Smart Generation and Transmission with Coherent, Real-Time Data

David E. Bakken, Anjan Bose, Carl H. Hauser
School of Electrical Engineering and Computer Science
Washington State University
Pullman, Washington, USA
{bakken,bose,hauser}@eecs.wsu.edu

Edmund O. Schweitzer III, David E. Whitehead, Gregary C. Zweigle
Schweitzer Engineering Labs, Inc.
Pullman, Washington, USA
{Ed_Schweitzer, Dave_Whitehead, Greg_Zweigle}@selinc.com

***Abstract*—In recent years much of the discussion involving "smart grids" has implicitly involved only the distribution side, notably advanced metering. However, today's electric grids have many challenges involving the rest of the grid—the bulk power system—that can be mitigated by making it more intelligent. An enabling technology for helping the bulk power system that has emerged in recent years is coherent, real-time data such as synchrophasors. In this paper we describe major challenges facing electrical generation and transmission today, including distributed generation (both microgrids and renewables), that availability of these measurements can help address. We overview applications utilizing coherent, real-time measurements that are in use today, or proposed by researchers. Specifically, we describe, normalize, and then quantitatively compare key factors for these power applications that influence how the delivery system should be planned, implemented, and managed. These include whether a person or computer is in the loop; and for both inputs and outputs: low latency, rate, criticality, quantity, and geographic scope. From this, we abstract the baseline requirements for performance and availability of a data delivery system supporting these applications and suggest implementation guidelines required to achieve them. Finally, we overview the state of the art in the supporting computer science areas of networking and distributed computing (including middleware), and analyze gaps in available network protocols, commercial middleware products, and utility standards in this area.**

Keywords—wide-area measurement systems; synchrophasors; middleware; smart grid; critical infrastructure protection

I. INTRODUCTION AND BACKGROUND

Large power grids around the world were built mostly or completely from the ground up. During the middle of the twentieth century, utilities integrated into larger power grids in order to improve reliability. In such a structure, an entire grid such as the USA Eastern Grid or Western Europe's UTCE *ipso facto* operates at the same frequency, and supply and demand must be balanced in real-time across each grid.

Unfortunately, limitations of combining utilities became apparent in part due to a large blackout in the northeastern USA and southeastern Canada in 1965. As a result of this, it was realized that utilities need to have better visibility into their operations beyond what can be sensed in a control center. From this, SCADA was born. Regrettably, communications technologies deployed in the grid are largely unchanged from the 1960s, although they have been augmented in a piecemeal basis by newer networking technologies. Four decades later, electric grids still have inadequate communications; we are literally "flying blind" [1].

This lack of adequate situational awareness by power grid operators has been a leading contributor to power disturbances cascading into large blackouts (for example, the ones in the USA/Canada and Italy/Switzerland in 2003). Today, large grids generally have many sub-parts that do not coordinate their actions and have only very basic (and slow) communication between these sub-parts. This lack of situational awareness means that, as has happened in a number of blackouts in the last decade, one or two events such as a transmission line fault? can happen an hour or two before the blackout occurs, but nobody has anything close enough to the entire "big picture" of the grid and thus the significance of these events that ultimately start the blackout is not understood by any person or computer. Such informational disconnects are inevitable, given that, for example, the grids in North America have 3500 participants that can affect grid stability, including the small company First Energy that was at the core of the 2003 blackout in North America [2].

The rudimentary communications also means that opportunities for better protection, control, and efficiency are either impossible or far too expensive (typically with "one-off," hardcoded communications for each application family or even individual application). For more background, see [3], [4].

There are many chicken-egg problems involved with modernizing the power grid, including its data delivery infrastructure. In our opinion, the best way to solve many of these is to holistically and simultaneously consider power grids' dynamics and their data delivery infrastructure, both their steady states and those perturbed by a power contingency or a failure or cyber-attack involving the data delivery infrastructure. One key recent technology is that involving sensor data given microsecond-accurate timestamps, then delivered in real-time to give a coherent picture of a grid for operators, and soon for closed-loop control and broader protection [5].

In this paper, we offer such a holistic view of the power grid and the use of such sensors, involving applied electric power and computer science researchers who have been looking at this problem in this manner for more than a decade. We first overview fundamental problems in the power grid that can be mitigated by such coherent, real-time data: reliability, efficiency, and integration of renewable resources such as wind and solar. We then describe a wide range of applications utilizing such coherent, real-time data in order to mitigate these fundamental problems. We then normalize and summarize the communications requirements, including not just traditional quality of service (QoS) metrics such as latency and rate but also broader metrics, which we call "*QoS+*," including geographic scope, criticality, and amount of data. Next, we describe how these QoS+ requirements must necessarily impact the data delivery system of power grids, including absolute requirements and highly recommended implementation guidelines. As part of this, we also compare how existing network-level technologies, middleware, and power protocols map onto those requirements and guidelines. Finally, we overview the decade-long research into the GridStat data delivery system and of the emerging NASPInet concept that GridStat has influenced.

II. Problems and Goals

We now overview fundamental problems in today's power grids that can be mitigated by coherent real-time data.

A. Reliability

Evolving to a non-carbon based electrical infrastructure will require the handling of high penetrations of non-traditional generation sources, which behave differently from the vast base of existing generators. The smart grid will have to be able to utilize these intermittent sources of generation without compromising reliability and efficiency. The ability to control and operate the grid with more precision will make it possible to do so.

The reliability of the power grid is its ability to deliver electric power from generation to load without disruption. Thus the grid must be able to withstand minor and major disturbances without losing any customers. Interruption of electricity supply is not only inconvenient to the user but it affects the overall economy (productivity) of the region.

Of course, reliability is usually enhanced by adding redundancy into the grid and providing enough margin for the loading of the grid. On the other hand, operating the grid at much lower levels than the limits, introduces inefficiency as the transmission system is not fully utilized. Thus there is always some compromise between reliability and efficiency, both of which have to be optimized.

The addition of intelligent analysis and control helps reliability in two ways. As the power system gets loaded closer to its limits, the monitoring tools can alarm the operator to limit violations and the analysis tools can alert the operator when the system is vulnerable to contingencies. The importance of these real-time functions was demonstrated during the 2003 USA-Canada blackout in which the system was getting more vulnerable over several hours, but neither the alarming system nor the state estimator was working properly to alert the operator to deteriorating operating conditions.

The other way reliability can be helped is with control and protection schemes that can prevent the system from instability or collapse. Again, during the 2003 blackout, once the last contingency occurred, the cascading over several US states and one Canadian province happened too quickly for operator intervention. Under such circumstances, the only way to avoid such cascading would be to utilize fast control or protective schemes to isolate impacted areas and/or adjust some controllable values.

B. Efficiency

The efficiency of a power grid is its ability to minimize the cost of generation, which is facilitated by the transfer of large amounts of power while incurring the least losses in the transmission system. Because the transmission lines have limits, the maximizing of efficiency requires a constrained and non-linear optimization problem, which is done in the day-ahead hourly energy market as well as in real time.

The availability of fast controls in the grid also enhances the efficiency because these controls can prevent instability of the system, thus allowing higher rates of power transfer over the transmission system, which raises the efficiency.

C. Renewable Integration

The smart grid has to accommodate renewable resources and in fact, the increased level of sensing, measurement, and control can monitor the interruptible wind and solar resources and can quickly bring up backup generation to counteract the loss of wind or solar. At higher penetrations of wind and solar, there are other problems such as the fact that solar photovoltaics do not have any inertia making the system more unstable. The sensing and control in the smart grid can help this situation.

III. Solutions for Enhancing Generation and Transmission Based on Coherent Real-Time Data

Time-synchronized measurement devices are becoming a standard part of the power system and provide microsecond time accuracy using GPS-based clocks. Measurements called *synchrophasors* are increasingly being made in the power grid. Synchrophasors are measurements that represent both the magnitude and phase angle of a 60 Hz voltage or current waveform at a particular time, synchronized to a system-wide reference such as a Global Positioning System clock. A few years ago, synchrophasor technology was found only in stand-alone instruments called phasor measurement units (PMUs). Today, synchrophasor technology is also found in meters, protective relays, and fault recorders, which dramatically lowers the cost of implementing synchrophasor-based control and protection strategies. Station phasor data concentrators (PDCs), which gather synchrophasors from several sources within a substation, and distributed synchrophasor control devices are important new system components, providing distributed aggregation and control functions. Furthermore, new communication architectures, which include in-network

data concentration, real-time distribution, and fast fault recovery provide an infrastructure with the necessary high reliability.

This section discusses applications that can bring increased reliability, stability, and security to the entire power grid, *if* communications are adequate. For each application family, we summarize its communications aspects, which are expanded upon in the rest of this paper.

A. State Estimation and Measurement

Knowing the system state is an important first step for reliable control of the power system [6]. In the electric grid, the state of the system is the voltage and angle of every bus in the system. Schwepp introduced the first state estimation system [7]. In traditional state estimators, the state is estimated from voltage (no angles) and power flow measurements using iterative, nonlinear algorithms that do not always converge.

Fast state calculation is increasingly important for the quick response time requirements of control loops that are coupled to new dynamic renewable energy sources. Direct synchrophasor measurement of the state at some buses allows use of faster state estimation techniques to be used to estimate the state at other buses. Synchrophasor-based state estimation offers another advantage as well. State estimators must keep track of dynamic power system topology in order to correctly estimate the system state. Using traditional methods, all measurements are not taken at the same instant in time. Although voltage magnitudes change slowly with time, there is the possibility that the measurements taken could be from two different system configurations. For example, consider a case with a breaker opening. If some measurements are from before the breaker change and some are from after the breaker change, then effectively there are two completely different systems. A conventional state estimator might include both sets of measurements when attempting to solve the state and fail to converge. With time-synchronized measurements, the precise timestamps enable aligning all measurements, including contacts, disconnect switches, and tap changer values, so accurate system states can be calculated.

As will be discussed later, with new distributed synchrophasor control devices, once the state is calculated locally in a substation, it is easy to share across a wide-area network using the time-stamps resulting from the synchrophasor-based state calculation [8]. This improves power system reliability and meets the increasing reliability expectations from electric power customers.

We now describe a series of state estimation and measurement algorithms, building up from little to much required communications, that improve on the current practice.

Time-synchronized measurements also create the capability to calculate the state within substations and localized regions. Overall system state estimation is then a matter of aggregating and reconciling the local estimates. Figure 1 shows an example of how local coherent measurements can improve system reliability. (For simplicity, the PDCs are shown connected directly to the power system buses. In most systems, the PDC connects through a PMU to the bus.) Each PDC, located at the substation level, collects the voltages, currents, associated phase angles, and electrical topologies of the system as required by the state calculation engine in the PDC. The data are also exchanged between the PDCs so that the state is refined based on measurements from adjacent substations. Although the communications here is relatively hardcoded and limited, even this has benefits. The data exchange provides redundant communications paths to a state estimator in the event that the primary communications channel is temporarily lost. Should direct communications be interrupted, an adjacent PDC forwards the data.

Figure 2 shows a two-level state estimator [9] that more systematically exploits synchrophasor capabilities across a wide area. This estimator simplifies the total state estimation process by detecting and correcting topology and data errors early in the estimation process. It also can greatly lower the quantity of data that needs to be sent to a control center, with respect to sending all of the PMU data. This particular scheme uses only two levels, but there is no inherent reason that similar techniques could not be done for more levels of a hierarchy; e.g., substation, utility sub-region, utility, ISO/RTO, NERC.

The communications QoS aspects of these state estimation and measurement schemes can be summarized as follows. The inputs, consisting of power flows, voltage and current magnitudes, and phase angles, are sent periodically. The required latency for these inputs is fairly forgiving because there is only a person in the loop; tenths of seconds or even seconds is adequate, and a rate of a few Hz or less suffices. The inputs are at the substation and the criticality of the inputs depends on the application using the estimated state. The output goes to the applications that require the estimated state and has much the same communications QoS requirements as the inputs do.

B. Operator Displays

Operator displays are the primary window by which people monitor the operational state of the grid. Most existing operator displays update slowly based on data collected from a SCADA (Supervisory Control and Data Access) system every few seconds. These data are insufficient to reveal some crucial dynamic phenomena, such as oscillations, that indicate undesirable operating conditions. For example, with so much new renewable generation being connected to the power system, it is difficult to analyze the power system in sufficient detail to predict some of these oscillations, so detecting them when they occur is crucial. Unfortunately, oscillations might not be detectable from the slowly updating SCADA data. Presenting operators with results of analysis based on synchrophasor measurements made at much higher rates offers a remedy for this. Many systems have been described in the literature [10], [11].

The communication latency constraints for wide-area visualization are not strict: updates every few seconds are sufficient and displays can lag by a few seconds, given that there is a person in the loop. However, the quantity of data gathered with synchrophasor measurements is large because of the high sampling rate and because measurements must traverse an entire utility or ISO (wide area). Further, it is not

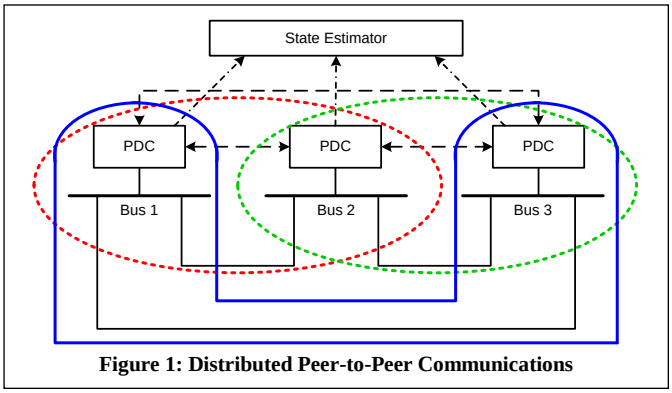

Figure 1: Distributed Peer-to-Peer Communications

critical that every measurement arrive; however, if there is a gap in communications, the operators may need up to the last several seconds retransmitted so they do not miss critical information during a fault or other problem. This kind of data transfer is different from some uses of sensor updates, where each update may be critical to deliver. We call this kind of transfer a bulk data transfer, and this application is labeled "Catch up for Operator Displays" in Table 2.

C. Distributed Wide-Area Control

Control of the power grid needs to be improved due to two major factors. First, the grid is inherently getting more stressed each year as increased demand and supply outstrips the addition of new long-distance transmission capabilities; there are more "miles times megawatts" being travelled each year. Second, renewable energy sources are much more variable, and their effect on the grid's stability much less known, than sources such as hydroelectric, coal, and nuclear with which operators and planners have greater experience.

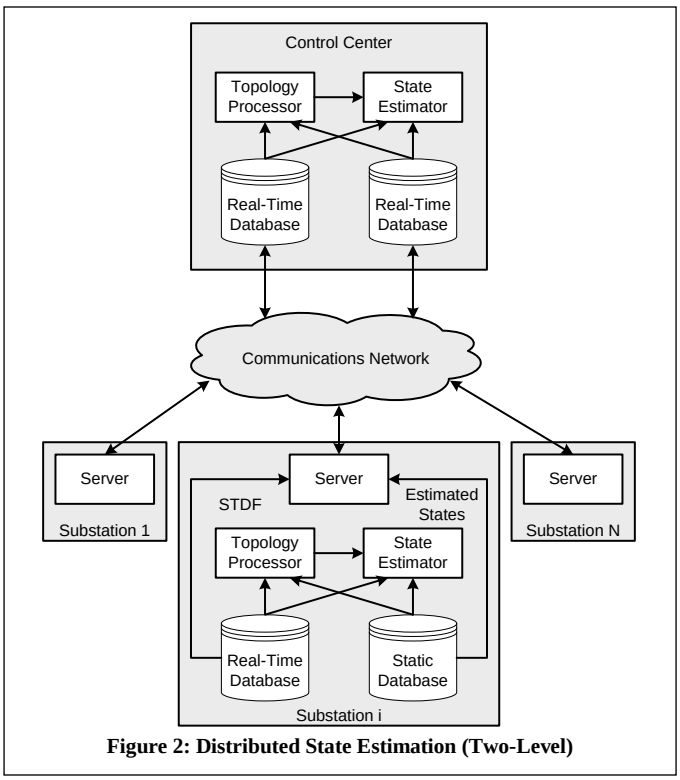

Figure 2: Distributed State Estimation (Two-Level)

This can be mitigated by moving from slower operator control towards more use of closed-loop feedback control. Southern California Edison has applied synchrophasors for wide-area dynamic voltage control by installing a PMU at the Big Creek Generation Station [12]. The PMU sends voltage measurements to the central control office where they are integrated into a static var compensator (SVC) controller. The SVC is located some distance away from the generation station. The SVC controller maintains local voltage at the SVC within proper operating range while simultaneously avoiding overvoltage at the generation station. The benefit of synchrophasors and PMUs for this system is not so much their time synchronization as it is their high speed and uniform sampling rate. The total measurement and communication latency requirement for this system was one second. This requirement was difficult to achieve with existing SCADA systems but relatively easy to achieve with time-synchronized phasors. Similar FACTS-based applications for HVDC systems and TCSC controllers are also proposed in [13].

Another control application is based on measuring power system modes—low-frequency electromechanical oscillations at characteristic frequencies. System disturbances, such as generation shedding or line tripping, can excite a mode. These oscillations become more pronounced when wind generation is added to the power system [14]. When oscillations are well damped, the system returns to a stable state after the disturbance; however, lightly damped or negatively damped oscillations cause instability. Clearly the power system is never operated in an unstable mode and is designed with large stability margins. The system topology, however, can change in unexpected ways during a disturbance, which can lead to an unstable system. Because of the power system size and the complexity of new renewable generation, it is difficult to predict all possible topologies, parameters, and associated modes. This limitation can be circumvented; however, synchrophasor technology unlocks the ability to directly calculate the frequency, magnitude, and damping factor of each power system mode.

For this family of applications, the input data are voltage and current. The allowable latencies for inputs vary from roughly 250 msec to a few seconds. The required rate for inputs varies, too: for voltage control it can be as slow as 1 Hz, while oscillation control may required 60 Hz. The input data delivery is critical, though missing data for up to a second does not cause problems given that this application family is handling

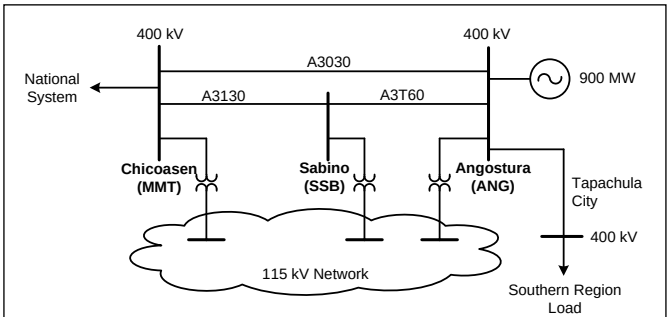

Figure 3: Block diagram of the CFE system. Synchrophasor measurements are taken at the Chicoasen and Angostura buses.

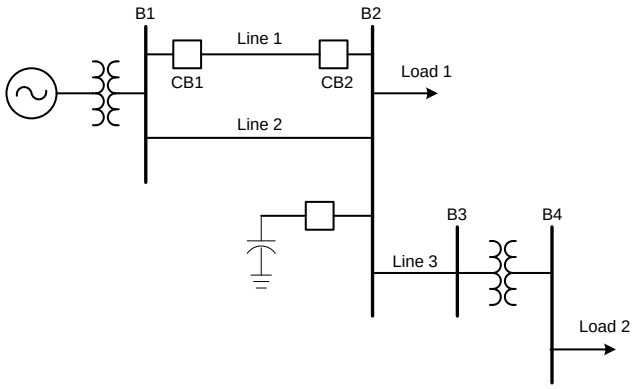

Figure 4: Power system model to demonstrate synchronous distributed control technology in real time.

fairly slow-moving phenomena. If input data are missing, there is no need to retransmit. The geography of inputs can vary widely, depending on the control scheme.

The output is a control signal to a reactive power controller or to a power system stabilizer, with latency requirements similar to those for the inputs. The rate of outputs can be slower than the inputs, because a control signal may only be needed every few seconds. However, a lower output rate makes the control loop slower so increasing it offers benefits in some configurations. The quantity of the output signals is small, though obviously goes up with increased output rates. The geographic scope of the outputs is similar to the inputs. Other possible control actions (outputs from the control application) include re-insertion of a transmission line, shedding generation, shedding load, or adjusting compensation devices such as shunt capacitors.

Distributed system integrity protection schemes are another class of wide-area control whose implementation is facilitated by ability to communicate synchrophasor data across the grid. A system integrity protection scheme (SIPS), also known as a remedial action scheme (RAS), provides the next level of protection after relays that respond to local power system emergencies [15]. One class of SIPS is contingency-based, where the scheme responds after a predefined event occurs, such as a breaker opening. Another SIPS design methodology is based on analog quantities such as frequency or voltage. With this design, the scheme responds if the frequency or voltage exceeds or drops below a threshold. For example, if an under-frequency condition develops, the system may shed load if the generation is unable to supply the required power.

A specific example of a synchrophasor-based RAS built to respond to angular instability is the Comisión Federal de Electridad (CFE, *México*) automatic generation shedding scheme as shown in Figure 3 [16]. The remote location of the Angostura generation station presents unique challenges for reliable system operation. If two of the lines between Angostura, Sabino, and Chicoasen are lost, the Angostura generators may experience angular instability. To prevent the 115 kV network from overloading, generation must be shed.

The RAS for this system was simplified using a relay with time-synchronized phasors at each end of the line to measure the angle and compare the difference against a threshold. This scheme requires a data exchange of 20 messages per second, which is easily met by a 19,200 baud fiber-optic serial connection between the relays.

Inputs to a SIPS include voltage and current, breaker status, and power. Its communications aspects can be extremely challenging. The input rate is the highest of all of the applications considered, and the latency must be very low. The criticality of its inputs (and outputs) is extremely high. For example, a SIPS might be installed in order to transfer more energy over a line than it can handle under all contingencies. Therefore, if a contingency happens it will have to respond by curtailing generation or load: if the SIPS fails, the contingency can cascade into a blackout [17]. The quantity and geographic scope for SIPS varies widely and is similar to that of distributed control.

Outputs from a SIPS are a condition-based control signal to initiate any of a number of actions to compensate for the contingency, e.g., tripping a breaker, generator or load. The outputs should be delivered with very low latencies. The criticality of the control actions is high though the quantity is low. The SIPS output control signals often have to be delivered less distance than the inputs because the SIPS logic tends to be located closer to the grid element that it is controlling.

Another emerging use of communication is to coordinate *synchronous distributed control actions*. Renewable generation is forcing more variability into the power system and one fundamental problem here is that when operators make system changes involving a number of compensatory control actions they do so by making one change at a time. This causes the grid to have unnecessary transient disturbances.

Synchronous distributed control [18] can reduce variability and keep the system stable. This technology uses the distributed time signal that is already available to relays in the power system and ties that time to specific operator commands. The commands are issued to the relays in advance of the anticipated operating time and validated for accuracy to ensure they have not been maliciously compromised. Then, at the preselected time, they execute in precise coordination.

Consider a traditional scenario for taking a line out of service. In Figure 4, Lines 1 and 2 are part of the transmission network. Line 3 connects the transmission and distribution networks. Bus B4 is a distribution bus. The transformer between buses B3 and B4 is a mechanical on-load tap change transformer. Consider what happens when Line 1 is taken out of service.

First an operator sends a command to open breakers CB1 and CB2, decreasing the voltage at Bus B2 due to the increased impedance from the generator through the remaining Line 2. As a result of this voltage decline, the transformer between B3 and B4 taps up to restore the distribution voltage to its target levels. If the transmission voltage at Bus B2 decays to a value below the desired minimum, the operators may insert the parallel capacitor into the system. This raises the transmission bus voltage but then requires the transformer to tap back down in order to avoid exceeding the distribution bus target voltage

levels. Figure 5 illustrates the system response to these changes.

These sequential operations result in unnecessary stress on the power system. Synchronous distributed control instead works as follows: the operator selects an appropriate *set* of commands to accomplish *all* of the desired changes; the commands are sent to a coordinator (e.g., a PDC) at each involved substation; the PDCs send appropriate subsets of the command list to Intelligent Electronic Devices (IEDs) to confirm that they are in states appropriate for carrying out the commands; after receiving confirmation from each IED that the sequences of commands are ready to run, the PDC indicates to the operator that the system is ready for initiation; the operator validates that all components are ready, no cyber security alarms have been received, and the change is still desired; the operator then arms the system and sends the start time to the PDC; the PDC and IEDs execute their commands at the preselected time.

This precision is possible only by using time-synchronized technology and results in minimal system impact as shown in Figure 6. The reduction in transients improves reliability and leaves additional margin for the uncontrollable dynamics of renewable sources.

The synchronized control system also improves overall safety. Local processing at the substations, for example in a PDC, will question a set of control commands that calls for vital actions, e.g., circuit-breaker tripping or reactive power insertion. Setting the control commands for a future time allows an interval when these commands can be re-analyzed for proper function. A distributed synchrophasor control device requests control validation from the system operations center or source of the synchronized commands. A local logic engine uses contingency analysis to validate the requested operation, such as confirming that opening a circuit breaker will not result in unacceptable voltage drop or even a voltage collapse. The system includes an alarm to alert the operator when a new series of controls is initiated. Only after validating the commands will the operator arm the system to execute at the desired time.

Yet another use of synchrophasor technology coupled with communications arises in *controlling renewable generation* itself. Two important cases are generation frequency control when distributed generation is islanded and generator isolation when islanded [19]. Figure 7 depicts a generation frequency control application in use by Abbott Pharmaceuticals that uses synchrophasor technology to control an islanded system [18]. When the plant and utility systems are connected, the grid controls system frequency and the governor controls generator power output. During an islanded state, the governor is switched to isochronous mode for frequency regulation.

When measurements at the two relays are not time-aligned, angle measurements are not available, so frequency data alone must be used to determine grid connection. The most difficult detection condition is when the island frequency and system frequency are nearly the same. Angle data helps disambiguate the situation in this case; furthermore, angle data is essential for safe reconnection after islanding occurs.

A similar detection challenge also faces a solar photovoltaic system (PV). Presently the IEEE 1547 Standard, "Interconnecting Distributed Resources With Electric Power Systems" specifies that the source must disconnect from a locally islanded system within two seconds. Such a requirement is important for safety reasons, quality of power, and out-of-phase reclosing avoidance. However, as the density of PV power increases, forced islanding reduces power system reliability. In the future, it will be important to keep PV online during power system events because the large quantity of generated power will help keep the system stable. For this reason, it is important to find better control methods that scale with the growing generation of PV power.

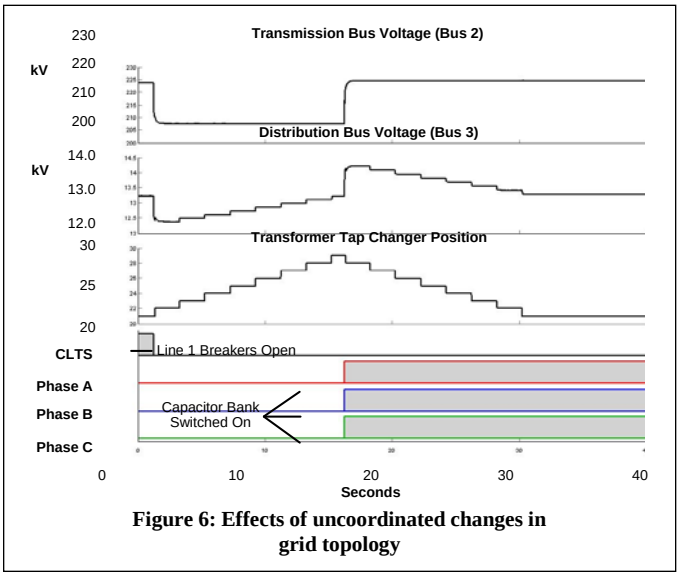

Figure 6: Effects of uncoordinated changes in grid topology

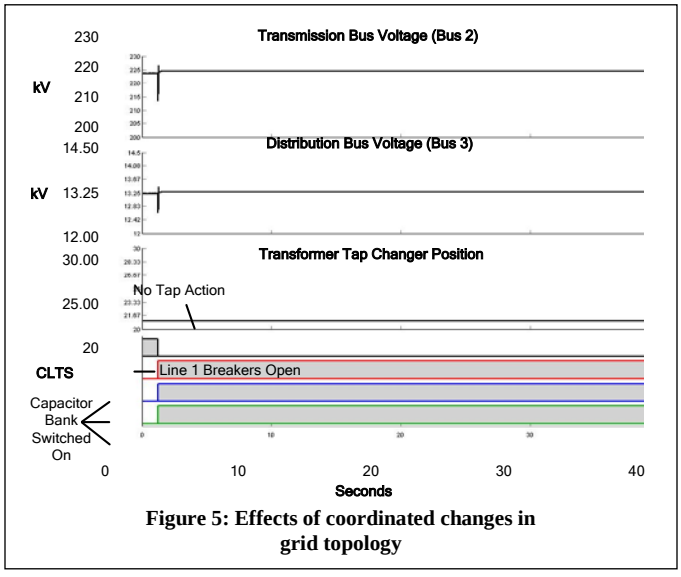

Figure 5: Effects of coordinated changes in grid topology

IEEE 1547 also requires disconnecting for sagging voltage under high demand. With a small amount of generation, this requirement is reasonable, but disconnecting a high-density solar generation source will cause the low-voltage condition to accelerate. Synchrophasors enable a wide-area view of the system and therefore enable solutions that can keep distributed generation, such as PV, online during transient conditions. Further, if the output control signals can be delivered with very low latency then reclosers can be made faster, which can greatly reduce power quality problems in a disturbance.

Most of the communications aspects for this islanding scheme, including inputs and outputs, are quite similar to those of SIPS, described above. The inputs can be considered to be very critical, but we note that often there will be local anti-islanding schemes deployed in case the communications fail. These local schemes are not as good because of their uncoordinated operation can cause power quality problems. Another strategy would be to automatically trip (disconnect) the distributed generation source if communications fail. This is reasonable if the amount of distributed generation (including renewable energy sources) is small.

Wide area distributed contols also have roles to play in improving *transient stability* and providing *ancillary services.*

Transient stability is a problem with many power systems in which the transfer limit on some transmission corridors are limited by the fact that short-circuits make the system unstable. Controls of various kinds – shedding load and/or generation, changing SVC or HVDC settings, etc. – are used to mitigate these instabilities, thus allowing higher limits on the transmission corridor. The main difficulty is that the instability occurs quite fast – within a second – thus requiring any control action to take place within 50 to 100 msec to maintain stability. Such fast controls are not possible without having a high bandwidth communication system.

Ancillary services is a catch-all term for services other than energy that are needed for the operation of the grid. The number and type of ancillary services tend to differ from region to region depending on how the local organizations decide to define these services for setting up markets, contracts and billing. In general, there are always some ancillary services for the provision of capacity reserves, for balancing generation with load, and for voltage control. The *load balancing service* requires a closed loop control that increases or decreases generator outputs to follow the changes in load. This closed loop control uses measurements of frequency, tie-line flows and generator levels as inputs every 2 to 4 seconds and sends output signals to the generators. *Voltage control* is often local but some regions are using area-wide voltage coordination, which requires communications similar to that of load balancing. The communications requirements for these services are modest and can be handled by present-day bandwidths connecting the control center. They establish the low end of the range of requirements for real-time control. Other roughly similar applications that have wide geographic scope include Dynamic Line Rating and Central Excitation Control [13].

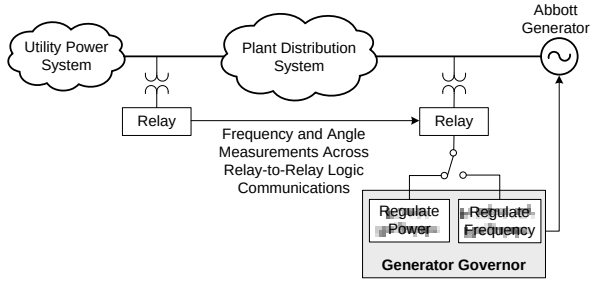

Figure 7: Frequency and angle measurements across relay-to-relay logic communications link.

D. Automated Contingency Drill-Down

Present best practices are for an operator to have a list of checks and actions prepared when a contingency is approached or reached. However, a data delivery system with the properties described later in this paper can greatly enhance the way that operators deal with an emerging or existing contingency. In responding, operators will seek additional data relevant for the situation, which may include seeking data at a higher rate or from different locations than those used in normal operation. As part of power contingency planning the data delivery system can be pre-configured to acquire and route the additional data to a response application [20]. The data delivery requirements are mostly similar to those of operator displays (though we do note that such drill-down techniques would be useful in conjunction with SIPS).

E. NASPInet Non-Realtime Delivery Applications

The control, protection, and visualization strategies presented earlier in this section all had requirements on the delivery of a single message. We conclude this section by noting two different classes of traffic that are planned for the future NASPInet [21], because of their implications for data delivery mechanisms as explained later in this paper. Both of these classes involve bulk transfer of data, compared to the rate-based sensor inputs or the rate-based or condition-based outputs described previously. They are similar in nature to the "Catch up" data for operator visualization described in Section III.B.

An *event* is a serious disturbance in the power grid, usually leading to a blackout at some scale. Utilities are required to log key sensor data in a database, typically called a historian, so regulatory authorities, such as NERC in North America, can ascertain the root cause of the problem. *Post-event data transfer*, then, involves transferring key related database entries for an event. The data messages of the transfer need not have any kind of latency guarantees, because the post-event analysis will be conducted offline. However, it is important to be able to transfer a reasonable amount of event data within a few hours or at most a few days. Of course, if the size of the required dataset is too large, it may not be possible to do this without interfering with important real-time data. But some communications resources must be reserved for this: arguably one of the most important post-event applications is *model validation*, which clearly must be supported.

Table 1: Normalized Values of QoS+ Parameters

Difficulty (5 : hardest)	Latency (msec)	Rate (Hz)	Criticality	Quantity	Geography	Deadline (for Bulk traffic)
5	5–20	240–720+	Ultra	Very High	Across grid or multiple ISOs	<5 sec.
4	20–50	120–240	Highly	High	Within an ISO/ RTO	1 min.
3	50–100	30–120	Medium	Medium	Between a few utilities	1 hr.
2	100–1000	1–30	Low	Low	Within a single utility	1 day
1	>1000	<1	Very Low	Very Low (serial)	Within a substation	>1 day

Another class of bulk transfer is for *research purposes*. Power researchers need access to access actual data in order to validate proposed new control and protection strategies. NASPInet will also be used for this background traffic. However, research traffic is handled on a best-effort basis when there is spare capacity (as will usually be the case if there are no power or IT disturbances). Unlike post-event data, there is no attempt to provide even soft guarantees on a completion time. However, barring power contingencies or IT anomalies, it is conceived that much useful research traffic will be supported.

IV. Power Application Requirements Mapped to Data Delivery Service Requirements

The power applications described in the previous section have a wide range of data delivery requirements in many dimensions. In this section we summarize those requirements to show the breadth of the requirement space and introduce the idea of a data delivery system for wide area measurement (henceforth WAMS-DD). This is key information that engineers who plan, build, and manage the data delivery infrastructure need to know in order to provide the required guarantees. Collecting and exploiting this information is not common practice in today's power grids; in our experience, there tends to be an assumption that the communications will be pervasive and data delivery will be "good enough."

A. Normalizing WAMS-DD Parameters

We now present these requirements in a qualitative form, normalized to indicate the level of difficulty, where 5 means most difficult and 1 means least challenging to provide. This enables comparison of different properties that have very different ranges, to get a sense of the wide ranges of difficulty or easiness involved for different power applications. It is important to note that a given application will not have all of its values in the same row: some requirements will be quite stringent (e.g., ultra-low latency) while others may be more forgiving (e.g., low volume of traffic).

Table 1 provides representative values of these data delivery requirements:

Latency: what latency is required for the delivery?

Rate: at what rate does/should the input be delivered, both now and in the future?

Criticality: how critical is this input [22], i.e., what is the severity of the consequences if data are not delivered for a short period of time?

Quantity: how much data needs to be delivered?

Geography: how far does the data have to travel?

Deadline: for bulk data transfer (defined shortly), when does the transfer have to be completed?

Some of these parameters are called *quality of service (QoS)* by networking researchers. We denote this entire collection, then, as *QoS+* to indicate that it includes other information needed in communications system design. QoS+ also refers to cyber-security issues, though these are beyond the scope of this paper.

B. Comparing WAMS-DD Parameters for Selected Power Applications

We now use the classes identified in Table 1 to summarize the QoS+ requirements for the power applications described in Section III. This is depicted in Table 2. The columns of this table are the different applications. The rows are the QoS+ attributes of the application's data delivery requirements along with three other kinds of information about the application:

Loop Entity: Where does the app's output go: a person (**P**); or a computer (**C**)?

Inputs and Outputs: for the inputs and the outputs is the data delivery: streaming sensor updates (**SS**); condition-based (**Co)** i.e., aperiodic events triggered by some condition; or bulk data transfer (**Bu**)? Note that the inputs and outputs for a given application can be different; for example, it can take in **SS** updates but only emit an output when those inputs show a certain condition (**Co**). Also note that **Co** and **Bu** inputs and outputs do not have a delivery rate and that a **Bu** input or output does not have a required latency (which in this table represents a per-message guarantee), but it has a (soft) deadline (which no other kind of data has).

NASPInet Class: what service class is this kind of traffic (see Section V.C.4).

It is crucial to observe that the requirements of even this small set of application families has great diversity. This means that the data delivery requirements are very broad, and many different kinds of traffic have to be managed in order for each

application to receive its required delivery guarantees. That is, this is exactly the opposite of "one size fits all" regarding data delivery!

Further, we note that the dynamics of the power grid can be affected by the dynamics of the data delivery. This is something that has rarely been studied (some examples we know of are [23], [24]), but needs to be developed as a best practice in the future. Otherwise, instabilities in the WAMS-DD may destabilize the power grid.

We now examine what these data delivery requirements are in greater detail, along with issues involved with implementing them.

V. Coherent Real-Time Data Delivery Enabling These Applications

Data deliver in the power grid today is, for the most part, hard-coded, tedious to implement and change, and does not provide any real end-to-end guarantees. The one exception to this is where isolated networks are used, for example for protection applications. Here, there are no real QoS mechanisms provided by the network, but rather massive over-provisioning provides low latencies and high availability (at least in the steady state, but not necessarily in the face of IT failures, bugs in software or hardware that cause spurious traffic, or cyber-attacks).

Table 2: Diversity of Data Delivery of Selected Power Applications

		Traditional State Estimation	Direct State Measurement	Operator Displays	Catch Up for Operator Displays	Distributed Wide-Area Control	Distribute SIPS	Synchronous Distributed Control	Renewable Gen. Islanding Control	Transient Stability	Ancillary Services	Automated Contingency Drill-Down	Post-Event (Analysis	Research
Paper Section		III.A	III.A	III.B	III.B	III.C	III.C	III.C	III.C	III.C	III.C	III.D	III.E	III.E
Loop Entity		P	P	P	P	C	C	C	C	C	C	P	P	P
Input[i] (most difficult input)	Kind	SS	SS	SS	Co	SS	SS	SS	SS	SS	SS	SS	Co	Co
	Lat.	1–2	1–2	1	1	2–4	4–5	2–4	2-3	5	1	1	1	1
	Rate	1–2	1–2	2–3	—	2–3	5	1-2	2–3	—	—	2–3	—	—
	Crit	1-5	1-5	1-5	1–2	5	5	5	4-5	5	1–3	5	1-5	1-5
	Quan	3–5	1–2	3–5	1–2	3–5	2–4	1-3	1-3	1–2	1–5	3–5	5	1-5
	Geog	5	1-5	5	5	1–5	1–5	1–5	2-3	4-5	3–5	3–4	3–5	3–5
	Dline	—	—	—	5	—	—	—	—	—	—	—	2–3	1
Output[j] (most difficult output)	Kind	SS	SS	SS	Bu	Co	Co	Co	Co	Co	SS	SS	Bu	Bu
	Lat.	1–2	1–2	1	—	3-5	5	3-5	3–5	5	1–2	1	—	—
	Rate	1–2	1–2	1	—	—	—	—	—	—	1–2	2–3+	—	—
	Crit	3	3	3	1–2	5	5	5	5	5	1–3	5	1–2?	1
	Quan	3–5	1–2	1	2–4	1-2	1	1	1	1	1	3–5	5	5
	Geog	1–2+	1–3+	1	1–2+	1–5	1–5	1–5	2-3	3–5	2	3–4	5	5
	Dline	—	—	—	5	—	—	—	—	—	—	—	2–3	1
NASPInet Class		—	B	D	—	B	A	A	A	A	A	D	C	E

This practice of using isolated networks will, in our opinion, soon become unsustainable as more applications, such as those outlined above, that can exploit coherent, real-time data delivery emerge. Further, so will the common practice today in the electricity industry of designing a new communication system for each new application or application family. Fortunately, the state of the art in distributed computing, real-time systems, and fault-tolerant computing does support providing strong guarantees with data delivered to many applications. If done right, a data delivery system leveraging this state of the art (and state of practice in other industries) can be a disruptive technology: greatly lowering the barrier to entry (in both time and money) to deploy new power applications. If done wrong, it will be something that will be discarded in 5 years because it cannot keep up with increasing demands. Further, these data delivery systems will have a long life, and no single network-level mechanism (for multicast or security or QoS) can be assumed to be everywhere. It is thus crucial that data delivery systems for the power grid's IT backbone have interoperability between different kinds of network mechanisms providing the same property such as delay guarantees [25].

In this section, we examine how WAMS-DD, which, if done right, will be an enabling technology for the new and emerging power. We first overview the requirements, which WAMS-DD must meet in the areas of performance and reliability. We then present implementation guidelines, based on best practices in other industries and in the field of distributed computing systems, that we believe must be followed in order to achieve these requirements in WAMS-DD. Next we compare how existing technologies meet these delivery requirements and design guidelines. This includes technologies and standards at the network layers (and below), the middleware layer(s), and related ones from the power industry. We also discuss relevant R&D for wide-area middleware. After this, we discuss the emerging NASPInet effort and our decade-long GridStat data delivery middleware, which meets these delivery requirements and design guidelines. Finally, we conclude this section with a brief discussion of pertinent cyber-security issues for next-generation data delivery services for the electric power grid.

A. Requirements for WAMS-DD

We now overview *delivery requirements* (DR) that WAMS-DD must meet [4], [21], not including the details of cyber-security related ones.

Requirement 1. *Hard, end-to-end (E2E) guarantees* must be provided over an entire grid. If the guarantees are soft or non-existent, then it is foolish to build protection and control applications that depend on the data delivery. The guarantees must thus be deterministic: met unless the system's design criteria have been violated (e.g., traffic amount, number of failures, and severity of cyber-attack).

Requirement 2. WAMS-DD *must* have a *long-lifetime* and thus be designed with future-proofing in mind. This is crucial in order to have its costs amortized over many projects, utilities, grids, etc. The goal of NASPInet, for example, is to last at least 30 years.

Requirement 3. *Multicast* (one-to-many) is the normal mode of communications, not point-to-point. Increasingly, a given sensor value is needed by multiple power applications.

Requirement 4. End-to-end guarantees must be provided for a *wide range* of QoS+. Data delivery for the grid is not "one size fits all" [22], as shown in Section IV. For example, to provide very low latencies, very high rates, and very high criticality/availability to all applications would be prohibitively expensive. Fortunately, many applications do not require these stringent guarantees, but their less stringent requirements must of course be met.

Requirement 5. Some merging and future SIPS, transient stability, and control applications require *ultra-low latencies*, delivered (one-way) on the order of a half or full power cycle (8-16 msec in the US) over hundreds of miles and possibly across most of a grid [15]. Thus, any forwarding protocols should not add more than a millisecond or two of latency (through all forwarding hops) on top of the speed of light in the underlying communication medium, which is roughly 100 miles/msec.

These latencies must be provided in a way that:

A. is *predictable*, and *guaranteed for <u>each</u> update message*, not a (much weaker) aggregate guarantee over longer periods of time, applications, and locations such as provided by multiprotocol label switching (MPLS) technology [26]. Each sensor update needs to arrive within its required guaranteed deadline. As we will see below, virtually all technologies widely deployed in today's best effort internet do not provide such per-packet guarantees. (ATM is a notable exception, but it does not provide multicast, and it is not a realistic end-to-end solution for the entire grid for other reasons.)

B. *tolerates* (non-malicious) *failures* in the WAMS-DD infrastructure. No system can tolerate unlimited kinds and numbers of failures. However, much like the power grid must continue in the face of one or more knowable contingencies, the IT infrastructure on which it increasingly depends must still provide these hard, end-to-end guarantees in the face of failures (up to design limits).

C. *tolerates* (malicious) *cyber-attacks*. Power grids are known to be subjects of extensive study and probing by multiple organizations that have significant information warfare capabilities, including nation states, terrorist organizations, and organized crime. WAMS-DD must adapt and continue to deliver data despite cyber-attacks of a designed severity (a bar, which should be designed to be increasable over the life of the system). Note that a bug in hardware or software that generations spurious traffic can have an effect similar to that of a cyber-attack.

Requirement 6. Extremely *high throughput* is required. Today's synchrophasor applications are generally limited to 30 or 60 Hz in the USA, largely because the communications systems they use are not designed to support higher rates. To not provide much higher sustainable throughput would greatly limit the number of

new applications that can help the grid's stability. Indeed, not just synchrophasors but digital fault recorders (DFR) and IEDs in substations provide a wealth of data, which is not tapped today. It is quite conceivable, and arguably likely, that "If you build it, they will come" and there will be many thousands of synchrophasors, DFRs, and other sources of sensor updates across a grid. Indeed, DFRs output (today only to disk) at 720 Hz and typically sample at 8 kHz, but their output is presently not used remotely due to communications limitations. If key DFR data could be delivered from a set of DFRs across a grid at 720 Hz many new opportunities open up for transient protection without using expensive dedicated networks, or for "drilling down" into the root causes of an ongoing power contingency using additional contingency-specific data, as described in Section D.

These six requirements must all be met if power engineers are to justify depending on data delivery. It is crucial that they be able to do so, however, given that the it is almost universally accepted that the grid is inherently getting less stable each year —for example, due to renewable energy sources (which have very different power characteristics) and load outstripping transmission construction—and there are emerging applications, which can help mitigate this. Providing the above capabilities in WAMS-DD can enable a much wider array of power applications to be deployed, with less time and cost, than if the electricity sector continues with "business as usual" on the data delivery front, or applies technology that will not meet these requirements.

We are not aware of any commercial or military market for a wide area data delivery infrastructure that either has such stringent requirements or exploits aspects of a WAMS-DD including ability to enforce complete perimeter control, ability to know the vast majority of the traffic ahead of time, much fewer kinds of traffic, and other factors incorporated into the implementation guidelines described next in this paper. The reason is quite simple: there is no other market for such stringent requirements, and so, predictably, vendors have not over-designed their products to meet these difficult requirements. In our opinion, however, these requirements are quite achievable, based on state-of-the-art in distributed real-time embedded (DRE) computing as long as a careful end-to-end analysis is done [27], and the core data delivery mechanisms are not saddled with unnecessary features.

B. Implementation guidelines for WAMS-DD

The requirements outlined in the previous subsection were kept to a bare minimum. In order to achieve them, however, we believe it will be necessary to utilize a number of *implementation guidelines* (IG), many which are quite different from what is provided in today's best-effort Internet and what has been the conventional wisdom in networking research. In this section we enumerate and explain these IGs.

Some of the IGs below (e.g., IG4 and IG5) are actually deemed requirements for NASPInet [21], but we describe them here as IGs because it is possible to build WAMS-DD without them (though we believe that would be inadvisable for an inter-utility backbone such as the proposed NASPInet). These IGs are drawn from a number or sources, including our knowledge of what the state of the art in distributed computing has demonstrated is feasible, best practices in other industries, and decades of experience gained in DARPA wide-area application and middleware projects of ours and others.

We note that these guidelines refer to best practices of how to ***build*** a WAMS-DD. Other guidelines (beyond the scope of this paper) will apply on how to ***use*** one and will need to be developed as best practices. For example, in our experience, many power engineers assume that with synchrophasors, they should have the phasor data concentrators (PDCs) inside their utility. However, this is a very bad idea for updates that need to be delivered with ultra-low latencies (DR5). A PDC aggregates many PMU signals, does error correction and angle computation, then outputs the collection of this information for a given PMU time slot (this is called *time alignment*). Such a PDC may have dozens of PMU signals coming into it, so doing time alignment means that the output has to wait until the slowest PMU update arrives. In this case, the updated sensor values will have suffered significant delays even before they leave the utility to be transported by a wide-area WAMS-DD such as NASPInet. Thus, for those updates that require ultra-low delivery latency, any PDC or other time alignment should be placed as close to the subscribers as possible (ideally in their local area network), even at the cost of a small amount of either wasted bandwidth and duplication of PDCs. Similarly, data that is required with extremely low latency should not have a database in its path: it can be entered into a database after it is sent out, but the database must not slow down the fast delivery path.

We also note that the scope of these IGs involves only the data delivery system for WAMS-DD. It does not include the supporting services that will be required for configuration, security, path allocation, resource management, etc. It will be important for the WAMS-DD that the use of these tools avoids hard-coding choices, but rather allows them to be specified in a high-level policy language (or at least a database) [28], [29], [30]. For an example of a hierarchical version of such services (a "management plane"), see [31], [32].

Table 3 provides an overview of the IGs and the DRs that require the given IG. We now explain each of the IGs in turn.

Table 3: Implementation Guidelines and the Delivery Requirements that Mandate Them

DR1: Hard E2E WAN guarantees	DR2: Future-Proofing	DR3: Multicast	DR4: Wide Range of QoS+…	4A: Latency & Rate	4B: Criticality/Afailability	4C: Cyber-Security	DR5: Ultra-Low Latencies…	5A: Per-Update & Predictable	5B: Tolerating failures	5C: Tolerating Cyber-Attacks	DR6: High Throughput	IG*x* Prerequisites	Summary of Implementation Guideline IG*x*
X								X	X				IG1: Avoid post-error recovery mechanisms
X				X				X	X	X	X		IG2: Optimize for Rate-Based Sensors
		X									X		IG3: Provide Per-Subscriber QoS+
		X									X		IG4: Provide efficient multicast
												2,3	IG5: Provide Synchronized Rate Down-Sampling
X					X			X			X		IG6: Don't depend on priority-based "guarantees"
X	X			X	X	X							IG7: Provide end-to-end interoperability across different/new IT technologies (multicast, QoS+)
X								X			X		IG8: Exploit *a priori* knowledge of traffic
X								X	X	X		8	IG9: Have systematic, quick internal instrumentation
X								X					IG10: Exploit smaller scale of the WAMS-DD
X								X				8-10	IG11: Use static, not dynamic, routing
X								X	X	X			IG12: Enforce complete perimeter control
X								X	X	X	X	12	IG13: Reject unauth. messages quickly & locally
											X	2,8	IG14: Provide only simple subscription criteria
								X			X	2	IG15: Support transient, not persistent, delivery
								X			X		IG16: Don't over-design consistency & (re)ordering
											X	2,8,14-16	IG17: Minimize forwarding-time logic
X	X			X	X	X							IG18: Support multiple QoS+ mechanisms for different operating conditions
								X			X	17	IG19: Inspect only message header, not payload
X								X			X		IG20: Manage aperiodic traffic

Guideline 1. *Avoid post-error recovery mechanisms.* Traditional protocols for the internet in general and reliable multicast protocols from the fault-tolerant computing research community use post-error recovery. In these protocols the receiver either sends a positive acknowledgement (ACK) when it receives a message, or it sends a negative acknowledgment (NACK) when it concludes that the message will not arrive. However, both add considerable latency when a message[1] gets dropped: three one-way latencies are required plus a relatively large timeout. The better alternative is to send sensor updates (messages) proactively over multiple disjoint paths, each of which meets the latency and rate requirements [33], [34]. Indeed, if multiple independent messages, each going over a QoS-managed path, cannot meet the delivery deadline, then sending ACKs or NACKs is very unlikely to help, and indeed will only make things worse.

[1] We use the term "message" rather than "packet," because in many cases we are describing middleware-layer mechanisms above the network and transport layers.

Note that the guideline to avoid post-error correction is only for data that has guarantees on a per-message basis. Bulk data transfer is similar to a remote file transfer and will almost certainly employ post-error correction. However, those mechanisms must be different from the ones that have to provide per-message guarantees.

Guideline 2. *Optimize for rate-based sensors.* WAMS-DD can be made with higher throughput and robustness if they are not over-engineered. General-purpose publish-subscribe systems offer a wide range of traffic types, because they are designed to support a wide range of applications. However, in a WAMS-DD, the vast majority of the traffic will be rate-based. Design accordingly.

Guideline 3. *Provide per-subscriber QoS+*. It is crucial that different subscribers to the same sensor variable be able to have different guarantees in terms of latency, rate, and criticality/availability. If not, then a lot of bandwidth will be wasted: all subscribers will have to be delivered that sensor's updates at the most stringent QoS+ that any of its subscribers requires.

Guideline 4. *Provide efficient multicast.* In order to achieve the highest throughput possible, it is imperative to avoid unnecessary network traffic. Thus, never send an update over a link more than once. Also, as a sensor update is being forwarded through the network, if it is not needed downstream in the multicast tree (e.g., those subscribers require it at a lower rate than other subscribers), the update message should be dropped. This can best be implemented using a *rate down-sampling* mechanism as is done in GridStat [33], [34].

These first four guidelines add up to a need for multicast routing heuristics that provide multiple, disjoint paths to each subscriber with each path meeting the subscriber's latency requirement. A family of heuristics developed for this multi-cast routing problem [35], [36] confirms the feasibility of the approach at the anticipated scale (see IG10) if routing decisions are made statically (IG11).

Guideline 5. *Provide synchronized rate down-sampling*. In providing rate down-sampling, it is import to not down-sample in a way that destroys the usefulness of some data. For example, synchrophasors are used to take a direct state measurement at a given microsecond. If some subscribers require only a small fraction of the updates for a set of synchrophasor sensors, it is important that the updates that reach the subscriber at each interval carry the same timestamp. For example, if a subscriber only requires a tenth of the updates from two different variables, then it would not be useable to get updates {#1, #11, #21, …} from one synchrophasor and updates {#2, #12, #22} from another synchrophasor, because the given measurements (e.g., #1 vs #2) do correspond to the same time (they are not the same snapshot), which is the main point of synchrophasors.

Guideline 6. *Don't depend on priority-based "guarantees"*. Publish-subscribe delivery systems typically offer a way to specify a priority, so if the traffic gets too heavy less important traffic can be dropped. However, this does not provide a hard end-to-end guarantee to subscribing applications, and even applications that are not of the highest criticality still need their DRs to be met. Instead of priorities, mechanisms must be used that exploit the characteristics of WAMS-DD (as outlined in these guidelines) to provide each subscriber firm assurances that its guarantees will be met so long as there are not more than the agreed upon number of failures or severity of cyber-attack.

Guideline 7. *Provide end-to-end interoperability across different/new IT technologies (providing multicast, latency, rate, etc).* A grid-wide WAMS-DD will *ipso facto* have to span many utility and network organizations. It is unlikely that the same mechanisms will be present across all these organizations. And, even if they are today, if the WAMS-DD gets locked into the lower-level APIs and semantics of a given multicast or QoS mechanism, it will be difficult to "ride the technology curve" and utilize newer and better mechanisms that will inevitably become available over the long lifetime of the WAMS-DD. This is a stated goal of the GridWise community, for example [37]. Fortunately, it is possible to use middleware to span these different underlying technologies in order to provide guarantees that span this underlying diversity.

Guideline 8. *Exploit a priori knowledge of predictable traffic.* Internet routers cannot make assumptions or optimizations based on the characteristics of the traffic that they will be subjected to, because they are intended to be general-purpose and support a wide range of traffic types. WAMS-DD, however, have traffic that is not just rate-based, but is almost all known months ahead of time (e.g., when an engineering survey is made of a new power application). This common case can be optimized, as described in later IGs below.

Guideline 9. *Have systematic, quick internal instrumentation.* In order to provide end-to-end guarantees across a wide area despite failures and cyber-attacks, IG8 must be exploited to provide systematic and fast instrumentation of the WAMS-DD. This allows much quicker adaptations to anomalous traffic, whether accidental or malicious in origin.

Guideline 10. *Exploit smaller scale of the WAMS-DD.* This is a crucial if the challenging delivery requirements are to be met over a wide area with reasonable cost. However, this requires rethinking the conventional wisdom in networking research and commercial middleware products.

NnDB will be orders of magnitude smaller in scale than the Internet at large[2], so it is feasible for the entire configuration to be stored in one location for the purposes of (mostly offline) route selection. Additionally, academic computer science researchers historically consider something that is $O(N^2)$ for path calculation with N routers or forwarding engines to be infeasible; see for example [32]. However, this assumption ignores two key factors for NnDB. First, N is not in the neighborhood of 10^8 as in the Internet, but rather is more likely $\sim 10^3$ at least for the next 5-10 years; this is Even $O(N^2)$ algorithms are feasible at this scale. Second, as a rule, power engineers do not decide that they need a given sensor's values seconds before they really need it, due in part to the fact that today's data delivery infrastructure requires them to recode hard-coded socket programs and then recompile. Rather, power engineers plan their power contingencies (and what data they will need in them) months ahead of time with detailed engineering studies, and similarly for their monitoring, protection, control, and visualization needs. Thus, the routing/forwarding decisions involved in path selection can be done offline well ahead of time, while still allowing for handling a modest number of subscription requests at runtime.

[2] For example, in the entire USA there are approx 3500 companies that participate in the grid [2]. We thus believe that the number of router-like forwarding engines that would be required for a NnDB backbone (at least in the case of broker-based publish-subscribe; defined later) is at most 10^4 and likely only around 10^3.

It is also feasible for router-like forwarding engines to store state for each flow. Having a router keep per-flow state has long been considered a bane to networking researchers, because it is considered to be prohibitively unscalable. However, with the much smaller scale, and the much more limited type of applications for a WAMS-DD, storing per-flow state is not only feasible but it is a requirement for providing IG3 (per-subscriber QoS+) with IG4 (efficient multicast); this is something that our GridStat project has been advocating for many years [31]. However, recently networking researchers are realizing the necessity of storing per-flow state to provide any reasonable kind of QoS [38]. Other recent efforts with roughly similar approaches include as CHART [39] and PHAROS [40].

Guideline 11. *Use static, not dynamic routing and naming.* Much stronger latency guarantees can be provided when using complete knowledge of topology coupled with static routing. Complete topology knowledge is a reasonable assumption in a managed NnDB, given that it will be a carefully managed critical infrastructure with complete admission control. Also, almost all of the sensors and power applications will be known well ahead of time, so optimizations for static (or slowly-changing) naming can potentially be useful and can be done while still providing more flexible and dynamic discovery services at a much lower volume. We note that networking and security researchers generally assume that the membership of multicast groups (or a set of subscribers) may change rapidly; see for example [32]. However, as noted above, that is not the case with WAMS-DD.

Guideline 12. *Enforce complete perimeter control*. All traffic put onto a WAMS-DD must pass admission control criteria (permissions based on both security and resource management) via a management system: the publisher registering a sensor variable (at a given rate) and the subscribers asking for a subscription with a given rate and end-to-end latency. This is essential to provide guarantees at a per-message granularity. It also enables quicker adaptations.

Guideline 13. *Reject unauthorized messages quickly and locally*. Messages that have gone around the admission control perimeter should be rejected as soon as possible, ideally at the next WAMS-DD forwarding engine, rather than going most or all the way across the WAMS-DD consuming resources along the way. Detection of such unauthorized packets is an indicator of anomalous traffic and evidence of a failure or cyber-attack that needs to be reported to the management infrastructure. When sufficient evidence over sufficient time is collected, an appropriate adaptation can occur.

Guideline 14. *Provide only simple subscription criteria.* This is exactly the opposite of what is usually done with general purpose publish-subscribe in either academic research or commercial products: both tend to favor complex subscription criteria, which are expensive to evaluate as each update is forwarded through the system (think of complex "topics") [41]. For example, in GridStat, the subscription criteria are latency, rate, and number of paths, and, as noted below, the forwarding decision is done completely based on rate, with static routing. Note also that the lower-level ID of a sensor variable could still be looked up through a complicated discovery service; this guideline is concerned with avoiding complex forwarding logic.

Guideline 15. *Support only transient delivery, not persistent delivery*. Most publish-subscribe systems offer persistent delivery, whereby if an event cannot be immediately forwarded, it is stored for some time and then the delivery retried. This harms throughput, however, as well as potentially the per-packet predictability (because it requires storing the data). In our experience, it is completely unnecessary for real-time visualization, control, and protection, due to the temporal redundancy inherent in rate-based update streams; the next update will be arriving very soon anyway, so the usefulness of a given update fades very quickly. Thus, it is inadvisable to complicate delivery mechanisms to support persistent delivery (though it can be provided "on the side" by other mechanisms). Furthermore, in the power grid, historian databases are already required for archiving data, so there is no reason to complicate the design or otherwise bog down the fastest and highest availability mechanisms of WAMS-DD to deliver historical data[3].

Guideline 16. *Don't over-design for consistency and (re)ordering*. Research in fault-tolerant multicast tends to provide different levels of ordering between updates from the same publisher, or between different clients of the same server, as well as consistency levels between different replicas or caches of a server. There is no need for anything like this in a WAMS-DD, Present data delivery software provides no kind of consistency at all, so power applications assume nothing in terms of consistency and ordering. The only requirement for such consistency that we have found is reflected in IG5 for synchrophasors, and the only ordering of any kind is where a PDC combines updates from different PMUs into one message to pass onwards. With devices such as synchrophasors that have accurate GPS clocks the order of events can be directly known and no delivery ordering mechanism is required other than that which is done by a PDC.

Guideline 17. *Minimize forwarding-time logic*. In order to provide the highest throughput, the forwarding logic that decides how a packet or update is to be forwarded on should be kept as simple as possible. On the GridStat project, forwarding decisions are made based solely on the subscription rate of subscribers downstream in the multicast tree [31], [34]. Given that the traffic is rate-based (IG2) and known ahead of time (IG8), and that subscription criteria are kept simple (IG14), and only transient delivery is supported (IG15), and that there are

[3] We note that such post-event historical data can be delivered by the same network links as the fast traffic with traffic isolation mechanisms; indeed, this is one of the main traffic categories for the emerging NASPInet.

no consistency semantics (IG 16), much logic can be pushed off to subscription setup time or even offline. This reduces the logic necessary when an update arrives at a forwarding engine (or P2P middleware mechanisms at an edge) and hence greatly increases throughput and decreases latency.

Guideline 18. *Support multiple QoS+ mechanisms for different runtime conditions.* A given mechanism that provides guarantees of latency and security, for example, will not be appropriate for all the runtime operating conditions in which a long-lived WAMS-DD may have to operate. This is because different implementations of a given QoS+ mechanism can require very different amounts of lower-level resources such as CPU, memory, and bandwidth [42].

Guideline 19. *Inspect only packet header, not payload*. In order to provide the highest throughput and lowest latency, ensure that subscription criteria and consistency semantics allow a forwarding decision to be based solely on a packet header. This is not possible for publish-subscribe middleware that has complicated subscription topics as is typical with commercial and research systems. For them, data fields in the payload also have to be inspected.

Guideline 20. *Manage aperiodic traffic*. Any traffic that is aperiodic (i.e., not based on rate but on a condition) must be isolated from rate-based periodic traffic and managed accordingly. This can be done deterministically, for example with (OSI Layer 1) optical wave division multiplexing (OWDM) hardware. Further, aperiodic traffic should be aggregated intelligently—ideally based on updateable policies rather than hardcoded settings—instead of sending all alarms/alerts to the next level up for processing.

It is important to recognize that you can't have the highest level of all the properties described in the Design Requirements for every sensor variable. As noted in [25]:

1. Different properties inherently must be traded off against others.
2. Different mechanisms for a given property are appropriate for only some of the runtime operating conditions that an application may encounter (especially a long-lived one).
3. Different mechanisms for the same non-functional property can have different tradeoffs of lower-level resources (CPU, bandwidth, storage)
4. Mechanisms most often can't be combined in arbitrary ways

Even if you somehow could have them all at once, it would be prohibitively expensive. Given these realities, and the fact application programmers rarely can be expert in dealing with the above issues, middleware with QoS+ properties supported in a comprehensive and coherent way is a way to package up the handling of these issues and allow reuse across application families, organizations, and even industries.

Similarly, it is important to note that meeting IG3 (and others) requires the data delivery system to be provided at the middleware layer. This is because network-level mechanisms know about packets and IP addresses, not middleware-layer sensor variables and the power applications that subscribe to their updates. There is thus no way that network-level mechanisms can provide different subscribers to the same sensor variable with different QoS+ guarantees, which is mandated by efficient multicast (IG4).

Finally, because of length constraints it is not possible in this paper to fully discuss the cyber-security issues that arise in a WAMS-DD. Clearly, a WAMS-DD providing universal connectivity creates cyber-security challenges beyond those arising in a conventional, single-utility SCADA system. Cyber-security also interacts with DRs and IGs: for example, techniques used for message confidentiality and authentication must not impose too much additional latency, yet the multicast requirement appears to limit use of symmetric-key cryptography for authentication.

C. Analysis of existing technologies for WAMS-DD

We now analyze how existing technologies and standards meet the above delivery requirements and design guidelines. Table 4 summarizes this coverage, which we explain next. The columns of this table are existing networking and middleware technologies, while the rows are the DRs and IGs outlined previously in this section. The table cells denote how well the given technology meets the given IG or DR. These have the following values: '**Y**': yes; '—': no; '**S**': some; '**L**': likely (but not confirmed); '**?**': unknown; 'D': doubtful (but not confirmed); 'F': future plans (architected for this); '**φ**': Not applicable (and does not provide).

We note that some of the values are not confirmed, because it is extremely difficult to glean detailed information about whether or not a commercial product provides a given DR or IG. In these cases, if we believe that it does or does not (for example, based on its intended domain), this is indicated by 'L' or 'D', respectively

1) Technologies and standards at the traditional network layers

Traditional network protocols, including the OSI-2 ("Data Link") layer (e.g., Ethernet), OSI-3 ("network") layer (e.g., IP) and the OSI-4 ("transport") layer (e.g., TCP, UDP, SCTP) do not provide any kind of end-to-end QoS+ guarantees or multicast. We will examine extensions of these layers to see how they meet the requirements and guidelines. We do not consider experimental or emerging network technologies such as CHART [39], PHAROS [40], and Anagram's Flow routers [38]. Such technologies may some day be helpful in providing QoS guarantees across parts of a WAMS-DD. However, they are unlikely to cover a significant portion of such infrastructures for a decade or more.

Table 4: Coverage of Delivery Requirements and Implementation Guidelines by Existing Technologies

IP	TCP, UDP, SCTP	IPv6 Flow Labels	IP Multicast	MPLS	VLANs & VPNs	ATM	SOSCOE [43]	BB COTS Pub-sub	P2P COTS Pub-Sub	Streaming SQL/CEP	Military RT Apps	SOA/Web Services	IEC 61850, OPC UA	DNP3, MMS	GridStat	Delivery Requirement or Design Guideline
—	—	?	—	—	—	Y	?	D	D	—	D	—	—	—	Y	DR1: Hard E2E WAN guarantees
—	—	S	—	—	—	—	Y	Y	Y	D	Y	Y	—	—	Y	DR2: Future-Proofing
—	—	?	Y	Y	?	—	Y	Y	Y	D	Y	S	—	—	Y	DR3: Multicast
																DR4: Wide Range of QoS+:
—	—	—	—	Y	?	Y	?	Y	Y	—	Y	—	—	—	Y	4A: Latency & Rate
—	—	—	—	—	—	—	L	Y	Y	—	Y	—	—	—	Y	4B: Criticality/Availability
S	S	—	S	S	?	—	L	Y	Y	—	Y	S	—	—	D	4C: Cyber-Security
																DR5: Ultra-Low Latencies:
—	—	—	—	—	—	Y	D	D	D	—	D	—	—	—	Y	5A: Per-message & predictable
φ	φ	φ	φ	φ	φ	?	D	D	—	—	–	S	—	—	Y	5B: Tolerating failures
φ	φ	φ	φ	φ	φ	—	D	?	?	D	S	—	—	—	F	5C: Tolerating Cyber-Attacks
Y	Y	Y	Y	Y	?	Y	D	Y	Y	Y	Y	—	—	—	Y	DR6: High Throughput
Y	—	φ	—	—	—	Y	D	?	—	D	?	—	—	—	Y	IG1: Avoid post-error recovery mechanisms
—	—	—	—	—	—	—	D	?	?	D	?	—	—	—	Y	IG2: Optimize for Rate-Based Sensors
φ	φ	—	—	D	—	—	?	?	?	?	?	D	—	—	Y	IG3: Provide Per-Subscriber QoS+
φ	φ	φ	Y	?	—	—	D	?	D	D	?	—	—	—	Y	IG4: Provide efficient multicast
—	—	—	—	—	—	—	D	—	—	D	–	—	—	—	Y	IG5: Provide Synch'd Rate Down-Sampling
φ	φ	—	φ	—	—	—	D	?	?	D	?	—	—	—	Y	IG6: Don't count on priority "guarantees"
—	—	—	—	—	—	—	L	Y	Y	S	Y	Y	—	—	Y	IG7: Provide E2E interoperability across diff./ new IT technologies (multicast, QoS+)
—	—	?	—	?	—	—	D	?	?	L	?	—	—	—	Y	IG8: Exploit *a priori* knowledge of traffic
—	—	φ	—	?	—	—	?	?	?	D	P	—	—	—	F	IG9: Have systematic, quick internal instrumentation
—	—	—	—	—	—	—	D	?	?	D	?	—	—	—	Y	IG10: Exploit smaller scale of the WAMS-DD
—	—	—	—	—	—	—	D	?	?	D	?	—	—	—	Y	IG11: Use static, not dynamic, routing
—	—	—	—		—	—	D	—	—	D	–	—	—	—	—	IG12: Enforce complete perimeter control
—	—	?	—		—	—	?	?	?	D	?	S	—	—	S	IG13: Reject unauth. packets quickly & locally
—	—	Y	—	—	—	—	D	D	D	—	D	—	—	—	Y	IG14: Provide only simple subscription criteria
Y	—	Y	Y	—	—	—	?			D		—	—	—	Y	IG15: Support transient, not persist., delivery
Y	Y	Y	Y	Y	Y	Y	?	?	?	D	?	S	—	—	Y	IG16: Don't provide unnecessary consistency
Y	Y	Y	Y	Y	Y	Y	D	?	?	—	?	—	—	—	Y	IG17: Minimize forwarding-time logic
—	—	—	—	—	—	—	L	?	?	—	?	—	—	—	Y	IG18: Support multiple QoS+ mechanisms for different operating conditions
Y	Y	Y	Y	Y	Y	Y	?	—	—	—	–	—	—	—	Y	IG19: Inspect only packet header, not payload
—	—	—	—	—	—	—	L	D	D	—	D	—	—	—	F	IG20: Manage aperiodic traffic

IPv6 Flow Labels: IPv6 flow labels [44] associate each "reservation" with an application-to-application network socket connection, which would contain many different sensor update streams with a wide range of required QoS+. Packets are processed in a flow-specific manner by the nodes that have been set up with flow-specific state. The nature of the specific treatment and the methods for the flow state establishment are out of scope of the specification.

IP Multicast: IP Multicast provides efficient multicast for a single, non-replicated flow. However, if multiple IP multicast groups are used as a replication mechanism there is no guarantee that the corresponding multicast trees will be disjoint, which is important not only for efficient multicast (IG4) but also for providing low latencies in the face of failures (DR5B). It also does not, by itself, have other end-to-end capabilities that are necessary for WAMS-DD

MPLS: MPLS is designed to give ISPs a set of management tools for bandwidth provisioning, not to provide fine-grained (per-update) QoS [45]. Its guarantees are very weak compared to the needs of a critical infrastructure: it gives aggregate economic guarantees over user, location, and protocol. Further, different ISPs can implement it in different ways, and there are no facilities for combining flows across different ISPs (as would be required in WAMS-DD) and being able to reason about what the end-to-end predictability of delays will be.

MPLS has some fault tolerance mechanisms (fast re-route feature, detour merging, and end-to-end path protection). However, MPLS provide a minimum latency of about 50 msec, which is far more than what is needed for emerging SIPS and Transient Stability applications described above. We are aware of no MPLS providers that will guarantee anything close to these requirements.

VLANs and VPNs: Virtual local area networks (VLANs) and virtual private networks (VPNs) intrinsically meet none of the DRs listed above as their purposes are orthogonal to the DRs. A VPN or VLAN could be part of WAMS-DD but VPN and VLAN technologies alone do not meet the requirements.

ATM: Asynchronous Transfer Mode (ATM) is a networking technology sometimes employed in wide area networks. It offers very strong latency guarantees on a per-message basis. However, it does not support multicast (DR3) and multiple disjoint paths (DR4B), and it does not follow any of the IGs other than IG1. Thus, ATM is not an end-to-end solution for WAMS-DD. However, given its strong latency guarantees at the right granularity, it can be part of a WAMS-DD that overlays ATM and other kinds of lower-level networking technologies.

2) Commercial middleware technologies and standards

There is a wide range of commercial, off-the-shelf (COTS) middleware frameworks providing different kinds of services with some relevance for WAMS-DD.

We first consider middleware supporting the publish-subscribe paradigm. There are two distinct architectures for publish-subscribe middleware, each with advantages and disadvantages.

Broker-Based: Broker-based (BB) publish-subscribe middleware relies upon an infrastructure of broker nodes, that receive events and forward them to neighboring brokers in their connected graph. BB pub-sub systems require a broker/server infrastructure to be installed, which often for small and medium scales cannot be amortized over enough applications to be justified. BB pub-sub systems have an advantage, however, in that they place intelligence inside the network, not just at the edges. This enables, for example, efficient multicast (IG4) and rate down-sampling throughout an NGDDS, not just at the edges.

Peer-to-Peer: Peer-to-peer (P2P) publish-subscribe systems place mechanisms for reliability and filtering only at the edges of an infrastructure. For the core backbone, they typically rely on a combination of IP multicast and Ethernet broadcast to be as efficient as possible.

P2P pub-sub systems have an advantage in smaller and medium sized deployments, but for larger scales the lack of mechanisms in the backbone core for rate down-sampling and fault tolerance limit their abilities to achieve extremely low latencies in the presence of failures.

A federated combination of P2P and BB publish-subscribe systems can offer the best of both worlds. Here, near the edges (e.g., within a single utility), P2P pub-sub is employed. Between utilities (or ISOs), BB pub-sub is used in order to support higher throughputs and the lowest possible latencies over distance. A federated amalgamation of P2P would feature a globally unique namespace for variables and utilities, and could seamlessly pass messages with standardized wire and message formats [25].

Another middleware category called *streaming queries* (also known as streaming SQL or complex event processing) consists of a network of computer nodes that manipulate data streams through continuous queries in order to selectively propagate data, merge streams with existing data, or store data in a distributed database. Such systems are not designed to provide hard end-to-end WAN guarantees (DR1) with per-message granularity (DR5A) while tolerating failures (DR5B). Given their intended application domain, they also do not follow most of the IGs.

More recently a number of vendors are offering middleware based on web technologies such as HTTP, XML, and "web services" for use in the power grid. We note that scalability and throughput of such systems is highly questionable due to the many integration layers they typically add to make it possible to glue together just about any application to another [46]. Ken Birman, a leading expert in reliable distributed computing notes in [47] (emphasis is ours):

> *It doesn't take an oracle to see that the mania for web services, combined with such rampant online threats, contains the seeds of a future debacle. We're poised to put air-traffic control, banking, military command and control, electronic medical records, and other vital systems into the hands of a profoundly insecure, untrustworthy platform cobbled together from complex legacy software components.*

Unfortunately, one can add the smart grid to this list: a number of utilities and organizations see web services as a key enabling technology for the smart grid (for example [48], [49], [50]).

3) Existing power technologies and standards

Middleware is rarely used in today's power grid, despite being considered a "best practice" in many other industries for a few decades [25]. We are aware of no networking technologies developed for the power grid that meet any of the DRs above. Popular technologies are intended for a substation scope. When moving from a LAN to a WAN there are many issues that arise, and often implicit design decisions, that cannot be solved by merely layering a new "WAN-appropriate" API over the existing facilities [42].

OPC-UA [51] was designed for a substation scope and is fairly crude. It uses TCP, which was not designed for predictable latency and does not support multicast. Subscribers and publishers "ping" each other to verify if the other is up, which not only does not scale but also ignores best practices for publish-subscribe systems.

IEC 61850 was also designed for a substation scope, including having messages mapped directly onto an Ethernet frame. Extending it beyond the substation to the wide area faces exactly the issues raised above in the DRs and IGs. However, its CIM can be of great use in a WAMS-DD, especially when (and if) the harmonization with C37.118 is completed, in particular in helping automate QoS+ settings and perhaps adaptation strategies for a wide variety of sensors and applications that use them. And, if the 61850 GOOSE APIs were successfully extended to the WAN, then 61850 may well be able to successfully use a WAMS-DD transport.

IEEE C37.118 is a standard for synchrophasors that includes standard message formats. Unfortunately, its present version has no separation between these formats and data delivery mechanisms for them. C37.118 is being revised to allow different data delivery mechanisms to be used. If successful, then C37.118 synchrophasor updates should easily be deliverable by any WAMS-DD transport.

4) NASPInet

The North American Synchrophasor Initiative (NASPI) is a government-industry consortium dedicated to effective deployment of synchrophasors in the US. It is the only effort worldwide that is dealing with end-to-end WAMS-DD issues at a more-than-superficial level. To support the use synchrophasors, NASPI has been developing the notion of NASPInet (Nn), which has two main components, the data bus (NnDB) and the phasor gateway (NnPG). The NnPG is the edge component of Nn, interfacing utility (or ISO) internal works, historians, etc. to the NnDB.

The NnDB is the electricity version of what is sometimes called an enterprise service bus (ESB), which provides communication services for business-to-business exchanges. NnDB satisfies the DRs described earlier in this paper. Five initial service classes have been identified for the NnDB in recognition of the fact that different kinds of traffic with different delivery requirements must be carried.

A. Feedback Control (e.g., small signal stability)

B. Feed-forward Control [4] (e.g., enhancing state estimators with synchrophasors)

C. Post-Event (post-mortem event analysis)

D. Visualization (for operator visibility)

E. Research (testing or R&D)

Each class has associated qualitative requirements for such properties as low latency, availability, accuracy, time alignment, high message rate, and path redundancy. While distinguishing the classes in this way is an important first step, it is nowhere what is needed to plan, design, and manage a WAMS-DD, as we discuss next. It also considers the lowest required latency to be 100 msec, which is insufficient for some of the applications described above.

A common misconception, in our experience, is that a customer such as a utility, an ISO, RTO, or NERC can simply specify that it wants from a telecom provider, for example a "Class A" network, and this is all that is needed. However, this will not result in a WAMS-DD that meets the requirements across multiple traffic classes. For example, if too much traffic of "easier" classes is on the network, then you will not get Class A guarantees. Rather, to provide the DRs identified above, one needs to do significant *resource management* within the data delivery service. Network management components must account for all traffic associated with each subscription using a given level of QoS+. This is embodied in a number of IGs, including IG8 (exploit traffic knowledge), IG9 (systematic, quick internal instrumentation), IG12 (complete perimeter control), IG13 (reject unauthorized packets quickly and locally), and IG20 (manage aperiodic traffic).

[4] We note that this term is not ours, is not in standard use, and may be misleading. This is because there is virtually no control being done in grids today using feed-forward techniques, due to well-known stability issues.

5) GridStat

GridStat is a data delivery service designed to support the DRs in this paper. Its research results have significantly influenced the shape of NASPInet [21]. The GridStat research started in 1999 by looking at the (usually unstated or very incomplete) QoS+ requirements of innovative power applications being developed by power researchers, and analyzing closely what the state of the art in applied distributed computing systems could support. After significant gaps were identified, the detailed design and then programming of GridStat began in 2001.

GridStat is a Broker-Based publish-subscribe system that meets all of the DRs from this paper (except for 5C: tolerating cyber-attacks, which has been planned for and is near-term future research). It also implements all but 3 of the IGs, which have similarly been planned for and are also near-term future research. More on GridStat overall can be found in several publications: general details [4], [34], [52], [53], [54]; QoS routing [35], [36]; securely upgradeable encryption and authorization infrastructures in [55], [56].

On a 2007-era PC, GridStat adds ~0.1 msec per overlay hop and handles ~20K forwards/sec at each forwarding engine. On 2003-era network processor hardware, it adds ~0.01 msec/hop and scales to a few million forwards/sec [57]. We plan to verify our belief that, with 2010 hardware, and using multiple network processor cards, it is feasible to achieve 50–100 million forwards/sec while rejecting unauthenticated messages, monitoring traffic patterns, and checking for evidence of intrusions and cyber-attacks. That would be a solid backbone for WAMS-DD.

VI. Related Work

In Section of this paper we have overviewed a large body of related research. One other technology of note is LIPSIN: a publish-subscribe multicast forwarding fabric; [58] LIPSIN gives results of some small-scale simulations and an initial forwarding engine prototype. Like GridStat, is assumes that the network topology is fully known. It names the links in the network with large (~250 bit names) in which 1-bits are sparse. This allows multicast packets to be source-routed by a Bloom filter carried in each packet. With this design, forwarding engines need only logically AND each of their outgoing link names with the Bloom filter in a received packet in order to decide on which links the packet should be sent. The paper discusses mitigations for inevitable issues such as false positives and cycles. The basic idea of LIPSIN operates at a lower level than GridStat's mechanisms for redundant real-time, multi-cast and possibly would allow implementation of higher-performance GridStat forwarding engines.

VII. Conclusions

Today's bulk power systems suffer from a number of fundamental problems—particularly reliability, efficiency, and renewable integration—that can be mitigated by coherent-real-time data delivery. In this paper we outlined these problems and a wide range of applications that can help solve them. We then considered the requirements that these applications need

from grid-wide data-delivery systems, the impact on how to construct such data delivery systems, and existing and emerging technologies that can help implement it.

REFERENCES

[1] The Economist, "Building the Energy Internet," 11 May 2004 (Technology Quarterly section).

[2] U.S.-Canada Power System Outage Task Force. Final Report on the August 14th, 2003 Blackout in the United States and Canada, 2004. https://reports.energy.gov/

[3] C. Hauser, D. Bakken, and A. Bose. "A failure to communicate," IEEE Power and Energy, 3(2), March/April, 2005, pp. 47–55.

[4] D. Bakken, C. Hauser, H. Gjermundrød, and A. Bose. *Towards More Flexible and Robust Data Delivery for Monitoring and Control of the Electric Power Grid*, Technical Report EECS-GS-009, Washington State University, May 2007.

[5] E. O. Schweitzer, III and D. Whitehead, "Real-world synchrophasor solutions," in 35th Annual Western Protective Relay Conference, Spokane, WA, October 21–23, 2008.

[6] A. Abur and A. G. Exposito, Power System State Estimation,Theory and Implementation. New York: Marcel Dekker, 2004.

[7] F. C. Schweppe and J. Wildes, "Power System Static-State Estimation, Part I: Exact Model," *IEEE Trans. on Power Apparatus and Systems*, vol. PAS-89, pp. 120–125, Jan. 1970

[8] E. O. Schweitzer, III and D. E. Whitehead, "Real-time power system control using synchrophasors," in 34th Annual Western Protective Relay Conference, Spokane, WA, October 16–18, 2007.

[9] [YSB09] T. Yang, H. Sun, and A. Bose, "Two-level PMU-based Linear State Estimator," in *Proceedings of the IEEE PES Power Systems Conference & Exposition (PSCE)*, Seattle, Washington, March 15-18, 2009, pp. 1-6.

[10] [MPA+04] Roy Moxley PE, Chuck Petras PE, Chris Anderson, and Ken Fodero II. Display and Analysis of Transcontinental Synchrophasors, *Western Power Delivery and Automation Conference*, 2004.

[11] Daniel J. Trudnowski, John W. Pierre, Ning Zhou, John F. Hauer, Manu Parashar "Performance of Three Mode-Meter Block-Processing Algorithms for Automated Dynamic Stability Assessment, IEEE TRANSACTIONS ON POWER SYSTEMS, VOL. 23, NO. 2, MAY 2008.

[12] Anthony Johnson, Robert Tucker, Thuan Tran, Dan Sullivan, Chris Anderson and Dave Whitehead ,"Static Var Compensation Controlled via Synchrophasors," Western Protective Relay Conference, Spokane, WA, 2007.

[13] [PT08] A. G. Phadke and J. S. Thorp. *Synchronized Phasor Measurements and Their Applications*. Springer, 2008.

[14] M. Klein, G. J. Rogers, and P. Kundur, "A fundamental study of inter-area oscillations in power systems," *IEEE Trans. Power Syst.*, vol. 6, no. 3, pp. 914–921, August 1991.

[15] Horowitz, S. Novosel, D. Madani, V. Adamiak, M. "System-Wide Protection," *IEEE Power & Energy Magazine*, 6(5), September 2008, 34-42.

[16] E. Martínez, N. Juárez, A. Guzmán, G. Zweigle, and J. León, "Using synchronized phasor angle difference for wide-area protection and control," in *33rd Annual* Western Protective Relay Conference, Spokane, WA, October 17–19, 2006.

[17] Jan Åge Walseth, Jan Eskedal and Øyvind Breidablik. "Analysis of Misoperations of Protection Schemes in the Nordic Grid 1st of December 2005." *Protection, Automation and Control World*, March 2010.

[18] Edmund O. Schweitzer III, David Whitehead,Greg Zweigle, Krishnanjan Gubba Ravikumar, "Synchrophasor-Based Power System Protection and Control Applications, Western Protective Relay Conference, Spokane, WA, 2009.

[19] J. Mulhausen, J. Schaefer, M. Mynam, A. Guzmán, and M. Donolo, "Anti-islanding today, successful islanding in the future," in 36th Annual Western Protective Relay Conference, Spokane, WA, October 20–22, 2009.

[20] Stian F. Abelsen and Harald Gjermundrød and David E. Bakken and Carl H. Hauser. "Adaptive Data Stream Mechanism for Control and Monitoring Applications". In Proceedings of 1st International Conference on Adaptive and Self-adaptive Systems and Applications (ADAPTIVE'09), Athens, Greece, November 2009, 86–91.

[21] North American Synchrophasor Initiative, "Quanta Statement of Work".

[22] Electric Power Research Institute (EPRI), The Integrated Energy and Communication Systems Architecture, Vol. IV: Technical Analysis, 2004.

[23] Sudipto Bhowmik, Kevin Tomsovic, and Anjan Bose. "Communication models for third party load frequency control." IEEE Transactions on Power Systems, 19:1, February 2004, 543–548.

[24] J. Nutaro, P.T. Kuruganti, L. Miller, S. Mullen, M. Shankar. Integrated Hybrid-Simulation of Electric Power and Communications Systems, in *Proceedings of the 2007 Power Engineering Society General Meeting*, 2007. IEEE , 1-8, June 2007

[25] David E. Bakken, Richard E. Schantz, and Richard D. Tucker. "Smart Grid Communications: QoS Stovepipes or QoS Interoperability," in *Proceedings of Grid-Interop 2009*, GridWise Architecture Council, Denver, Colorado, November 17-19, 2009. Available http://gridstat.net/publications/TR-GS-013.pdf.

[26] E. Rosen, A. Vishanathan, R. Callon. RFC-3031: Multiprotocol Label Switching Architecture. The Internet Society, 2001. http://datatracker.ietf.org/doc/rfc3031/

[27] J. Saltzer, D. Reed, and D. Clark. "End-to-End Arguments in System Design". Transactions on Computer Systems, Association of Computing Machinery (ACM), 2(4), November 1984, 277–288.

[28] Schantz RE, Loyall JP, Atighetchi M, Pal PP. Packaging Quality of Service Control Behaviors for Reuse. Proceedings of ISORC 2002, The 5th IEEE International Symposium on Object-Oriented Real-time distributed Computing, April 29 - May 1, 2002, Washington, DC.

[29] Pal PP, Loyall JP, Schantz RE, Zinky JA, Shapiro R, Megquier J. Using QDL to Specify QoS Aware Distributed (QuO) Application Configuration. Proceedings of ISORC 2000, In *Proceedings of the 3rd IEEE International Symposium on Object-Oriented Real-time distributed Computing*, March 15 - 17, 2000, Newport Beach, CA.

[30] David Bakken. *Quality of Service Design Considerations for NASPInet*. Presentation to the North American Synchrophasor Initiative (NASPI) Work Group meeting, Scottsdale, AZ February 4, 2009.

[31] K. Harald Gjermundrød, Ioanna Dionysiou, Carl Hauser, Dave Bakken, and Anjan Bose "Flexible and Robust Status Dissemination Middleware for the Electronic Power Grid". *Technical Report EECS-GS-003*, School of Electrical Engineering and Computer Science, Washington State University, September 2003.

[32] Rakesh Bobba, Eric Heine, Himanshu Khurana, and Tim Yardley. "Exploring a Tiered Architecture for NASPInet," in *Proceedings of the IEEE Conference on Innovative Smart Grid Technologies*, Gaithersberg, MD, January 2010.

[33] K. Tomsovic, D. Bakken, M. Venkatasubramanian, and A. Bose, Designing the Next Generation of Real-Time Control, Communication and Computations for Large Power Systems. In *Proceedings of the IEEE* (Special Issue on Energy Infrastructure Systems), 93(5), May 2005.

[34] K. Harald Gjermundrød, David E. Bakken, Carl H. Hauser, and Anjan Bose. "GridStat: A Flexible QoS-Managed Data Dissemination Framework for the Power Grid," *IEEE Transactions on Power Delivery*, 4(1), 2009, 136–143.

[35] V.S. Irava, and C. Hauser, "Survivable low-cost low-delay multicast trees," Proceedings of the IEEE Global Telecommunications Conference (GLOBECOM), Nov. 2005.

[36] V.S. Irava, "Low-cost delay-constrained multicast routing heuristics and their evaluation," PhD Dissertation, Washington State University, August 2006.

[37] GridWise Architecture Council, *Interoperability__Constitution Whitepaper (v1.1)*, December 2006.

[38] Lawrence Roberts. "A Radical New Router". *IEEE Spectrum*, July 2009, 35–39.

[39] Jack Brassil, Rick McGeer, Raj Rajagopalan, Puneet Sharma, Praveen Yalangadula, Sujata Banerjee, David P. Reed, Sung-Ju Lee, Andy Bavier, Larry Peterson, Stephen Schwab, Larry Roberts, Alex Henderson, Bob Khorram, Shidong Zhang, Soonyong Sohn, Brian Mark, John Spies, Nicki Watts, The CHART System: A High-Performance, Fair Transport Architecture Based On Explicit-Rate Signalling, *ACM SIGOPS Operating Systems Review*, January 2009.

[40] Kristin Rauschenbach, Regina Hain, Alden Jackson, John Jacob, Will Leland, John Lowry, Walter Milliken, Partha Pal, Ram Ramanathan, Cesar Santivanez, "Dynamic provisioning system for bandwidth-scalable core optical networks," in *Proceedings of MilCom 2009*.

[41] Patrick Th. Eugster, Pascal A. Felber, Rachid Guerraoui, and Anne-Marie Kermarrec. "The Many Faces of Publish-Subscribe". *ACM Computing Surveys*, Vol. 35, No. 2, June 2003, pp. 114–131.

[42] [ZBS97] Zinky J., Bakken D., Schantz R. Architectural Support for Quality of Service for CORBA Objects. Theory and Practice of Object Systems, April 1997.

[43] www.soscoe.com

[44] J. Rajahalme, A. Conta, B. Carpenter, and S. Deering. RFC3697: IPv6 Flow Label Specification. The Internet Society, 2004. http://www.faqs.org/rfcs/rfc3697.html

[45] Adrian Farrel (ed*). Network Quality of Service Know it All*, Elsevier, 2009.

[46] Birman, K. "Like it or not, Web Services are Distributed Objects!," *Communications of the ACM*, 47:12, 60–62

[47] Birman, K. "The Untrustworthy Web Services Revolution". *IEEE Computer*, 39:2, Feb., 2006, 98-100.

[48] PJM, *PJM 2007 Strategic Report*, April 2, 2007, http://www2.pjm.com/documents/downloads/strategic-responses/report/20070402-pjm-strategic-report.pdf.

[49] IEEE Guide for Monitoring, Information Exchange, and Control of Distributed Resources Interconnected with Electric Power Systems. IEEE Std 1547.3™-2007.

[50] Open Secure Energy Control Systems (OSECS), www.osecs.com.

[51] W. Mahnke, S. Leitner, and M. Damm. *OPC Unified Architecture*, 4 May 2009.

[52] K. Harald Gjermundrød. *Flexible QoS-managed status dissemination middleware framework for the electric power grid.* PhD Dissertation, Washington State University, August 2006.

[53] Ioanna Dionysiou, Dehorah Frincke, Carl Hauser, and Dave Bakken, "An Approach to Trust Management Challenges for Critical Infrastructures," *Lecture Notes in Computer Science 5141*, Springer, Berlin, 2007.

[54] Carl H. Hauser, David E. Bakken, Ioanna Dionysiou, K. Harald Gjermundrod, Venkata S. Irava, Joel Helkey, and Anjan Bose. "Security, Trust and QoS in Next-generation Control and Communication for Large Power Systems." *International Journal of Critical Infrastructures (Inderscience)*,2007.

[55] E. Solum, C. Hauser, R. Chakravarthy. Modular over-the-wire configurable security for long-lived critical infrastructure monitoring systems, *Proc. of the 3rd ACM Int'l Conf. on Distributed Event-Based Systems (DEBS 2009)*, Nashville, TN, July 2009.

[56] R. Chakravarthy, C. Hauser, and D. Bakken. Long-lived authentication protocols for critical infrastructure process control systems, *Fourth IFIP WG 11.10 Int'l Conf. on Critical Infrastructure Protection*," Washington, D.C., March, 2010.

[57] K. Swenson. Exploiting network processors for low latency, high throughtput, rate-based sensor updated delivery. MS Thesis, Washington State University, 2009.

[58] Petri Jokela, András Zahemszky, Christian Esteve Rothenberg, Somaya Arianfar, Pekka Nikander. "LIPSIN: line speed publish/subscribe inter-networking". In Proceedings of SIGCOMM 2009, ACM, August 2009, Barcelona, 195–206.