers will still see a consistent picture of the evolving network state, in real-time, because they could work from the direct state measurements across regions based on the shared underlying synchrophasor input data streams. As a result, operators are able to coordinate actions both implicitly (because they react on the basis of the identical situational information), and explicitly (when a contingency requiring collaborative planning or coordinated reaction arises, the individuals who need to take action can base that coordinated response upon an agreed-upon network state). But if each ISO/RTO *separately* collects data and then runs its own private state estimation and analytics, then during a crisis they might still disagree on system state. The key to successful sharing is that they should see the *same* data, should agree on the wide-area network state, and then base actions on a consistent, agreed-upon, joint perspective.

Today's most pressing opportunity centers on wide-area regional state estimation and coordination, but tomorrow, the same concept could be taken even further. Sharing on an even larger scale would permit cooperative reasoning to be taken even further, enabling a new kind of "over the horizon" capability, with great potential for further benefits. Not only could the national grid be better managed in this way, it could also be better protected against disruptions such as large-scale weather events or earthquakes, deliberate attempts to destabilize the grid, etc.

1.3 Problem Statement of the Existing Implementation of PMU Data Exchange

Similar to the ISO-NE, most synchrophasor data networks in North American were implemented in a bottom-up tree structure that facilitates unidirectional data flow. Data generated from PMUs are sent through communication networks and several layers of PDCs using point-point real-time network flows (IP Unicast) to ISO/RTO PDCs, from which data are passed onto various applications. As seen in Fig. 2, PDCs are designed to time-align measurements and combine different PMU/PDC streams into one stream, the intent of which is to save communication bandwidth and simplify configuration. Another

reason for time-aligning data is that many of today's applications operate on vectors of PMU values obtained at a single timestamp. Several data protocols are in use today including IEEE C37.118-2005, IEEE C37.118-2011, C37.118.2 and IEC 61850-90-5.

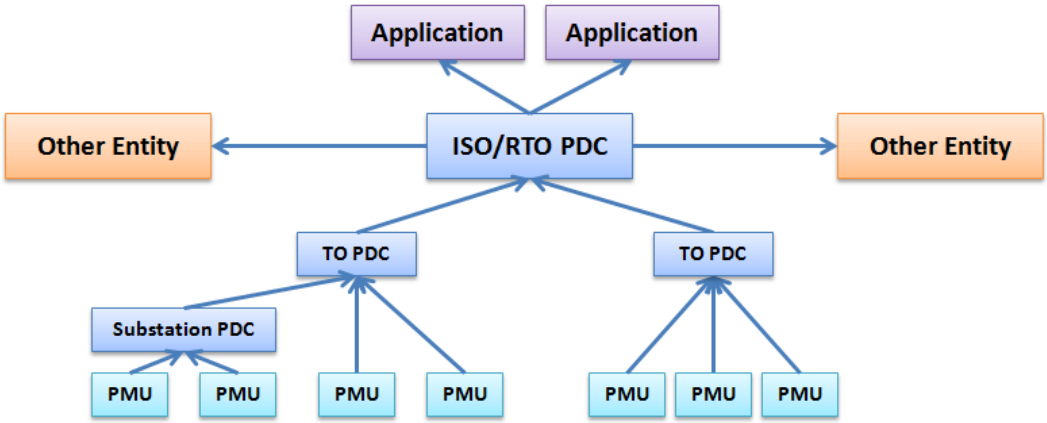

Figure 2. Typical PMU/PDC structure

Although widely used, today's implementation of PMU data exchange has a number of drawbacks:

- This chained PDC network accumulates and increases the total time delays. Small communication network delays arise as a side-effect of the way that modern computer networks handle overload when traffic from multiple sources traverses a shared network link or router. A PDC needs to wait if no data has been received from some PMUs because transient congestion on a stream can easily delay it. The PMU or communication failures will result in a maximum latency (the configured PDC waiting time). When data reaches the next level, the receiving PDC inherits the previous latencies and may experience further delay. This delay is exacerbated because PDCs must wait for multiple streams, and thus the worst delay on any single incoming stream tends to be the performance-limiting factor for the combined output stream.
- There is substantial complexity to managing a PDC hierarchy. For example, the PDC operator must configure the PDC to coordinate waiting time across the hierarchy: not an easy task.
- The time alignment at each PDC hierarchy might be unnecessary. As a recommended practice, most entities have each PDC perform time alignment and only send the concentrated data stream to PDCs at the next level or advanced applications. However, from an application's point of view, time-alignment only needs to occur once, at the last step. Therefore the relaying PDCs do not need to time-align data (additional delays), except for the end ones that feed applications. A decade or more ago, network bandwidth was a severe concern, and time-alignment seemed appealing because a PDC output is more compact than a set of PMU and PDC inputs, but today's networks are so rapid that this kind of bandwidth reduction is not a limiting factor. Given that bandwidth is no longer a limiting factor, it is preferable for each PMU to send data directly to the relevant last-hop PDCs (for purposes of administrative control, this might still involve *routing* the data through one or more PDCs, but without delaying and time-aligning the data stream prior to forwarding the output).
- With purely bilateral exchanges, data streams will be duplicated because each ISO/RTO has multiple peers. For example, NYISO, PJM and MISO currently request different but overlapping PMU data sets from the ISO-NE. ISO-NE therefore is required to configure three different output streams, sending each over the EIDSN. Similar situations arise at NYISO, PJM and MISO.
- Each entity has to model and maintain its own and other regions' PMU data. For example, ISO-NE has to model PMU data from NYISO, PJM, and MISO in the PhasorPoint application for wide area situational awareness. Similarly NYISO has to model PMU data from PJM, MISO and ISO-NE in their advanced visualization applications.
- Today most entities focus on sharing PMU data, instead of sharing common displays among regional

grid operators. Since each ISO/RTO uses different vendors and applications for wide area monitoring and situational awareness, the grid operators may not be able to communicate straightforwardly when they call each other after identifying a wide area problem.

With the continued growth of synchrophasor technology by US utilities and the inefficiency of the current implementation of PMU data exchange for wide area monitoring, we need to look at new solutions which are building on mature and tested technologies, and use them in a small development process that focuses on leveraging the best of breed solutions in ways that have been successful in the past.

In the next section, we present a solution to this problem. Researchers from the ISO-NE, Cornell University and Washington State University have built a platform based on the existing cloud computing infrastructure, "hardening" it for the purpose of PMU data exchange and wide area monitoring. The resulting platform offers a shared infrastructure, is highly robust, and represents a response to the various concerns elaborated above.

2. Why the Cloud?

2.1 Overview of Cloud Computing Technology

The term cloud computing itself is extremely "flexible," and is used in conjunction with multiple styles of computing and communication. According to U.S. National Institute of Standards and Technology (NIST), cloud computing can be defined as "*a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storages, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*"[4]. The cloud computing paradigm centers on easy access to large-scale computing resources via a network (i.e., Internet) at relatively low cost, and with the capability of quick reconfiguration to add resources or release them, using a pay-as-needed cost model. Cloud computing can be a basis for a robust and efficient solution that might briefly use computing resources far beyond the economic reach of most organizations, and that has outstanding operational efficiency, resilience, and sustainability.

While cloud failures do occur, it is very rare for an outage to impact more than one data center at a time. Thus, much as for other power grid reliability goals, one can use redundancy to anticipate and overcome the types of failures seen in operational cloud settings. Further, cloud security has become stronger and stronger: one can already protect cloud systems against a wide range of attacks, and with new hardware security options emerging at a fast pace, it won't be long before cloud systems can even be protected against intrusion by the data center operator itself.

Various drivers are contributing to the surge of cloud computing technology, including the dramatic decrease in costs for hardware optimized to support cloud computing, the simultaneous increase in computing power and storage capacity, the advent of multiple-core and multi-thread computing architectures, customer demand for stronger and stronger security, and the exponential growth in computing and data storage demands on the part of scientific and commercial applications.

To accommodate the needs of different customers, cloud computing services can be provided in different ways. According to the levels of the capacity and service provided, cloud computing services can generally be divided into at least the following three categories [5][6]]:

- Infrastructure as a Service (IaaS): A cluster of virtualized resources (e.g., CPU capacity, data storage, network bandwidth and basic operating system, etc.) are provisioned. Once being provisioned, customers run software stacks on the top of the virtualized resources and are able to re-size the scale of resources as needed.
- Platform as a Service (PaaS): An environment on which customers can easily develop and deploy applications on a large scale of computing infrastructure to meet changing demands.

- Software as a Service (SaaS): Online applications are delivered as web services to users which allows for alleviation of the burden of software maintenance and upgrade on the customers' side.

According to the physical location and distribution of cloud computing services, a cloud can be further classified as [5][6]

- Public cloud: A third-party provider offers and maintains a multi-tenant cloud infrastructure and/or services to customers on a subscription basis that allows for the optimum utilization of computing resources. Here the physical server might be simultaneously used by other clients.
- Private cloud: Cloud computing infrastructure that uses cloud software but runs within a company's own data center and is dedicated to internal users and/or partner users.
- Hybrid cloud: Mixed usage of the client's own computing resources and resources hosted on a public cloud. Such an approach allows the customer to employ public cloud services for specific purposes that the customer's own capacity is inadequate to address, securely combining the cloud data and output with the customer's own data and systems.

Commonly recognized benefits of cloud computing technology include:

- Operational efficiency and cost savings
- Operational resilience through state-of-the-art backup services and security schemes as well as fully redundant hardware infrastructure and services
- Scalable and Flexible
- Almost unlimited storage
- Shortened development life cycle
- Ease of interoperation
- Shared data and applications for easy collaboration among different entities

Increasing attention has been drawn to potential security and privacy issues with cloud computing implementation. The risk is evident: in a cloud setting, a third-party provider hosts data that the end-users traditionally have stored on their internal computers behind many layers of firewalls. However, customer demand for ironclad cloud security has fueled a wave of creative work, resulting in powerful new security mechanism with better and better data and computation protection, and remarkably good options for secure interoperability, and for integration of cloud and private systems.

Today's top-tier cloud service providers such as Amazon Web Service (AWS) deploy state-of-the-art security schemes for enhanced data security and privacy protection [7]; while the out-of-box configuration is often not an exact match to the needs of the power sector, our experience shows that the underlying security building blocks are powerful and flexible, and that the needed solutions can be created by configuring them properly. These features center on the AWS virtual private cloud (which uses software encryption to protect data and implement firewalls). If one trusts the software used to implement them, the resulting security is extremely good.

Meanwhile, new and even more exciting options are emerging, in which one would need to trust the hardware but not the software or the data center owner. For example, with Intel's SGX architecture [8][9], a cloud hosts computing "enclaves" that even the data center owner itself cannot penetrate: the data center hosts the system and provides computing cycles and storage, yet the computation and stored data is encrypted at all times, using keys that reside only at the customer's site. Thus, even an intruder who has fully compromised the cloud data center would have no ability to observe the data streams, network state, or other analytics carried out in a system like GridCloud: that attacker's power would be limited to crashing or overloading computers. We plan to do experiments with SGX, but in fact SGX is just one example among an array of proposed hardware technologies for security.

Our conclusion is that for a properly configured system, Internet and cloud security is already surprisingly good today, and far better than one might imagine without looking closely at the choices. Moreover, we see strong reasons for optimism that security will continue to improve in the future.

2.2 GridCloud Platform

The GridCloud platform was created by Cornell University and Washington State University, as a research effort sponsored by the DOE's ARPA-E GENI program. GridCloud aims to achieve a highly scalable, reliable and secure computing infrastructure strong enough to operate a nationally critical infrastructure such as the power grid. It extends a standard commercial cloud-computing infrastructure, configuring it for maximum security and enhancing real-time and fault-tolerance features through use of redundancy and replication. GridCloud thus seeks a balance between the cloud's widely-cited cost-effectiveness and flexibility and the special requirements of PMU data sharing and wide area monitoring.

The core technologies used in GridCloud draw upon decades of experience and systems work by our team on issues of reliability, consistency, fault-tolerance and real-time availability. A book-length treatment of the issues that arise, the underlying theory, and the protocols employed to achieve correct behavior is available [10]. For our purposes here, we will discuss higher level solutions we created using those techniques, but without delving deeply into the theory or the structure of the software tools we created using that theory.

The resulting solution is quite flexible. In particular, GridCloud can be configured to run on any desired number of cloud data centers by using data and code replication both for fault-tolerance and performance. The system defines a standard data collection infrastructure, which can include standard databases and solutions such as OpenPDC. By hosting shared applications on machines that are rented only as needed, GridCloud can offer the ISOs and TOs authorized access to the platform - a highly effective collaborative infrastructure.

GridCloud proves that mission-critical real-time solution could be hosted on today's cloud by careful attention to security policy together with additional technology aimed at high-availability and real-time performance. The system supports a mix-and-match approach to deployment and has following key components:

2.2.1 The GridCloud Security Architecture

The GridCloud system obtains security by careful configuration of Amazon's virtual private cloud and network link encryption options. No new mechanisms were required. The security solution trusts the core security features offered by Amazon in support of its virtual private cloud and network link-layer security, including implementation of the software, and Amazon's correct operation of their infrastructure. Further, it trusts Amazon's system operators not to attack the active system, for example by freezing a copy of the system and then inspecting it offline using forensic analysis software tools. Such actions are not possible for other cloud customers, and they are intrusive enough that they would trigger audit and other monitoring alarms, but are certainly attacks that data center owner intent on breaking into a virtually private cloud could conduct. Thus GridCloud is secure against a large class of possible attacks, but its security model could be defeated if the cloud-hosting company were using flawed software, or had been severely compromised in one of the ways we've outlined.

In the future, if GridCloud begins to take more advantage of hardware security, the "attack surface" of the system can be reduced in many ways. On the other hand, like any technology, it will still ultimately be dependent upon the hardware on which it runs, and core elements of the software used to operate that hardware.

2.2.2 The GridCloud Data Collection Layer

The GridCloud data collection layer captures incoming PMU data over one or more cryptographically secured network links, writes all received data into files to create a historical record and also relays it to the state estimation subsystem. This system automatically establishes secure, replicated connections and will automatically reestablish connectivity if disruption occurs. Redundant data collection is employed to overcome network timing issues and ensure that critical data will still flow into the system even if some network links fail.

2.2.3 The GridCloud Archival Data Storage Subsystem

The GridCloud data collectors operate by writing the received data into files and simultaneously forwarding the data for state estimation processing. GridCloud stores data in a variety of formats, favoring standards. For example, we store streams of PMU data into append-only logs in the IEEE C37.118 format. We store network state estimates as a table, in which each row describes an element of the network model for the grid and gives that element's estimated state. The storage system also records both the original timestamp of the PMU data, and the platform time at which the data was actually archived. By so doing, we can sense situations in which a GPS time receiver malfunctions and streams containing data with incorrect timestamps, and can also support time-based retrieval (snapshots of the exact system state at a designated instant in time, accurate to the millisecond).

Before deciding to implement our own solution, we evaluated a number of candidate file systems relative to the requirements posed for this archival storage role. Thorough review and experiments revealed limitations that precluded use of existing solutions: prior systems lacked ways to exploit embedded timestamps, and the prior approaches to retrieving snapshots embodied temporal inaccuracies, performance issues, and inconsistencies when files are being updated rapidly [11][12]]. Accordingly, we developed a new solution: the Freeze Frame File System (FFFS) [13] which offers a quick way to access data at any desired temporal resolution, from any point in the past or present history of the system, and employs a memory-mapped access structure and hardware assisted data transfers (RDMA) to achieve high performance in support of smart grid computational tasks. FFFS also provides logical consistency in the sense of Chandy and Lamport [14].

2.2.4 Cloud Manager

Management of the GridCloud infrastructure posed challenges: existing cloud management tools focus on end-users who are accessing web pages, a scenario remote from wide area monitoring. Accordingly, the GridCloud required a new robust system management tool. Cloud Manager (CM) is an extension of the UNIX "Makefile" infrastructure. The tool supports the standard dependency-triggered model used in Make (and the standard syntax, too), but extends the model to also react to system events such as nodes failing or joining, events being sensed in the incoming data stream, etc. CM encodes such events as XML files and when something occurs, the file update triggers reactive behavior as specified in the CM script, which can be customized for each use-case. Actions are similarly initiated: CM outputs a file, and we use this to drive the desired response. We use an off-the-shelf constraint solver to carry out the optimizations needed for purposes such as job placement on the available cloud nodes, deciding which data collector will handle which data stream, etc.

CM also manages the desired suite of applications. At present, CM application support is focused on a state estimator and visualization tool, but new applications can easily be added to the system and either configured to run continuously, 24x7, or on demand. CM can also automatically configure applications with redundancy. We've worked with either one or two Amazon data centers, and in one case, went further and also triple-replicated our entire cloud infrastructure within one Amazon AWS data center, to understand the extent to which doing so could conceal scheduling and other latency factors.

3. Proof-of-concept Cloud Hosted Wide Area Monitoring System

In 2015, ISO-NE established a collaboration with Washington State University and Cornell University to employ GridCloud as a proof-of-concept infrastructure, collecting synchrophasor data in a wide-area monitoring deployment and then carrying out cloud-hosted state estimation in real-time. This is the first such application of cloud computing in the electricity industry, which has been cautiously turning to the cloud, migrating various roles as the needed reliability, security and privacy concerns have been addressed. The project seeks to demonstrate a reliable, secure, and fault-tolerant cloud-hosted synchrophasor platform and wide area monitoring system and to carefully evaluate performance,

scalability and cost, but not to actually use the system in production.

3.1 Conceptual Overview of the Cloud-hosted Synchrophasor Platform

Figs. 3 and 4 give a conceptual overview of the resulting cloud-hosted synchrophasor platform. The synchrophasor data are streamed from PMUs in the substation directly to the cloud or from PDCs at the TO or ISO/RTO to the cloud-hosted data collectors. The cloud data collectors then route the PMU data to various cloud-hosted applications, such as Regional Data Repository and Real-Time Phasor State Estimator (RTP-SE). The RTP-SE performs an interconnection-wide, time-synchronized assessment of the voltage magnitudes, phase angles, and transmission line flows and provides common visualization displays for real-time collaboration among grid operators. The regional entities can also subscribe to each other's PMU data or retrieve historical data from the central repository. Because data feeds, the data archive, and the state estimator are shared, distinct entities can see consistent information at all times.

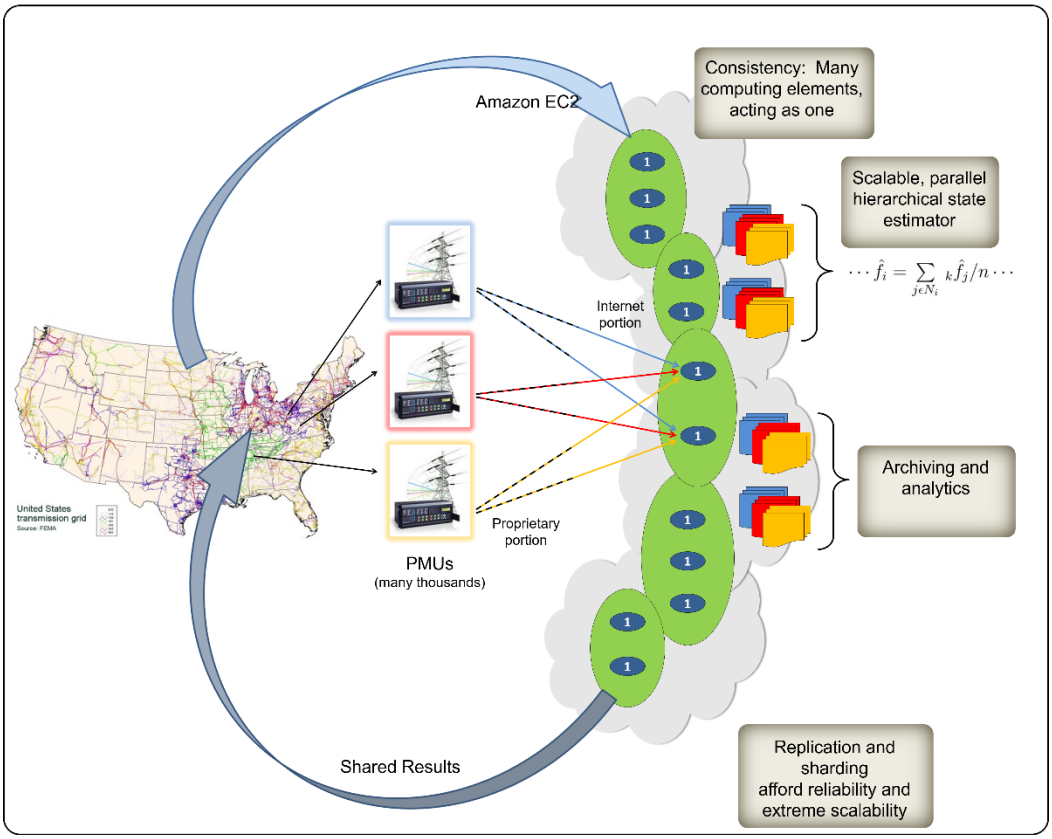

Figure 3. GridCloud Infrastructure Diagram

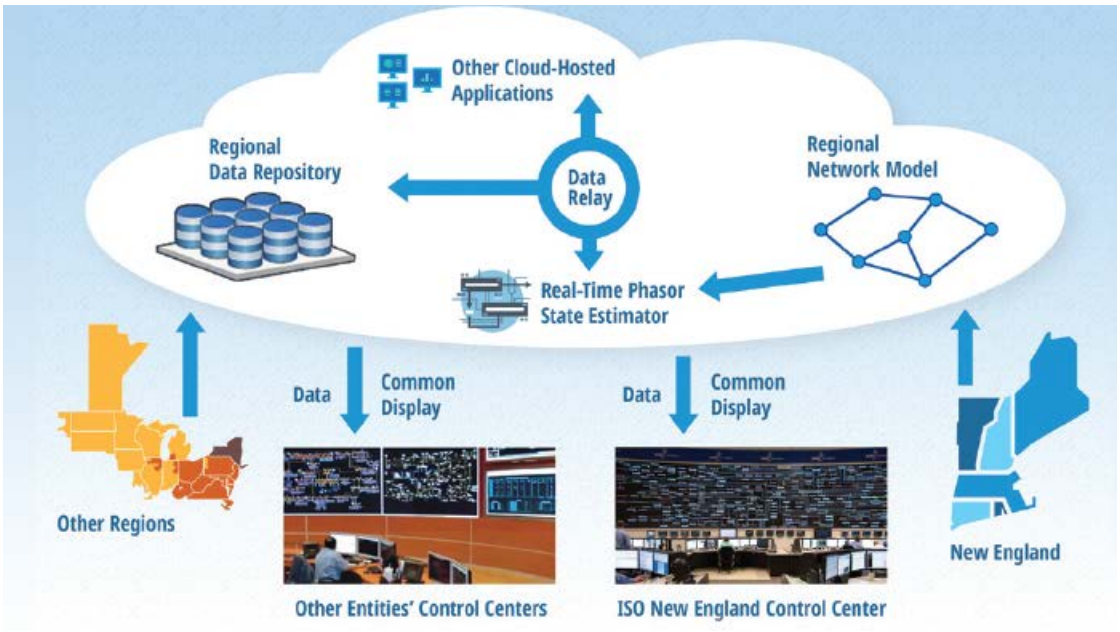

Figure 4. Overview of the cloud-hosted synchrophasor platform

The cloud based RTP-SE is a core component which performs state estimation at one second intervals or faster - a much higher update rate than the current SCADA-based EMS state estimation which occurs at three minutes intervals in the ISO-NE. With sufficient computational resources the RTP-SE could be updated even at the PMU data rate so that the power system dynamics are captured.

The RTP-SE offers the capability to:

- Provide time-synchronized state estimation that leverages PMU data and accurate GPS time information to eliminate state estimation errors caused by time-skew;
- Calculate "pseudo" PMU measurements at unmeasured buses and lines and fill in missing PMU data;
- Cross-validate the PMU data at different substations, allowing data quality enhancement;
- Employ cutting-edge linear state estimation methods, which are fast, have no convergence issues, and work properly even under islanding conditions;
- Use a high sampling rate to achieve dynamic visibility of disturbances, disturbance propagation, frequency response, and system oscillation;
- Establish real time models to determine stability margin.

Currently there is very little industry-wide standardization around power system visualization tools or algorithms for situational awareness for a cooperative environment among entities. Different control centers have deeply rooted traditions about one-line schematics, color conventions, overloading alert/alarms, etc. The cloud-hosted RTP-SE offers an opportunity for operating entities to collaborate on addressing this vitally important and mutually needed capability. Operation staffs in different control centers can be trained to understand and interpret the information in the same way so that they can collaborate easily and effectively when contingencies requiring cooperative action arise.

3.2 Proof-of-concept Experiment Setup

Using the GridCloud system as described above, WSU, Cornell and ISO-NE carried out a deployment, then undertook an experimental evaluation to assess the performance of the system, specifically focusing on both internally and externally observable latencies, on the time to recover from failures, and on whether the system consistently delivered the same data and results when running multiple, parallel instances. In addition the demonstration system was instrumented to evaluate any additional latency introduce by the use of encryption of communication channels.

The experiment uses real PMU data captured by ISO-NE. Twenty-one seconds of recorded historical data were replayed at 30 measurements/second with their timestamps altered to reflect the current time. The 21-seconds of data were "looped" to provide continuous data streams. The results below are from a 25-minute run of the system.

There were a total of 73 PMUs. We split the data across two data source machines, with 31 PMUs streaming from ISO-NE and 42 PMUs streaming from Cornell to the cloud, to mimic a situation in which two or more ISOs might collaborate to share data via the cloud platform. Each data source was emulated and the data was transmitted via a PMU data replay and streaming software, designed to transmit each measurement at the proper instant in time.

The overall experimental design is seen below in Figure 5. PMU data was sent from Cornell and ISO-NE to the AWS cloud via SSH tunnels in IEEE C37.118 format. These data are captured by a set of data collectors in the cloud, then archived in the cloud, as well as processed by the RTP-SE application software in the middle and a synchrophasor data visualizer on the right. The RTP-SE results were also relayed back to the Cornell in the form of additional C37.118 feeds, representing the reconstructed power system states after the state estimation solutions. The raw PMU streaming from ISO-NE was also replayed back to the Cornell in the form of C37.118 format, representing raw PMU data exchange between different entities.

To evaluate the costs of redundancy for fault-tolerance, all data was mirrored, with one copy sent to a GridCloud system running on an AWS data center in Virginia, and the other, in Oregon.

The RTP-SE used in the experiment was the Linear State Estimator (LSE) created at Washington State University by Anjan Bose and his colleagues [15]. This is a complex application that includes an internal communication bus and OpenPDC, as a front-end for ensuring that only time-aligned data are passed to the actual application, and of course an implementation of the linear state estimator algorithm. Cloud Make is used to track the health of the LSE's components and, if anything fails, can cleanly shut down and then reboot the entire LSE application.

The power system network model used by the LSE was based on a PSS/E planning model that matched the operational topology at the time when the PMU data was captured by the ISO-NE.

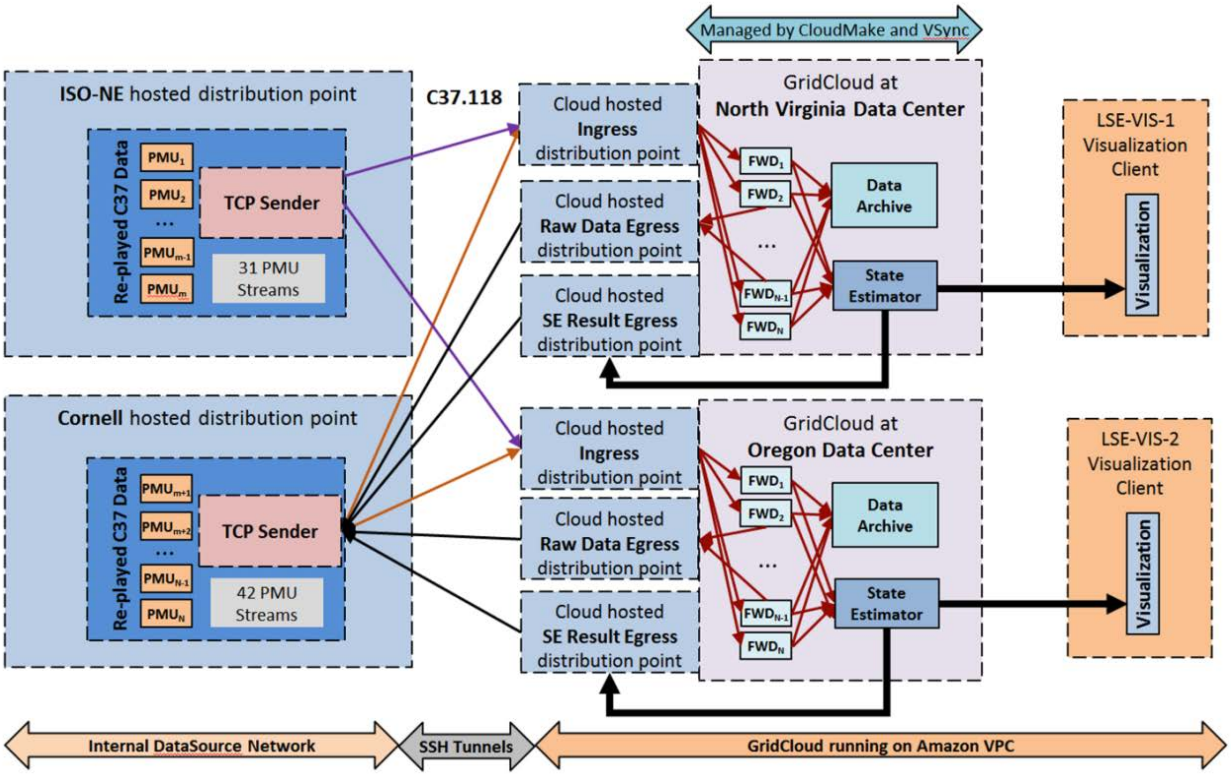

Figure 5. Overall design of the Proof-of-concept project

3.3 Experiment Findings

3.3.1 Security

We utilize three defenses at different parts of GridCloud in order to protect the system against a variety of security threats. First, we employe SSH tunnels as secure connections between data sources (where PMU data is replayed) and datacenters (where PMU data is processed). The SSH tunnels add less than 2ms latency (less than 1%) to Round Trip Time (RTT) on average, relative to unencrypted TCP. Based on these measurements we consider SSH tunnels to be a cost-effective solution for securing the external (to the datacenters) communications.

GridCloud also needs to be protected from other users running in a Public Cloud environment. To address this issue, GridCloud utilizes Amazon's Virtual Private Cloud (VPC). Relative to the basic AWS Elastic Computing Cloud (EC2) service, VPC offers better isolation guarantees, meaning that other applications running in the same physical machines cannot deduce any important information about GridCloud. Of course, this improvement does not come for free and reduces performance by a noticeable margin. Table 1 summarizes the results of GridCloud platform with and without VPC enabled in AWS. We measured the latency between the data sources and the Linear State Estimation application (without considering computational cost of LSE itself). The average additional latency, when VPC is used, is 16ms (approximately 6.5%). We consider this overhead acceptable, since the VPC configuration greatly improves the protection of sensitive data against malicious behavior by other cloud users.

Latency (DataSource-LSE)	Without VPC	With VPC
Average	245ms	261ms
1st Percentile	211ms	228ms
99th Percentile	255ms	270ms

Table 1: One-Way Latency between DataSources and Linear State Estimator at Oregon with and w/o VPC enabled

A third security concern arises when maintaining data confidentiality for information stored into persistent storage (SSD or Hard Disk). GridCloud uses AWS S3 storage with AES 256 encryption to protect against attacks of this type, and AWS, for its part, takes steps to ensure that no person with physical access to a storage device would also have a way to obtain the storage encryption keys. Our experiments showed that there is no measureable performance impact when GridCloud enables the S3 encryption option.

3.3.2 Latency

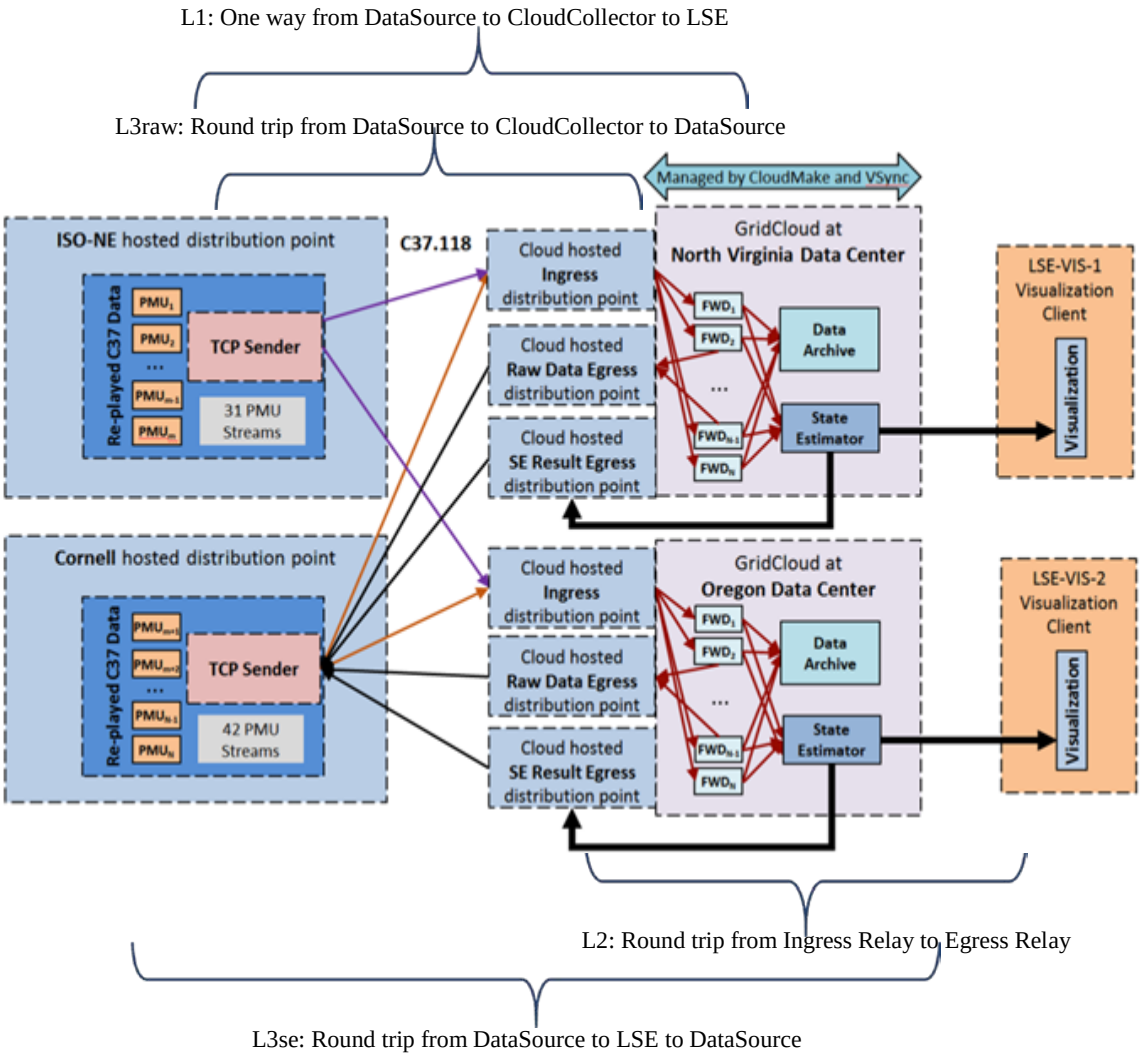

Figure 6. Latencies measured in GridCloud platform

Figure 6 shows three different types of latencies that we measured in the experiments. L1 is the one-way latency from data sources up to the point where data are available for application (Linear State Estimation in our measurements). L1 measurements are summarized at Table 1. In less than 300ms, more than 99% of the data produced by PMUs is available to the LSE application.

It is useful to understand how much time is spent in the network between data sources and datacenters

and inside the datacenter. L2 is the round-trip latency between the point when data enters the datacenter and when the corresponding processed data exits the datacenter. Figure 7 shows that average value of L2 is around 300ms. From previous work we know that most of that is due to the LSE computation itself. Since L2 is measured entirely internal to each datacenter, as expected there are no major differences between the two datacenters.

	Virginia	Virginia-Internal	Oregon	Oregon-Internal
ISONE Raw-Low Min	20		88	
ISONE Raw-Low 1st Percentile	22		89	
ISONE Raw-Low Average	**25**		**102**	
ISONE Raw-Low 99th Percentile	58		152	
ISONE Raw-Low Max	611		696	
ISONE Raw-High Min	22		90	
ISONE Raw-High 1st Percentile	25		99	
ISONE Raw-High Average	**46**		**127**	
ISONE Raw-High 99th Percentile	82		179	
ISONE Raw-High Max	612		697	
Cornell Raw-Low Min	17		90	
Cornell Raw-Low 1st Percentile	17		91	
Cornell Raw-Low Average	**18**		**115**	
Cornell Raw-Low 99th Percentile	20		191	
Cornell Raw-Low Max	49		407	
Cornell Raw-High Min	18		91	
Cornell Raw-High 1st Percentile	18		92	
Cornell Raw-High Average	**19**		**120**	
Cornell Raw-High 99th Percentile	20		199	
Cornell Raw-High Max	49		413	
SE Results Min	279	242	351	240
SE Results 1st Percentile	294	267	370	273
SE Results Average	**325**	**300**	**409**	**317**
SE Results 99th Percentile	384	348	490	393
SE Results Max	911	469	962	642

L3 Raw: Virginia ISONE Raw-Low Min through Cornell Raw-High Max; Oregon ISONE Raw-Low Min through Cornell Raw-High Max

L3 SE: Virginia SE Results Min through SE Results Max; Oregon SE Results Min through SE Results Max

L2: Virginia-Internal SE Results Min through SE Results Max; Oregon-Internal SE Results Min through SE Results Max

Figure 7. L2 and L3 results for North Virginia and Oregon datacenters

L3 is the full-path latency between the data sources and the data consumers located outside the cloud. L3 Raw is the latency for raw unprocessed data while L3 SE is the latency for LSE results. As can be seen in Figure 7, Oregon latencies are approximately 60ms larger than the Virginia ones, since Oregon datacenter is much further away from data sources (network latency is higher). Moreover, it is evident that the Linear State Estimator contributes around 270ms of latency (comparison between L3 Raw and L3 SE). Figure 8 and Figure 9 provide additional details on the latency distribution of L3 Raw and L3 SE.

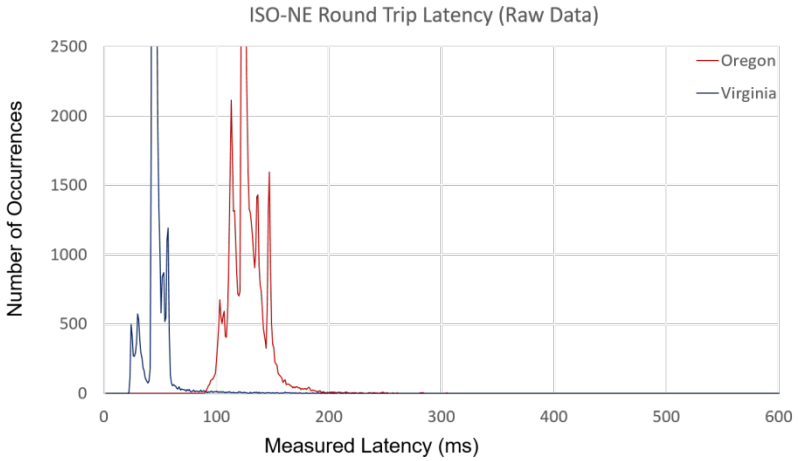

Figure 8. L3 Raw latency to Oregon and Virginia datacenters

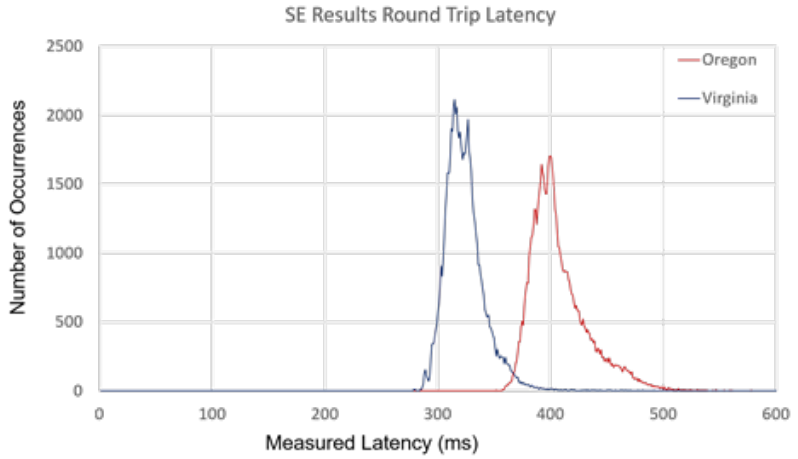

Figure 9. L3 SE latency to Oregon and Virginia datacenters

3.3.3 Data Consistency and Fault Tolerance

When using standard data mirroring approaches with two different datacenters, inconsistencies can arise. GridCloud is designed to minimize this problem, but cannot fully mask it: if a failure occurs, some raw data streams might not reach one or both datacenters. Furthermore, LSE calculations are based on data that arrives *on time* at the datacenter. A LSE computation is very likely to produce different outputs when there are missing inputs. In our case, if LSEs do not have the full array of data, they do not do any computations. Thus, we expect to have missing LSE result points rather than incorrect results.

The results of our 25 minutes experiments can be seen in Table 2. All the raw data were delivered in both datacenters (N.Virginia, Oregon). There are some missing LSE results in both datacenters (mostly in Oregon), but the rate of data missing is very small. We also witnessed a couple of different LSE results (between the two data centers) for the same stream and same timestamp. This should not normally happen, but seems to arise because of a long delay each time the 21-second data loop repeats: in all cases, these occurrences were seen on the first data point of a looped 21-second epoch. We also confirmed that all data received by each data center was stored (indeed, we confirmed that both data centers had identical data at the end of the run). This seems to suggest that the rare inconsistencies

that arose in our experiments were due to some form of long delay when data sources reset for a new 21-second epoch: the same data ultimately reached both data centers, but in one, a data point arrived too late and was excluded from the LSE computation, while in the other, it was included.

Data Type	Identical packets in both Datacenters	Packets received in N. Virginia but not in Oregon	Packets received in Oregon but not in N. Virginia	Packets received at both Datacenters but with different data value (same timestamp)	Packets dropped at both Datacenters
Raw	100%	0%	0%	0%	0%
LSE	99.952%	0.033%	0.013%	0.002%	0%

Table 2. Consistency of Raw and LSE Data across N.Virginia and Oregon datacenters.

Although we did not experiment with node crashes during the study, we did experiment with complete data center shutdowns. Restart required approximately 175s, during which no data was lost because the Oregon data center continued to operate while the Virginia one was recovering. The vast majority of the 175s was consumed by EC2 rebooting the Windows LSE instance. Then, the delays of reconfiguration and initial setup of the LSE instance are non-significant compared to 175s. At the end of the 175s period, the GridCloud system was back to full function (state estimation included) and full redundancy.

3.3.4 Operational Cost

Table 3 shows the operational cost for GridCloud, in dollars per hour for each element of the system. Our experiments used one AWS instance (Cloud Relay) as a gateway for incoming and outgoing traffic at each datacenter. We used 2 instances for routing the data inside the datacenter (Forwarder). Finally, one instance was used for the LSE application, another for the Cloud Manager Leader, and seven instances for archiving data (3 for raw data, 4 for LSE results). If we also include the visualizer, the total cost of deploying GridCloud at this scale is $2.47/hour/datacenter.

Instances	Type	Number	Price	Total
Cloud Relay	C3.large	1	$0.11	$0.11
Cloud Manager	C3.large	1	$0.11	$0.11
Visualizer	C4.largeWin	1	$0.19	$0.19
LSE	C4.xlargeWin	1	$0.39	$0.39
Raw Archiver	C3.xlarge	3	$0.21	$0.63
LSE Archiver	C3.xlarge	4	$0.21	$0.84
Forwarder	C3.large	2	$0.11	$0.22
Total		13		$2.47

Table 3. Breakdown of Operational Cost/Hour

4 Challenges and Future Research Directions

Our preliminary work with GridCloud confirms that the system is a robust basis for capturing PMU data, archiving it, and for carrying out state estimation. Additional applications will be added in the future, as well tools for carrying out exploratory data analysis on the archive of past system states.

We see a number of challenges ahead. First, as more ISO/RTO partners are added to the system, a wide range of regulatory and security-policy questions will inevitably arise. Our belief is that the system is

adequately secure and robust to respond to these requirements, but until such steps are carried out, it is impossible to be certain. In particular, we note that the primary policy document for our area of work, namely the NERC CIP v5, was not designed with cloud-hosted PMU data monitoring in mind.

A second question concerns leveraging advanced security technologies, notably the Intel SGX architecture, which can protect an application even against a malicious or intrusive cloud host. With SGX, we believe that GridCloud could operate so securely that Amazon itself would be unable to see any of the data captured or archived within it, even by freezing and examining the states of compute nodes, disks, or carrying out other forms of attack on the system. However, to fully convince ourselves that this is the case, we would need to carefully evaluate the exact manner in which SGX was employed, to confirm that the configuration itself is correct and trustworthy. Further, to be practical, the performance and cost impact would need to be measured and determined to be within an acceptable range. SGX is just one of several options that might be considered.

A third question involves scalability of the platform with growing numbers of users and growing numbers of PMU data streams. Although our preliminary experiments suggest that GridCloud could scale to a deployment that would instrument and monitor the entire North American continent, careful monitoring of performance and costs will be required as the system expands over time.

Finally, we are interested in the concept of GridCloud as a "big data" platform for the bulk power community, offering tools aimed at better understanding power management, improving operational efficiencies, and tracking down the sources of disruptions or instabilities. Such tools could leverage some of the existing technologies available in today's cloud, but would need to also go beyond today's most common solutions simply because power-grid analytics are unique in many ways: the real-time nature of the problem, the specific problem formulations that are of interest, and the kinds of analysis required, all depart from anything seen in common web hosting platforms or big-data systems used to understand human behavior, advertising placement, or other questions of interest to the primary cloud community today. With further work, considerable progress can certainly be made on this question.

Reference:

[1] "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," https://www.npcc.org/Library/Blackout%20Recommendations/BIT_Conclusions.pdf

[2] "Conclusions of the NPCC 2003 Blackout Investigative Team Assessing the Power System Collapse of August 14, 2003," https://www.npcc.org/Library/Blackout%20Recommendations/BIT_Conclusions.pdf

[3] "Smart Grid Investment Grant (SGIG) Program Final Report," https://www.smartgrid.gov/document/us_doe_office_electricity_delivery_and_energy_reliability_sgig_final_report.html

[4] The NIST Definition of Cloud Computing, https://www.nist.gov/sites/default/files/documents/itl/cloud/cloud-def-v15.pdf

[5] Cloud Computing, https://en.wikipedia.org/wiki/Cloud_computing

[6] Zhang, Q., Cheng, L. & Boutaba, R., "Cloud Computing: state-of-the-art and research challenges," Journal of Internet Services and Applications (2010) 1: 7-18. DOI:10.1007/s13174-010-0007-6.

[7] www.amazon.com

[8] Anati, Ittai, et al. "Innovative technology for CPU based attestation and sealing," Proceedings of the 2nd international workshop on hardware and architectural support for security and privacy. Vol. 13. 2013.

[9] Costan, Victor, and Srinivas Devadas. "Intel SGX Explained," IACR Cryptology ePrint Archive 2016 (2016): 86.

[10] Kenneth P. Birman. Guide to Reliable Distributed Systems. Building High-Assurance Applications and Cloud-Hosted Services. (Texts in Computer Science). Springer. 2012.
[11] Shvachko, Konstantin, et al. "The hadoop distributed file system," 2010 IEEE 26th symposium on Mass storage systems and technologies (MSST).
[12] Weil, Sage A., et al. "Ceph: A scalable, high-performance distributed file system," Proceedings of the 7th symposium on Operating systems design and implementation. USENIX Association, 2006.
[13] Song, Weijia, et al. "The Freeze-Frame File System," Proc. of the ACM Symposium on Cloud Computing (SoCC'16). 2016.
[14] Chandy, K. Mani, and Leslie Lamport. "Distributed snapshots: Determining global states of distributed systems," ACM Transactions on Computer Systems (TOCS) 3.1 (1985): 63-75.
[15] Tao Yang, H. Sun and Anjan Bose, "Two-level PMU-based linear state estimator," 2009 IEEE/PES Power Systems Conference and Exposition, Seattle, WA, 2009, pp. 1-6.