Towards More Effective & Resilient Power Apps Exploiting Better Comms. & Computation

Prof. Dave Bakken

School of Electrical Engineering and Computer Science
Washington State University
Pullman, Washington, USA

Grenoble INP
Grenoble, France
16 December 2019

Big Picture

- Electric grids are getting more stressed each year
  - Transmission line growth inadequate
  - New kinds of generation to integrate (solar, wind, …)
  - Semi-independent microgrid
  - “Prosumers” producing electricity at the distribution edges
  - Chance or cyber or physical attacks
  - Imperfect/incomplete modeling everywhere

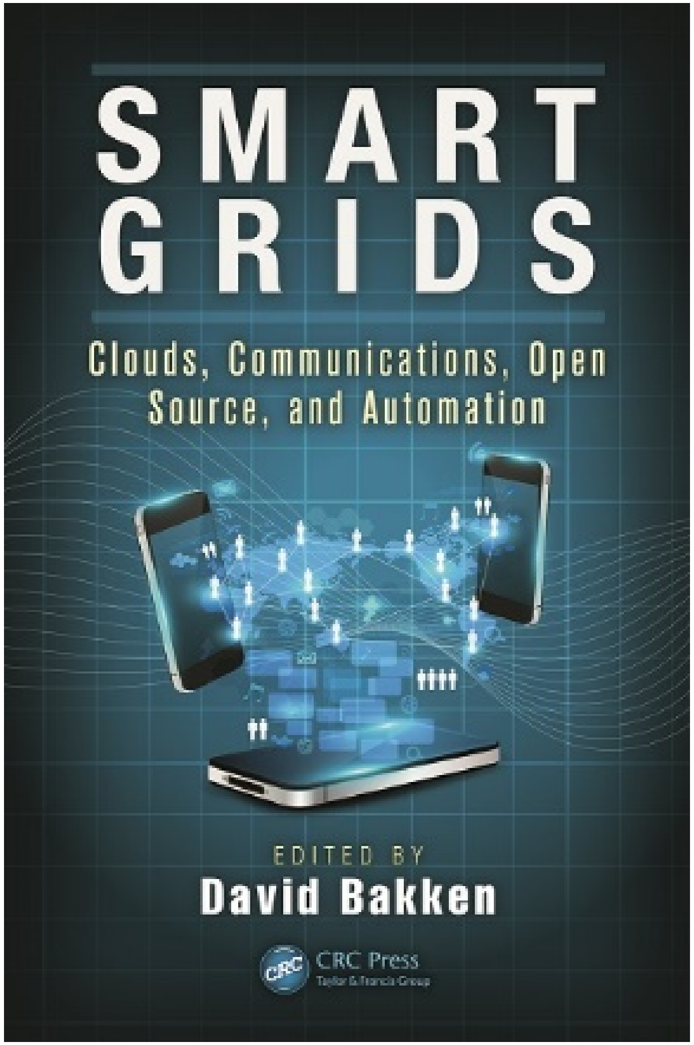
WASHINGTON STATE UNIVERSITY

Better Comms & Computation can <u>greatly</u> mitigate the above

gridstat.net

Context

- IANAPP (power person): Computer Scientist
  - Core background: fault-tolerant **distributed computing**
  - Research lab experience (BBN) with wide-area middleware with QoS, resilience, security, …. for DARPA/military
  - Working with Anjan Bose since 1999 on **wide-area** data delivery issues appropriate **for RAS and closed-loop applications**
    - **GridStat (1999-present)**
    - **GridSim (2009-2014)**
    - **GridCloud (2012-present)**
    - **DCBlocks (2014-present)**



May, 2014 | ISBN: 1482206110

So what do I REALLY do?

From my too-long, smart aleck email .signature:

WSU GridStat Project: Since 1999, cheerfully and audaciously dragging the wide-area data delivery services of the electric power grid -- kicking and screaming -- into the mid-1990s. ETA: 2020-2025 for 10% penetration of mid-90s technology & 2045 for 50% penetration of (then) some half-century-old ICT technology.

Sources of Info (1)

- D. Bakken, A. Bose, C. Hauser, D. Whitehead, and G. Zweigle. “Smart Generation and Transmission with Coherent, Real-Time Data. *Proceedings of the IEEE*, 99(6), June 2011.
- 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. Online http://gridstat.net/publications/TR-GS-013.pdf.
  - **Best Paper Award for “Connectivity” track**. This is the official communications/interoperability meeting for the pseudo-official “smart grid” community in the USA, namely DoE/GridWise and NIST/SmartGrid.

Sources of Info (2)

- ToSG-Workshop.org
- Chapters in D. Bakken and K. Iniewski, ed. *Smart Grids: Clouds, Communications, Open Source, and Automation,* CRC Press, May 2014, ISBN 9781482206111.
  - G. Zweigle, “Emerging Wide-Area Power Applications with Mission Critical Data Delivery Requirements”.
  - D. Bakken *et. al*. “GridStat: High Availability, Low Latency and Adaptive Sensor Data Delivery for Smart Generation and Transmission.”
  - T. Gamage *et. al*. “Power Application Possibilities with Mission Critical Cloud Computing.”

Outline

- **Questions for Power Engineers & Researchers**
- WAN Apps with Extreme Comms Requirements
- IT Guidelines for Achieving these Requirements
- GridStat: Industrial Internet for Electricity (IIE)
- GridStat Cyber-Physical Example
- Cyber-Security for Closed-Loop Apps
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

A Crucial and Wide Open Issue

- **How do anomalies in ICT affect power "stability"?**
- Wide open: almost completely unstudied ☹
  - Lars Nordstöm of KTH best power researcher at this
  - Still a huge opportunity for research topics!
- Trying here to plant seeds to break chicken-egg
  - Power researchers can assume much "better" data delivery to come up with "better" apps
  - Computer scientists can come up with even better data delivery but need to know killer app requirements and acceptable tradeoffs (there are *always* tradeoffs!)
  - Data Analytics scientists can come up with better analytics given the tradeoffs and assumptions above

Comms Baseline: You Can Assume

- Data delivery over WAN **can** be (with GridStat etc):
  - Very **fast**: less than ~1 msec added to the underlying network layers across an entire grid
    - **Even in the presence of failures!.....**
  - Very **available**: think in terms of up to 5+ 9s (multiple redundant paths, each with the low latency guarantees)
  - Very **cyber-secure**: for long-lived embedded devices and won’t add too much to the low latencies
    - E.g., RSA adds >>60 msec so not for RAS or closed-loop
    - Shared keys (61850-90-5): subscriber can spoof publisher ☹
    - GridStat solution not vulnerable and only adds ~1msec
  - Tightly **managed** for **very strong guarantees** (~~MPLS~~)
  - **Adaptive**: can change pre-computed subscriptions ~INSTANTLY (and dynamic requests FAST)

Questions to Ask Yourself

- So how can power researchers exploit this better communications infrastructure?
- What **rate** and **latency** and **data availability** does my power app *really* need for remote data?
  - Why fundamentally does it need that?
  - How sensitive is it to occasional longer delays, periodic drops (maybe a few in a row), or data blackouts for longer periods of time?
- **Can I formulate & test hypotheses for the above?**

Beyond Steady-State-Only Thinking

- Previous is just for steady state: different in some contingency/mode situations?
- How important is my app *in that given contingency/mode, compared to other apps?*
  - E.g., simple “importance” number [0,10]
  - How much worse QoS+ (latency, rate, availability) can I live with in steady state and in given contingencies?
    - But would **still get strong guarantees** at that lower quality
    - How much benefit do different levels really give me?
  - Can I program my app to run at different rates, or is there a fundamental reason it has to run at one?
- What extra data feeds (or higher rates etc) could I use in a contingency/mode (could get in << 1sec)

Bad Data

- How vulnerable is my power app to bad data?
  - State estimation obviously has handed for many decades
- But
  - How much bad data
  - Does how much bad (grammar alert!)
  - In what circumstances?
- E.g. can I specify assumptions about bad data?
  - Number: absolute or (better) as a function of the problem size (state/configuration/#PMUs/etc)
  - Location and timing: randomly distributed or worst case?
  - Error degree: randomly off (what probability distribution) or worst case (from an adversary)?

Bad Manners

- How vulnerable is my power app or RAS scheme or stability assumptions to worst-case malicious behavior?
- E.g. not just false data (which may be able to be detected) but taking over command of a relay or other devices
  - How many of these, and of what kind, could cause problems?
- Thinking cyber-physical here
  - What are some worst case combinations of a physical attack (rifle, chaff, modifying sensors, ..) and a cyber attack (colluding customer meters, taking over relays, DDOS to throttle delivery of sensor data and commands,
  - And worst case under what situations?

Bad Manners (cont.)

- “The event I fear most is a physical attack in a successful cyber-attack in conjunction with responders' 911 system or *on the power grid,*”
  - Ronald Dick, director of the FBI's National Infrastructure Protection Center, *Washington Post*, Front Page Article, June 27 2002, (*emphasis* added)

A Cloudy Forecast

- What could I do with cloud computing, assuming it is made mission critical, i.e.:
  - Keeps same fast throughput
  - Does not allow deliberate “inconsistencies”
    - e.g., a replica does a state update never received by others
  - Is much more predictable with CPU perf., ramp-up time, …
  - (BTW, ARPA-E GridCloud proj. w/Cornell+WSU doing for >2 years)
    - Pilot starting with ISO-New England, likely others soon
  - **Not all CPUs in datacenter, some (managed) in substations… (Cisco Fog?)**
- How could I use
  - Tens/Hundreds of processors in steady state
  - **>>Thousands when approaching/reaching contingencies**
  - Data from ALL participants in a grid enabled quickly when approaching a crisis
- More in the “Killer Apps” discussion in the backup slides

CIP-Managed Compute+Comms+Security

- Computations + communications + security *can* be
  - Mission critical to power grid specs
    - Closed-loop WAN app requirements **WAY harder** than air traffic control, railways, military, …
  - Changed rapidly in a coordinated manner
    - Providing app developers **much higher-level building blocks**
  - Managed in a network operations center 24x7
    - **Much like a power control center!!!**
    - Needed if power grid stability really does depend on comms and computation and cyber-security
    - No more hard-coded and unmonitored comms infrastructures causing headaches when glitches occur!

Outline

- Questions for Power Engineers & Researchers
- **WAN Apps with Extreme Comms Requirements**
- IT Guidelines for Achieving these Requirements
- GridStat: Industrial Internet for Electricity (IIE)
- GridStat Cyber-Physical Example
- Cyber-Security for Closed-Loop Apps
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

Wide Range of QoS+ Requirements

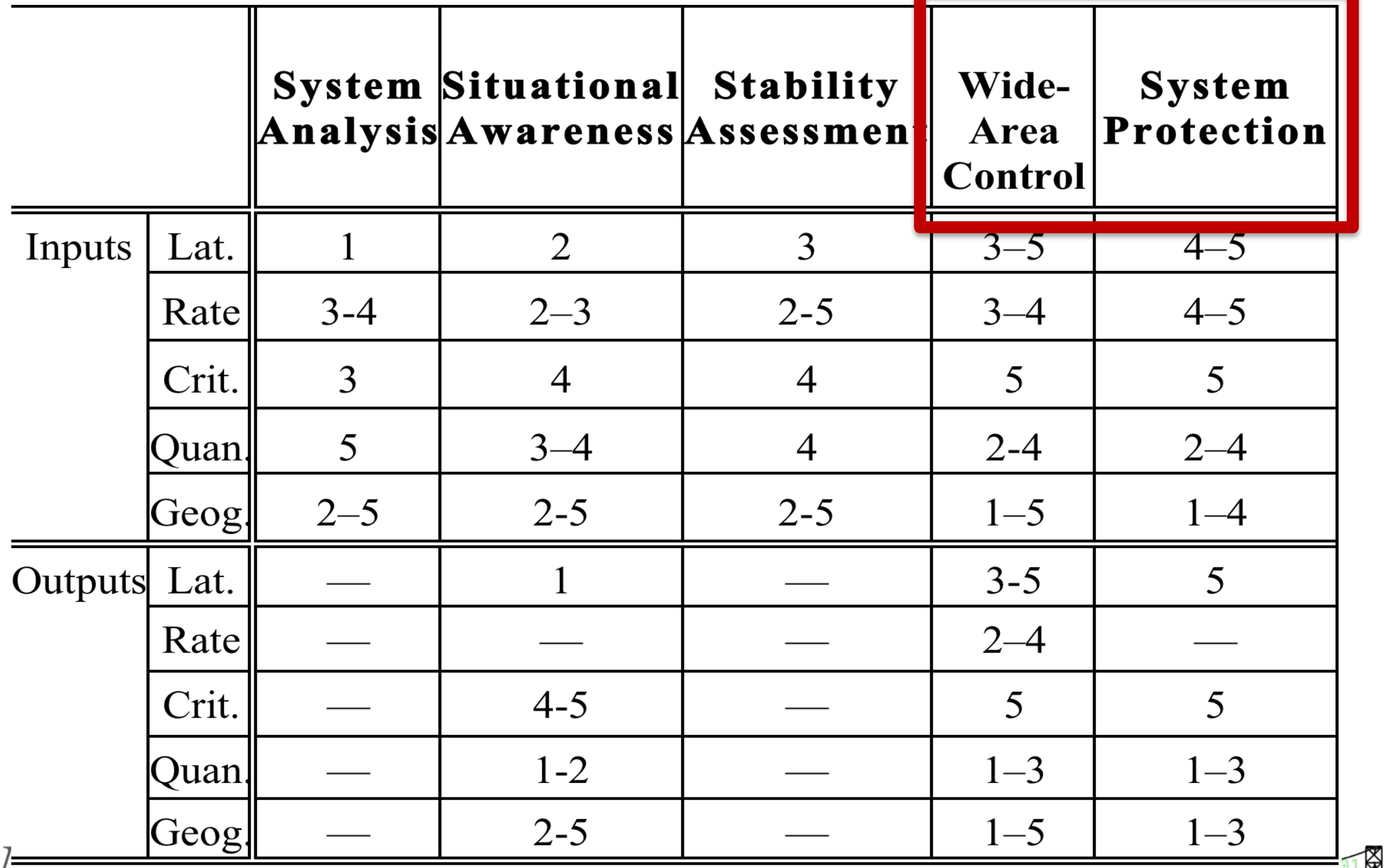
- **QoS+**:
  - network/middleware “QoS” (latency, rate), availability/criticality
  - Also things an implementer/deployer of WAMS-DD[1] needs to know: geographic scope, quantity.
- Comparing Apples and Apples:
  - Normalize each from 1 (very easy) to 5 (very hard)
- Wide ranges
  - Across application families
  - Sometimes within them (each configuration is different)

[1] WAMS-DD=WAMS Data Delivery

Normalized Values of Parameters

Difficulty (5 is hardest)	Latency (ms)	Rate (Hz)	Criticality/ Availability	Quantity	Geography
5	5-20	>240	Ultra	Very High	Across grid or multiple ISOs/RTOs
4	20-50	120-240	Very High	High	With an ISO/RTO
3	50-100	30-120	High	Medium	Between a few utilities
2	100-1000	1-30	-	Low	Within a utility
1	>1000	-	-	Very Low	Within sub.

Diversity of Extreme Apps



		System Analysis	Situational Awareness	Stability Assessment	Wide-Area Control	System Protection
Inputs	Lat.	1	2	3	3–5	4–5
	Rate	3-4	2–3	2-5	3–4	4–5
	Crit.	3	4	4	5	5
	Quan.	5	3–4	4	2-4	2–4
	Geog.	2–5	2-5	2-5	1–5	1–4
Outputs	Lat.	—	1	—	3-5	5
	Rate	—	—	—	2–4	—
	Crit.	—	4-5	—	5	5
	Quan.	—	1-2	—	1–3	1–3
	Geog.	—	2-5	—	1–5	1–3

Outline

- Questions for Power Engineers & Researchers
- WAN Apps with Extreme Comms Requirements
- **IT Guidelines for Achieving these Requirements**
- GridStat: Industrial Internet for Electricity (IIE)
- GridStat Cyber-Physical Example
- Cyber-Security for Closed-Loop Apps
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

Internet vs. NASPInet/WAMS-DD

Characteristic	Internet	NASPInet/WAMS-DD
Network Size	~10^9 hosts worldwide	10^5 hosts on a grid, $10^{3\text{-}4}$ “routers”
Per-Flow State	Death (RIP RSVP)	**Very feasible**
Network Design Goal	Best-effort delivery for any user and purpose	**Strong guarantees of QoS+ in several dimensions for grid-specific users and purposes**
Admission Cntl Perimeter	None	**Complete**
Fraction of Managed Traffic	None/Very Little	Almost all. All traffic subject to policing. >>90% periodic.
Central Knowledge of Topology	Not attempted: large scale and dynamicity	Feasible: small scale and relatively slow changes
Topology changes (w/o failures)	Often & without warning	Not often & virtually always with warning
Frequency of Route Changes	Frequent; can converge slowly	Infrequent → **use static topo**.

Internet vs. NASPInet/… (cont.)

Characteristic	Internet	NASPInet/WAMS-DD
Latency Achievable	Slow to Medium	Very fast
Latency Predictability	Poor	Very Good to Excellent
Recovery Delay after dropped packet (with “reliable” delivery)[1]	**High** (timeout waiting for data or acknowledgement; then two one-way msgs)	Zero (redundant copies sent over disjoint paths arrives virtually at same time) ➔ **DO NOT USE post-error recovery, be proactive!**
Forwarding Unit	Uninterpreted packet	Update of a sensor variable
Traffic Predictability	Low	Very High
Elasticity of QoS requirements	None/Low	Potentially Medium-High (if power apps programmed per previous questions)

[1] I.e., latency in the case of a failure, which is what ultimately needs to be “guaranteed”

Other IT Guidelines for WAMS-DD

- Don’t depend on priority-based “guarantees” (including “fairness”) for hard real-time
  - They really don’t guarantee anything!
  - This rules out MPLS (but 6 classes is worse for MPLS…)
- Exploit *a priori* knowledge of traffic
- Optimize for rate-based sensor updates
- Use static, not dynamic, routing
  - Can still have adaptations to update routing tables
- Reject unauthorized messages quickly & locally
- Provide per-subscriber QoS+ (~~IP Multicast~~)
- Don’t over-design consistency and (re)ordering

Outline

- Questions for Power Engineers & Researchers
- WAN Apps with Extreme Comms Requirements
- IT Guidelines for Achieving these Requirements
- **GridStat: Industrial Internet for Electricity (IIE)**
- GridStat Cyber-Physical Example
- Cyber-Security for Closed-Loop Apps
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

Middleware in One Slide

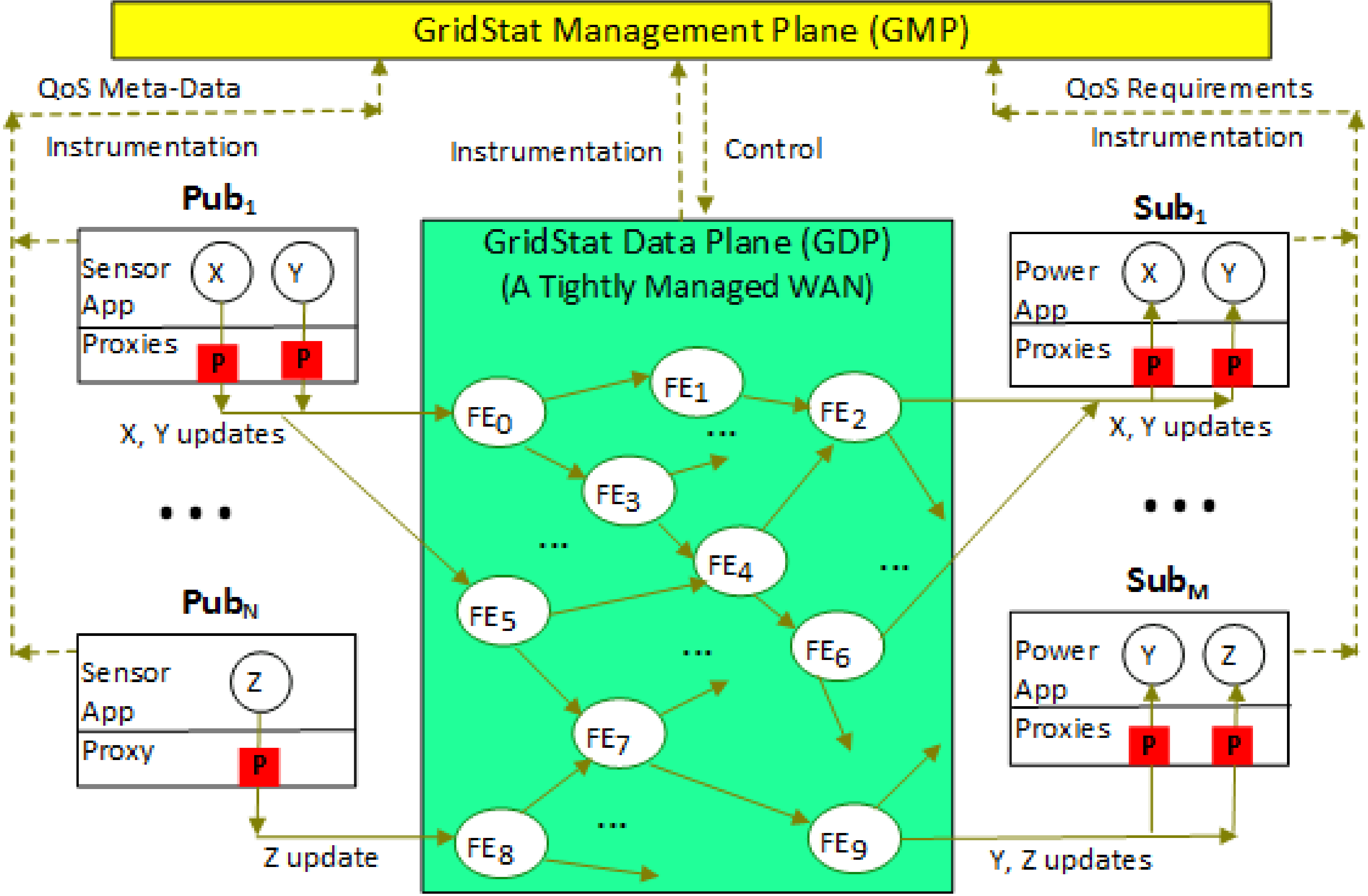
- Middleware == "**A layer of software above the operating system but below the application program that provides a common programming abstraction across a distributed system**"
- Middleware exists to help manage the complexity and heterogeneity inherent in distributed systems
- Middleware provides **higher-level building blocks** ("abstractions") for programmers than the OS provides
  - Can make code much more portable
  - Can make them much more productive
  - Can make the resulting code have fewer errors
  - **Programming analogy — MW:sockets ≈ HOL[1]:assembler**
- Considered best practices in other industries for 15-20 years! (Ouch!)

[1] HOL≡Higher Order Language

What is GridStat?

- Bottom-up re-thinking of how and why the power grid’s real-time data delivery monitoring services need to be
- Comprehensive, ambitious WAMS-DD middleware for power grid in coding since 2001
  - Rate-based publish-subscribe middleware with
    - Predictably low latency
    - Predictably high availability
    - Predictable adaptation
  - Different subscribers to same variable can get different **QoS+ {rate, latency, #paths}**
- Influencing (glacial) NASPInet “effort”

GridStat: Rate-Based Forwarding



Note: GMP, not programmer, finds paths

What Really is GridStat?

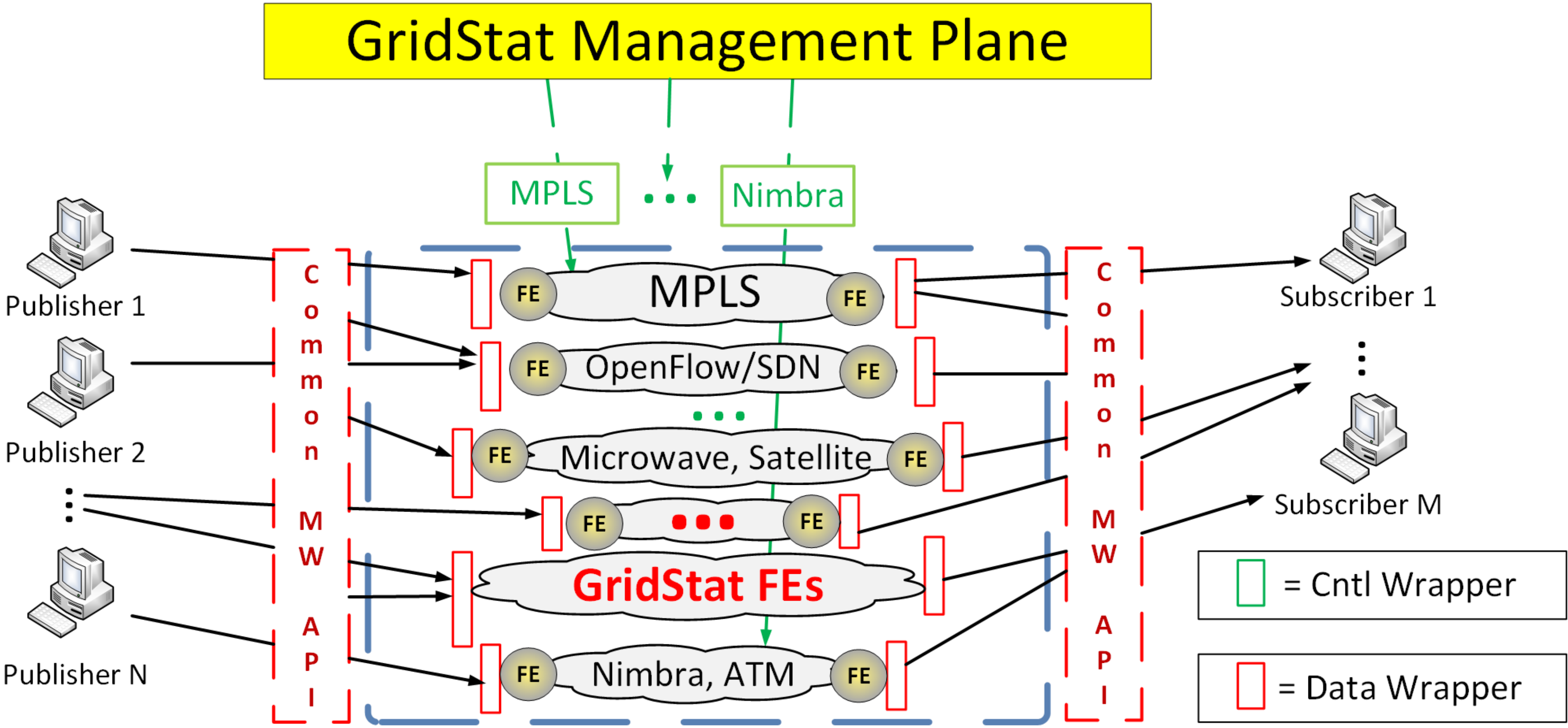
- GridStat at two layers
  - APIs & services (including management, monitoring, …) at edges (e.g., last DNMTT comment)
    - I.e., Middleware overlay with mechanisms only at edges (P2P)
  - Augmented with core software defined network (SDN) utilizing rate-based, in-network router-like Layer-3 *forwarding engines* (FEs)
    - Also then richer management that exploits them
- Even with only 10% penetration of FEs you have much more control and monitoring over data delivery, and at fine granularity

Overlay Middleware Integrating Legacy (Sub)Systems

Flow start (publisher) could also be RTU, substation router, OpenPDC, etc. i.e. not just a single sensor.

GS subscriber could be RTU, substation router, OpenPDC, …

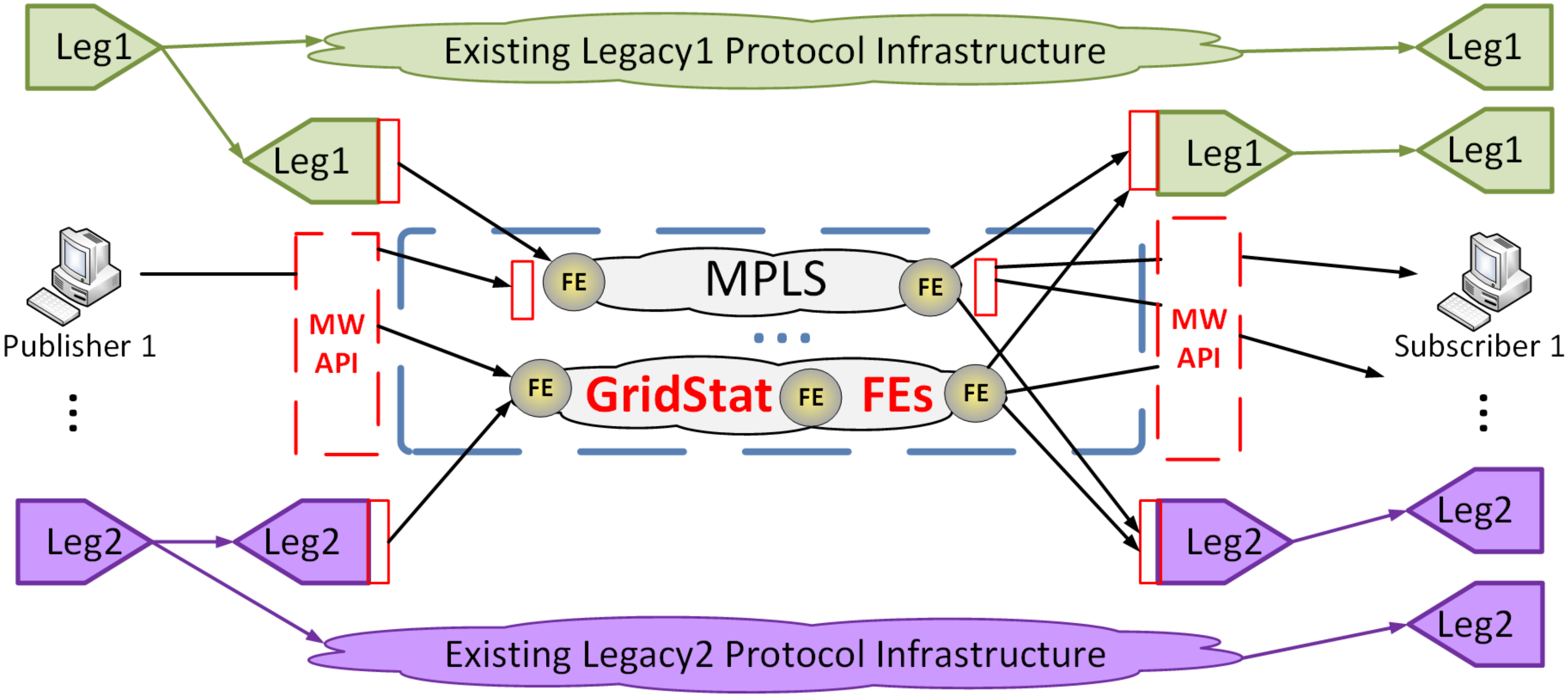
GridStat is
- Real-Time
- Overlay (sometimes)
- Middleware



“...” could be BPL/PLC, 4G teleco, 801.1TSN, even best-effort internet, etc.

Getting There from Here

Issue: how to integrate existing legacy protocols and transition to newer ones and not break apps’ data delivery?



Legend Leg ≡ Legacy ▷ ≡ Sender ◁ ≡ Receiver □ ≡ Data Wrapper

Washington State University

gridstat.net

Overview of GridStat Implementation & Perf.

- Coding started 2001, demo 2002, real data 2003, inter-lab demo 2007-8
  - But power industry moves very, very slowly……
    - “Utilities are trying hard to be first to be second” Jeff Dagle, PNNL
    - “Utilities are quite willing to use the latest technology, so long as every other utility has used it for 30 years” unknown
  - And NASPI is pretty dysfunctional in a number of dimensions
- Implementations
  - Java: < 0.05 msec/forward, 500k+ forwards/sec
    - **Adds less than 1 msec over underlay network (fiber, etc) over entire grid**
  - **C version should be 10X-20X faster (efficient I/O0**
  - Network processor: 2003 HW ~.01 msec/forward, >1M fwds/sec
    - Current network processors are ~10x better, and you can use >1 …
  - Future: ASIC (on a chip)
    - Should be competitive w/IP routers in scale: doing much less, on purpose!
- Note: no need to use IP for core …… (ssshhhhh!): less jitter and likely more bullet-proof (no IP vulnerabilities)

Washington State University

gridstat.net

Outline

- Questions for Power Engineers & Researchers
- WAN Apps with Extreme Comms Requirements
- IT Guidelines for Achieving these Requirements
- GridStat: Industrial Internet for Electricity (IIE)
- **GridStat Cyber-Physical Example**
- Cyber-Security for Closed-Loop Apps
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

GridStat Modes

- Observation
  - Path allocation algorithms complex, not for a crisis 10^3+
  - But power grid adaptations planned ***way*** ahead of time
- GridStat supports **operational modes**
  - Can switch (preloaded) forwarding tables very fast
  - Avoids overloading subscription service in a crisis
  - Can have modes for different contingencies/modes/etc

WASHINGTON STATE UNIVERSITY

gridstat.net

Data Load Shedding

- Electric Utilities can do **load shedding** (I call **power load shedding**) in a crisis (but can really hurt/annoy customers)
- GridStat enables **Data Load Shedding**
  - Subscriber's **desired & worst-acceptable QoS+ (rate, latency, redundancy)** are already captured; can easily extend to add priorities (e.g., [0,10] above)
  - In a crisis, can shed data load: move most subscribers from their desired QoS to worst case they can tolerate (based on priority, and eventually maybe also the kind of disturbance)
  - Works very well using GridStat's operational modes
  - Note: this can prevent **data blackouts**, and also does not irritate subscribers
- Example research needed: systematic study of *data load shedding* possibilities in order to prevent *data blackouts* in contingencies and disturbances, including what priorities different power apps can/should have…
- Enables critical infrastructures: **adapt data comms infrastructure to benign IT failures, cyber-attacks, power anomalies, changing req, …**

Multi-Level Contingency Planning & Adapting

- Electricity example: Applied R&D on coordinated
  1. Power dynamics contingency planning
  2. Switching modes to get new data for contingency
  3. New visualization window specific for the contingency and its new sensor data

  involving contingencies with

  A. Power anomalies

  B. IT failures

  C. Cyber-attacks
- State of art and practice today: 1 & A only, offline
- Very possible: {1,2,3} X {A,B,C} and online

Outline

- Questions for Power Engineers & Researchers
- WAN Apps with Extreme Comms Requirements
- IT Guidelines for Achieving these Requirements
- GridStat: Industrial Internet for Electricity (IIE)
- GridStat Cyber-Physical Example
- **Cyber-Security for Closed-Loop Apps**
- Optional Backup
  - GridSim & GridControl Brief Overviews
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE

A Note on Closed-Loop Cyber-Security

- Extra low added latency crucial for closed-loop apps
  - Otherwise delivery latency too high to work
- RSA is too slow (~60 msec on 2012-era PC)
- Using shared key (publisher and its subscribers) has a serious security vulnerability publicized by WSU
  - Nothing prevents a subscriber from spoofing a publisher
  - Shared keys (and RSA) are the choices for IEC 61850-9005
  - Example from 61850 workshop: presenter ducked out of next session after my pointing above out ☹
- Real-time multicast authentication is an extremely hard problem (much harder than point-point)

Overview of GridStat Cyber-Security

- GridStat has been a founding member of **TCIP and TCIPG** centers for cyber-security for the grid, 2005+.
  - Over $20M invested from NSF, DHS, DOE (also UIUC, others)
- Stackable and changeable security modules at pubs and subs (2007)
  - Long-lived required ability to change modules as crypto technology evolves
  - Modules for encryption & authentication & obfuscation of data
- Authentication of management plane entities pairwise (2009, 2011+)
- Node security protecting data in management plane nodes (2012)
  - Secure key storage (quorum based, Byzantine fault-tolerant, …) ProFokus

GridStat Multicast Authentication

- Researchers: Prof. Carl Hauser and Kelsey Cairns
- Started with TV-OTS (Illinois), an experimental data authentication protocol that appears to fit PMU data delivery needs:
  - Safe with Multicast (one-to-many) environments
  - Low Latency
  - Independently verifiable signatures
  - Optimized for rate based communication (but not nearly fast enough for closed-loop apps with original TV-OTS)
- Achieved by *probabilistic security* (not perfect security)

GridStat Multicast Authentication (cont.)

- DETERLab experiments:
  - Measure signing and verification latency
  - Test achieved security with Security Analyzer node
- Low latency signing, lower latency verifying
- Lossy network has negligible effect
- Given message forgery probability ranging from 1.2e-7 to 9.6e-17
  - Faster cases less secure
- In tightly-controlled WAMS-DD: ~1 msec added
  - Ergo GridStat+Security ≤ ~2 msec over fiber/underlay!

Final Thoughts on GridStat Security

- Would be very hard for an adversary to inject unauthorized traffic anyway even without authentication
  - Has to know exactly where to insert
  - Almost always would get dropped by next FE because not in forwarding table
- Recent MS thesis (patented) on rate-based monitoring can make adversary's job even harder

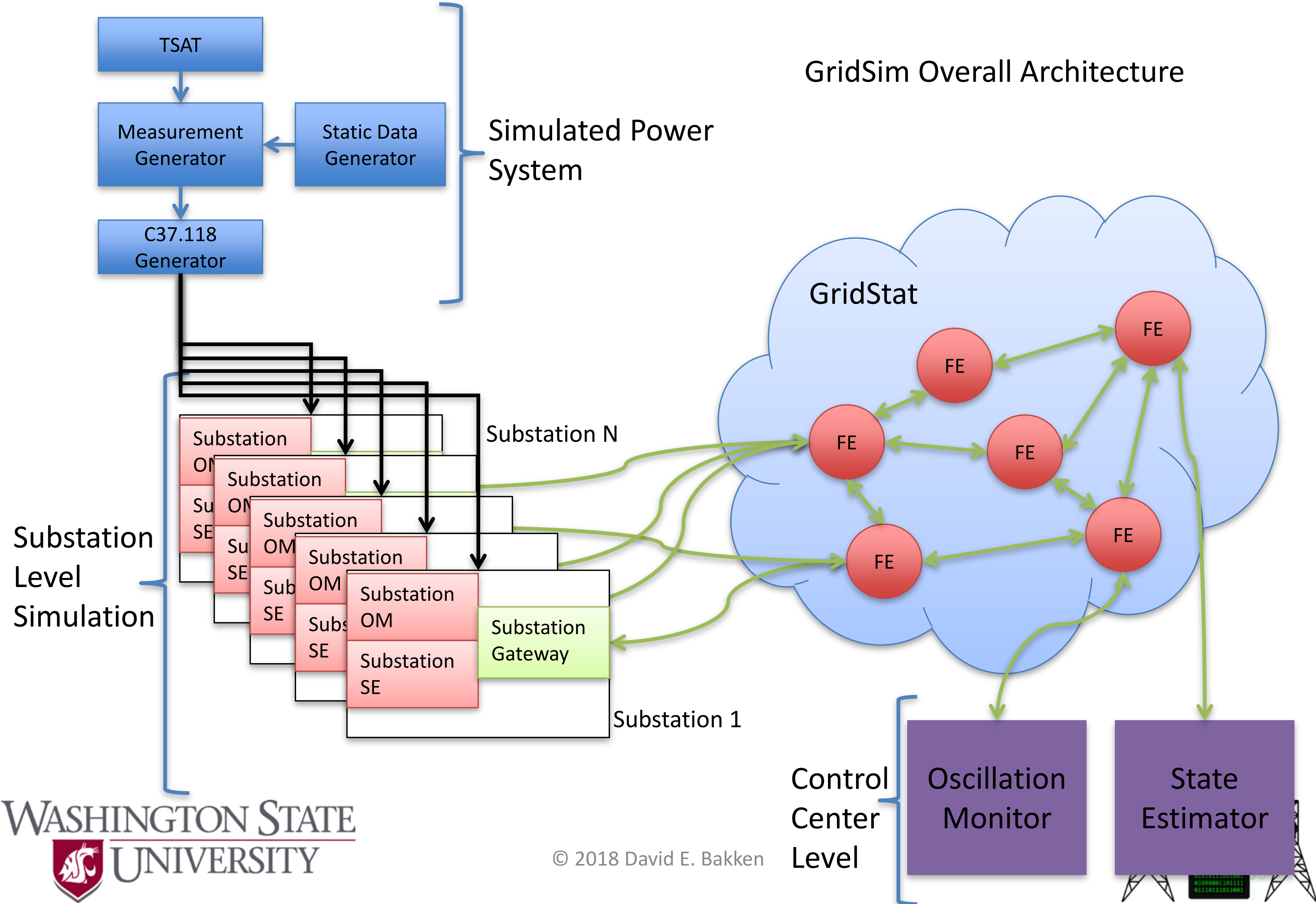
Conclusions

- Requirements for wide-area communications in the power grid is the most extreme of any critical inftastructure
- GridStat is real-time pub-sub middleware designed to meet these needs
- Power grid is great opportunity for computer scientists to have impact
  - Takes many years of learning curve ☹

Outline

- **Optional Backup**
  - **GridSim & GridControl Brief Overviews**
  - Killer Apps for Cloud Computing
  - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - Why Middleware is Needed for IIE
  - Ratatoskr RPC/GridStat with tuneable timeliness, redundancy, safety

GridSim Overall Architecture



TSAT

Measurement Generator

Static Data Generator

C37.118 Generator

Simulated Power System

Substation Level Simulation

Substation N

Substation OM

Substation SE

Substation Gateway

Substation 1

GridStat

FE

FE

FE

FE

FE

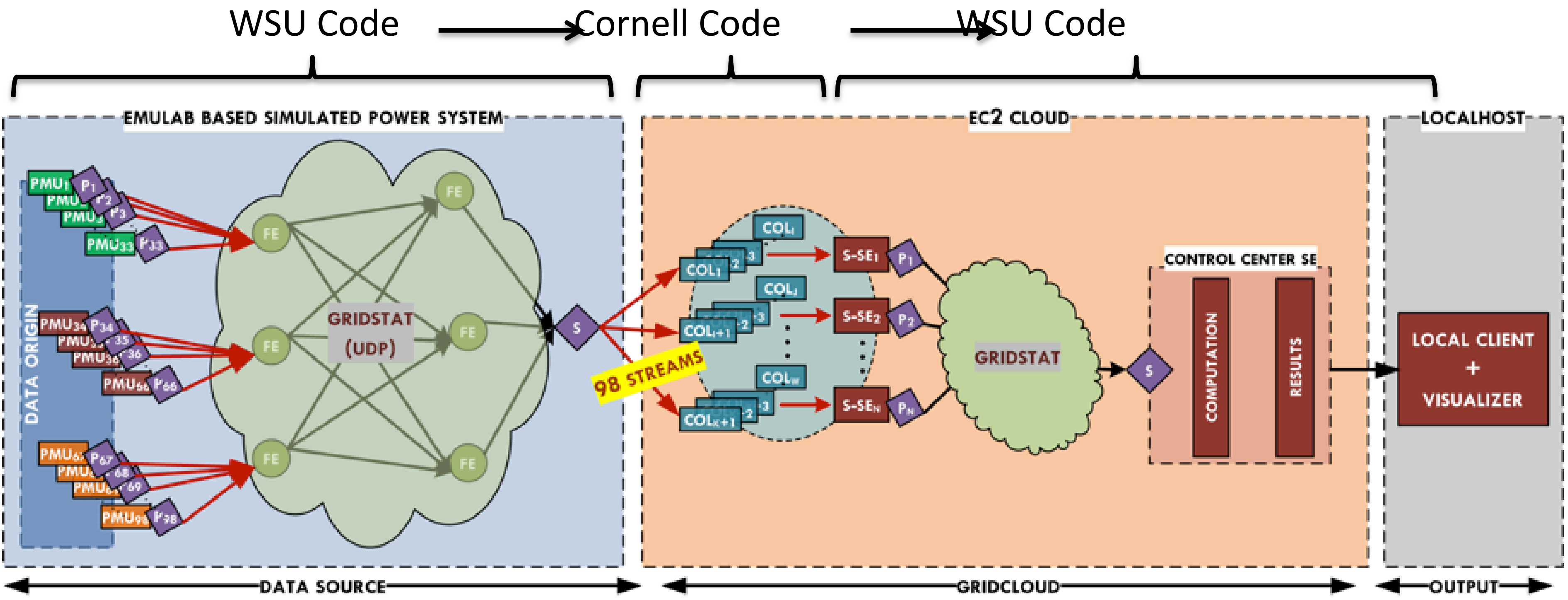
FE

Control Center Level

Oscillation Monitor

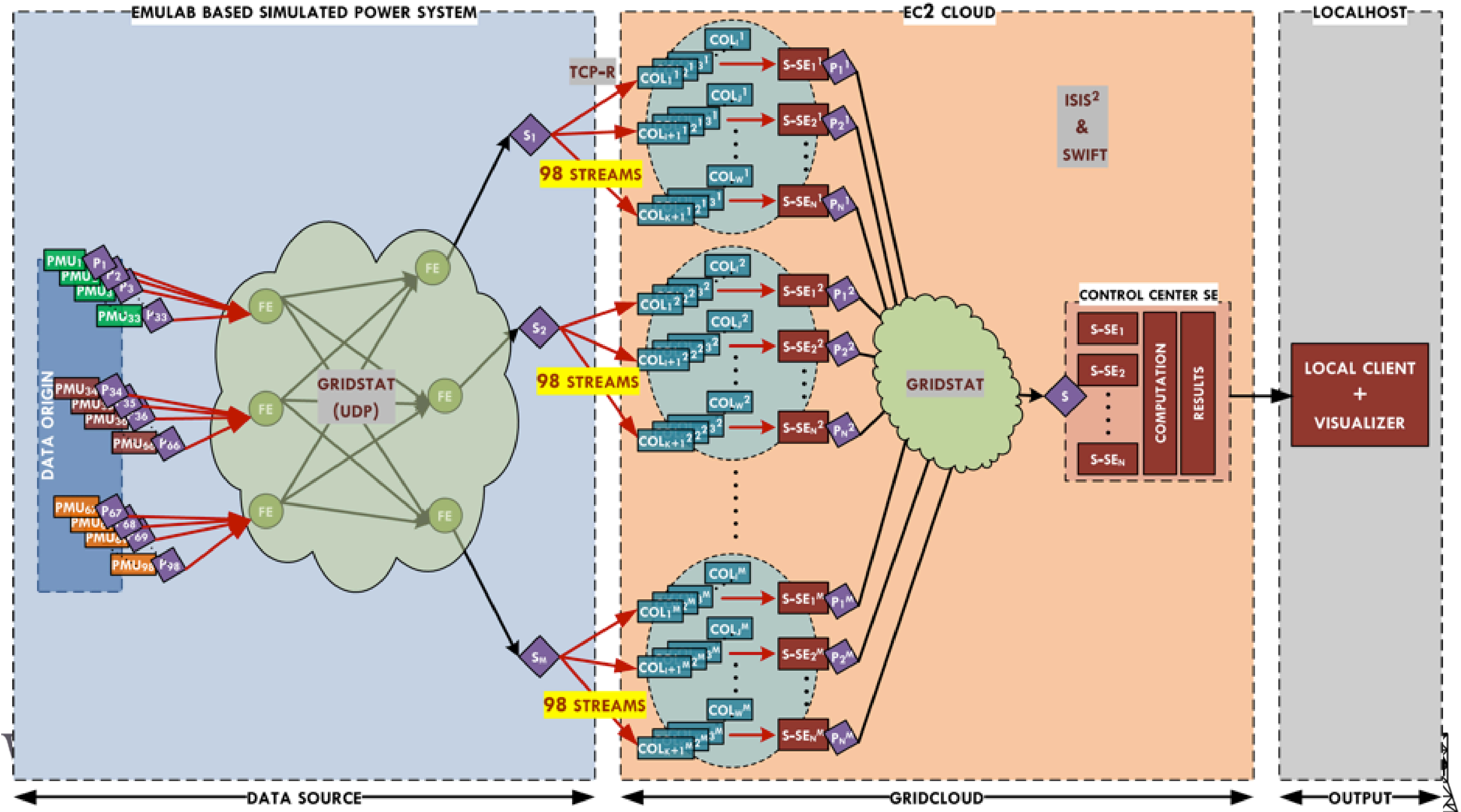
State Estimator

2013: Updated Cloud Architecture (1 Replica)



WSU Code → Cornell Code → WSU Code

Current Cloud Architecture (3 Replica)



Outline

- **Optional Backup**
  - GridSim & GridControl Brief Overviews
  - **Killer Apps for Cloud Computing**
    - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
    - Why Middleware is Needed for IIE
    - Ratatoskr RPC/GridStat with tuneable timeliness, redundancy, safety

Cloud Computing: The "Next New Thing"

- Big data centers (probably hosted by power industry vendors or NERC or DHS/DoE, not Amazon or Google)
- These permit "consolidation"
  - 10x or better reductions in cost of operation
  - Far better equipment utilization and management
  - New styles of elastic computing, potential to compute directly on massive data collections
  - Adds up to a new way of computing that forces us to undertake new kinds of thinking
- But deliberately designed to trade off consistency for scalability

GridControl

- Combining GridStat plus Cornell cloud computing technology (GridCloud)
  - See slides from NASPI meeting February 2012
  - GridStat+GridCloud+ (replicated) Anjan’s LHSE in EC2 for 2+ years
  - ISO New England is starting a pilot project with GridCloud (then also GridStat)
- Challenging questions with *highly elastic* apps
  - Rapid elasticity at scale
  - Predictability of such elasticity
  - Consistency with such elasticity
- Now outlining 8 killer apps that GridCloud enables

#1: Mitigation Control

- Rare combination of events do happen
  - Have lead to many blackouts when not mitigated!
- E.g., *N-3 contingency* (3 failures) never planned for
  - Infrequent but hugely expensive to analyze
  - GridCloud commissions thousands of nodes analyzing candidate mitigation steps in parallel
  - Best approach (actionable steps) is given to operators
- Acknowledgements: Prof. Mani Venkatasubramanian (WSU)

#2: Oscillation Alarm Processing

- Grids oscillate between regions
  - Negatively damping can lead to blackout
  - E.g., Oregon/California in USA July 1996: 0.3 Hz (!!)
- GridCloud commissions massive parallel computations exploring huge permutation space
  - Looking for trends and correlations of alarm data
  - Also huge number of model-based simluations too
  - Finds root cause much faster than possible today in much broader set of conditions
- Acknowledgements: Prof. Mani Venkatasubramanian (WSU)

#3: Post-Tripping Fault Diagnosis

- Protection scheme trips a relay, but why?
  - Underlying cause must be ascertained *post facto*
- GridCloud commissions massive computations to identify the fault(s) that provoked the trip(s)
  - Many different kinds of fault diagnosis algorithms, all could be run in parallel
  - Possible integration candidate: openFLE (fault location engine) from Grid Protection Alliance
- Acknowledgements: Prof. Anuraug Srivastava (WSU)

#4: Multi-Resolution Frequency Disturbance Visualization

- Grid operates in very narrow range unless stressed
  - *Frequency excursions* outside this give clues to problems
- *Frequency disturbance recorder* (FDR): new device recording frequency disturbances at high rates
  - E.g., internal sampling of FNET device (in our lab): 1440 Hz
- GridCloud commissions thousands of parallel frequency rendering computations
  - Provide operators a rich suite of visualizations with which to better understand nature and cause of present excursion
- Acknowledgements: Prof. Yilu Liu (University of Tennessee, Knoxville)

#5: Multi-Dimensional Computations over Both Space and Time

- Two existing GridSim apps can be combined in rich ways possible only with cloud computing
- Hierarchical linear state estimation: rich coverage of (geographical) space
  - At one snapshot in time
  - Obvious extensions over more space with more PMUs
- Oscillation monitoring
  - Uses moving window of time (a few seconds typically)
  - Over streaming data
  - Produces a single number: damping factor
  - Obvious parallel computations over different sets of data with different time windows and algorithms

#5: Multi-Dimensional Computations over Both Space and Time (cont.)

- Combination: provide rich set of two-dimensional (space, time) data to any desired location
  - Enables extremely powerful *new families of applications* operating coherently over both space and time
  - At each location: different time windows, different algorithms, different sets of data
  - If available, people would inevitably think of *many* uses for this data
  - Note: in other contexts (not cloud), Prof. Santiago Grijalva argues for a 3rd dimension: contingency
- Acknowledgements: Prof. Anjan Bose (WSU)

#6: Ultimate Scale: Tertiary Monitoring Centers

- Balancing authorities (144 in North America) must have remote backup control centers
  - Hot backups with same data and apps
- TVA found great value in having a tertiary control center
  - Limited to monitoring: control outputs computed but not used
  - Obvious candidates for the cloud
  - But this is barely scratching the surface here…

#6: Ultimate Scale: Tertiary Monitoring Centers (cont.)

- Major problem today: balancing authorities have almost no visibility anywhere in grid except for a few places in a few neighbors
  - “*Flying blind*”, The Economist, 2004
- Why not just share more?
  - Data stored at another utility is problematic for owner
- Storing in cloud could alleviate this
  - Only access a subset of data and/or derived info
  - Access opened up when grid sufficiently stressed

#6: Ultimate Scale: Tertiary Monitoring Centers (cont.)

- Above is static with default steady state
- Could also drill down on demand with elastic computations
  - Using higher-fidelity algorithms
  - Using higher-resolution data
- Acknowledgements: Russell Robertson (Grid Protection Alliance), for the TVA example (though not the cloud possibilities)

#7: Robust Adaptive Topology Control (RATC)

- Use software to optimize grid topology switching as the control resource
- Technology: use topology control to enhance operations and manage disruptions in grid
- Massively parallel computations to
  - Detect, classify, and respond to grid disturbances
  - Ensure the grid maintains efficient operations while guaranteeing reliability
- Acknowledgements: Prof. Mladen Kezunivoc, Texas A&M University.
  - Funded by the ARPA-E GENI program

#8: Prosumer-Based Distributed Autonomous Cyber-Physical Architecture

- **Prosumer**: An economically motivated power system participant that can consume, produce, store, or transport electricity
  - Interact with other prosumers through services – generation, consumption, storage, and transportation
    - E.g. A utility prosumer aggregating heterogeneous home user prosumers to provide consumption and storage services to a distribution ISO prosumer
  - Drastically increased data acquisition rates, autonomy, distributed control capability

#8: Prosumer-Based Distributed Autonomous Cyber-Physical Architecture (cont.)

- GridCloud commissions massive parallel computations exploring huge permutation space
  - Heterogeneous data aggregation for utility level device management that accounts for instantaneous interoperability
    - Home users can change their strategies (e.g. local storage is not available)
  - Scenario generators for prosumers at different level (in scale)
  - Data organization and processing
- Acknowledgements: Prof. Santiago Grijalva (Georgia Institute of Technology, Georgia)
  - Funded by ARPA-E GENI program

Outline

- **Optional Backup**
  - GridSim & GridControl Brief Overviews
    - Killer Apps for Cloud Computing
  - **A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …**
    - Why Middleware is Needed for IIE
    - Ratatoskr RPC/GridStat with tuneable timeliness, redundancy, safety

Power Culture, not ICT Culture

- Every person can only specialize in a few areas!
- Engineers are confident problem solvers!
  - Some knowledge of computer networking and programming
    - "A little knowledge is a dangerous thing", *Thomas Huxley*
  - Their managers, regulators, & research funding personnel power not ICT
- Middleware best practices in other industries, elec. sector its rare
- Very often end up with
  - Hard-coded solution that is very inflexible, has to be re-implemented for each new power application program for each utility
    - "Application-level protocols" in network parlance
  - Not utilizing the state of the practice in other industries
  - Not handling the interoperability and building blocks necessary
- Utility ICT staff unaware of cutting-edge data delivery or other relevant R&D (e..g, DARPA in 1990s), middleware, etc
  - Ergo don't push vendors to offer much better ☹
  - Vendors generally not very aware either

IP Multicast, Int-Serv Guaranteed Svc

- IP Multicast (IPMC)
  - Spams every “subscriber” at highest rate anyone wants it at
  - Can cause address instability; banned from some cloud computing environments
    - **Dr. Multicast: Rx for Data Center Communication Scalability.** Ymir Vigfusson, *et al*. *ACM SIGOPS 2010*, pp. 349-362.
- Int-Serv
  - Guaranteed Service only guarantees max, not average and does not handle jitter

OpenFlow (OF) & SW-Defined Net’s (SDN)

- Good per-flow network QoS
- But at net not MW level
  - Need management layer and some APIs above OF
- Incomplete: Still need to handle other non-network QoS+ properties: redundancy, confidentiality, authentication, ….
- Can be a lowest-common-denominator approach
- Interoperability and subsetting [see GridStat chap. of my book]
  - S. McGillicuddy, “Not all OpenFlow Hardware is Created Equal: Understanding the Options”, Open Network Foundation, 25 September 2013, available via www.opennetworking.org.
- No rate downsampling
- Utilities often don’t have a green field opportunity: have to be able to integrate many non-OF network assets, too

MPLS

- Weak statistical guarantees over {location, user, long time}
  - Meant to help ISPs coarsely provision bandwidth w/QoS, not for providing specific QoS for given data variable
  - E.g., Harris' FAA network has 30 minute statistical guarantees
  - Highly inappropriate (reckless) for closed-loop apps
- Only 8 categories (3 bits) of QoS treatment, yet many (hundreds, ?thousands) of QoS combinations very useful
  - Its not one size (or 8 sizes) fits all!
  - Flow with higher priority may not get better treatment than lower
  - Still have to track and manage ALL traffic if any hope
- But widely used (with IPMC) by utilities lately, because you can buy it from a router vendor
  - Because it has (some flavor of) QoS and 1→many superficially similar to what is needed!

IEC 61850: The Good (LOTS!)

- HUGE benefit compared to wires in substation
- Data model elegant
  - Opens up a lot of opportunities to exploit this semantic information in conjunction with power models, data delivery topologies, adaptation, default configuration or QoS settings, ….
- Substation Configuration Language (SCL) elegant

IEC 61850: The Bad

- **Complexity**
  - Far more complex than it has to be given the problem it is tackling
  - Double the size/bandwidth of IEEE C37.118 with no extra useful info
  - Feels to me like a spec doc by a 1975 Mechanical Engineer specifying HW not a 1995 (or later) SW Engineer specifying SW
- **Hype**
  - Almost sounds like it will cure cancer at times
    - PJM engineer: 4 substations (ISO has ~30% of the USA footprint)

IEC 61850: The Bad (2)

- **Performance**
  - Subscriber apps have to be able to detect missing and duplicates (no sophisticated fault-tolerant multicast)
  - GOOSE authentication via RSA signatures: way too expensive for many embedded devices
    - UIUC paper (Jaianqing Zhang and Carl Gunter, IEEE SmartGridComm 2010)
    - WSU paper (Hauser et al paper from HICS 45 (2012))
    - Later shared key extensions allow subscriber to spoof publisher ☹
  - GOOSE messages very CPU-intensive with ASN.1 integer fields etc, expensive for many embedded devices
  - Have to be careful that the multicast (Ethernet broadcast) does not overload small embedded devices
  - Note: **61850-90-5 is NOT middleware** (not even close)

IEC 61850: The Bad (3)

- **Misc**
  - $3K just to read the spec
  - Design by Committee before Full Implementation
  - Way better standardization models: IETF and OMG

"We reject: kings, presidents, and voting.

We believe in: rough consensus and running code."

– *David Clark, Internet pioneer*

“Any time you standardize beyond the state of the practice you are in trouble.”

– *Richard Schantz, father of middleware*

IEC 61850: The Bad (4)

- **Misc (cont.)**
  - PMUs often need many:one (to a PDC) not 1:many communication
  - Lack of a reference implementation and reference test suite
    - Have to test devices pairwise
    - Standard so huge many vendors don’t implement all of it; most vendors violate the standard in some way

IEC 61850: The Ugly

- Data Model is portable, but no configuration and other tools that are vendor-agnostic ☹
  - Can you spell this: V-E-N-D-O-R L-O-C-K-I-N ???
  - But it's a "standard" so the suits and MBAs want it
- WANs are very different from LANs: partial failures & widely-varying performance (incl. network jitter)
- 61850 assumes the same interface for a LAN will magically work in a WAN
  - Known by distributed computing practitioners and applied researchers to be false since <= 1990
    - See the "A Note on Distributed Computing" by Waldo et al

IEC 61850: The Ugly (2)

- 61850-90-5 is the WAN extension
  - Dec 2010 draft says communications redundancy is "crucial"
  - But the draft has less than one page on it (Sec 8.8) that has no meaningful details
  - IETF RFC 2991 it relies on has nothing about end-to-end latency, availability, exploiting a more controllable utility infrastructure, tradeoffs below, etc
  - Advanced multicast is hard, fault-tolerant is harder, real-time is harder yet, with security (not ruining perf.) worse
  - Wide range of properties could trade off, incl. latency, jitter, consistency, throughput, resource consumption, availability, ...
  - Do implementers (or drafters) know what this space of possible properties is, what tradeoffs their given implementations make? Very unlikely…
  - Do utilities/ISOs know what tradeoffs they are being sold, and how appropriate they are for them? Unlikelier!

IEC 61850: The Ugly (3)

- Bottom line: a lead control engineer from a large utility (with very forward-thinking, advanced ICT) to me
  - 2009: “No way in hell am I letting it [61850] outside my substations”
  - 2011: (ruefully) “I was overruled from above, because its ‘a standard’.”
    - But a standard for doing what? With what properties traded off?

Email from that Same Utility

[**Emphasis** is mine…. There are standards, and then there are STANDARDS …..]

I have little insight into the particulars, but I've been involved in conversations about aligning the IEC 61850 with the CIM (an elusive goal), plus some sidebar conversations on the "immaturity" of the standard (although its been kicking around for 10 years). I think the underlying reason for this perception is the **vendor equipment-specific configuration tools** for 61850 and how **each vendor cherry-picks the standard with little regard to its impact on the overall substation configuration problem faced by a utility**.

There is a need for a vendor-agnostic toolset that mirrors the utility engineering process for constructing (or upgrading) a substation, and the long-term maintenance of the substation configuration. This process goes through several hands over several years, starting with a substation designer and ending with project engineers. The designer typically has templates to follow for the design, necessarily at a high level to explain (and sell) the design. **The electrical equipment vendors associated with the utility at the beginning of the design may not be the same when the time comes to purchase equipment**. [… continued]

Email from that Same Utility (2)

[… continued] Thus the need for the vendor-agnostic toolset to support the design process and "seamlessly" transition to vendor-specific 61850 implementations as purchase orders are cut. Having all the tools CIM compliant would be a nice touch, but the two standards are not easily made compatible. There is much work to be done to solve the 61850 design/maintenance tool problem.

There are a lot of communication protocols in the electric grid domain, each reflecting the needs (and IT maturity state) of the time - from Modbus to DNP3 to 61850 to GridStat. **Unfortunately a utility cannot green-field a new grid as each new protocol is developed**, it has to ensure its deployed assets remain useful while trying to realize the benefits offered by maturing Information and Communications Technologies. That is a major driver behind the XYZ Advanced Lab - to determine which technologies have the potential to improve the XYZ grid's "ities" : reliability, stability, profitability, etc.

Outline

- **Optional Backup**
  - GridSim & GridControl Brief Overviews
    - Killer Apps for Cloud Computing
    - A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - **Why Middleware is Needed for IIE**
    - Ratatoskr RPC/GridStat with tuneable timeliness, redundancy, safety

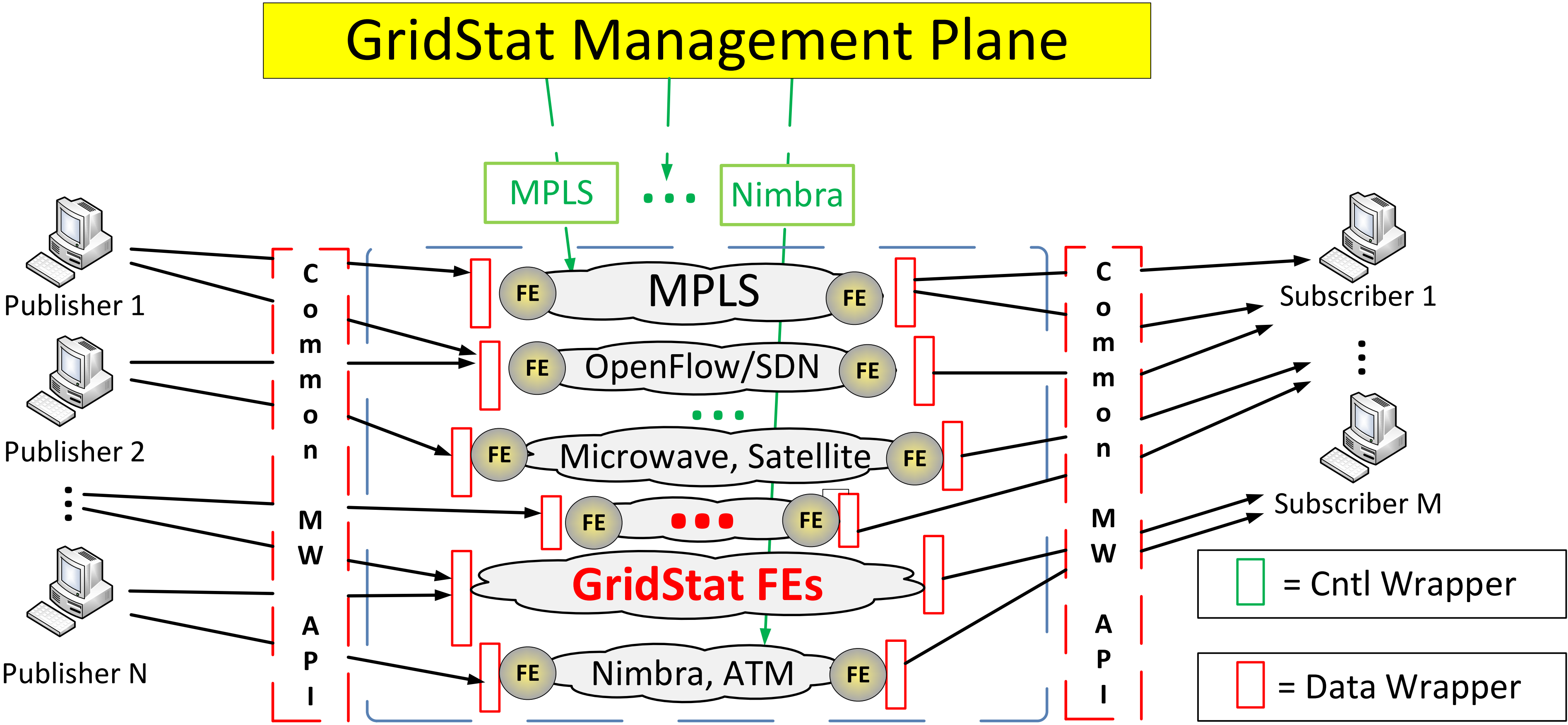
Middleware (MW) Integrating Legacy (Sub)Systems

Flow start (publisher) could also be RTU, substation router, OpenPDC, etc. i.e. not just a single sensor.

GS subscriber could be RTU, substation router, OpenPDC, …

GridStat is

- Real-Time
- Overlay (sometimes)
- Middleware



"…" could be BPL/PLC, 4G teleco, best-effort internet, etc.

The “Smart Grid” Community Agrees

- 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. (shown earlier, with URL)
  - Argued that, by the stated goals of the US “smart grid” leadership (GridWise and NIST), **middlware is needed for WAMS-DD**
  - Argued that if you care about QoS, needed even more
  - **Best Paper Award for “Connectivity” track**. This is the official communications/interoperability meeting for the pseudo-official “smart grid” community in the USA, namely DoE/GridWise and NIST/SmartGrid.
  - Paper also describes what middleware is and how standardized

Other Reasons Middleware is Needed

- Very hard problem for computer scientists to get right: combining mechanisms for network QoS and security and other QoS+ properties
  - Almost impossible for non-expert to get right
  - Without middleware, each application has to rewrite
  - Much better to let middleware developer do this, and **apps reuse**
- Future proofing: not locked into given mechanism
- Only way to do monitoring/policing on a per-update per-sensor basis (rather than ~1.5 Mbps RTU-CC)
- …. see Bakken book chapter on GridStat for 5 pages on this!

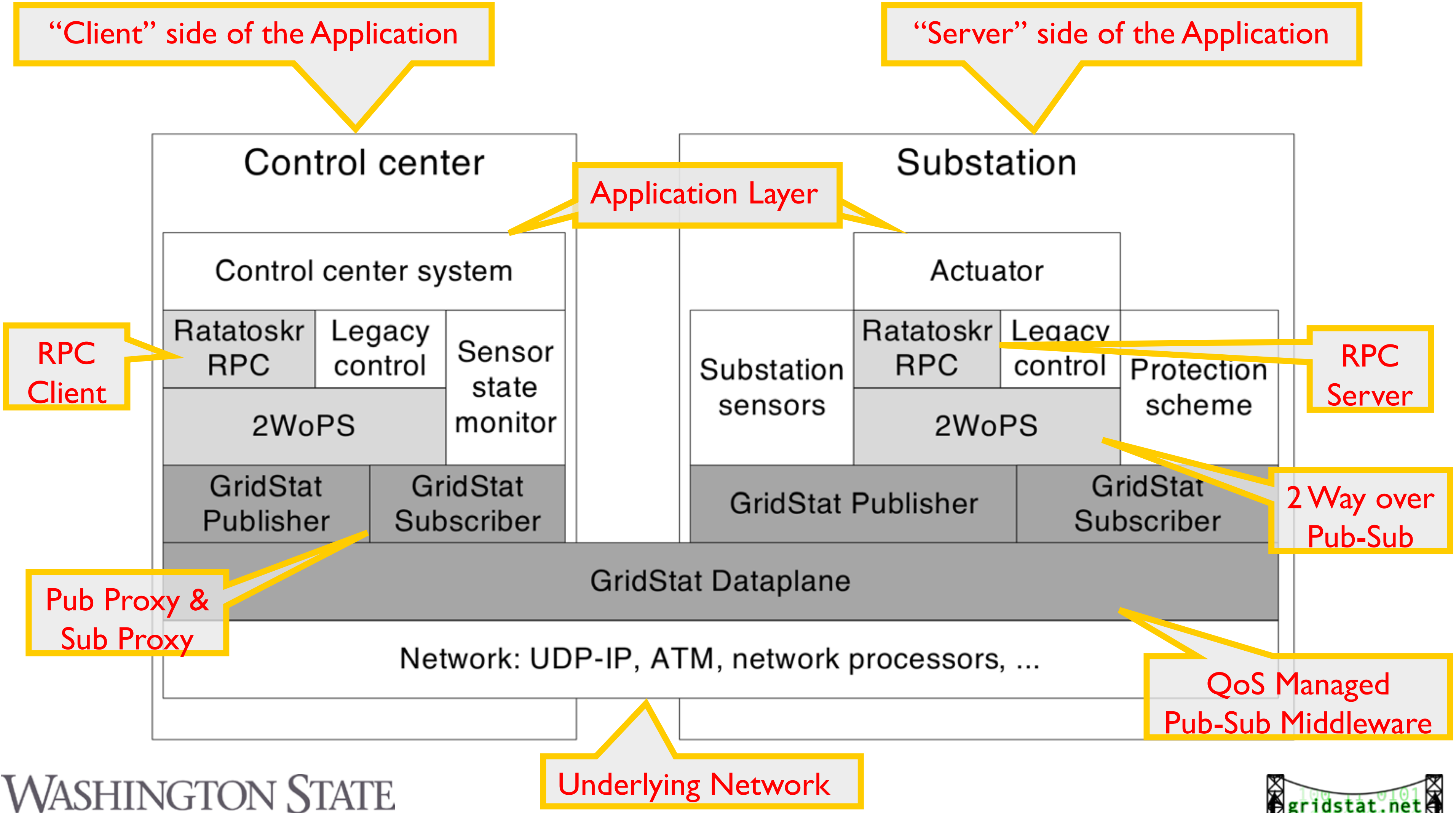
Outline

- **Optional Backup**
  - GridSim & GridControl Brief Overviews
  - – Killer Apps for Cloud Computing
  - – A Distributed Computing Critique of Some Power Protocols: IEC 61850(-90-5), MPLS, …
  - – Why Middleware is Needed for IIE
  - **Ratatoskr RPC/GridStat with tuneable timeliness, redundancy, safety**

Ratatoskr RPC Mechanism

- RPC mechanism targeted to set actuators in the Electric Power Grid, with tunable QoS for
  - Timeliness
  - Redundancy
  - Safety
- Uses a two-tiered architecture on top of a QoS managed publish-subscribe middleware
- Tier 1
  - Provides a 2 Way over Publish-Subscribe (2WoPS) communication protocol
- Tier 2
  - Provides RPC client and server functionality

Ratatoskr Architecture



"Client" side of the Application

"Server" side of the Application

Control center

Substation

Application Layer

Control center system

Actuator

Ratatoskr RPC

Legacy control

Sensor state monitor

Substation sensors

Ratatoskr RPC

Legacy control

Protection scheme

RPC Client

RPC Server

2WoPS

2WoPS

2 Way over Pub-Sub

GridStat Publisher

GridStat Subscriber

GridStat Publisher

GridStat Subscriber

GridStat Dataplane

Pub Proxy & Sub Proxy

Network: UDP-IP, ATM, network processors, ...

QoS Managed Pub-Sub Middleware

Underlying Network

Timeliness

- Ratatoskr RPC mechanism requires an underlying QoS managed Pub-Sub middleware
- One of the QoS dimension is end-to-end delay of a message delivery, hence timeliness
- The timeliness guaranties that the Pub-Sub middleware provides are exposed to the RPC call interface
  - User specifies the desired and required end-to-end delay when setting up the RPC "connection"

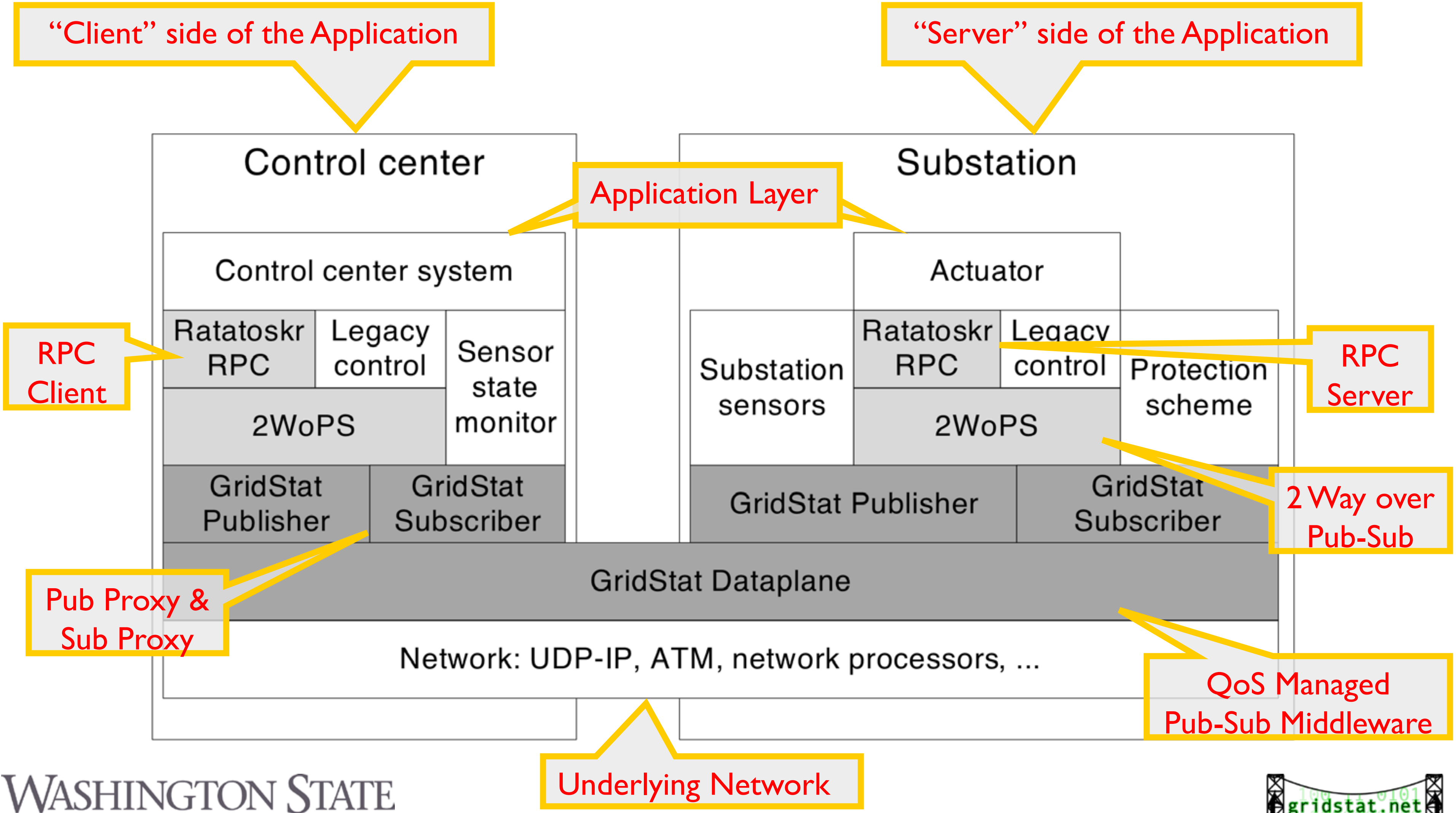
Redundancy

- Three Redundancy Techniques provided
  - Spatial Redundancy
    - How many disjoint network paths should the messages be sent over
  - Temporal Redundancy
    - How many copies of the same message should be sent and with what delay between each send
  - ACK/Resend
    - Require an explicit ACK to be sent back to the sender
    - Can specify how many resends of the message should be done if no ACK is received
- The user can combine any of the techniques as needed by the application
  - Use Spatial and Temporal for time delay sensitive application
  - Use ACK/Resend for non time critical application

Ratatoskr RPC Mechanism

- RPC mechanism targeted to set actuators in the Electric Power Grid, with tunable QoS for
  - Timeliness
  - Redundancy
  - Safety
- Uses a two-tiered architecture on top of a QoS managed publish-subscribe middleware
- Tier 1
  - Provides a 2 Way over Publish-Subscribe (2WoPS) communication protocol
- Tier 2
  - Provides RPC client and server functionality

Ratatoskr Architecture



"Client" side of the Application

"Server" side of the Application

Control center

Substation

Application Layer

Control center system

Ratatoskr RPC

Legacy control

Sensor state monitor

2WoPS

GridStat Publisher

GridStat Subscriber

Actuator

Substation sensors

Ratatoskr RPC

Legacy control

Protection scheme

2WoPS

GridStat Publisher

GridStat Subscriber

GridStat Dataplane

Network: UDP-IP, ATM, network processors, ...

RPC Client

RPC Server

2 Way over Pub-Sub

Pub Proxy & Sub Proxy

QoS Managed Pub-Sub Middleware

Underlying Network

WASHINGTON STATE UNIVERSITY

gridstat.net

Timeliness

- Ratatoskr RPC mechanism requires an underlying QoS managed Pub-Sub middleware
- One of the QoS dimension is end-to-end delay of a message delivery, hence timeliness
- The timeliness guaranties that the Pub-Sub middleware provides are exposed to the RPC call interface
  - User specifies the desired and required end-to-end delay when setting up the RPC “connection”

Redundancy

- Three Redundancy Techniques provided
  - Spatial Redundancy
    - How many disjoint network paths should the messages be sent over
  - Temporal Redundancy
    - How many copies of the same message should be sent and with what delay between each send
  - ACK/Resend
    - Require an explicit ACK to be sent back to the sender
    - Can specify how many resends of the message should be done if no ACK is received
- The user can combine any of the techniques as needed by the application
  - Use Spatial and Temporal for time delay sensitive application
  - Use ACK/Resend for non time critical application

Safety

- In some large scale infrastructures the client may not have the latest “state/view” of the condition
  - If an actuator is set wrongly serious consequences can occur
- Similarly the effect of setting an actuator may not be what was intended
- Pre-and Post conditions are built into the call semantic
  - Pre-conditions are conditional expressions that are evaluated at the server side before the execution of the RPC call
    - In case the condition evaluates to false, call is aborted
  - Post-condition are conditional expressions that are evaluated after (delay user defined) the RPC call is executed
    - In case the condition evaluates to false, this is reported back to the application through an exception

Safety

- In some large scale infrastructures the client may not have the latest “state/view” of the condition
  - If an actuator is set wrongly serious consequences can occur
- Similarly the effect of setting an actuator may not be what was intended
- Pre-and Post conditions are built into the call semantic
  - Pre-conditions are conditional expressions over GridStat-updated varaibles that are evaluated at the server side before the execution of the RPC call
    - In case the condition evaluates to false, call is aborted
  - Post-condition are conditional expressions over GridStat-updated variables that are evaluated after (user-defined failure) the RPC call is executed
    - In case the condition evaluates to false, this is reported back to the application through an exception
    - Idea here is condition gives some physical end-to-end confirmation that the call succeeded and that the actuator had not failed