GridCloud Cloud-hosted high-assurance system to monitor the electric power grid

Demonstration and Experiments

sponsored by the Department of Energy ARPA-E program

Question

How can today's computing clouds support **high-reliability, real-time power grid monitoring software**?

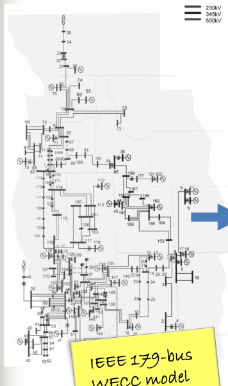
Requirements

- Work at the scale of major North American Interconnects
- Accept and process thousands of sensor data streams, at rates of multiple samples per second per stream

Proposed Answer

- Develop a platform that exploits the cloud's raw **speed** and **elasticity** and hides its **complexity** from power grid application developers
- Use **redundant communication** and **computation** to **mask failures** and **performance transients** that frequently occur in the cloud
- Use Isis^2's advanced distributed computing techniques to provide **consistent views** of replicated computations and data

GridCloud Demonstration: Hierarchical Linear State Estimation for 179-bus WECC system model using thousands of simulated PMU data streams

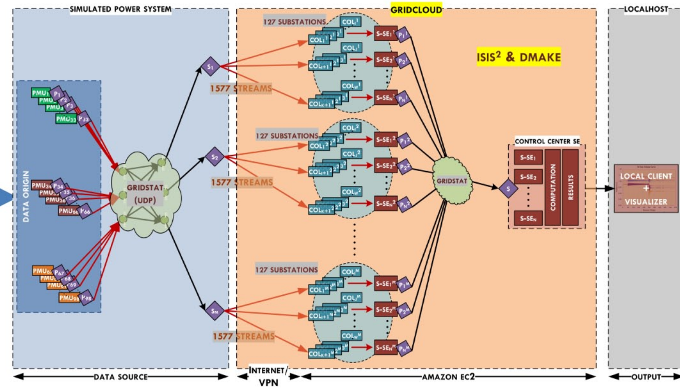


IEEE 179-bus WECC model

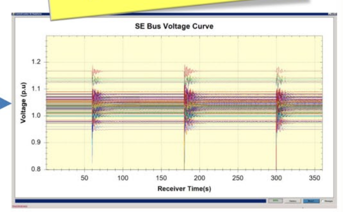
	Aggregate Data Rate (Mb/s)	Total Connections	Substation SEs
179-bus (3 replicas)	**141**	**4,731**	**381**
Coming Spring '14 WECC 6K-bus (3 replicas)	~570	~18,000	~1,000

Powertech TSAT simulator → Recorded PMU Data →

1577 PMUs
360 seconds
1.4 GBytes

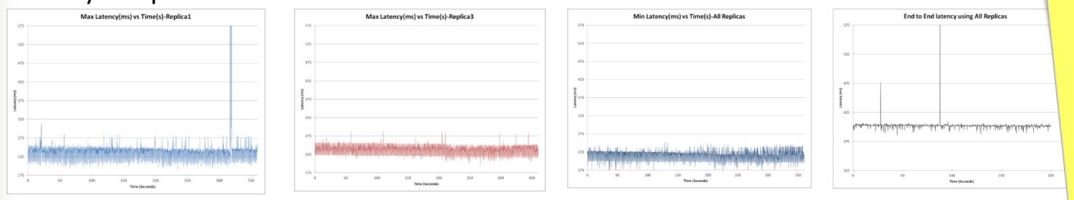


At 5 solutions/sec, SE results expose wide-area system modes following faults



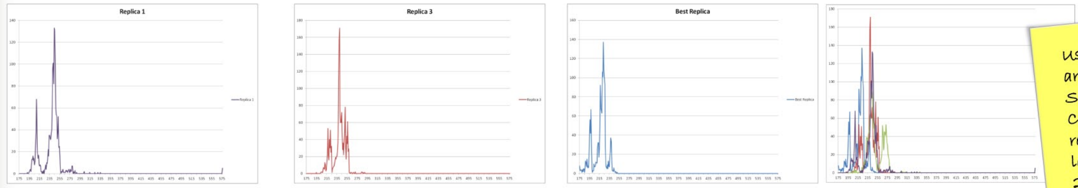
GridCloud performance: latency and jitter of data delivery and computation

Latency samples



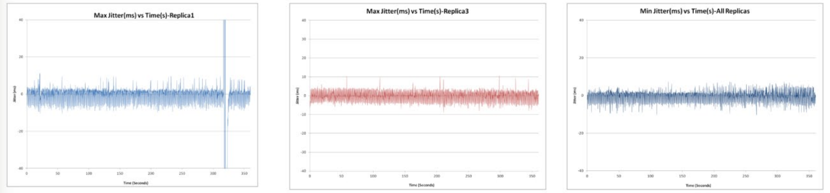
End-to-end latency is the time from when PMU samples are sent until the Control Center SE result is available: about 400ms

Latency histograms



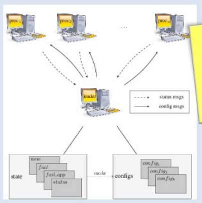
Using first-to-arrive substation SE result at the Control Center reduces typical latency by about 25ms and eliminates outliers

Jitter samples

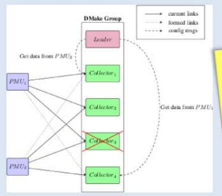


DMake

- Configures 49 machines, 1577 processes
- Uses 1 static IP address for dedicated "leader" machine, rest are dynamic
- Identifies and relaunches failed processes in less than 300ms
- Identifies failed machines and moves work to spare node or new node
- Load-balances work among the collector machines



DMake components and flows



DMake-directed failover

Lessons Learned

About cloud instance sizing

- Good performance requires sufficient resources in cloud instances: Amazon EC2 *C3: Large* and *C3: Extra Large*

About process creation and monitoring

- The demonstration uses thousands of processes on tens of cloud virtual machine instances: DMake is invaluable for creating and monitoring these processes

About the SE

- 1 to 5 solutions/sec is apparently feasible at grid scale
- Solutions are available about 400ms after measurements are made
- Replicated data delivery and substation computation are needed to reliably achieve this performance
- Variability of end-to-end latency is now attributable to performance of the unreplicated Control Center SE

Cornell **WSU**

Theo Gkountouvas | Carl Hauser | Dave Anderson

www.cs.cornell.edu/Projects/Gridcontrol/

Cornell University

Washington State University
World Class. Face to Face.

February 19, 2014