

# GPA Products Overview

## *Tutorial Session 1*

J. Ritchie Carroll

August 12, 2014

# GPA Products

- Grid Solutions Framework
- openPDC
- SIEGate (supersedes openPG)
- openHistorian
- openXDA
- PMU Connection Tester
- GEP Subscription Tester
- *New: PDQTracker*
- *New: substationSBG (Secure Buffered Gateway)*

# GPA Project Relationships

## Grid Solutions Framework (GSF)

<http://gsf.codeplex.com/>



## GSF Implementations:

openPDC / openHistorian / SIEGate

<http://openpdc.codeplex.com/>

# Multiple Open Source Projects

## Codeplex Hosted

- Grid Solutions Framework
  - <http://gsf.codeplex.com/>
- Secure Information Exchange Gateway (SIEGate)
  - <http://siegate.codeplex.com/>
- Open Source Phasor Data Concentrator (openPDC)
  - <http://openpdc.codeplex.com/>
- Open Historian
  - <http://openhistorian.codeplex.com/>
- PMU Connection Tester
  - <http://pmuconnectiontester.codeplex.com/>

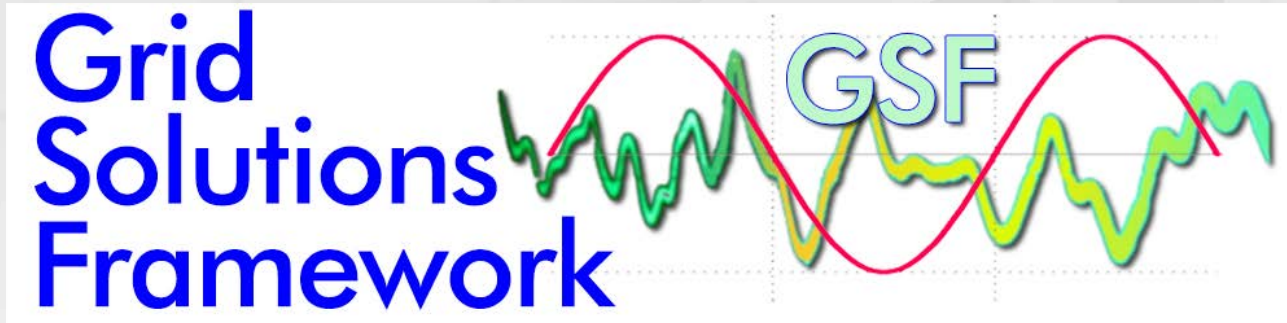
# Benefits of CodePlex Hosting

- Team Foundation Server project source control
  - This directly integrates with Visual Studio
- Project contributor forks or patches
  - This allows contributors to suggest formal code updates
- Project release downloads
  - This allows us to control major releases and track downloads
- Discussion forums & mailing lists
  - This allows users to help users and request community help
- Wiki and documentation pages
  - This allows up-to-date online documentation
- Bug and feature request tracker
  - This allows users to post issues for resolution

# Accessing Online Documentation

- All online documentation is continually updated by both GPA and contributors.
- Typically you need only go the project's CodePlex site in question and click the "Documentation" tab to get started with system documentation.
- For example, here is the [openPDC Documentation Link](#) - on this page you can navigate to:
  - Getting Started
  - Frequently Asked Questions
  - Major Component Overviews
  - How-to Guides, etc.

# GPA Development Framework



# Grid Solutions Framework (GSF)

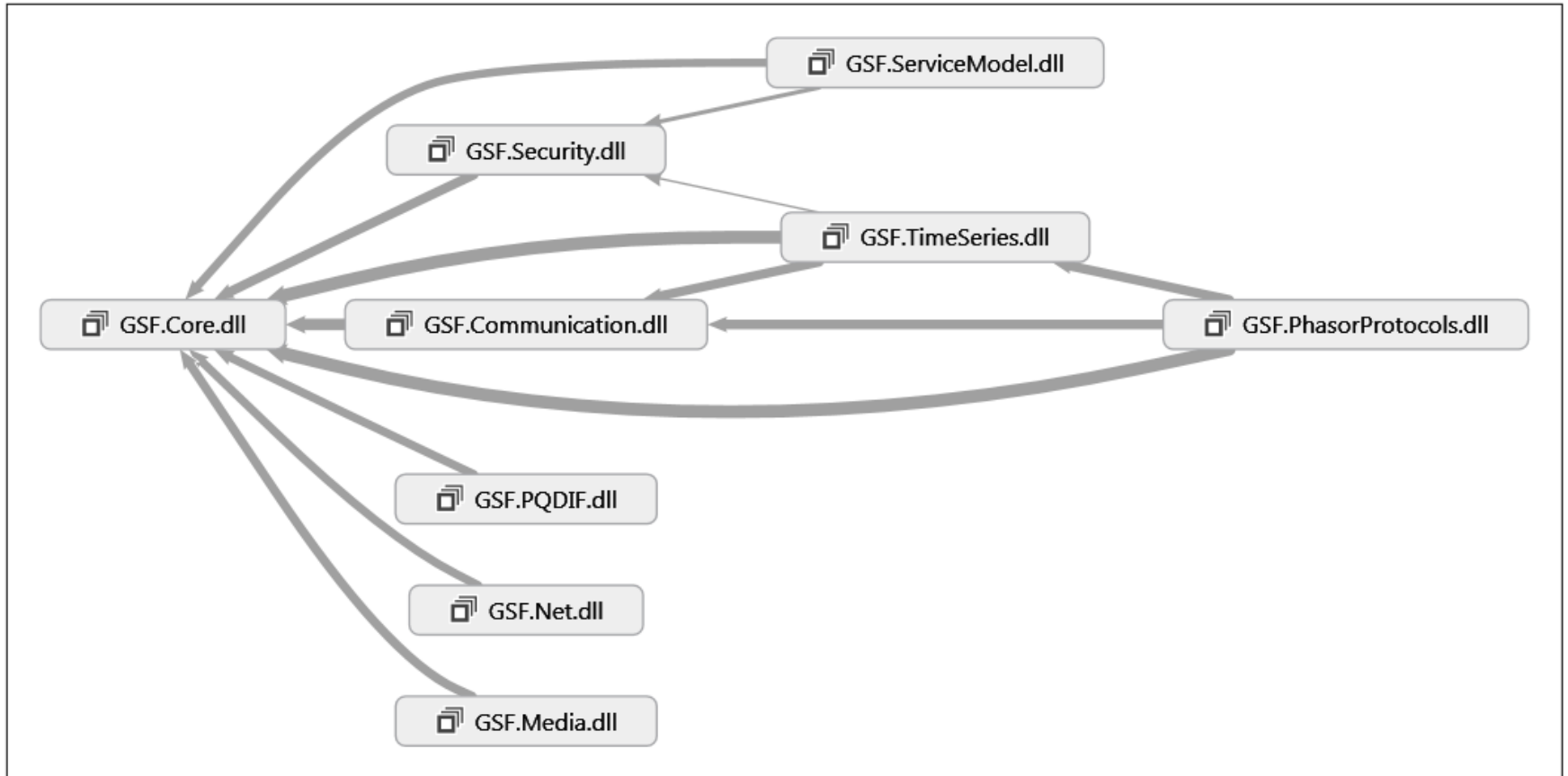
- New software development platform that was initially created as a combination of the Time-Series Framework and the TVA Code Library with a goal to increase performance and security
- Full namespace refactoring and projects targeted to compile with the new Microsoft 4.5 Framework (Released August 2012)
- New core features and improvements are only implemented in the <http://gsf.codeplex.com/>



# Grid Solutions Framework Purpose

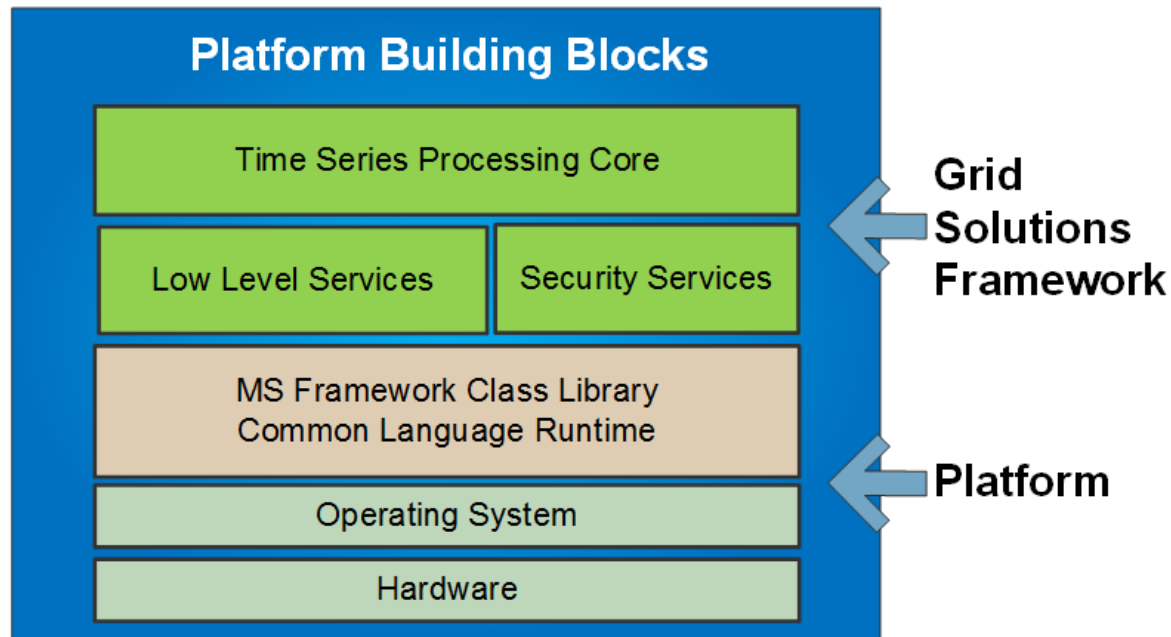
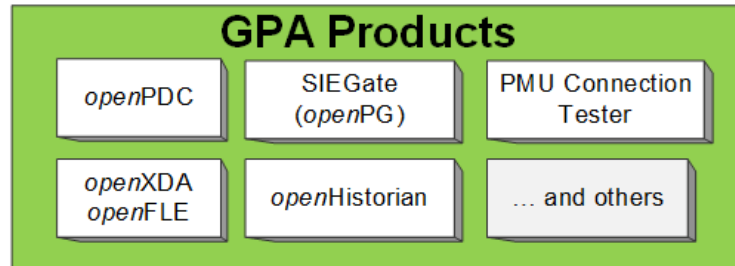
- General purpose open source library of .NET code used by many utilities and various open source projects that contains a large variety of code useful for nearly any .NET project.
- Consists of hundreds of classes that extend and expand the functionality included in the .NET Framework making more complex .NET features easier to use and adds functions not included in the .NET Framework.
- Used since it provides a standard development platform, improves development speed and increases reliability.

# GSF Primary Assemblies



**66 Total Assemblies, Over 329,000 Lines of Code and Comments,  
200,000 Lines of Code without Comments**

# All Latest Products are Built using GSF



# GSF Time-series Library

- Core collection of classes used to manage, process and respond to dynamic changes in fast moving streaming time-series data in real-time.
- Allows applications to be architected as measurement routing systems using “Input”, “Action” and “Output” adapter layer.
- Any application can host the framework which will allow a system to become a “real-time measurement bus”.

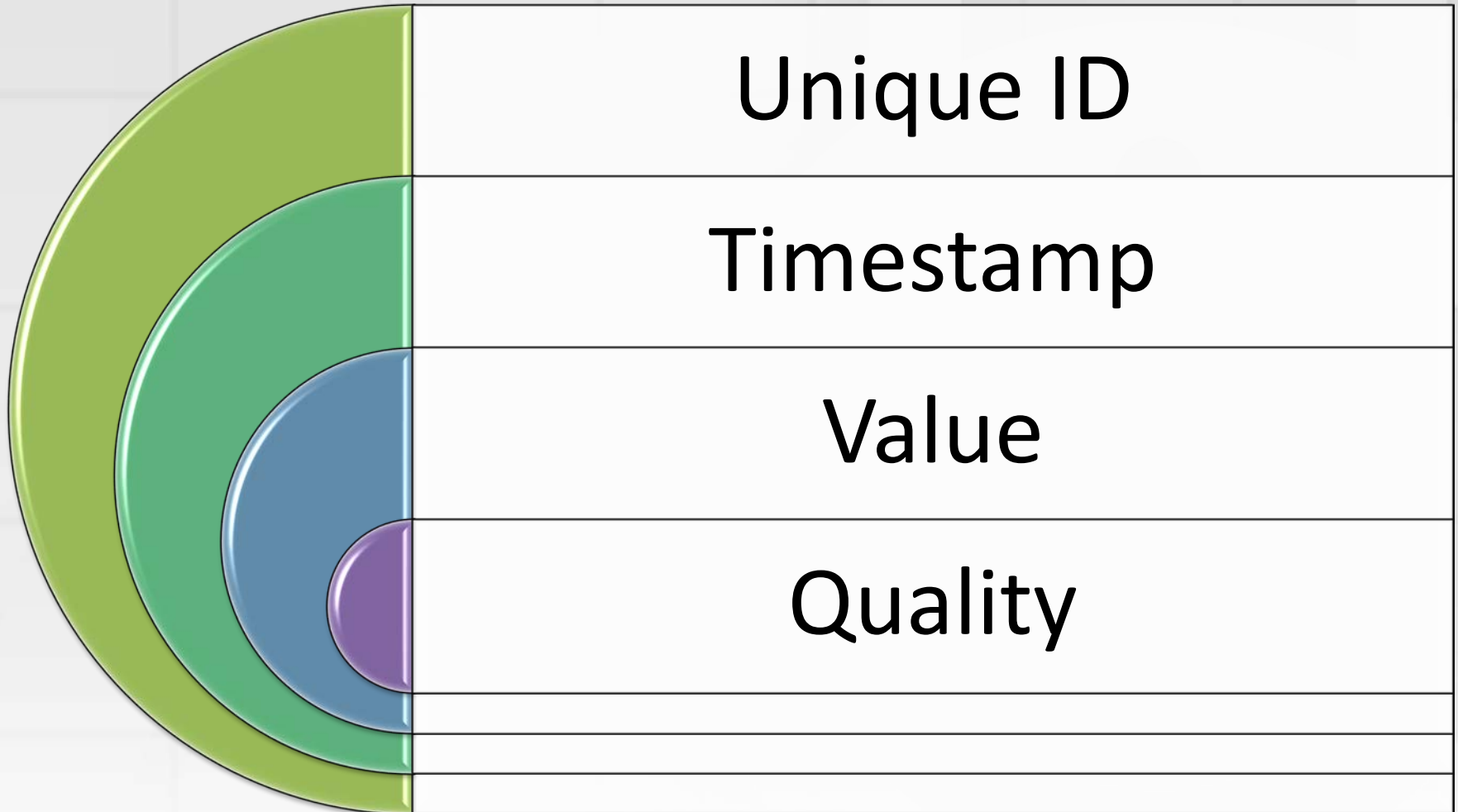
# Measurements

- Numeric quantities that have been acquired at a source device are often known as points, signals, events, or time-series values. Inside GSF they are known as ***measurements***:
  - Examples include: temperature, voltage, vibration, location, luminosity and phasors.

# Understanding “Measurements”

- A “measurement” as it is understood in the Grid Solutions Framework has many aliases:
  - Signal
  - Point
  - Tag
  - Time-series Value
- The primary components of the measurement are:
  - Timestamp
  - Value
  - Identification

# Measurement Structure

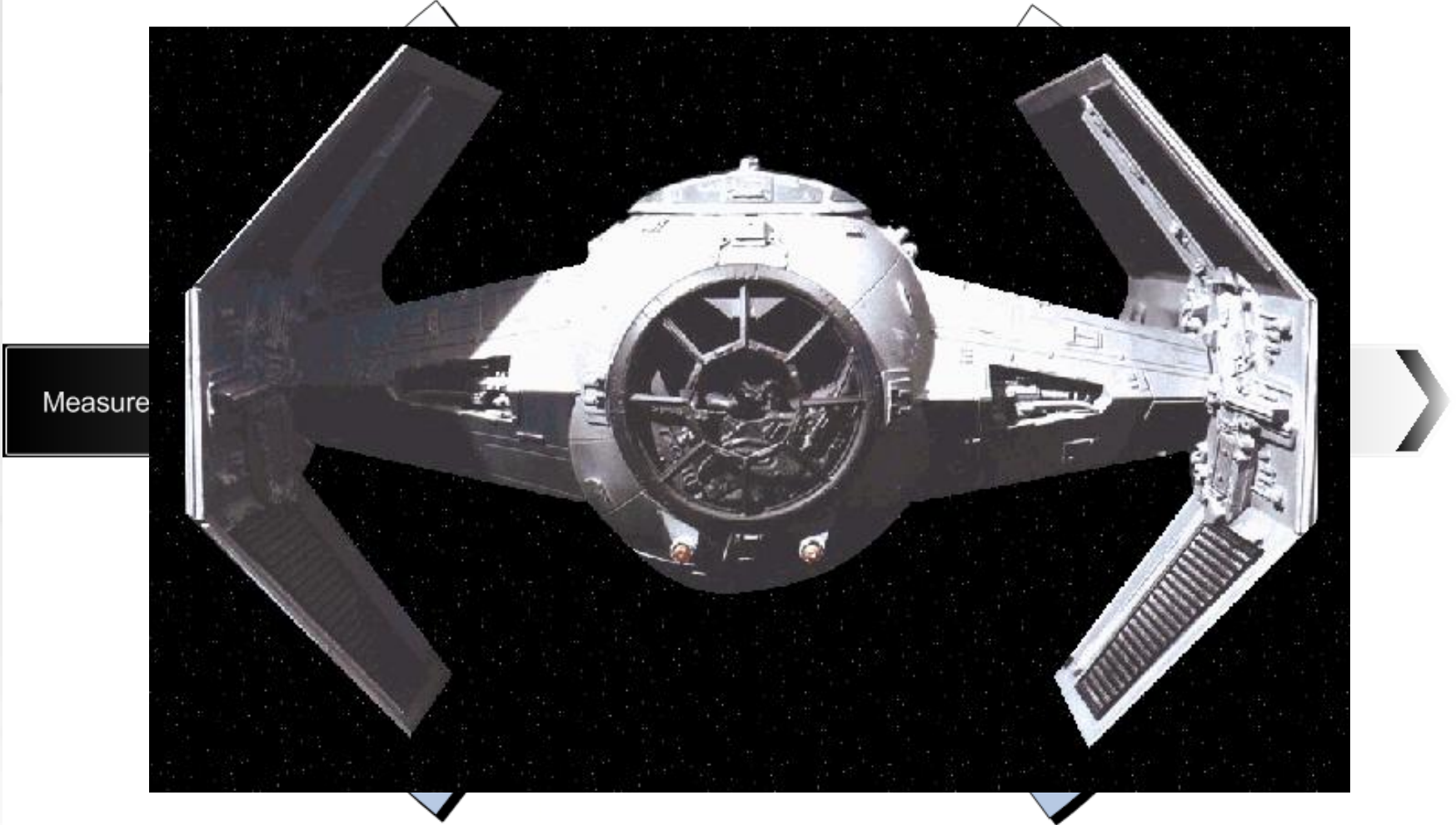


# Measurement Identification

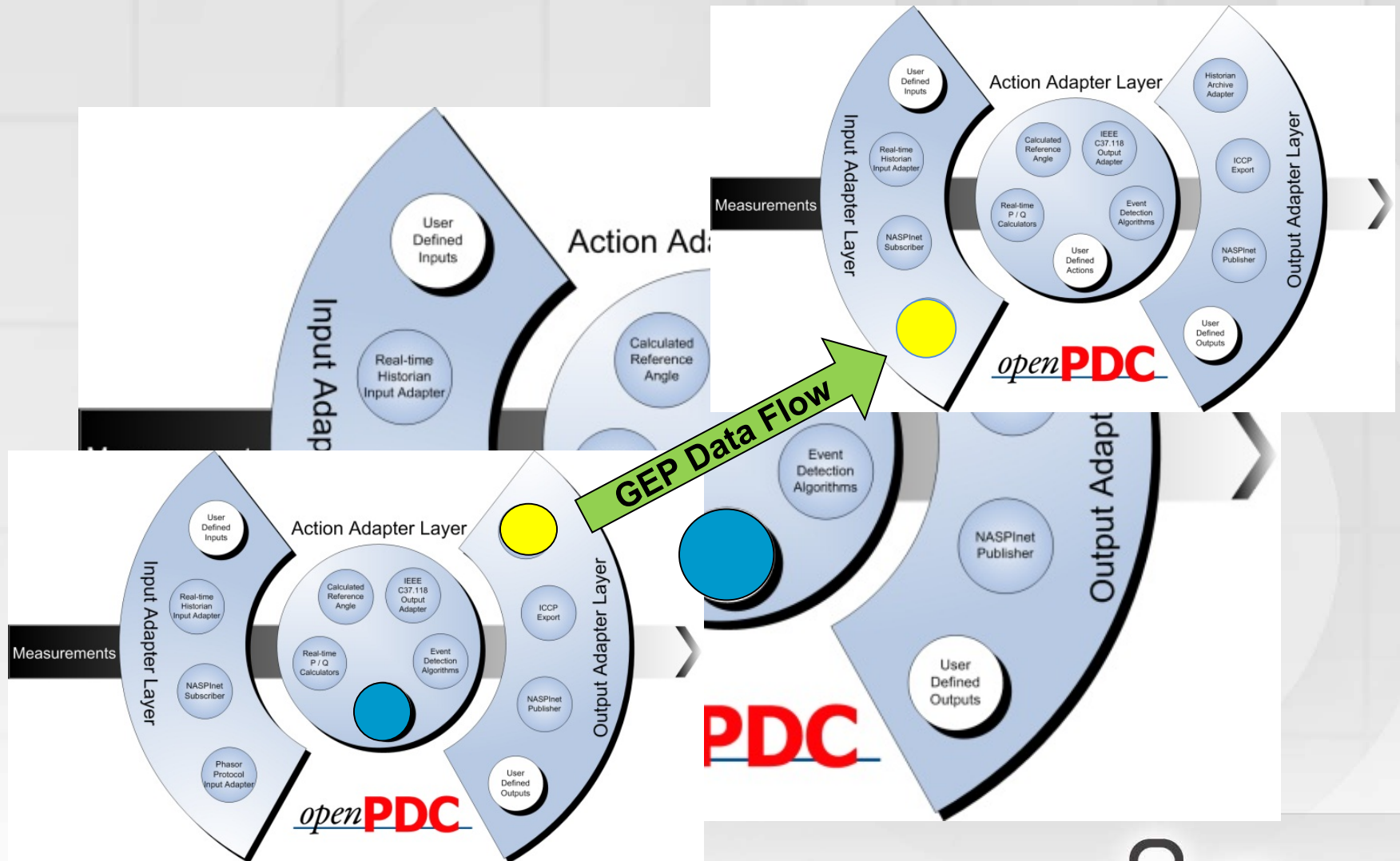
- Guid:
  - 128-bit randomly generated integer that is statistically going to be unique in the world, examples:
    - 7ACDEE91-661B-42A0-82C1-081090D0CA38
    - 532863E4-8C3A-4F84-8366-0C8A4711EA6F
    - 4E3548FD-470E-45DF-8C44-138936805BB6
- Measurement “Key”:
  - Two part identifier represented by a “Source” string and a numeric “ID”, examples:
    - PPA:2
    - STAT:42
    - SHELBY:39



# Overview of the Adapter Architecture Layer



# Scalable Adapter Distribution



## Interface Obligations

**IInputAdapter**

Producer

- Creates New Data

**IActionAdapter**

Consumer /  
Producer

- Creates New Data
- Processes Data

**IOutputAdapter**

Consumer

- Processes Measurements

## Ideal Behaviors

Parsing  
Mapping  
Creation

Processing  
Sorting  
Creation

Processing  
Queuing  
Dissemination

# Input Adapters

*Purpose:*  
**MAP**

- Collect and parse streaming data, assign incoming measurements an ID.

# Output Adapters

*Purpose:*  
**QUEUE**

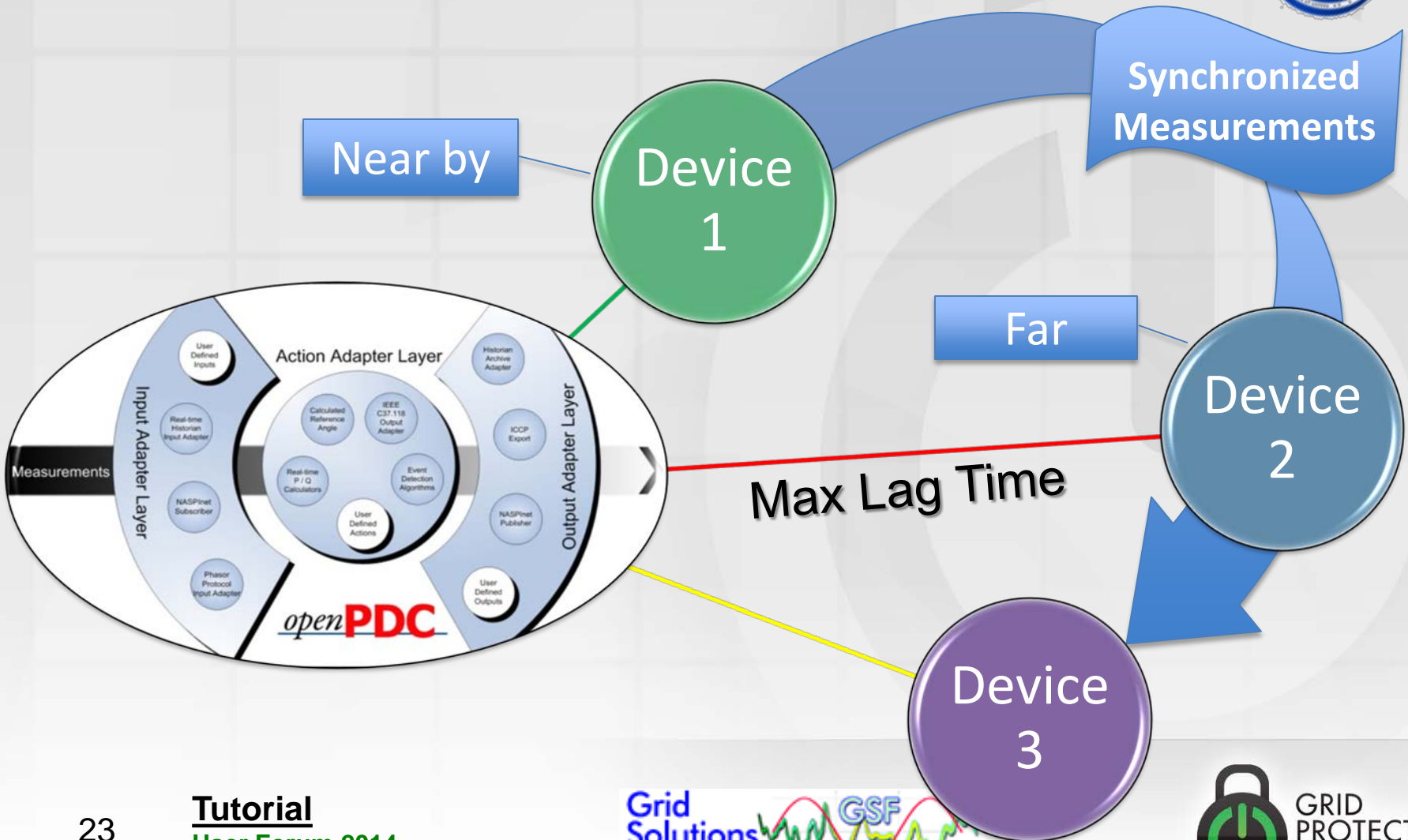
- Queue up measurement data for transmission to archival systems.

# Action Adapters

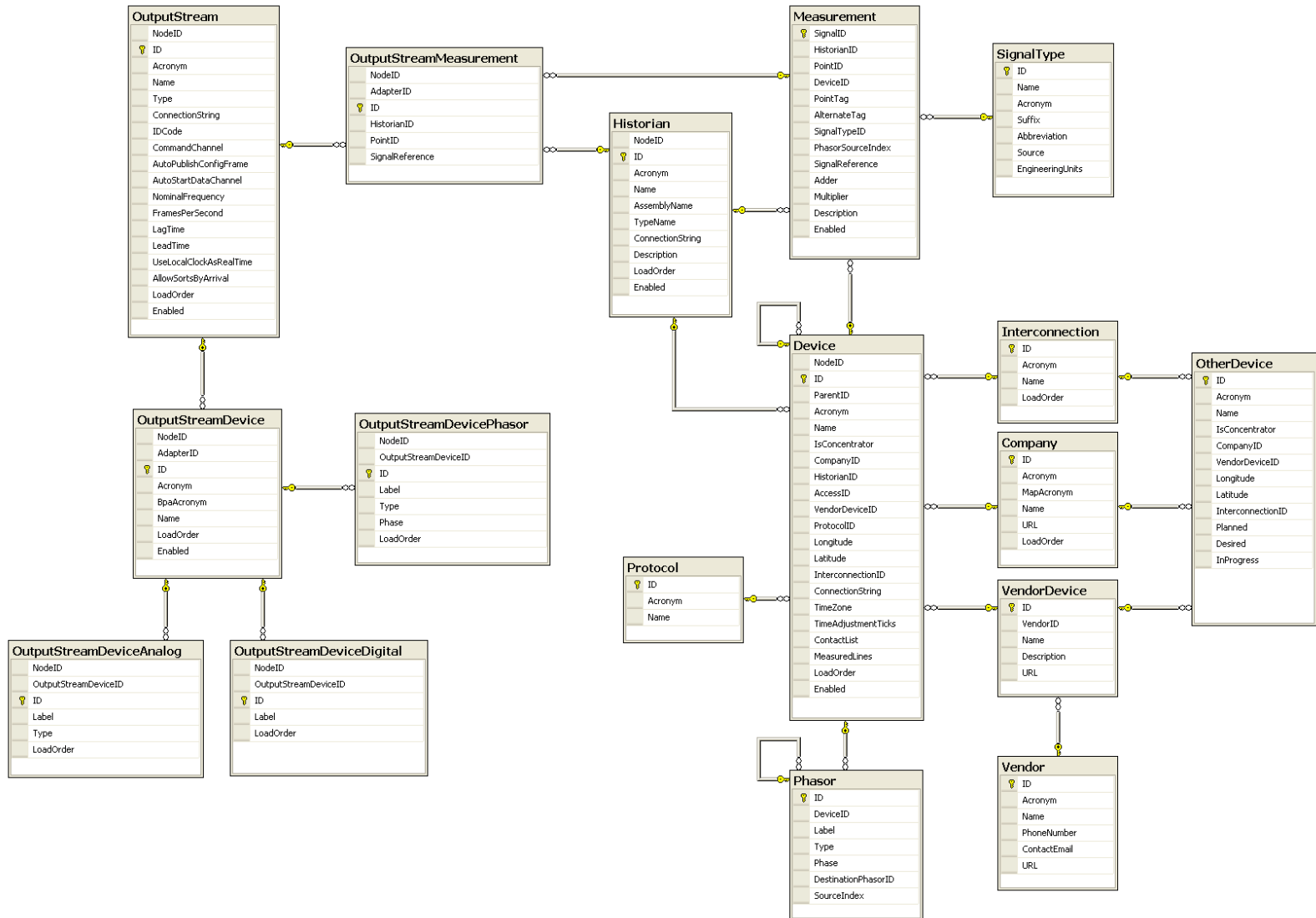
*Purpose:*  
**SORT**

- Sort measurement data by time and process the data for same time-slice.

# Concentration



# The Configuration Data Structure





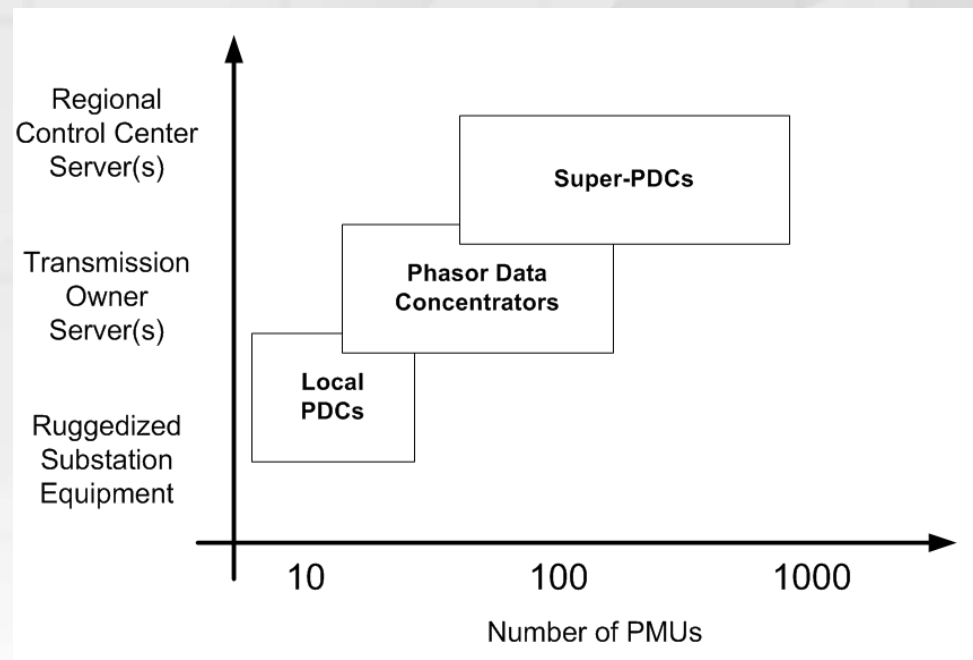
# Phasor Data Concentrator



# What is a PDC?

- *Phasor Data Concentrator (PDC)* – Receives and time-synchronizes phasor data from multiple PMUs to produce a real-time, time-aligned output data stream. A PDC can exchange phasor data with PDCs at other locations. Through use of multiple PDCs, multiple layers of concentration can be implemented within an individual synchrophasor data system.

From NERC RAPIR Report Draft, June 2010



# How is a PDC typically used?

- To create a time-synchronize measurement data set
  - In the substation
  - For the Transmission Operator
  - For the Reliability Coordinator
- To distribute phasor data to applications
- To parse C37.118 for use by other systems

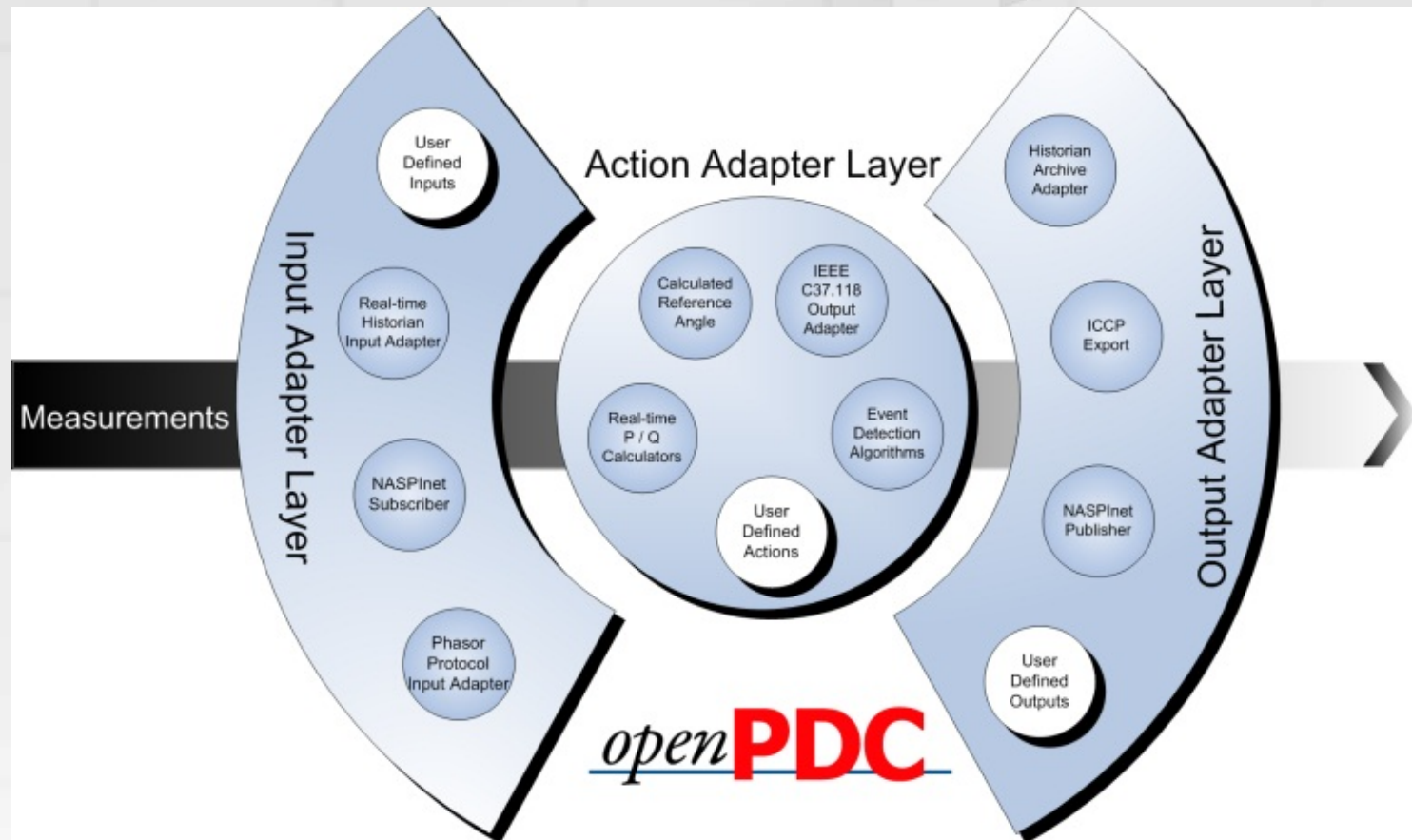
# Who “touches” a PDC?

- A PDC is like an RTU-Data Concentrator for a SCADA system
- PDC’s are back-office tools, administered by specialists, that are likely to soon be part of critical infrastructure
- For compliance and good configuration control, PDC change is tightly managed

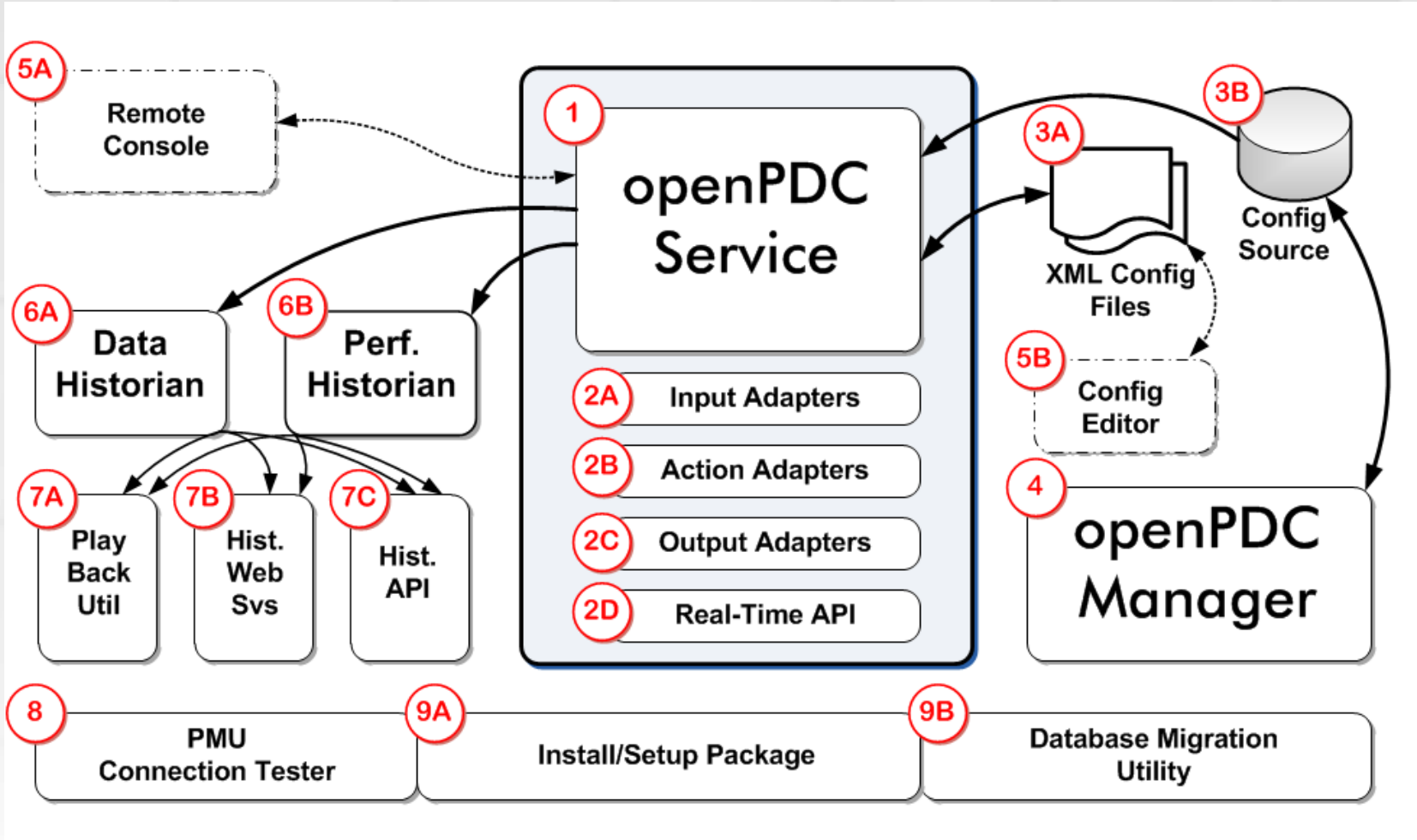
# Who are some PDC Vendors?

- GPA – openPDC
- Alstom Grid – openPDC & Psymetrix
- Electric Power Group – ePDC
- Schweitzer
- General Electric
- Kalkitech

# openPDC is adapter based



# openPDC Components



# openPDC Features

- High performance for the largest of installations
- Extreme configuration flexibility
- Preserves data integrity of incoming data streams
- Produces down-sampled real-time data streams
- Independently handles real-time and archival functions
- Horizontally and vertically scalable
- Low-latency, preemptive frame publishing
- Included performance historian logs highly granular operational statistics
- Extensible through the creation of input, action or output adapters
- Many instances can be remotely configured through a single configuration application
- A growing and active open source community



# openPDC Specifications

- Input Protocols
  - IEEE C37.118-2005
  - IEEE C37.118-2011 (Beta)
  - IEC 61850-90-5
  - SEL Fast Messaging
  - Macrodyne N and G
  - IEEE 1344-1995
  - BPA PDC Stream
  - UTK FNET
  - DNP3 (Beta)
  - Gateway Exchange Protocol (GEP)
- Output Protocols
  - IEEE C37.118-2005
  - BPA PDC Stream
  - Gateway Exchange Protocol (GEP)
  - Inter-Site Data (ISD) purchased from Alstom Grid

# openPDC Specifications

(continued)

- **Communications Standards**
  - TCP – IPv4 and IPv6
  - UDP Unicast and Multicast, IPv4 and IPv6
  - Serial (input only)
- **Operating System**
  - Windows Server 2008, R2 recommended
- **Hardware Requirements**
  - Multi-processor / multi-core systems recommended
  - Tested on single core, fanless systems with as little as 2 GB of RAM
- **Configuration System**
  - A relational database is recommended to house configuration data. Supported databases are:
    - MS SQL Server
    - MySQL
    - Oracle
    - SQLite

# PMU Connection Tester

## *Included with openPDC*

**PMU Connection Tester**

File Help

Connection Parameters

Tcp Udp Serial File Default IP Stack: IPv4 Protocol

Local Port: 8800 Host IP: 127.0.0.1 Remote Port: 5000

Enable Multicast / Remote Udp

IEEE C37.118-2005 Disconnect

Device ID Code: 235 Version 4.1.3.24840

Command: Disable Real-time Data Send

Configure Alternate Command Channel Defined

PMU: ID Code: 1

SHELBY

Phasor: (Selected is reference angle)

V: 500 kV Bus 1 +SV

Phasors: 5 Nominal Frequency: 60 Hz

Analog: 0 Digital: 1

Power: -208.1425 MW

Vars: -35.5455 MVars

Configured frame rate: 30 frames/second

59.9744  
59.9698  
59.9652  
59.9606  
59.9560

180  
90  
0  
-90  
-180

500 kV Bus 1 +SV  
500 kV Bus 2 +SV  
Cordova +SI  
Dell +SI  
Lagoon Creek +SI

Graph Settings Messages Protocol Specific

Real-time Frame Detail

Frame Type: DataFrame AA 01 00 44 00 EB 4C D9 71 A2 00 D5 55 54 00 00 48 92 63

Time: 2010-11-09 16:06:58.833 9C BF 34 A9 D9 48 91 E3 E4 BF 34 8C 6F 43 6A C5 33 40 11

Frequency: 59.9780 Hz 12 27 43 FF 37 54 BF 34 C7 E3 43 55 24 C3 BF C7 BE 0D 42

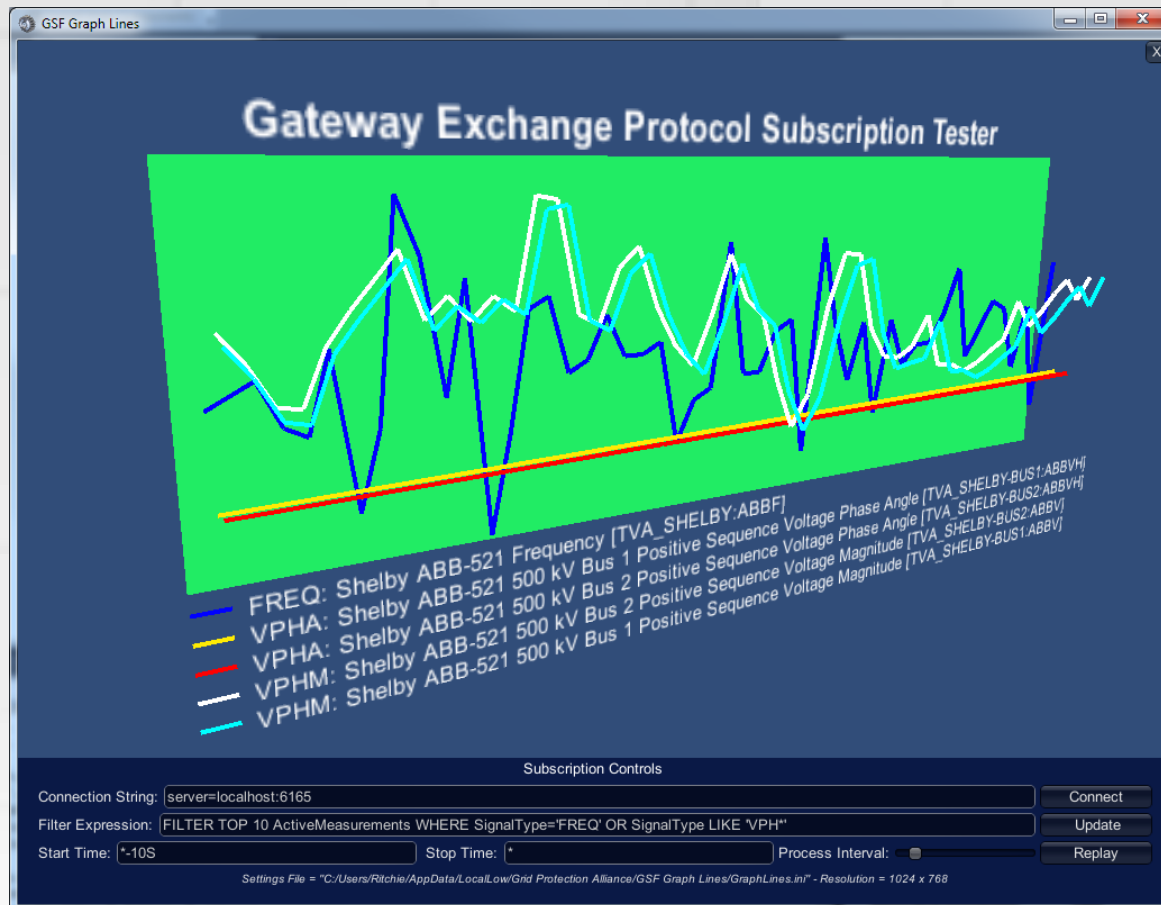
Angle: -40.4345866977867° 6F E9 79 3E 0F 5C 29 00 00 25 0B

Magnitude: 299.8049 (519.2773) kV

Display: Hexadecimal

Total frames: 139 Frames/sec: 0.0000 Total bytes: 9798 Bit rate (mbps): 0.0000 Queued buffers: 0

# GEP Subscription Tester *Included with openPDC*



# Who else uses the openPDC?

- In operational service at TVA since 2004
- Other North American production deployments include WECC, OG&E Dominion, Southern Company, Duke, ISO-NE, FP&L, AESO, PG&E and others
- Large community. There have been over 2,000 downloads of the openPDC since version 1.5 was released.

# openPDC Manager Home Screen

openPDC Manager - GPA\arkrohne

openPDC Manager

Current Node: Default

Home Devices Outputs Gateway Adapters Monitoring Manage

System Overview

CORDOVA \_CORDOVA:F

Quick Links

- Graph Measurements
- Stream Statistics
- Add New Device
- Manage Devices
- Manage Output Streams
- Remote System Console
- Restart openPDC

Current Configuration

Instance Type 64-bit

Server Time 2012-11-06 16:26:13.045

Local Time 2012-11-06 16:26:13.045

Current User GPA\arkrohne

Version Information

Server 1.5.159.0

Manager 1.5.159.0

Database Information

Type SQLite

Name openPDC.db

System Status

System Health (Last Refreshed: 16:26:13.045)

Counter	Last	Average	Maximum	Units
CPU Utilization	1.21	1.82	10.13	Average % / CPU
I/O Data Rate	16.96	64.22	1840.37	Kilobytes / sec
I/O Activity Rate	279.76	162.08	1035.85	Operations / sec
Process Handle Count	903.00	891.32	931.00	Total Handles
Process Thread Count	60.00	58.82	63.00	System Threads
CLR Thread Count	47.00	45.29	51.00	Managed Threads
Worker Threads	1.00	1.08	2.00	Active in Pool
I/O Port Threads	0.00	0.00	0.00	Active in Pool
Thread Queue Size	0.00	0.00	0.00	Waiting Threads
Lock Contention Rate	0.00	0.01	0.20	Attempts / sec
Process Memory Usage	379.42	323.69	379.42	Megabytes
CLR Memory Usage	33.20	20.82	39.52	Megabytes
Large Object Heap	12.93	7.10	12.93	Megabytes
Exception Count	4.00	2.66	4.00	Total Exceptions
Exception Rate	0.00	0.02	0.40	Exceptions / sec
IPv4 Outgoing Rate	3.60	7.76	32.95	Datagrams / sec
IPv4 Incoming Rate	100.17	98.52	148.93	Datagrams / sec
IPv6 Outgoing Rate	0.20	2.55	23.13	Datagrams / sec
IPv6 Incoming Rate	1.00	0.53	2.39	Datagrams / sec

Statistics calculated using last 120 counter values sampled every 5.0 seconds.

# openPDC Manager Home Screen

The screenshot shows the openPDC Manager interface. A green box with the text "Select Instance to Configure" has an arrow pointing to the "Current Mode" dropdown menu, which is currently set to "Default". Another green box with the text "Select Point to Display in Real Time" has an arrow pointing to the "\_CORDOVA:F" dropdown menu. The main area displays a line graph showing a fluctuating blue line over time. Below the graph is a table of system statistics.

Counter	Units
Average % / CPU	
Kilobytes / sec	
Operations / sec	
Total Handles	
System Threads	
Managed Threads	
Active in Pool	
Active in Pool	
Waiting Threads	
Attempts / sec	
Megabytes	
Megabytes	
Megabytes	
Total Exceptions	
Exceptions / sec	
Datagrams / sec	
Datagrams / sec	
Datagrams / sec	
Datagrams / sec	

Statistics calculated using last 120 counter values sampled every 5.0 seconds.

# Connect to a Device

The screenshot shows the 'openPDC Manager' application window. The title bar reads 'openPDC Manager - GPA\arkrohne'. The main menu includes 'Home', 'Devices', 'Outputs', 'Gateway', 'Adapters', 'Monitoring', and 'Manage'. The 'Current Node' is set to 'Default'. The active page is 'Manage Device Configuration', which is divided into 'Phasors' and 'Measurements' sections.

**Phasors Section:**

- Acronym: MULTI
- Name: (empty)
- ID Code (Access ID): 235
- Company: Select Company
- Protocol: IEEE C37.118-2005
- Connection String: transportprotocol=Udp;localport=5000;server=233.123.123.123;remoteport=5000;interface=0.0.0.0;
- Data Loss Interval: 5
- Allowed Parsing Exception: 10
- Delayed Connection Interval: 5
- Longitude: (empty)
- Interconnection: Select Interconnection

**Measurements Section:**

- Concentrator: Select Device
- Time Zone: Select Time Zone
- Frames Per Second: 30
- Historian: PPA
- Device Vendor: Select Vendor Device
- Alternate Command Channel: (empty)
- Time Adjustment Ticks: 0
- Parsing Exception Window: 5
- Measurement Reporting Interval: 100000
- Latitude: (empty)

**Runtime Settings:**

- Skip Disable Real-time Data
- Allow Use Of Cached Configuration
- Auto Start Data Parsing Sequence
- Concentrator
- Runtime ID: 9
- Connect On Demand
- Enabled

Buttons: Initialize, Delete, Clear, Save

**Concentrator Device List:**

Acronym	Name
BULLRUN	Bullrun
COLLINSVILLE	Collinsville
CORDOVA	Cordova
CUMBERLAND	Cumberland
HENDERSON	Henderson



# Connect to a Device

The screenshot shows the 'Manage Device Configuration' window in openPDC Manager. The interface includes a navigation menu (Home, Devices, Outputs, Gateway, Adapters, Monitoring) and a main configuration area. Three callout boxes with green arrows point to specific fields:

- Name of Connection:** Points to the 'Acronym' field, which contains the value 'MULTI'.
- Connect to another PDC:** Points to the 'Connection String' field, which contains the text: 'transportprotocol=Udp;localport=5000;server=233.123.123.123;remoteport=5000;interface=0.0.0.0;'. A green plus sign icon is visible next to this field.
- Set tolerances for Error Reporting and Reconnection Attempts:** Points to the 'Allowed Parsing Exception' field, which contains the value '10'.

Other visible fields include 'Name', 'ID Code (Access ID)' (235), 'Company', 'Protocol' (IEEE C37.118-2005), 'Data Loss Interval' (5), 'Delayed Connection Interval' (5), 'Longitude', 'Interconnection', 'Time Zone', 'Concentrator', and 'Time Adjustment Ticks' (0). There are also checkboxes for 'Skip Disable Real-time Data', 'Allow Use Of Cached Configuration', and 'Auto Sta'. A 'Runtime ID' of 9 and an 'Initialize' button are also present.

At the bottom, a 'Concentrator Device List' table is visible:

Acronym	Name
BULLRUN	Bullrun
COLLINSVILLE	Collinsville
CORDOVA	Cordova
CUMBERLAND	Cumberland
HENDERSON	Henderson

# Input Configuration

openPDC Manager - GPA\arkrohne  
Current Node: Default

Home Devices Outputs Gateway Adapters Monitoring Manage

Input Device Configuration Wizard

- Step 1: Configure Connection Settings
- Step 2: Select Device Configuration Settings
- Step 3: Select Devices to Configure \* Device acronym already exists in the database.

<input checked="" type="checkbox"/>	Acronym	Name	Longitude	Latitude	Digitals	Analogs
<input checked="" type="checkbox"/>	BULLRUN	Bullrun	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	COLLINSVILLE	Collinsville	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	CORDOVA	Cordova	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0

<input checked="" type="checkbox"/>	Label	Type	Phase
<input checked="" type="checkbox"/>	500 kV Bus 2 +SV	V	+
<input checked="" type="checkbox"/>	500 kV Bus 1 +SV	V	+
<input checked="" type="checkbox"/>	Haywood +SI	I	+
<input checked="" type="checkbox"/>	Shelby +SI	I	+
<input checked="" type="checkbox"/>	Benton +SI	I	+
<input checked="" type="checkbox"/>	Freeport +SI	I	+

<input checked="" type="checkbox"/>	CUMBERLAND	Cumberland	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	HENDERSON	Henderson	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	LOWNDES	Lowndes	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	MARSHALL	Marshall	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	RIDGEDALE	Ridgedale	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	SHELBY	Shelby	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	VOLUNTEER	Volunteer	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
<input checked="" type="checkbox"/>	CALLAWAY	Callaway	-98.6	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0

Previous Finish

# Input Configuration

openPDC Manager - GPA\arkrohne

openPDC Manager

Current Node: Default

Home Devices Outputs Gateway Adapters Monitoring Manage

Input Device Configuration Wizard

Step 1: Configure Connection Settings

Step 2: Select Device Configuration Settings

Step 3: Select Devices to Configure

Click on Row to Expand

Can Edit (override) C37-118 Labels

Acronym	Latitude	Digitals	Analogs
BULLRUN	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
COLLINSVILLE	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0
CORDOVA	37.5	<input type="checkbox"/> 0	<input type="checkbox"/> 0

Label	Type	Phase
500 kV Bus 2 +SV	V	+
500 kV Bus 1 +SV	V	+
Haywood +SI	I	+
Shelby +SI		
Benton +SI		
Freeport +SI		

Label	Type	Phase
CUMBERLAND		
HENDERSON		
LOWNDES		
MARSHALL		
RIDGEDALE		
SHELBY		
VOLUNTEER	-98.6	37.5
CALLAWAY	-98.6	37.5

Previous Finish

# Review Real Time Values

The screenshot displays the 'openPDC Manager' interface. The title bar shows 'openPDC Manager - GPA\arkrohne'. The main menu includes 'Home', 'Devices', 'Outputs', 'Gateway', 'Adapters', 'Monitoring', and 'Manage'. The 'Monitoring' tab is active, showing 'Real-time Device Measurements'. The interface includes a 'StatusFlag Reference', 'Display Settings', 'Refresh Interval: 10 sec', and 'Last Refresh: 21:24:05.086'. The data is organized into sections for 'CUMBERLAND' and 'DANIEL-BIGCRK', each with a sub-section for 'IEEE C37.118-2005'. Each section lists various PPA units with their corresponding real-time values and units.

PPA ID	Measurement	Value	Unit
PPA:145		361.817	Amps
PPA:147		213.849	Amps
PPA:149		548.207	Amps
PPA:138	Cordova Status Flags	0	Hex
<b>CUMBERLAND</b>			
Cumberland IEEE C37.118-2005			
PPA:152	Cumberland Frequency Delta (dF/dt)	0.27	
PPA:151	Cumberland Frequency	59.993	Hz
PPA:155		-16.896	Degrees
PPA:157		-16.889	Degrees
PPA:159		174.301	Degrees
PPA:161		4.83	Degrees
PPA:163		-15.707	Degrees
PPA:165		-16.051	Degrees
PPA:154		300066.406	Volts
PPA:156		299668.531	Volts
PPA:158		276.571	Amps
PPA:160		469.68	Amps
PPA:162		1379.755	Amps
PPA:164		1023.152	Amps
PPA:153	Cumberland Status Flags	0	Hex
<b>DANIEL-BIGCRK</b>			
Daniel-Bigcrk IEEE C37.118-2005			
PPA:506	Daniel-Bigcrk Frequency Delta (dF/dt)	0	
PPA:505	Daniel-Bigcrk Frequency	59.995	Hz
PPA:509		-11.887	Degrees
PPA:511		-137.335	Degrees
PPA:513		103.244	Degrees
PPA:515		-19.874	Degrees
PPA:517		-137.884	Degrees
PPA:519		100.206	Degrees
PPA:521		-15.326	Degrees
PPA:523		-19.177	Degrees
PPA:508		133581.922	Volts
PPA:510		134329.969	Volts
PPA:512		133689.625	Volts
PPA:514		309.822	Amps
PPA:516		318.581	Amps
PPA:518		321.968	Amps
PPA:520		133743.266	Volts
PPA:522		316.742	Amps
PPA:507	Daniel-Bigcrk Status Flags	0	Hex

# Review Real Time Values

openPDC Manager - GPA\arkrohne

openPDC Manager

Current Node: Default

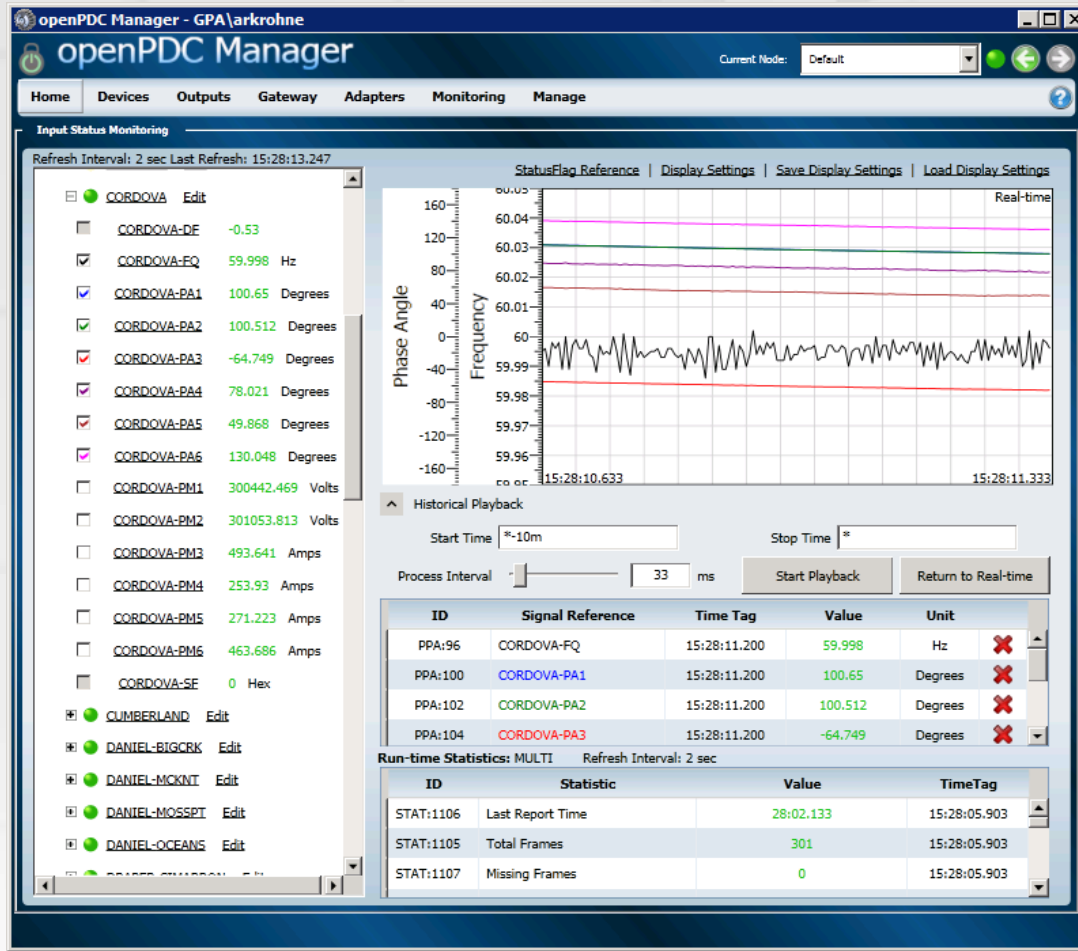
Home Devices Outputs Gateway Adapters Monitoring Manage

Real-time Device Measurements

StatusFlag Reference Display Settings Refresh Interval: 10 sec Last Refresh: 21:24:05.086

Device ID	Measurement	Value	Unit
PPA:145		361.817	Amps
PPA:147		213.849	Amps
PPA:149		548.207	Amps
PPA:138	Cordova Status Flags	0	Hex
<b>CUMBERLAND</b>			
IEEE C37.118-2005			
PPA:152	Cumberland Frequency Delta (dF/dt)	0.27	
PPA:151	Cumberland Frequency	59.993	Hz
PPA:155		-16.896	Degrees
PPA:157		-16.889	Degrees
PPA:159		174.301	Degrees
PPA:161		4.83	Degrees
PPA:163		-15.707	Degrees
PPA:165		-16.051	Degrees
PPA:154		300066.406	Volts
PPA:156		299668.531	Volts
PPA:158		276.571	Amps
PPA:160		469.68	Amps
PPA:162		1379.755	Amps
PPA:164		1023.152	Amps
PPA:153	Cumberland Status Flags	0	Hex
<b>DANIEL-BIGCRK</b>			
IEEE C37.118-2005			
PPA:506	Daniel-Bigcrk Frequency Delta (dF/dt)	0	
PPA:505	Daniel-Bigcrk Frequency	59.995	Hz
PPA:509		-11.887	Degrees
PPA:511		-137.335	Degrees
PPA:513		103.244	Degrees
PPA:515		-19.874	Degrees
PPA:517		-137.884	Degrees
PPA:519		100.206	Degrees
PPA:521		-15.326	Degrees
PPA:523		-19.177	Degrees
PPA:508		133581.922	Volts
PPA:510		134329.969	Volts
PPA:512		133689.625	Volts
PPA:514		309.822	Amps
PPA:516		318.581	Amps
PPA:518		321.968	Amps
PPA:520		133743.266	Volts
PPA:522		316.742	Amps
PPA:507	Daniel-Bigcrk Status Flags	0	Hex

# View Real Time Data



# View Real Time Data

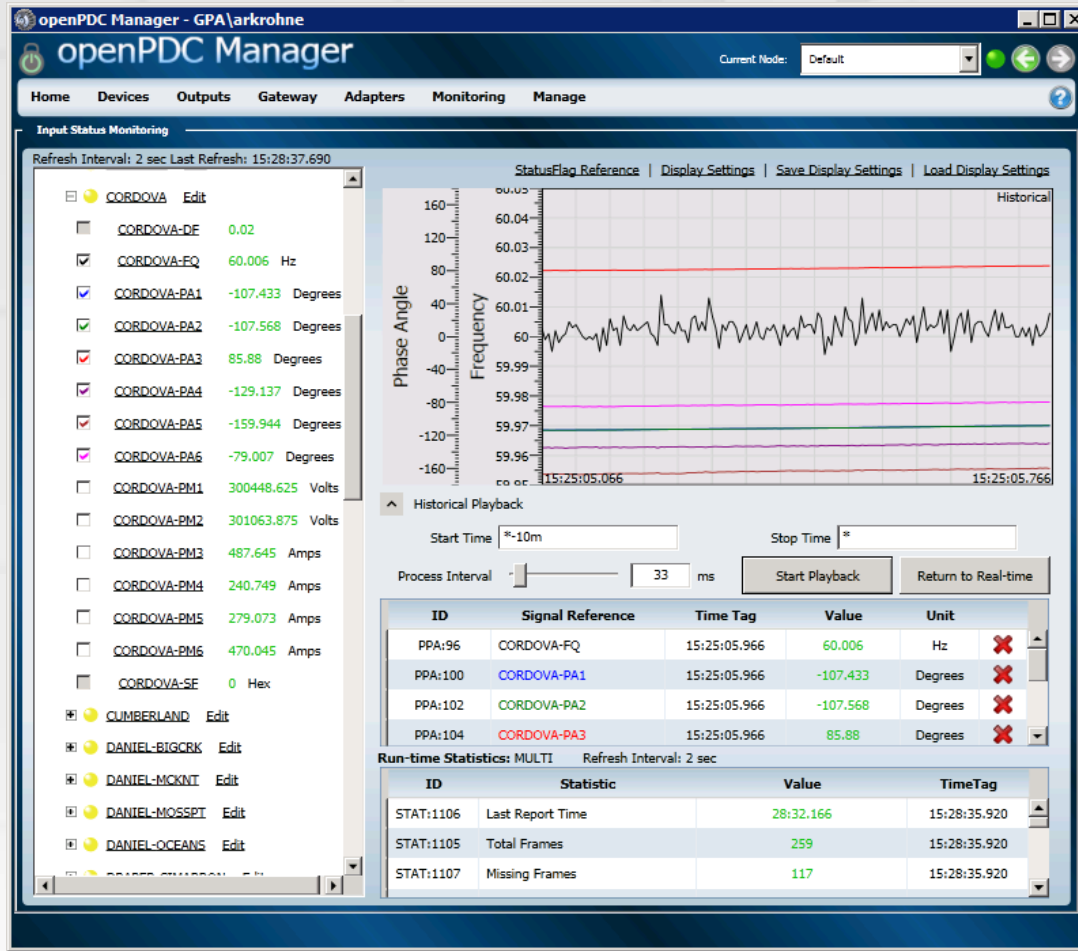
The screenshot shows the openPDC Manager interface with the following components:

- Device List (Left):** A list of devices including CORDOVA (with sub-devices like CORDOVA-DF, CORDOVA-EQ, CORDOVA-PA1-PA6, CORDOVA-PM1-PM6, CORDOVA-SF) and DANIEL (DANIEL-BIGCRK, DANIEL-MCKNT, DANIEL-MOSSPT, DANIEL-OCEANS).
- Graph (Center):** A real-time plot showing Phase Angle (Y-axis, -160 to 160) and Frequency (Y-axis, 59.96 to 60.05). The X-axis represents time from 15:28:10.633 to 15:28:11.333. A green arrow points to the graph with the label "Real Time".
- Historical Playback (Below Graph):** Controls for Start Time (\*-10m) and Stop Time (\*), with buttons for Start Playback and Return to Real-time.
- Real-time Data Table (Bottom Right):** A table showing selected data points.
 

Time Tag	Value	Unit
15:28:11.200	59.998	Hz
15:28:11.200	100.65	Degrees
15:28:11.200	100.512	Degrees
15:28:11.200	-64.749	Degrees
- Run-time Statistics (Bottom):** A table showing system statistics.
 

ID	Statistic	Value	TimeTag
STAT:1106	Last Report Time	28:02.133	15:28:05.903
STAT:1105	Total Frames	301	15:28:05.903
STAT:1107	Missing Frames	0	15:28:05.903

# View Historical Data





# View Historical Data

The screenshot shows the openPDC Manager interface with the 'Monitoring' tab selected. On the left, a tree view lists devices under 'CORDOVA' and 'CUMBERLAND'. The main area displays a 'Historical' graph with multiple data series. A green box labeled 'Review last 10 minutes' has an arrow pointing to the 'Start Time' field set to '\*-10m'. Another green box labeled 'Historical' has an arrow pointing to the graph area. A third green box labeled 'Simulated Time Frame' has an arrow pointing to the 'Time Tag' column in the data table below the graph.

ID	Signal Reference	Time Tag	Value	Unit	
PPA:96	CORDOVA-FQ	15:25:05.966	60.006	Hz	✘
PPA:100	CORDOVA-PA1	15:25:05.966	-107.433	Degrees	✘
PPA:102	CORDOVA-PA2	15:25:05.966	-107.568	Degrees	✘
PPA:104	CORDOVA-PA3	15:25:05.966	85.88	Degrees	✘

ID	Value	TimeTag
STAT:1105	166	15:28:35.920
STAT:1105	Total Frames	259
STAT:1107	Missing Frames	117

# openPDC Console

- The openPDC console can be used to remotely monitor the details of openPDC operation
- It can be run independently of the openPDC Manager
- Typical Commands
  - **Clients** Shows list of connections to service
  - **Health** Shows health report
  - **List** Displays list of devices connections
  - **Help** Displays list of commands

```
C:\Program Files\openPDC\openPDCConsole.exe
State of process "HealthMonitor" has changed to "Processing".
State of process "HealthMonitor" has changed to "Processed".

Counter                Last          Average       Maximum       Units
-----
CPU Utilization        0.23          3.60          7.55          Average % / CPU
I/O Data Rate          14.17         427.52        2262.70       Kilobytes / sec
I/O Activity Rate      854.55        19439.51      31276.70      Operations / sec
Process Handle Count   1000.00       996.69        1149.00       Total Handles
Process Thread Count   66.00         68.85         89.00         System Threads
CLR Thread Count       51.00         53.57         73.00         Managed Threads
Worker Threads         7.00          6.05          12.00         Active in Pool
I/O Port Threads      0.00          0.00          0.00          Active in Pool
Thread Queue Size     0.00          0.00          0.00          Waiting Threads
Lock Contention Rate  0.00          0.00          1.00          Attempts / sec
Process Memory Usage   699.86        600.34        752.81        Megabytes
CLR Memory Usage       31.78         42.25         186.93        Megabytes
Large Object Heap     5.28          5.70          13.92         Megabytes
Exception Count       152.00        78.66         152.00        Total Exceptions
Exception Rate        0.00          0.23          4.20          Exceptions / sec
IPv4 Outgoing Rate    10.78         7.28          105.56        Datagrams / sec
IPv4 Incoming Rate    26.14         90.01         254.86        Datagrams / sec
IPv6 Outgoing Rate    0.00          2.08          25.58         Datagrams / sec
IPv6 Incoming Rate    0.20          0.32          2.80          Datagrams / sec

Statistics calculated using last 120 counter values sampled every 5.0 seconds.

[Input Adapter Collection]
Process statistics for 14 hours 6 minutes 14 seconds total runtime:
Time span      Measurements      Per second
-----
Entire runtime  12,746,161        251
Last minute    375,366           6,254

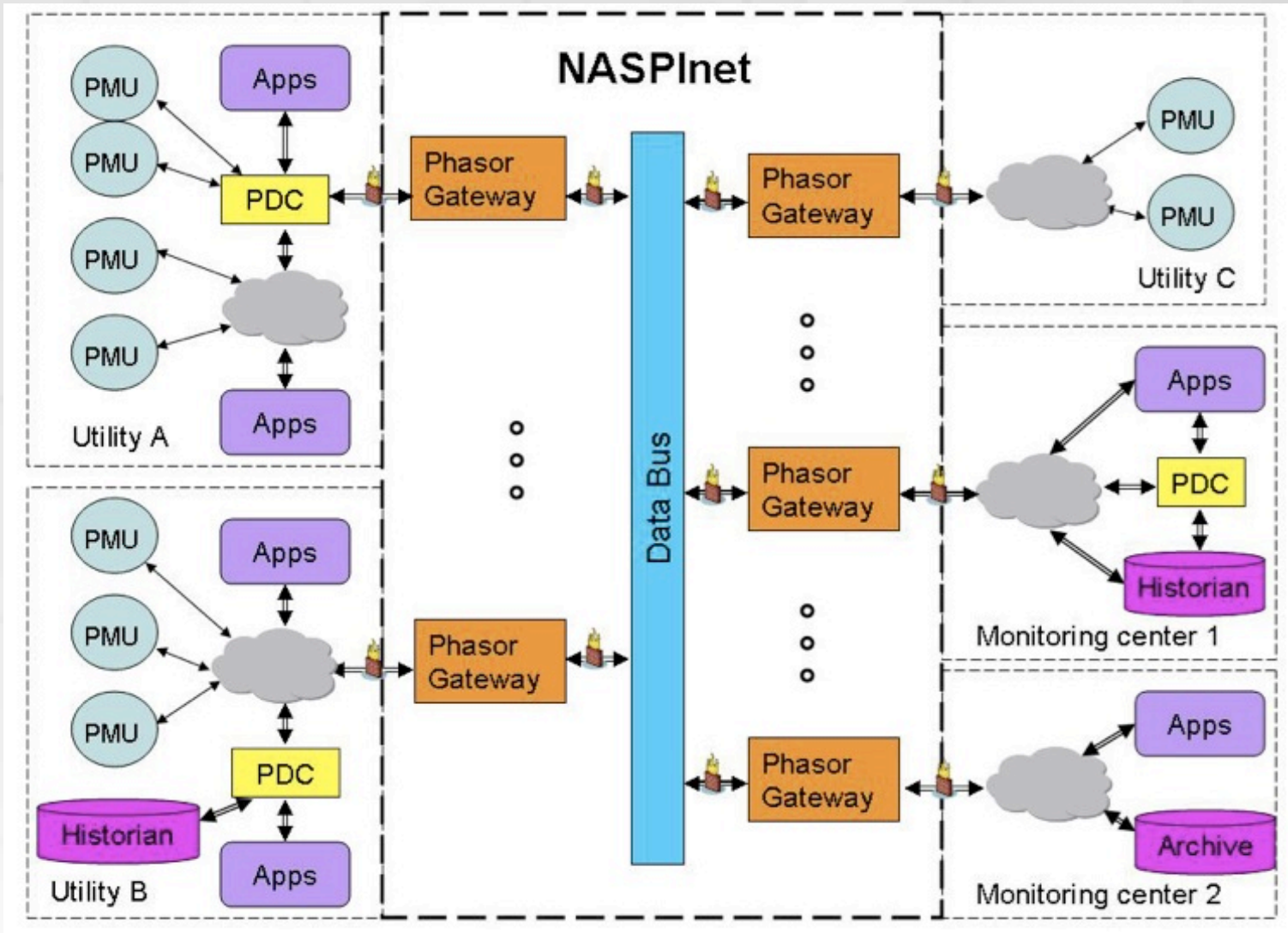
[Output Adapter Collection]
Process statistics for 14 hours 6 minutes 14 seconds total runtime:
Time span      Measurements      Per second
-----
Entire runtime  33,177,141        653
Last minute    376,764           6,277

[PPA] 32,900,437 measurements have been processed so far...
[MULTI] 12,800,152 measurements have been processed so far...
```

# Secure Information Exchange Gateway

# SIEGate

# The term "Gateway" came from NASPInet



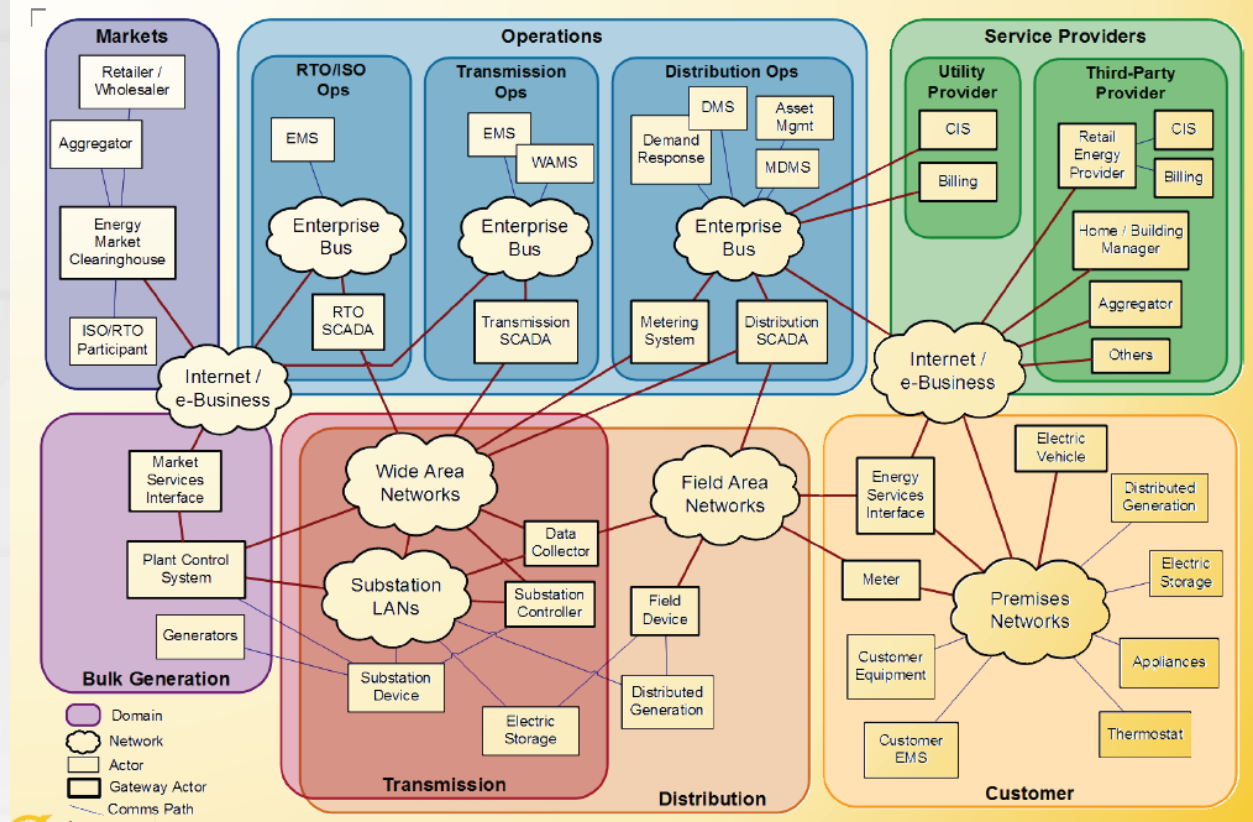
Taken from NASPInet Specification, 2007

# What is a gateway?

- Creates a hardened security buffer between critical internal systems and external ones
- Protects the confidentiality and integrity of reliability and market sensitive BES data
- Facilitates and reduces the cost of BES data exchange, including synchrophasor data -- both the actual data and the supporting metadata information for this data as well

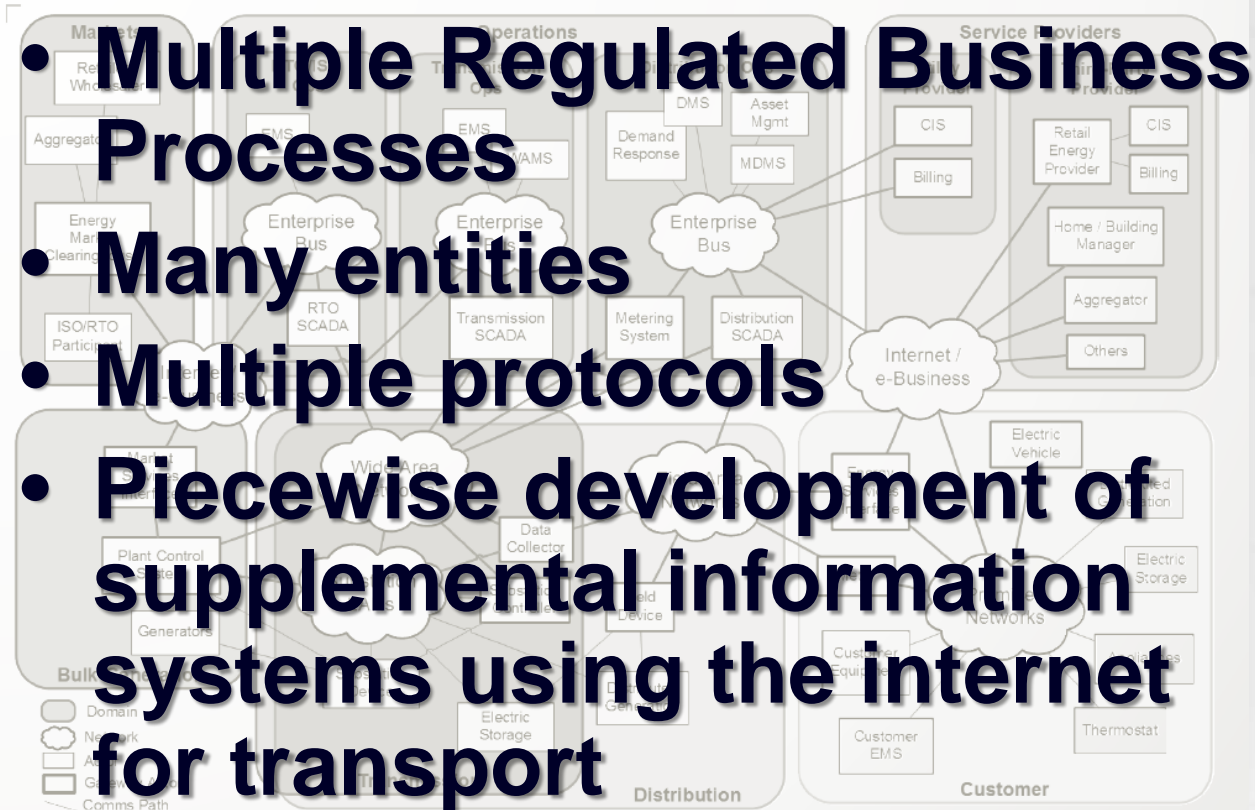
# Current State of BES Data Exchange is Complex

Smart Grid Architecture (Source: NIST)



# Current State of BES Data Exchange is Complex

Smart Grid Architecture (Source: NIST)



# PDC vs. SIEGate

## Distinguishing Features

- PDC – optimized for time-alignment of many inputs
  - Accepts inputs from PMUs and other IEDs using the broadest range of formats and protocols
  - Provides time-alignment of data (with delays and loss after time-out)
  - Allows implementation of adapters that require rapid access to time-aligned data
  - Publishes multiple time-concentrated streams
  - Reports and alarms on quality of measurements (signals) and input device status
- SIEGate – optimized for directed data transfer of granular information that facilitates a security-layered network design
  - Manages asynchronous communication of specific measurements (signals) with other SIEGate nodes
  - Relays data upon receipt without further delay
  - Can effectively manage the joining of two semantic models
  - Reports and alarms on status of communication of data with other gateways



# SIEGate Project Objective

*To develop and commercialize a flexible appliance to enable the secure exchange of all types of real-time reliability data among grid operating entities.*

**SIEGate will be a security-centric edge-device that**

- Resists cyber attacks
- Preserves data integrity and confidentiality

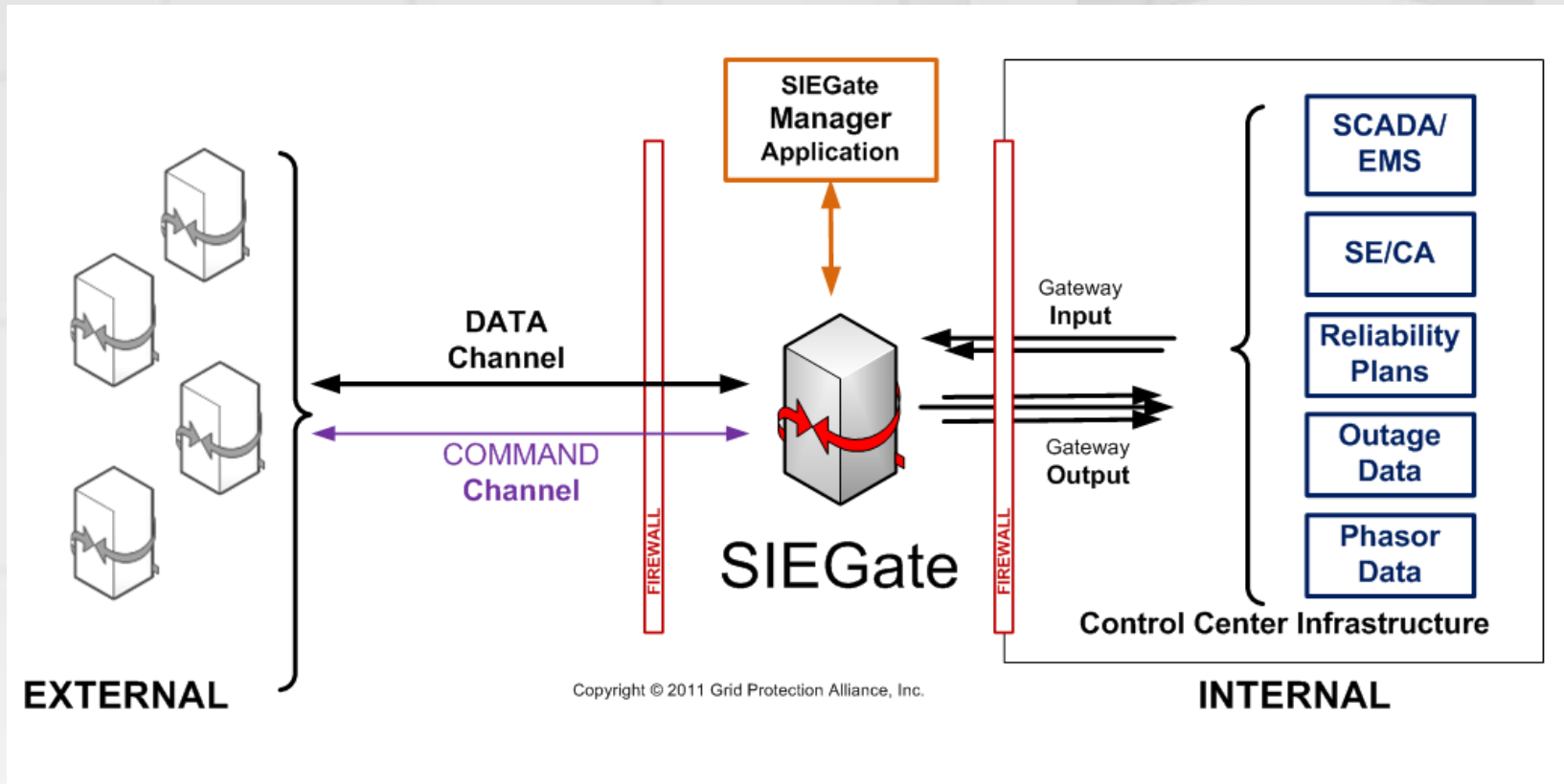
**and that integrates and interoperates easily with existing control room technology.**

# SIEGate Version 1.0

## High Level Requirements

- Security Throughout
  - At multiple levels: hardware, OS, application
- High Performance
  - Meet real-time requirements
  - Scalable to meet growing capacity needs
- Support for subset of power protocols
  - DNP3, IEEE C37.118, IEC 61850-90-5, also Modbus, ICCP and SDX expected

# SIEGate Implementation



# SIEGate Core Functionality

- Reliably exchange high-sample rate signal values and timestamps (measurements) with other gateways so that this information moves between with minimum time delay
- Enable gateway administrators to easily select the measurement points which are to be made available to owners of other gateways
- Enable gateway administrators to easily select the points that they chose to consume (i.e., the subset of the points made available to them) from other gateways

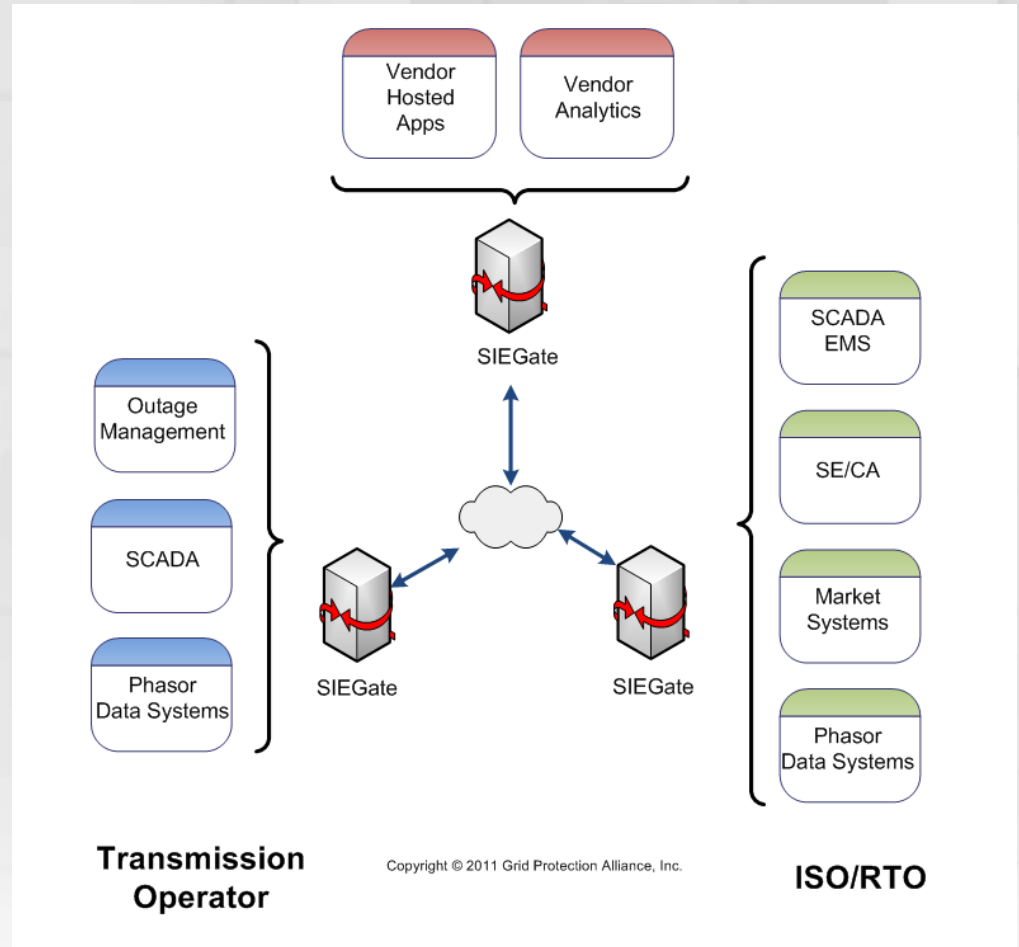
# SIEGate Core Functionality

(continued)

- Detect, log and **alarm** on communications issues
- Be implementable as a **high-availability** solution that can meet NERC **CIP compliance** requirements
- Support **encrypted communication** among gateways as well as **minimize bandwidth** requirements for gateway-to-gateway data exchange
- Utilize **standard communications, networking and server hardware**
- Be **easily extensible** to support the development of custom interfaces to the gateway owner's internal infrastructure and/or new phasor data protocols

# SIEGate Uses

- Case 1
  - RC to RC
- Case 2
  - TOp to RC
  - BA to RC
- Case 3
  - TOp to Distribution Ops
  - BA to BA
  - TOp to TOp
- Case 4
  - RC/Top/BA to Wide Area Service Provider (SANFR)



# SIEGate Data Classes

- Real Time Measurements
    - Phasor Data
    - SCADA Data
  - Batch Data
    - Disturbance Data
    - Planning Data
- Possible Future Classes:*
- Emergency Data (*extremely important data*)
  - Control Commands

# SIEGate Security Profile

- Availability – HIGH
- Integrity – VERY HIGH
- Confidentiality -- MODERATE



# Alarming and Notifications

- Bad data quality
- Security exceptions
  - E.g., Integrity failures, connection failures, access control
- Attestation failures
- Configuration changes
- System health

# Who “touches” a SIEGate?

- The SIEGate application is like an ICCP node in a control center
- As a back-office tool, SIEGate is administered by specialists, and likely to become part of critical infrastructure
- For security and compliance, change is tightly managed

# Who uses SIEGate?

- Entergy and TVA are active current users with MISO, PJM and Southern Company scheduling installations
- WECC has installed and tested the openPG (the predecessor of SIEGate)
- Dominion and Duke have expressed a desire to install to examine capabilities

# *open* Historian

GRID PROTECTION ALLIANCE

# What is a data historian?

- A non-relational database that is optimized for handling time-based process data
  - Data must be in the form of (time, value)
- Effectively handles very large volumes of data
- High performance read/write operations
- Easy migration of older data to less expensive, second tier storage media

# Why install a historian?

- Relational systems are not a good fit for phasor data
  - Do not scale well (record overload & retrieval responsiveness)
  - Cost - higher storage consumption per point
  - Data backup processes can be problematic (outages and network congestion)
- Typical Historian uses in a Control Room Architecture
  - SCADA/EMS Data Storage
  - Primary Phasor Data Storage
  - Second Tier Phasor Data Storage

# Who are historian vendors?

- **GPA**
- **OSIsoft PI**
- **eDNA**
- **Honeywell Uniformance PHD**
- **GE Proficiency Historian**
- **Industrial SQL Server Historian**

# Who “touches” a data historian?

- A historian is like an enterprise-wide relational system (e.g., work management) that’s just for operational, or process control, data. It requires diligent administration to enable enterprise-wide use
- A historian is used as the common point for systems to consume operational data in near-real-time; i.e., within about 1second of real-time
- Many engineers and analysts interact directly with a historian to obtain historical operating data



# openHistorian 1.0 vs. 2.0

## Version 1.0

- ▶ Two instances of the archiver are embedded in the openPDC and openPG
  - Data Historian
  - Performance Historian
- ▶ Configuration managed through openPDC or openPG Manager
- ▶ Includes two tools for data extraction/display
  - Data Extraction Tool
  - Data Trending Tool

## Version 2.0

- ▶ Includes both archiver and server components
- ▶ Completely redesigned storage engine
  - Broader range of data types
  - Greater time precision
  - Improved storage efficiency
  - Improved performance
- ▶ Flexibility in implementation with integrated support for other open storage systems
- ▶ Includes an integrated suite of tools for data extraction and display

# openHistorian 2.0 Design Goals

- **Complete redesign of current historian to enable the openHistorian to be the nexus for operational data at all sampling rates**
  - ACID protects data integrity  
*Atomicity, Consistency, Isolation, Durability*
  - High Performance
  - Maximum storage efficiency
  - High-availability
  - Compliant
  - Flexibility in deployment for rapid integration

# Planned openHistorian 2.0 Components

- Archival Services
- Extraction Services and API
- Administrator's Console
- Web-based graphing/trending display
- Engineer's Trending Tool and Screen Builder
- Operator's Display
- Alarming / Notification Services

# openHistorian 2.0 Features

- Optimized for management of process control and other time-series data
- Very large volumes of data can be efficiently stored and be made available on line
- Both lossless and swinging-gate compression options available
- Real-time data streams can be exported for both the provided web-based display or other application needs
- Horizontally scalable
- Easy to install, easy to configure
- Low cost of ownership
- Performance logging and alarming

# openHistorian 2.0 Features

(continued)

- Condition-based collection
- Data scaling on extract based on a set of scaling factors that apply over a time range
- Name (tag) translation and support for 61850 naming
- COMTRADE file exports

# openHistorian is ACID Compliant

- **Atomicity** - requires that database modifications must follow an "all or nothing" rule. Each transaction is said to be atomic
- **Consistency** - ensures that any transaction the database performs will take it from one consistent state to another
- **Isolation** - refers to the requirement that no transaction should be able to interfere with another transaction at all
- **Durability** - that once a transaction has been committed, it will remain so

**ACID protects data integrity.**

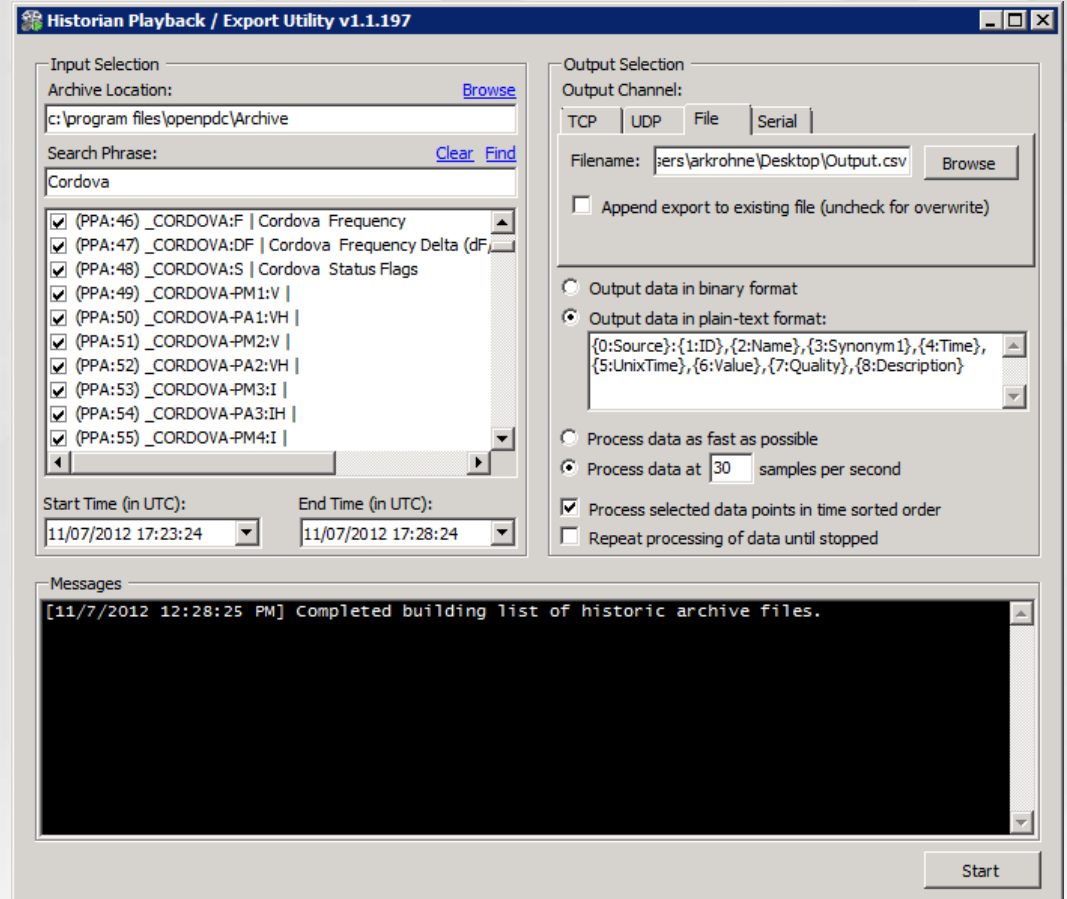
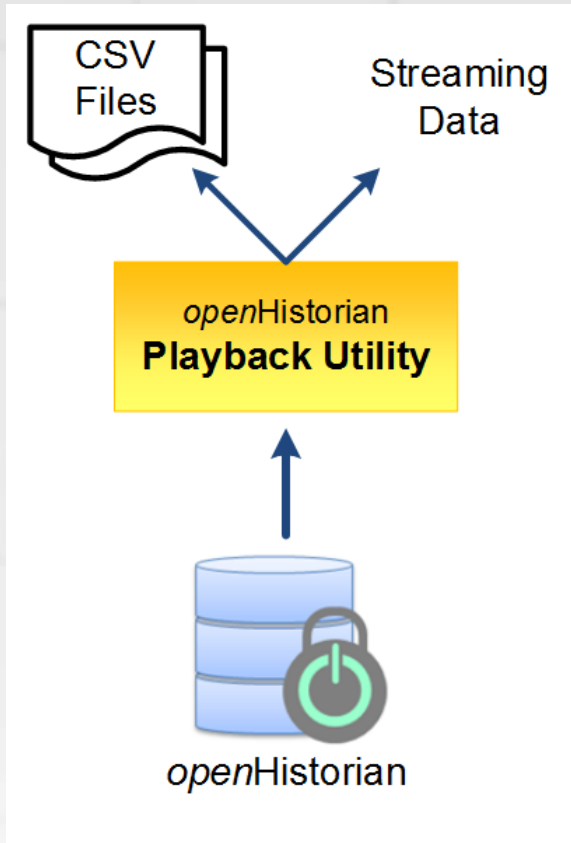
# Who else uses the openHistorian?

- 1.0 Implementations:
  - TVA has been a long term user (since 1995)
  - Dominion
  - PG&E
  - Entergy
  - Anyone hosting an openPDC
- 2.0 Alpha Implementations:
  - OG&E

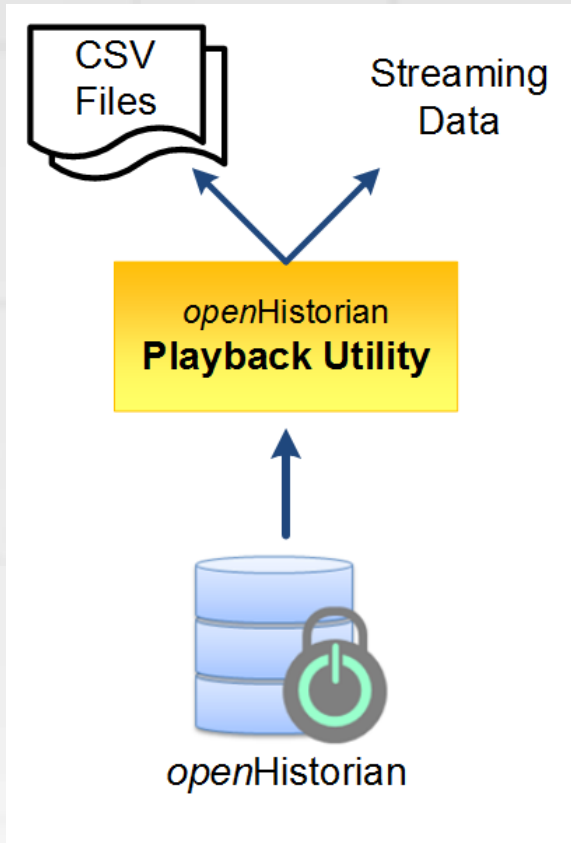
# openHistorian Screenshots



# Data Extraction



# Data Extraction



Historian Playback / Export Utility v1.1.197

Input Selection

Archive Location: c:\program file

Search Phrase: Cordova

Select Points

Output Selection

Output Channel: TCP | UDP | File | Serial

Filename: jers\arkrohne\Desktop\Output.csv

Output data in plain-text format:

```
{0:Source},{1:ID},{2:Name},{3:Synonym1},{4:Time},{5:UnixTime},{6:Value},{7:Quality},{8:Description}
```

Process data at 30 samples per second

Process selected data points in time sorted order

Repeat processing of data until stopped

Messages

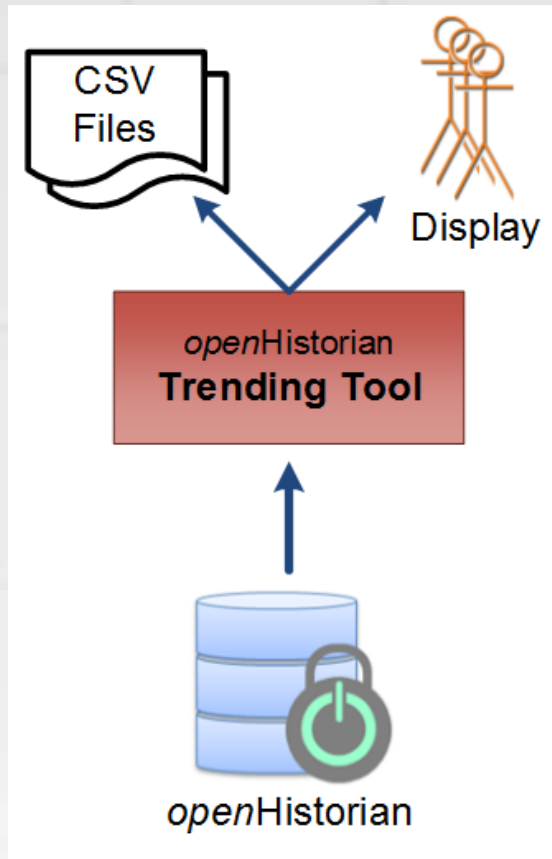
```
[11/7/2012 12:28:22 PM] Completed building list of historic archive files.
```

Select Timeframe

Create Output Format

Start

# Data Display



Historian Data Viewer

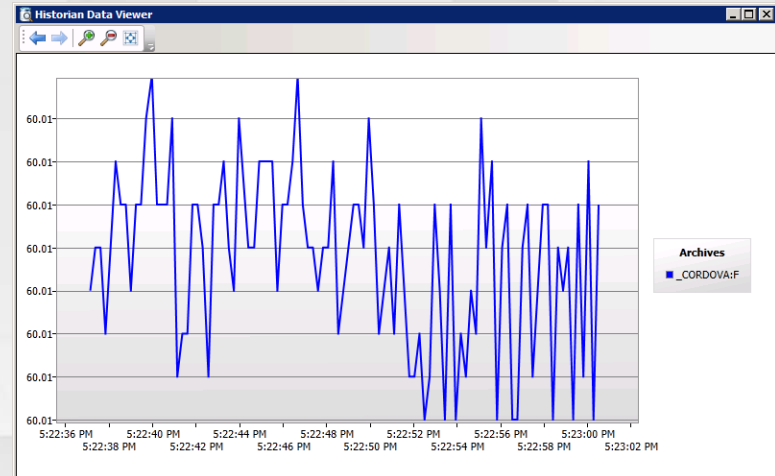
File Options

Search: Cordova Select All Deselect All

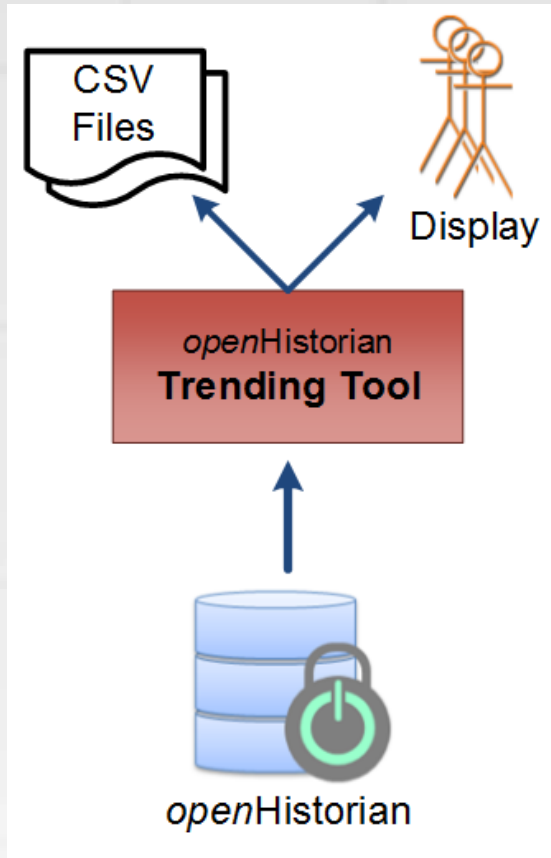
Export	Display	Pt. #	Name	Description
<input type="checkbox"/>	<input type="checkbox"/>	46	_CORDOVA:F	Cordova Frequency
<input type="checkbox"/>	<input type="checkbox"/>	47	_CORDOVA:DF	Cordova Frequency Delta (df/dt)
<input type="checkbox"/>	<input type="checkbox"/>	48	_CORDOVA:S	Cordova Status Flags
<input type="checkbox"/>	<input type="checkbox"/>	49	_CORDOVA-PM1:V	
<input type="checkbox"/>	<input type="checkbox"/>	50	_CORDOVA-PA1:VH	
<input type="checkbox"/>	<input type="checkbox"/>	51	_CORDOVA-PM2:V	
<input type="checkbox"/>	<input type="checkbox"/>	52	_CORDOVA-PA2:VH	
<input type="checkbox"/>	<input type="checkbox"/>	53	_CORDOVA-PM3:I	
<input type="checkbox"/>	<input type="checkbox"/>	54	_CORDOVA-PA3:IH	
<input type="checkbox"/>	<input type="checkbox"/>	55	_CORDOVA-PM4:I	
<input type="checkbox"/>	<input type="checkbox"/>	56	_CORDOVA-PA4:IH	
<input type="checkbox"/>	<input type="checkbox"/>	57	_CORDOVA-PM5:I	
<input type="checkbox"/>	<input type="checkbox"/>	58	_CORDOVA-PA5:IH	
<input type="checkbox"/>	<input type="checkbox"/>	59	_CORDOVA-PM6:I	
<input type="checkbox"/>	<input type="checkbox"/>	60	_CORDOVA-PA6:IH	

Last 5 minutes Chart resolution: 100 samples

Choose an interval relative to current time



# Data Display



The screenshot shows the **Historian Data Viewer** application window. A search bar contains the text "Cordova". Below the search bar is a table with columns: Export, Display, Pt.#, Name, and Description. A green box labeled "Select Points" has an arrow pointing to the table. At the bottom of the window, there are controls for "Last" (set to 5 minutes) and "Chart resolution" (set to 100 samples).

Export	Display	Pt.#	Name	Description
<input type="checkbox"/>	<input type="checkbox"/>	46	_CORDOVA:F	Cordova Frequency
<input type="checkbox"/>	<input type="checkbox"/>	47	_CORDOVA:DF	Cordova Frequency Delta (df/dt)
<input type="checkbox"/>	<input type="checkbox"/>	48	_CORDOVA:S	Cordova Status Flags
<input type="checkbox"/>	<input type="checkbox"/>	49	_CORDOVA-PM1:V	
<input type="checkbox"/>	<input type="checkbox"/>	50	_CORDOVA-PA1:VH	
<input type="checkbox"/>	<input type="checkbox"/>	51	_CORDOVA-PM2:V	
<input type="checkbox"/>	<input type="checkbox"/>	52	_CORDOVA-PA2:VH	
<input type="checkbox"/>	<input type="checkbox"/>	53	_CORDOVA-PM3:I	
<input type="checkbox"/>	<input type="checkbox"/>	54	_CORDOVA-PA3:IH	
<input type="checkbox"/>	<input type="checkbox"/>	55	_CORDOVA-PM4:I	
<input type="checkbox"/>	<input type="checkbox"/>	56	_CORDOVA-PA4:IH	
<input type="checkbox"/>	<input type="checkbox"/>	57	_CORDOVA-PM5:I	
<input type="checkbox"/>	<input type="checkbox"/>	58	_CORDOVA-PA5:IH	
<input type="checkbox"/>	<input type="checkbox"/>	59	_CORDOVA-PM6:I	
<input type="checkbox"/>	<input type="checkbox"/>	60	_CORDOVA-PA6:IH	

