Synchrophasor Data Systems at Dominion Virginia Power

2012 Grid Protection Alliance
User's Forum

August 22, 2012
Atlanta, GA

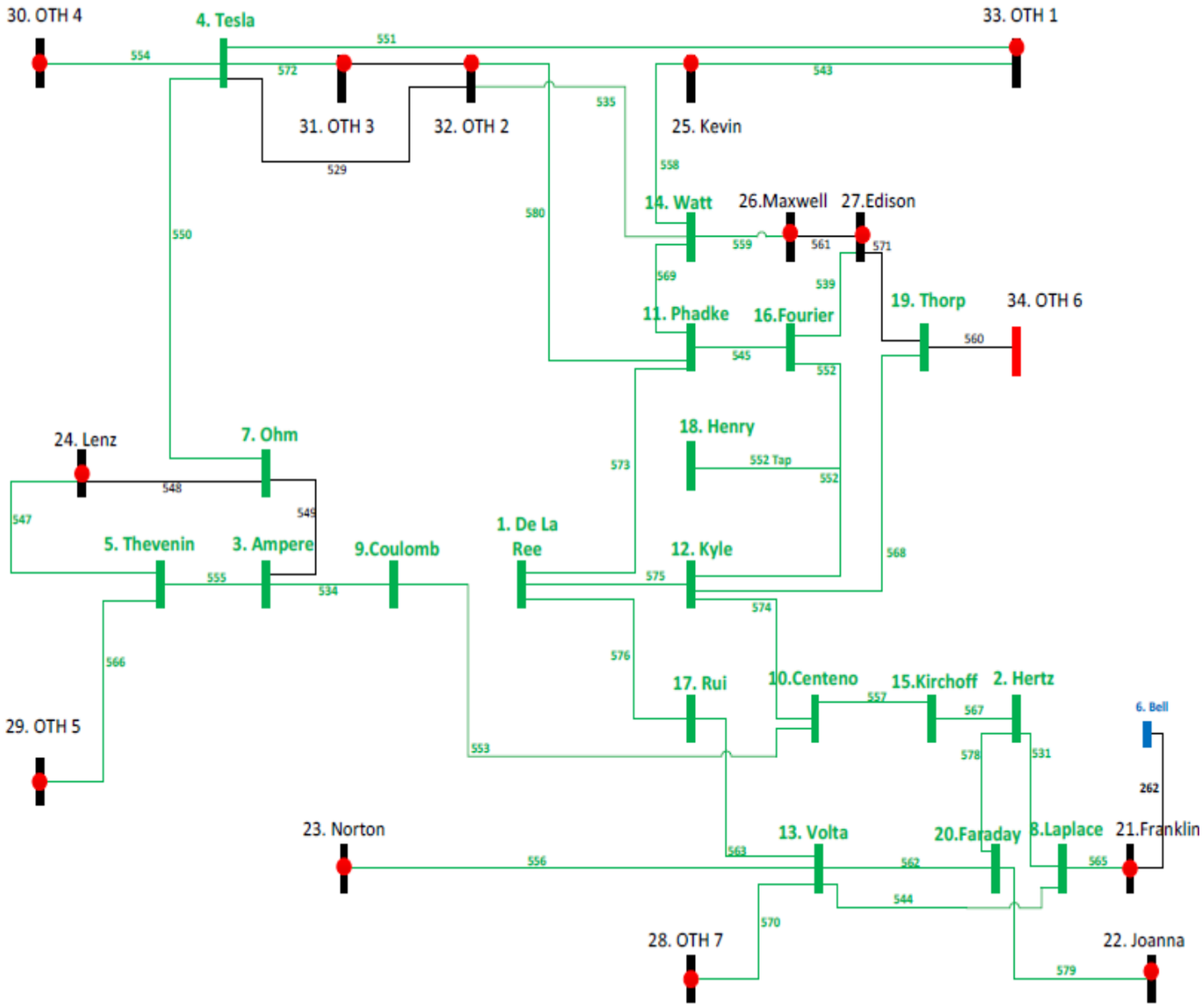
Kevin Jones
kdjones@vt.edu

Kyle Thomas
kyle.thomas@dom.com

Initial Synchrophasor Objectives

- Obtain wide-area monitoring of Dominion's EHV system with synchrophasors from PMU installations throughout the system
- Develop and implement novel synchrophasor applications:
  - Synchrophasor-based tracking three-phase state estimator
  - Transducer Calibration
  - Detect and analyze unbalanced conditions in the EHV network
  - Islanding strategies during catastrophic system event
  - Visualization tools for the synchrophasor deployment & applications

Initial Synchrophasor Deployment



Complete observability of Dominion's internal 500kV network from:

- 22 EHV Substations
- 100+ Relays as PMUs
- 98% of EHV Bkr Contacts
- 1000+ Measured Phasors (A, B, C, Pos Seq)
- 600+ Estimated Phasors (A, B, C, +, -, 0 seq)

Substation PMU & PDC Design

Synchrophasors are from SEL digital relays:

- Relays are either stand-alone PMUs or dual-use devices

- Each synchrophasor substation has a local PDC:
  - Stores highest res. data (60 mes/ sec) for short term (one month)
  - Output down-sampled (30 mes/ sec) data to SOC

First PMU Installation 2010

Substation PDC

SEL-3373

Substation PMUs

SEL-487 & SEL-421

Scalability of Dual-Use Relay PMUs and Substation PDCs

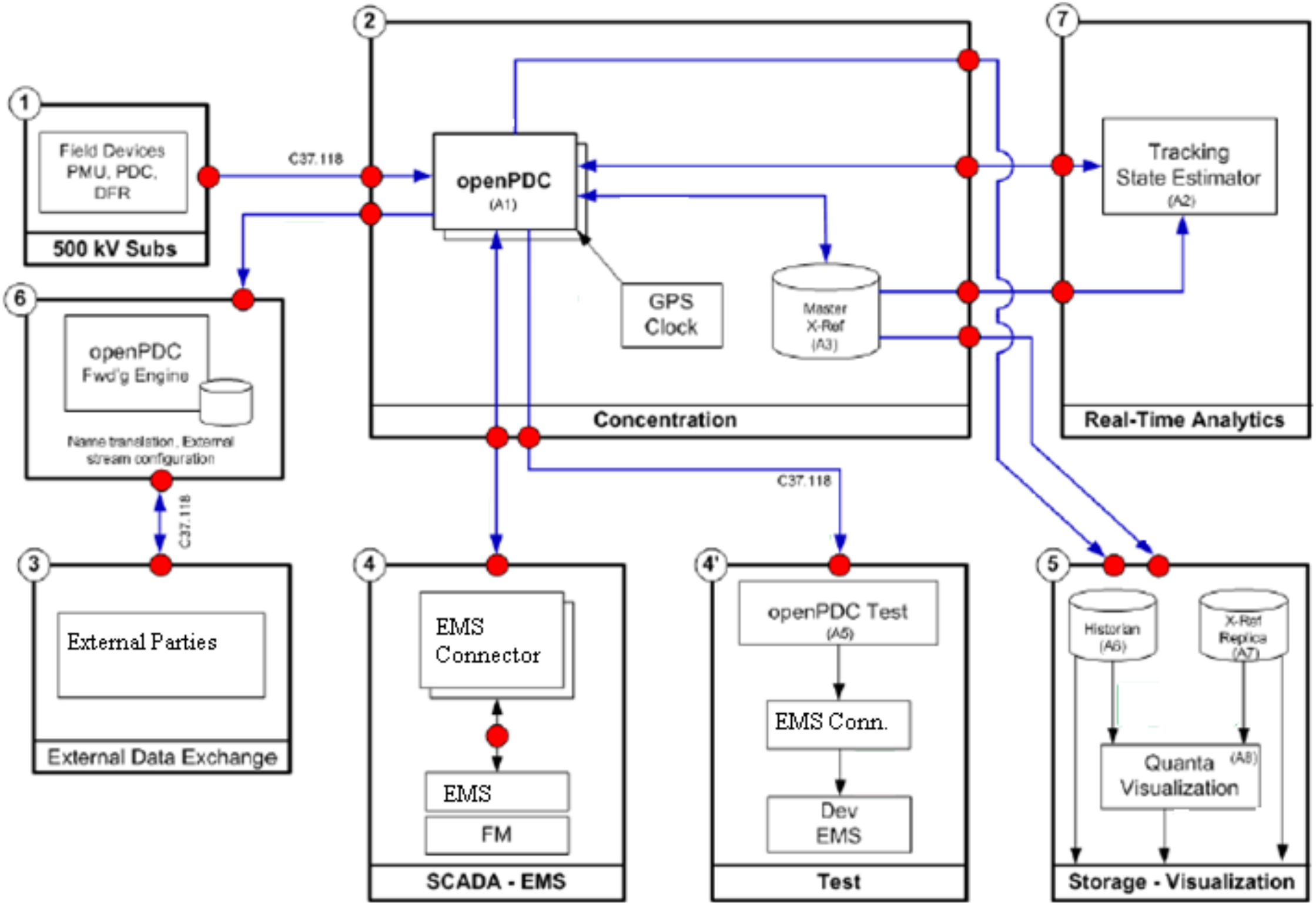
- Most of our PMUs are dual-use relays, performing protection and synchrophasor functions

- As digital relays continue to replace EM relays, there is an opportunity to take advantage of scheduled replacements by using protective relays with synchrophasor capabilities

- And with local substation PDCs, PMUs can be automatically placed in-service with local storage, using existing reliability improvement projects

- Then when the grid network system & bandwidth is ready, local PDCs only require a simple configuration update to send output stream to Operation Centers

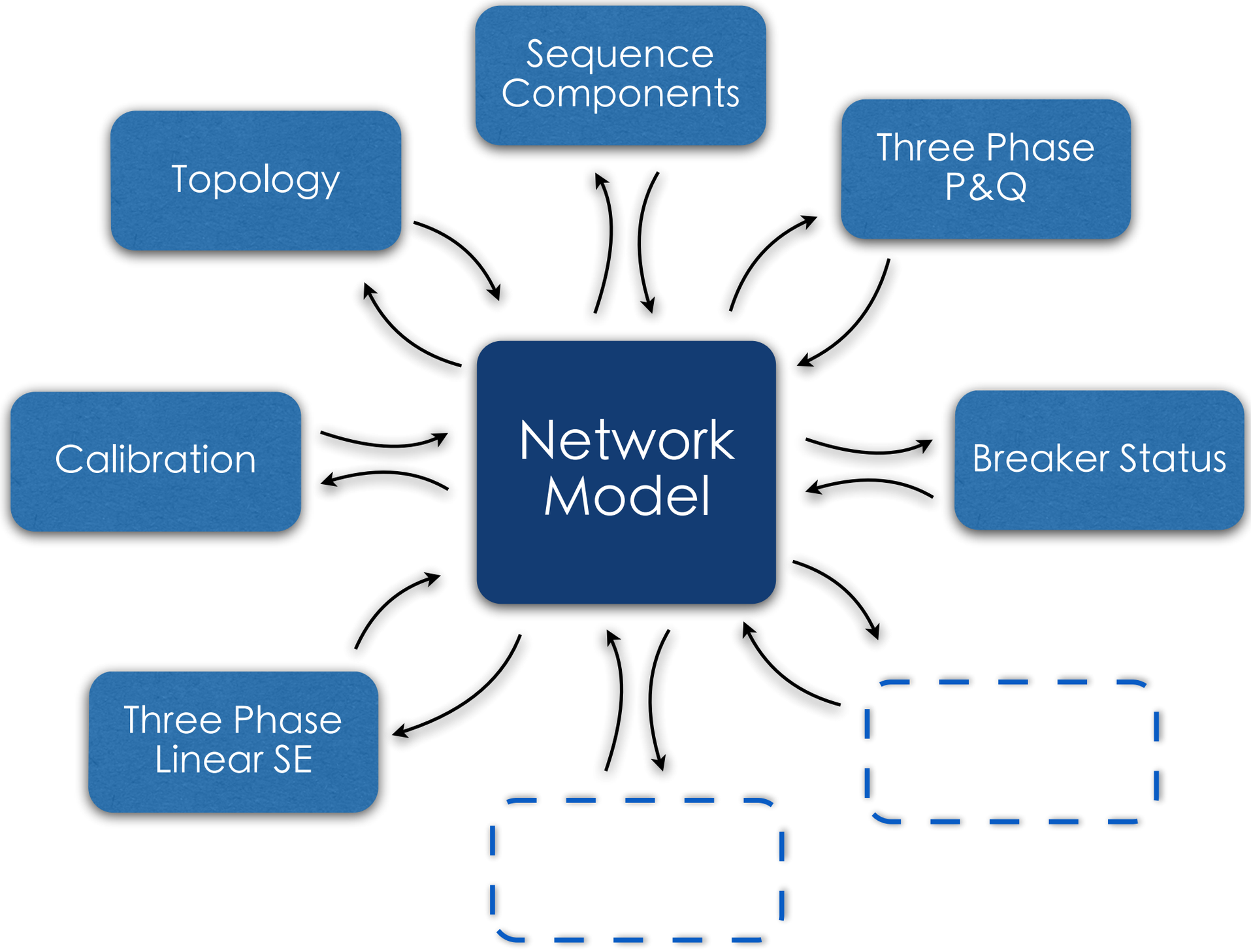
SOC Synchrophasor Architecture

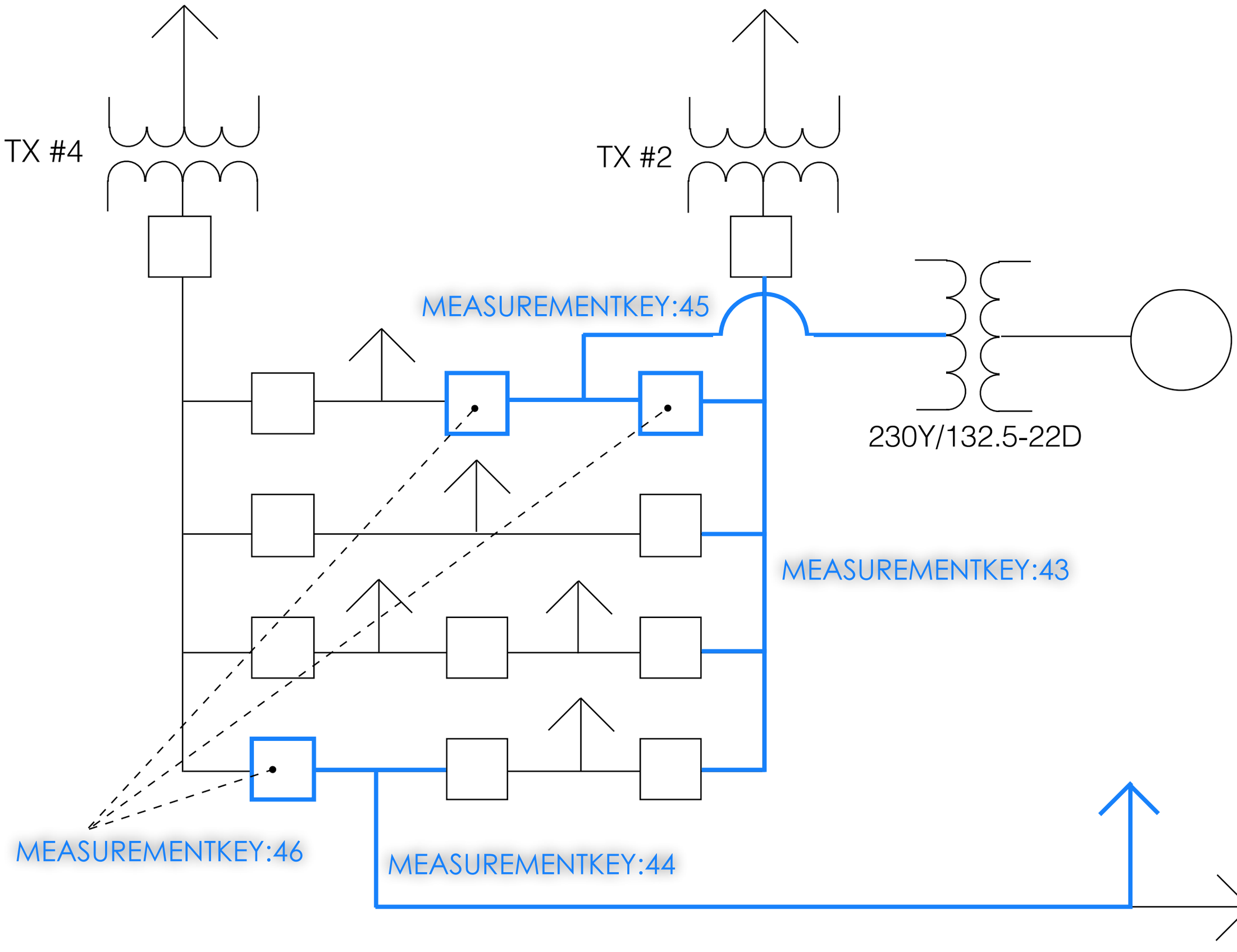


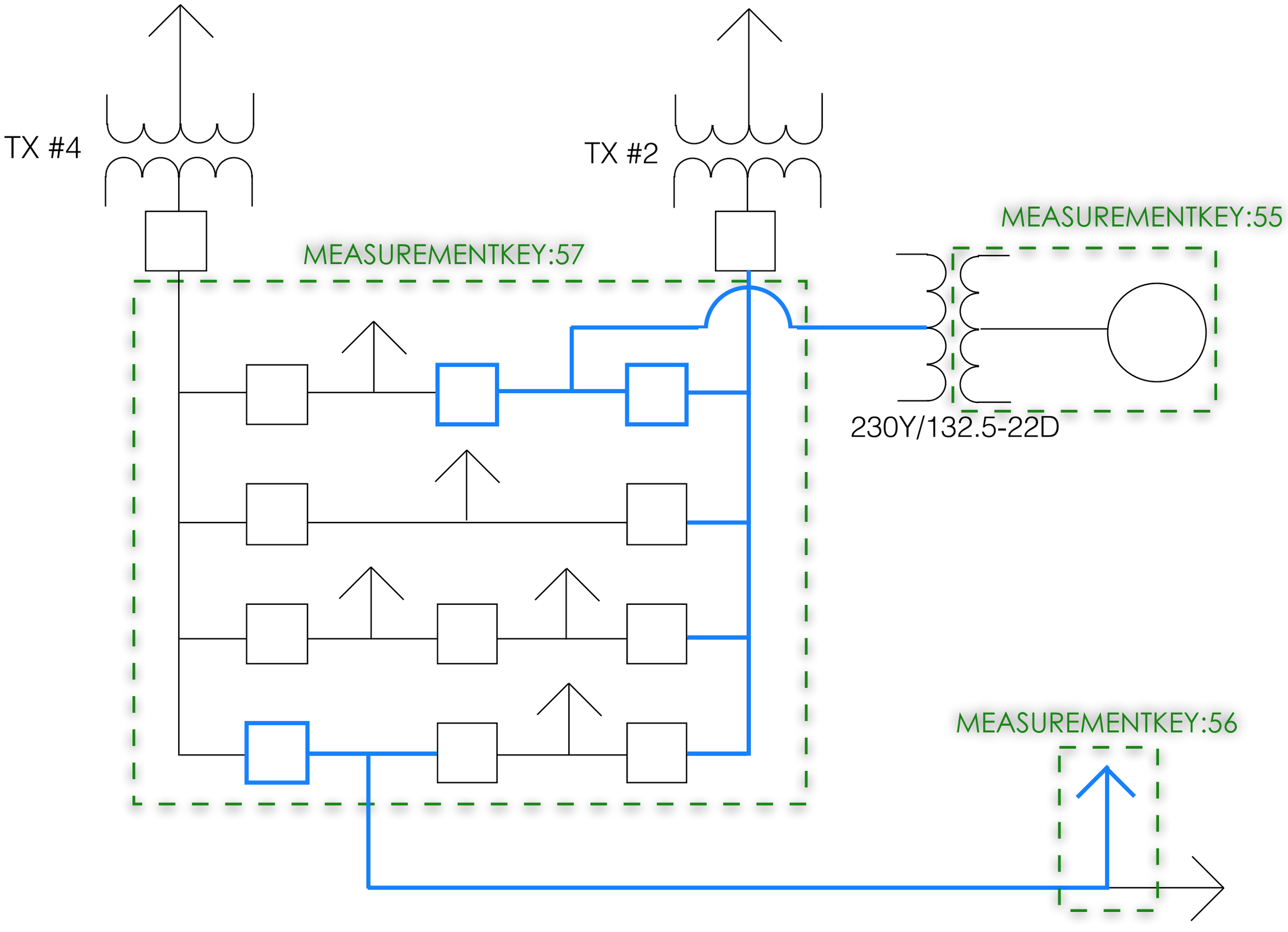
openPDC As Central PDC Solution

- openPDC was the best solution for our many requirements we knew up-front
  - RTO requirements: *name translations & phase angle rotation*
  - Platform to create the custom synchrophasor applications

- As implementation progresses, new requirements continue to be realized and implemented using openPDC
  - Local substation PDC "wait-time" setting relied on openPDC's Quality flag and latency statistics
  - New (and improvements to) synchrophasor applications


Sequence Components
Topology
Three Phase P&Q
Network Model
Calibration
Breaker Status
Three Phase Linear SE


TX #4
TX #2
MEASUREMENTKEY:45
230Y/132.5-22D
MEASUREMENTKEY:43
MEASUREMENTKEY:46
MEASUREMENTKEY:44


TX #4
TX #2
MEASUREMENTKEY:57
MEASUREMENTKEY:55
230Y/132.5-22D
MEASUREMENTKEY:56

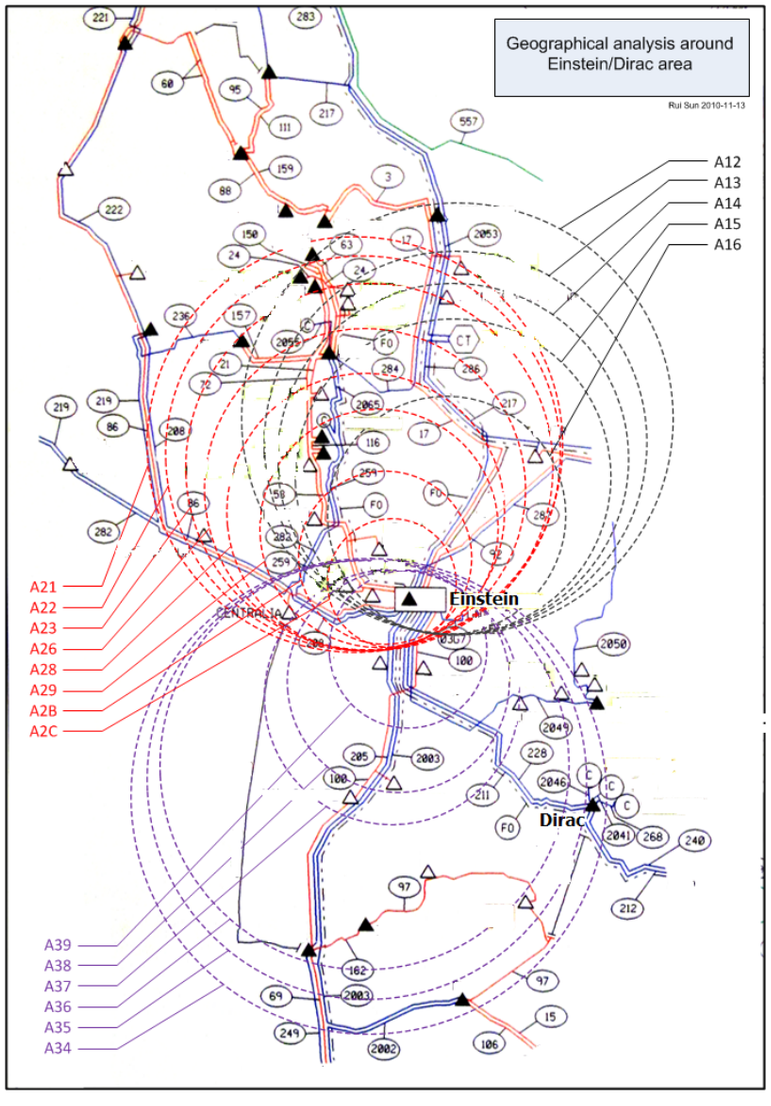
```
// Extract the raw measurements from the frame
foreach (IMeasurement measurement in frame.Measurements.Values)
{
    if (networkModel.InsertValueForKey
    (measurement.Value, measurement.Key.ToString())) { }
    else
    { OnStatusMessage("Key is invalid: " + measurement.Key.ToString()
                                        + " " + measurement.Value.ToString()); }
}
```

$$\mathbf{x} = \left[\left(\mathbf{H}^{\mathbf{T}} \mathbf{H} \right)^{-1} \mathbf{H}^{\mathbf{T}} \right] \mathbf{z}$$

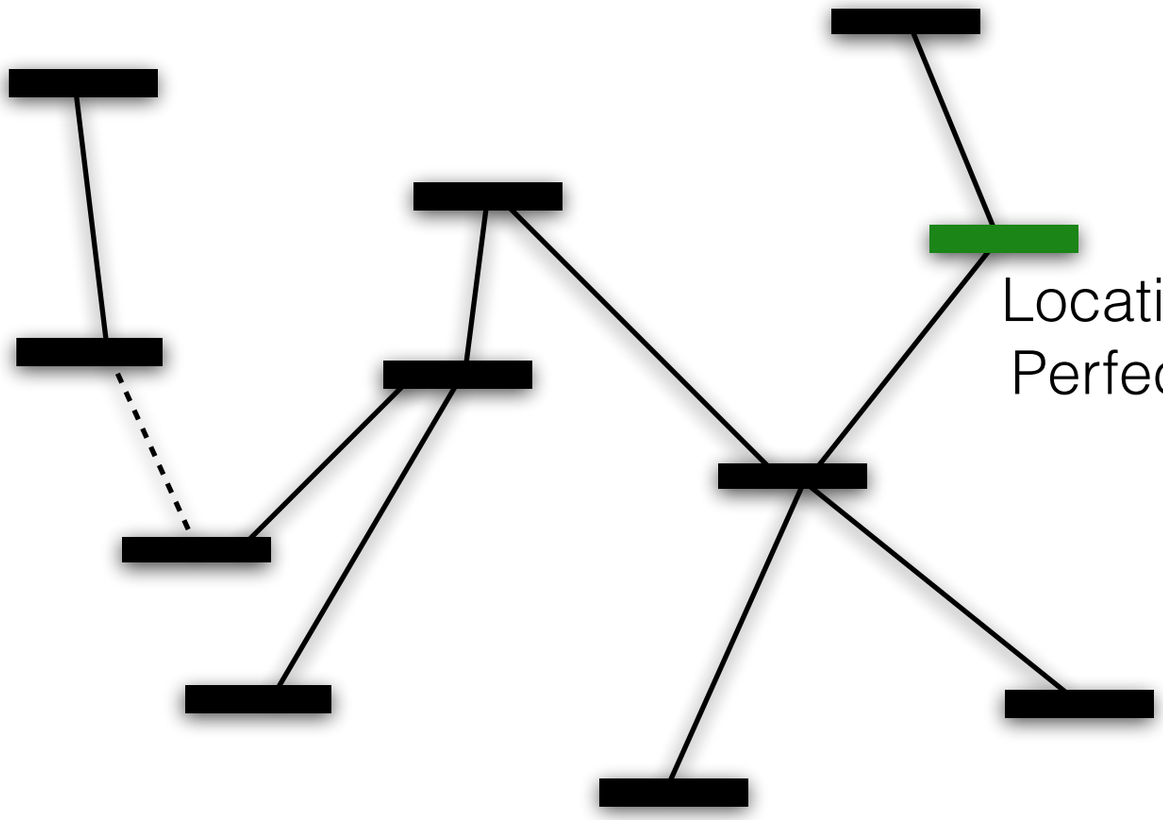
```
// Calculate the system state
networkModel.StateVector =
      PsuedoInverseOfSystemMatrix * networkModel.MeasurementVector;
```

Islanding

- Decision tree calculated offline
- Positive sequence phasors across the system
- 3 geographical regions
- Severity index



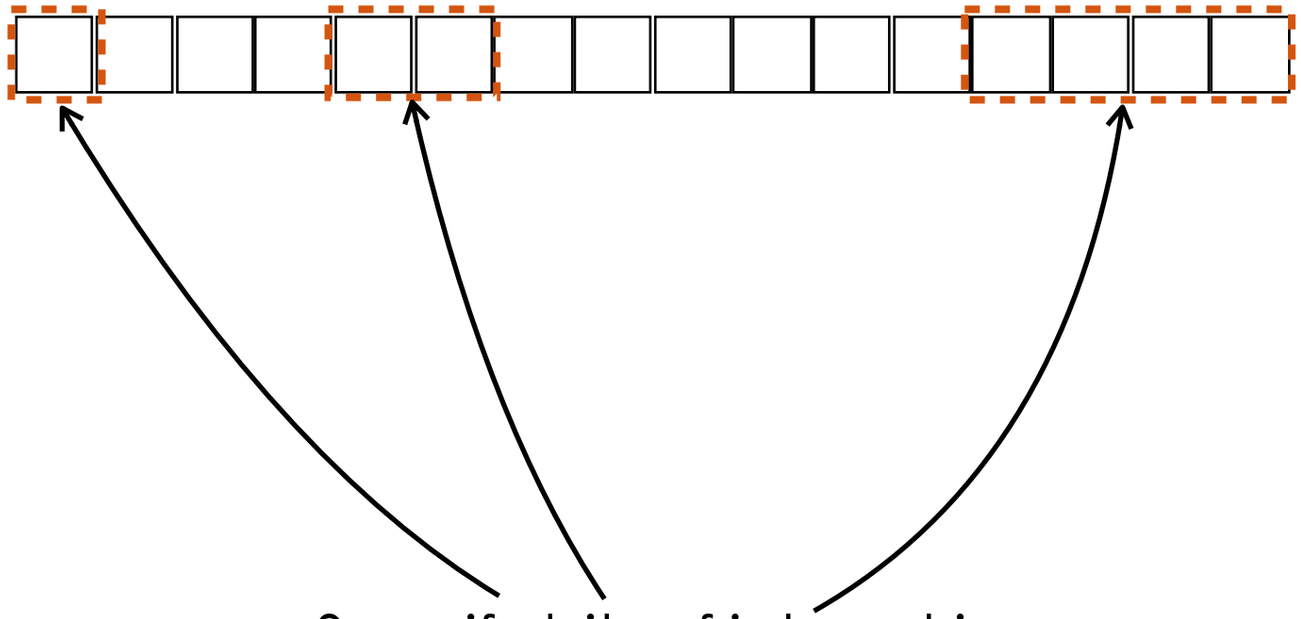
Calibration



- Needs a perfect voltage measurement
- Build a tree of measurements
- 16 system-wide samples across 24 hours
- Linear solution

Status Flag Parser

- Selectively parse bits from PMU status word
- Convey measurement information to applications & operators
- Example: Relay used as PMU fed test signals during relay testing

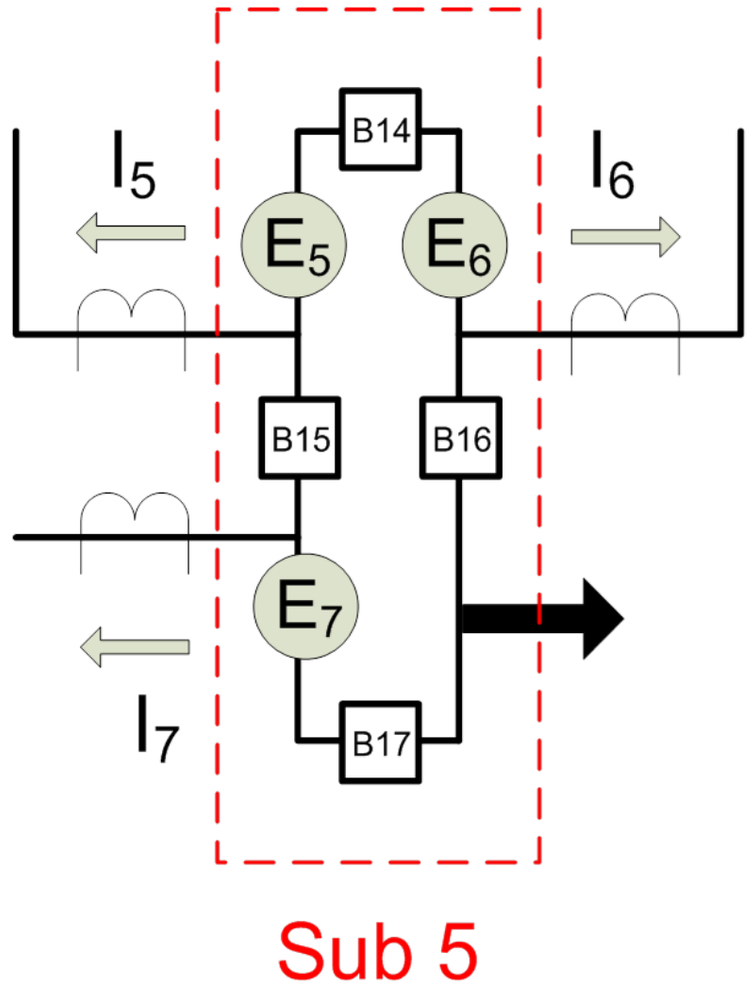


Specify bits of interest in Connection String

Syntax: `bitName=true;`

Topology Processing

- Breaker statuses from PMU digital word
- Current flow magnitudes
- Determines branch status



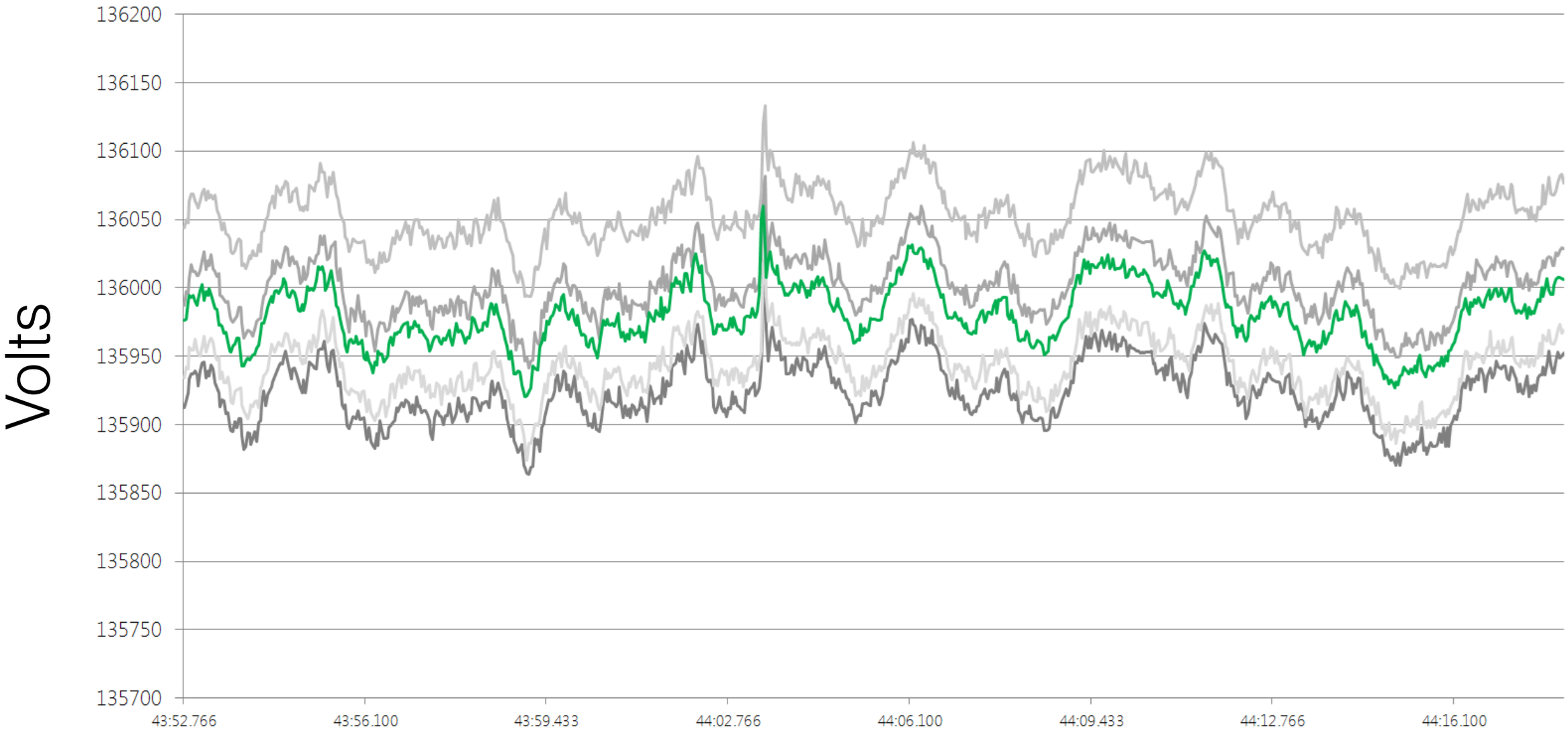
Three Phase Linear SE

- A, B, C, +, -, 0
- Directly measuring the system state
- No scan times
- No divergence

$$\mathbf{z} = \begin{bmatrix} \mathbf{II} \\ \mathbf{YA} + \mathbf{Y_s} \end{bmatrix} \mathbf{x} + \mathbf{e}$$

$$\mathbf{x} = \left[\left(\mathbf{H}^{\mathbf{T}} \mathbf{H} \right)^{-1} \mathbf{H}^{\mathbf{T}} \right] \mathbf{z}$$

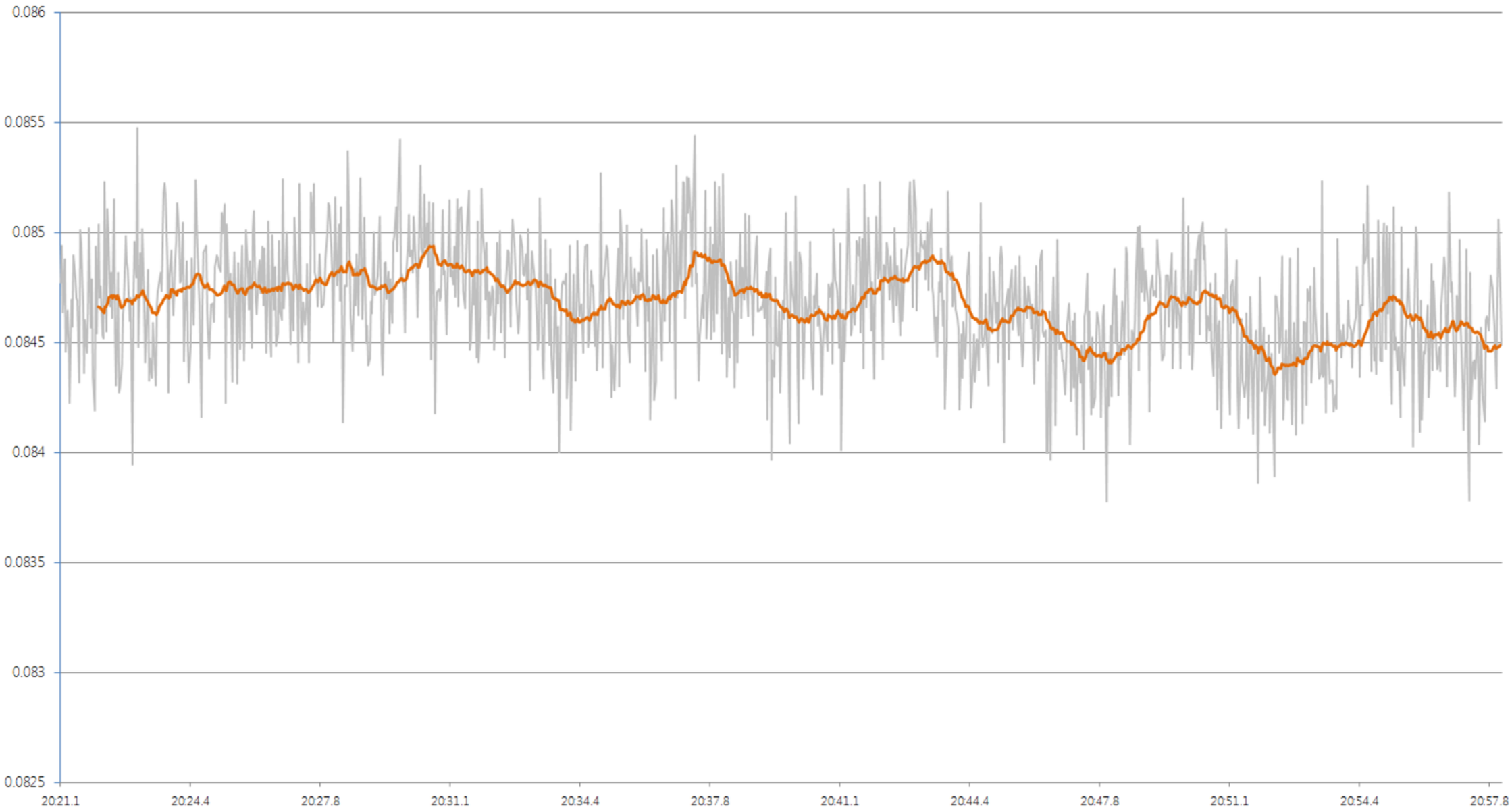
Phase A Voltage Magnitude of 230kV Bus



Phase A Measurement

Phase A Estimate

Ratio of Negative Sequence to Positive Sequence of 230kV Line



1 second moving average

Overview

- Three-Phase Linear State Estimation
- Negative Sequence Monitoring
- Topology Processing
- Instrument Transformer Calibration
- Islanding Detection
- Three-Phase Complex Power Flow Calculator
- Zero Sequence Monitoring
- Breaker Status Parser
- Status Flag Parser
- PJM Naming Convention & Phasing Convention Mapping

Some Small Lessons

Console output frequency in Connection String

```
// Count the number of frames to control console output
numberOfFrames++;

// Rollover the number of frames so it does not reach infinity
if (numberOfFrames > 2592000) // 24 Hours
{ numberOfFrames = 0; }
```

⋮

```
 if (numberOfFrames % outputFrequency == 0)
 {
     OnStatusMessage("Some useful troubleshooting info");
 }
```

Some Small Lessons

An Adapter which just prints output measurements

```
foreach (IMeasurement measurement in frame.Measurements.Values)
{
    if (numberOfFrames % outputFrequency == 0)
    {
        OnStatusMessage(measurement.Key.ToString() +
                        " " + measurement.Value.ToString());
    }
}
```