Ritchie Carroll
Grid Protection Alliance

Tim Yardley
Erich Heine
University of Illinois

SIEGate

Secure Information Exchange Gateway

DOE - Cybersecurity for Energy Delivery Systems

GPA User's Forum – August 14, 2013

DOE Project -- SIEGate

Secure Information Exchange for Grid Operations

A generalized, security hardened appliance for the exchange of real-time grid operating information.

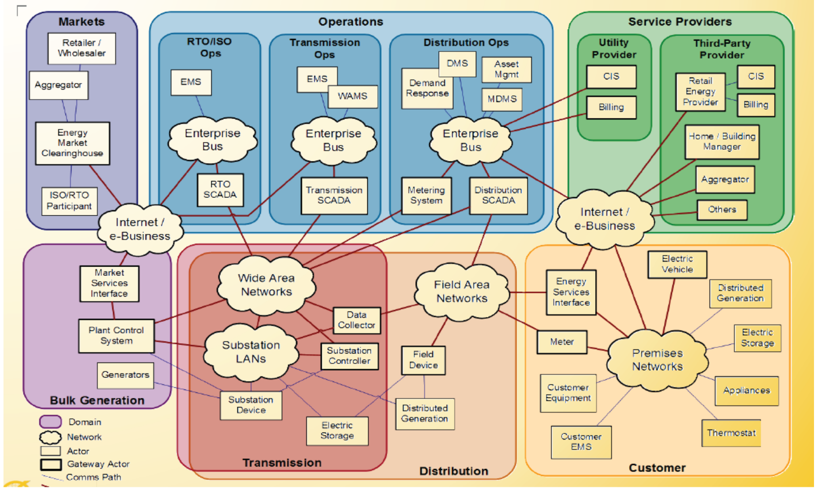
- Open source
- Productized by Alstom
- Security tested by PNNL
- Demonstrated by PJM
- NERC provides cost share via NASPI project

DE-OE536

SIEGate: Technical Approach and Feasibility

- **Current Situation**
  - Complex communication interactions using multiple protocols, many without security
  - Need a unified secure communication mechanism

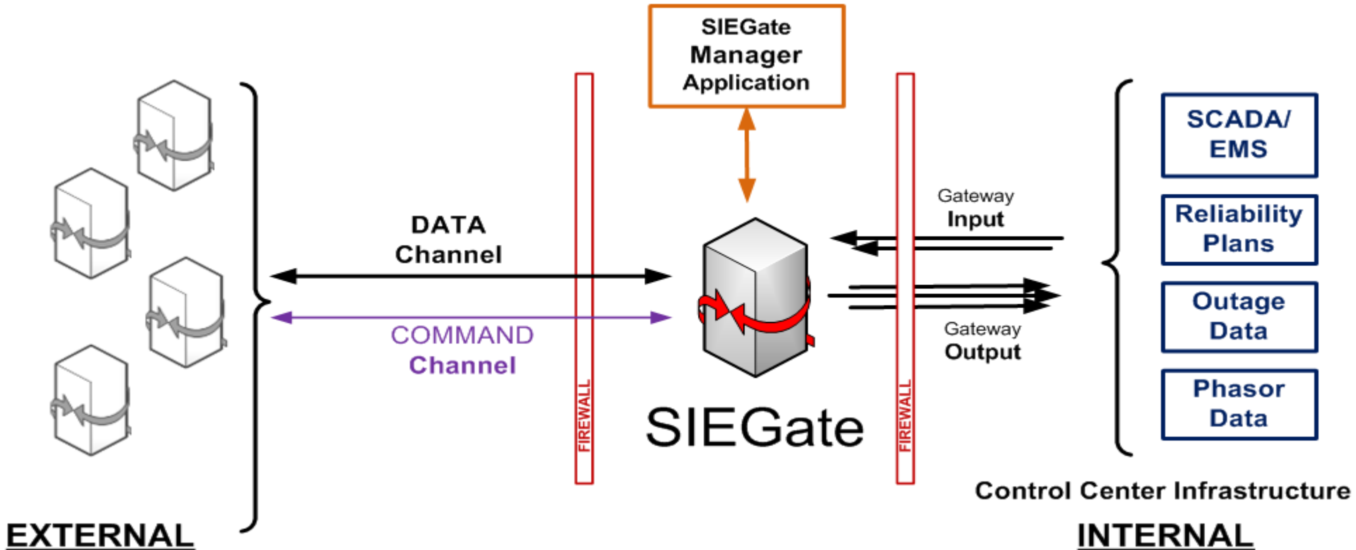
Smart Grid Architecture (Source: NIST)



- **Development Approach**
  - **Security built throughout**
    - Defense in depth
  - **High Performance**
    - Real-time, Built to purpose, Reliable, Flexible
  - **Multiple Protocols**
    SCADA, Synchrophasors, File-based
  - **Free, Open Source**
    Accelerated innovation
    Unencumbered commercialization
    Proven foundational codebase

SIEGate: Technical Approach and Feasibility

- **SIEGate adds value for grid operators**
  - Reduces configuration management costs
  - Improves security posture through a single point of interface
  - Reduces risk of non-compliance



Two Development Versions

- **Engine Development (ED) Version**
  - Incorporates a new "advanced core" that improves internal system security
  - Available as source-code for download
  - Currently undergoing refinement and testing
- **Feature Development (FD) Version**
  - Includes all the features and functionality of SIEGate with the exception of the "advanced core"
  - Ready for use and evaluation within pre-production control center environments
  - Release Candidate 2 posted and available for download and installation

Feature Development Release Candidate

- **Support for Multiple Data Classes**
  - Real-time data, e.g., SCADA and phasor data
  - File-based data exchange
  - Notifications for broadcast to all nodes
- **TLS Connections for SIEGate data exchange**
- **Tools to setup trusted SIEGate unions and to automate configuration synchronization**
- **Many improvements over the openPG in management and configuration**
- **Built using new .NET 4.5 Grid Solutions Framework**

With Release of “FD” Version of SIEGate, openPG has been retired

- **SIEGate provides all openPG functionality – and more.**
- **SIEGate has stronger security and better performance**
- **SIEGate is actively undergoing additional testing and refinement**

PNNL Testing (“ED” Version)

- **Test plan established**
  - Structured evaluation
    - Code review
    - Data validation
    - Key Management
  - Exploratory Tests
- **Review and testing initiated in June 2013**

PJM Demonstration (“FD” Version)

- **Demonstration plan developed**
  - Bench testing within PJM Lab
  - Data exchange between PJM & GPA
  - Security Testing
  - Optionally, data exchange with others
- **Installation scheduled for August 29th**

Entergy-TVA-MISO Implementation

- **SIEGate (FD) Release Candidate 2 in service at Entergy and TVA as of August 6, 2013**
- **No issues discovered in installation or configuration**
- **TVA Gateway configured so that Entergy may subscribe to any of approximately 800 measurements from TVA**
- **SIEGate installed at MISO and is being configured for exchange of phasor data with Entergy**

Erich Heine
University of Illinois

SIEGate Technical Details

SIEGate: Summary

- **Objective**

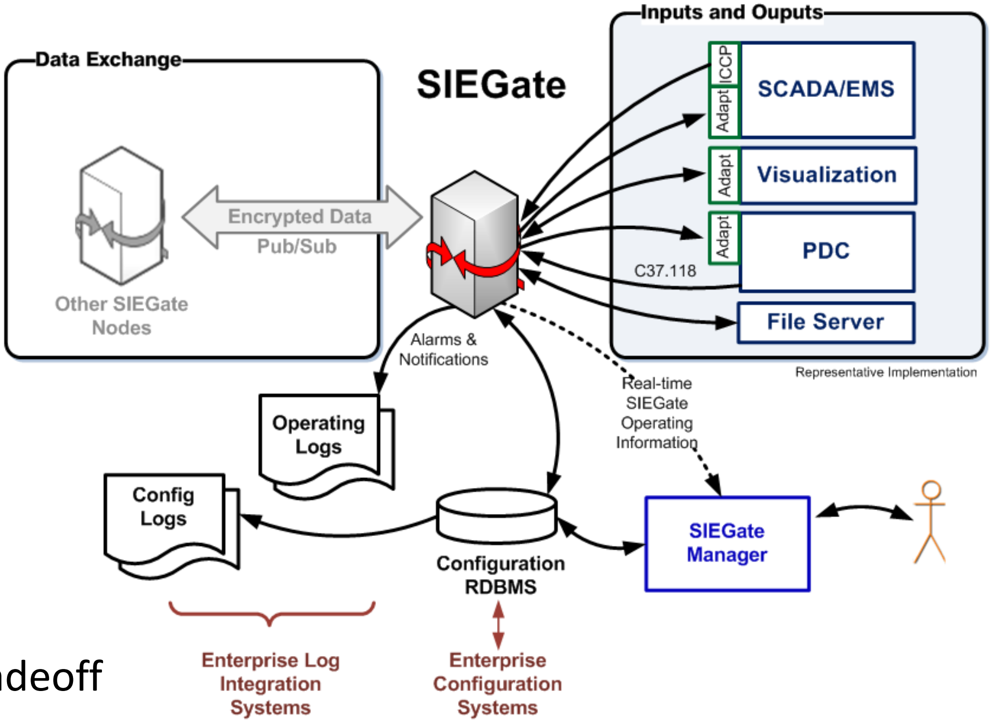
  To commercialize an appliance that enables the secure exchange of all types of reliability and market data among grid operating entities and provide a next-generation platform for GPA Open* products

- **Design Approach**
  - Lower risk by building upon the open source phasor gateway
  - Create an extensible platform
  - Design security throughout
  - Balance real-time and security needs
  - Conduct thorough bench tests to identify and fix security defects

- **Technical Goals**
  - Maintain time-series framework compatibility
  - Provide enhanced performance with new core
  - Leverage intelligent responsibility separation
- **Development Partners**
  - Grid Protection Alliance; University of Illinois
- **Test and Demonstration Partners**
  - Pacific Northwest National Laboratory, Alstom Grid, and PJM Interconnection

SIEGate: Technical Design Challenges

- **Performance given system complexity**
  - Support multiple data types efficiently and securely
  - Support multiple priorities
  - Minimize latency and maximize throughput
- **High availability assurance**
  - Horizontal and vertical scalability
  - SIEGate stability and reliability
  - Graceful performance degradation
- **Security assurance**
  - Maximize security performance
  - Minimize security breach impact
  - Configurable security levels
  - Security versus simplicity/usability tradeoff



Engine Development

- **Adapting SIEGate to core design principles**
  - Modular design
  - Single responsibility principle
  - Flexibility for easy adaptation
  - Enhanced security – from the ground up
    - Services, Locked-down installation, Hardened OS

SIEGate: Technical Design Principles

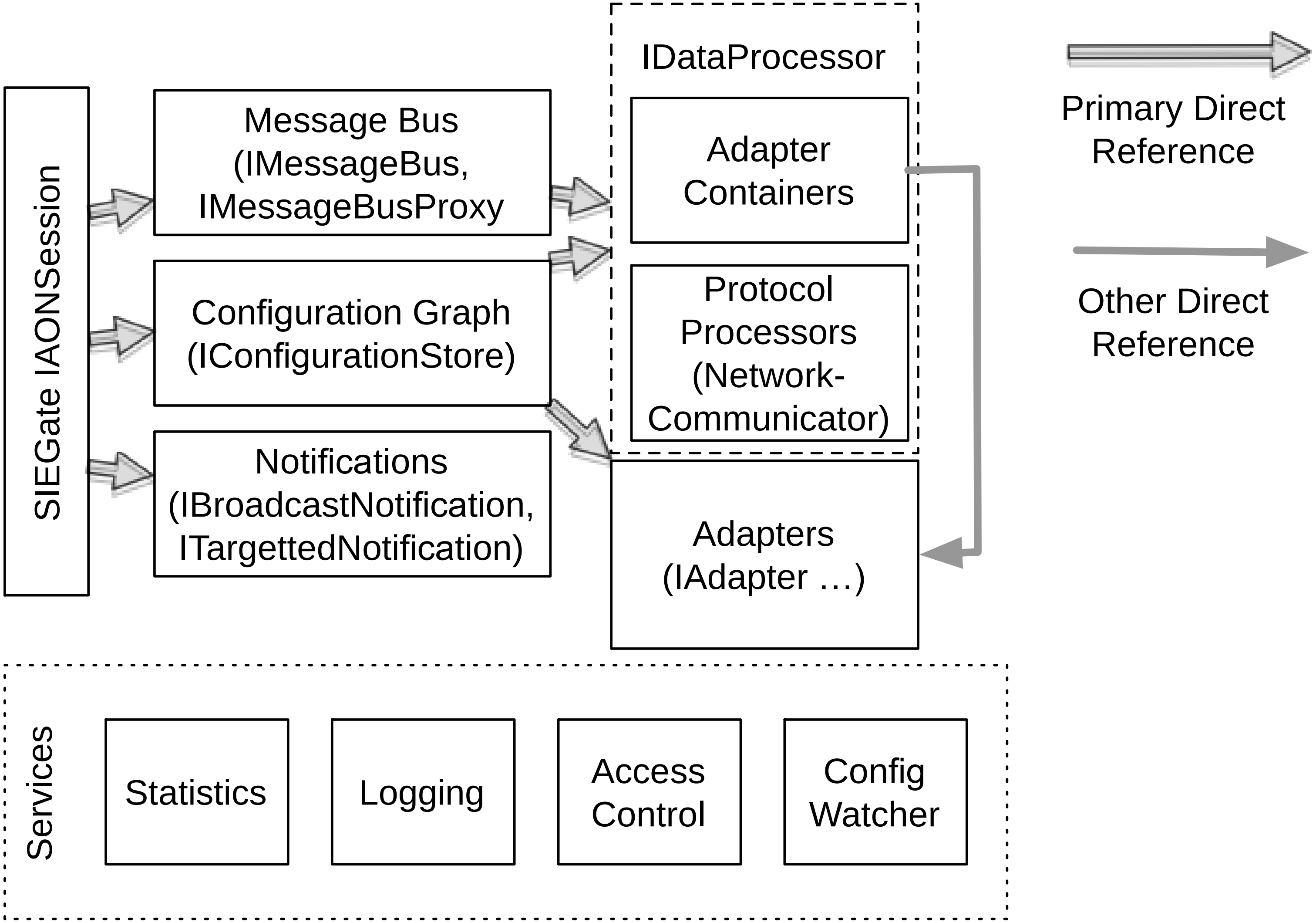
- Minimize thread-locking and contention
- Pre-compute criteria for decisions rather than on-the-fly
- Simplify resource access
- Choose heavy memory usage over heavy CPU
- Discard unneeded data as early as possible
- Provide extensibility & offloading
- Adhere to the single responsibility principle
- Maintain a layered approach to security and defenses
- Design components to operate with least privilege
- Leverage existing, tested components
- Pluggable component architecture

ENGINE DEVELOPMENT DEEP DIVE: MODULARITY IN CORE

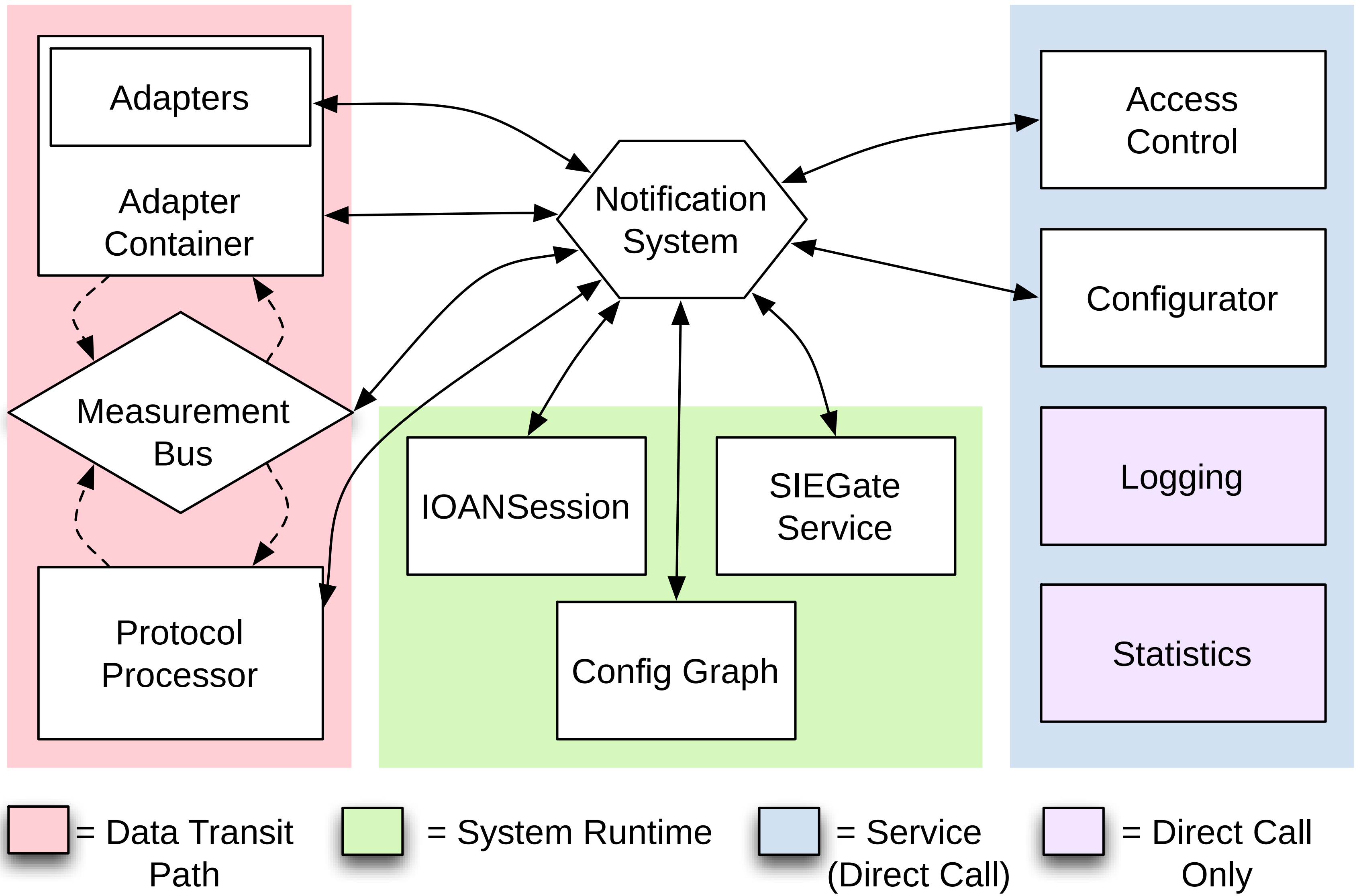
Modular Design Benefits (Development)

- **Increases flexibility**
  - Allows drop-in replacement of any component
  - Modules interface rather than intrude
- **Reduces code complexity**
  - Simplifies dependencies
  - Call graph has fewer "cycles"
  - Reduces "trampolining" between modules
- **Reduces coding errors**

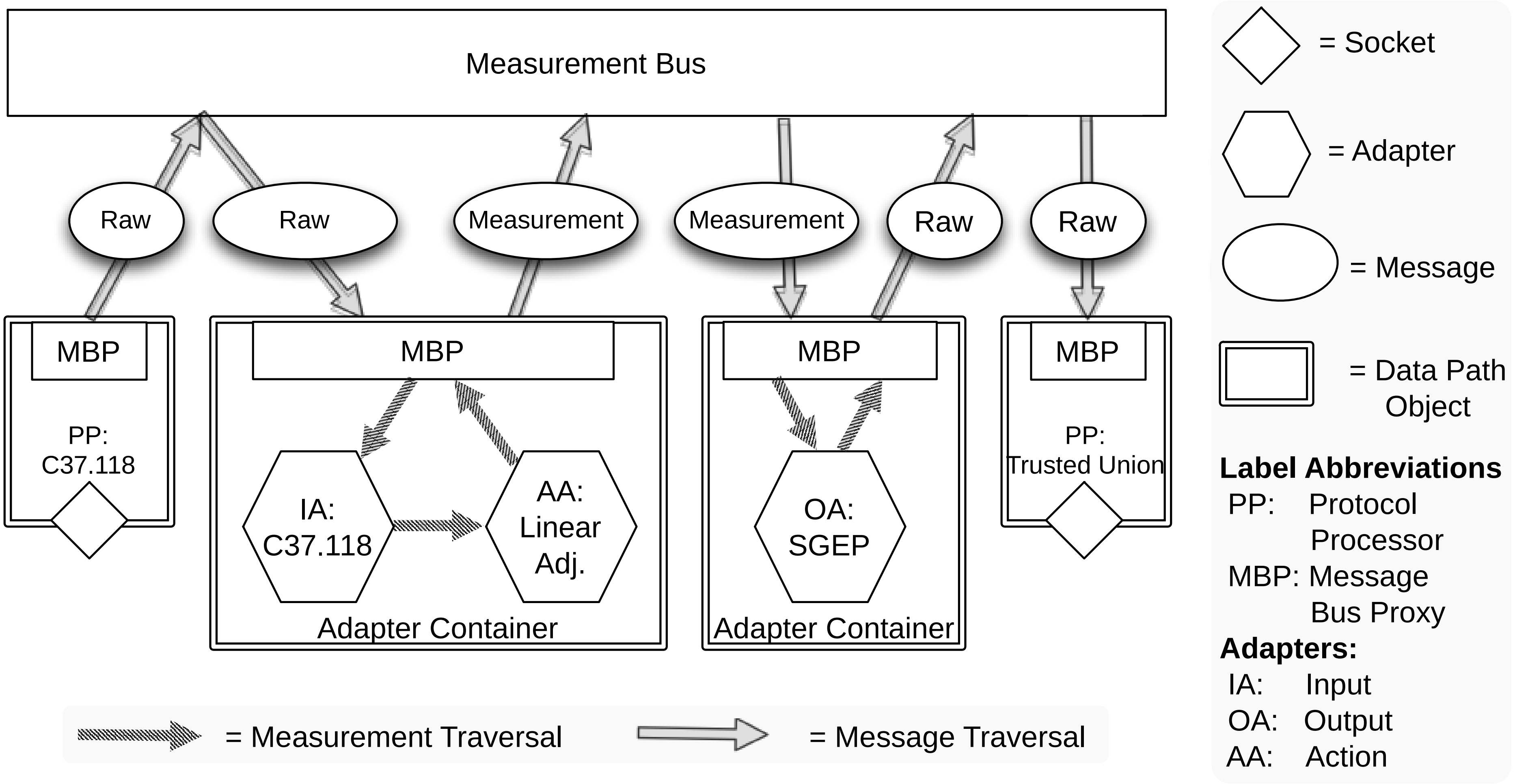
Modular Design



Decoupling the Data Path from Internal Messaging



Modular Data Path



Modular Design Benefits (Security)

- **Allows consistent policy application**
  - Access control is handled by a single service
  - No inconsistencies between how rules are applied
  - General credential service allows system wide revocation
- **Single Responsibility Principle**
  - Any defect occurs in only one code location
  - Any new security feature can be applied to everything immediately
- **Carefully restricted API to limit access**
  - Isolates functionality to minimize attack surface
  - Minimizes information leakage

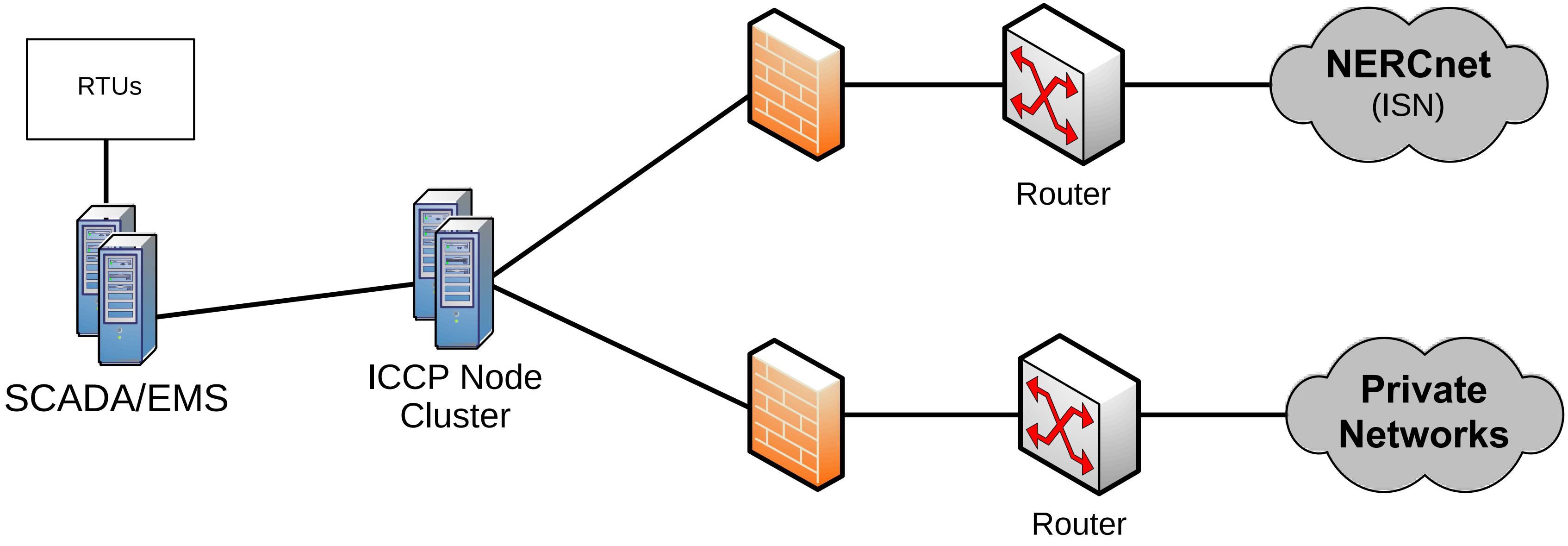
Furthering Security via Modularity: Going forward with Engine Development

- **Assembly Separation**
  - Per assembly security capabilities
  - Limit code available to the runtime to reduce attack surface
- **Composability for deployment specific security choices**
  - Public vs. private transport considerations
  - Computation speed vs. data security
- **Extensible security modules for enterprise integration**
  - Corporate credential managers
  - Network access rules

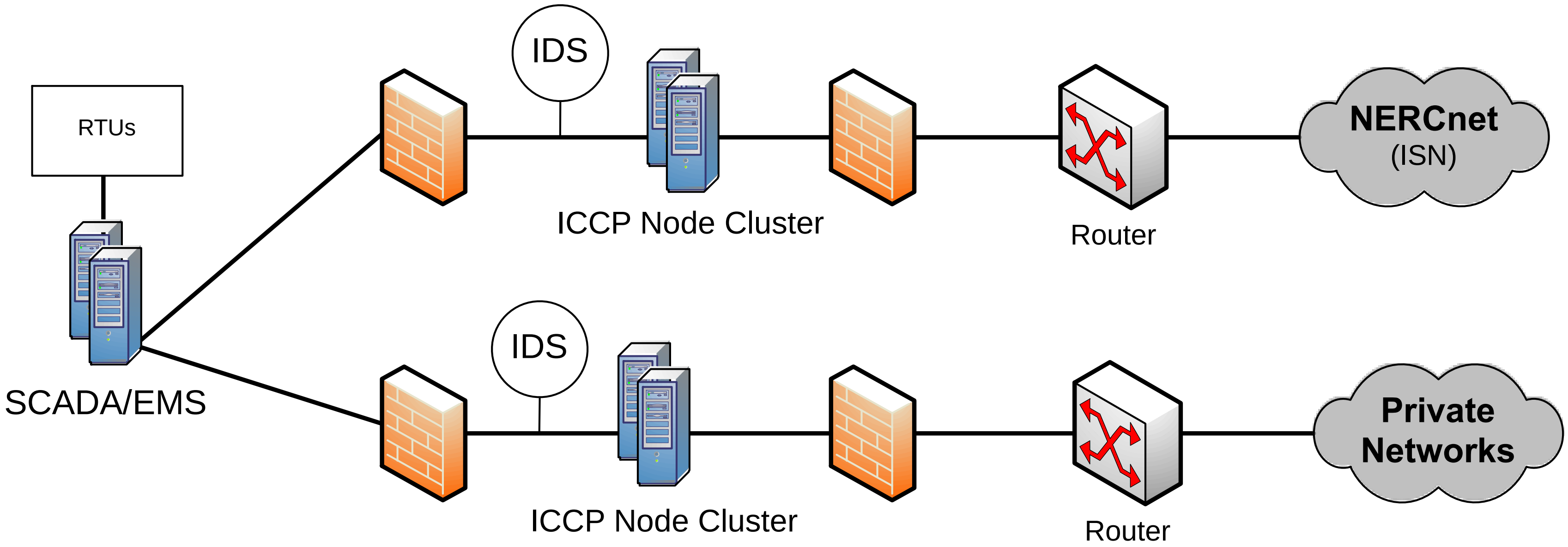
Tim Yardley
University of Illinois

Deployment Architecture Discussion

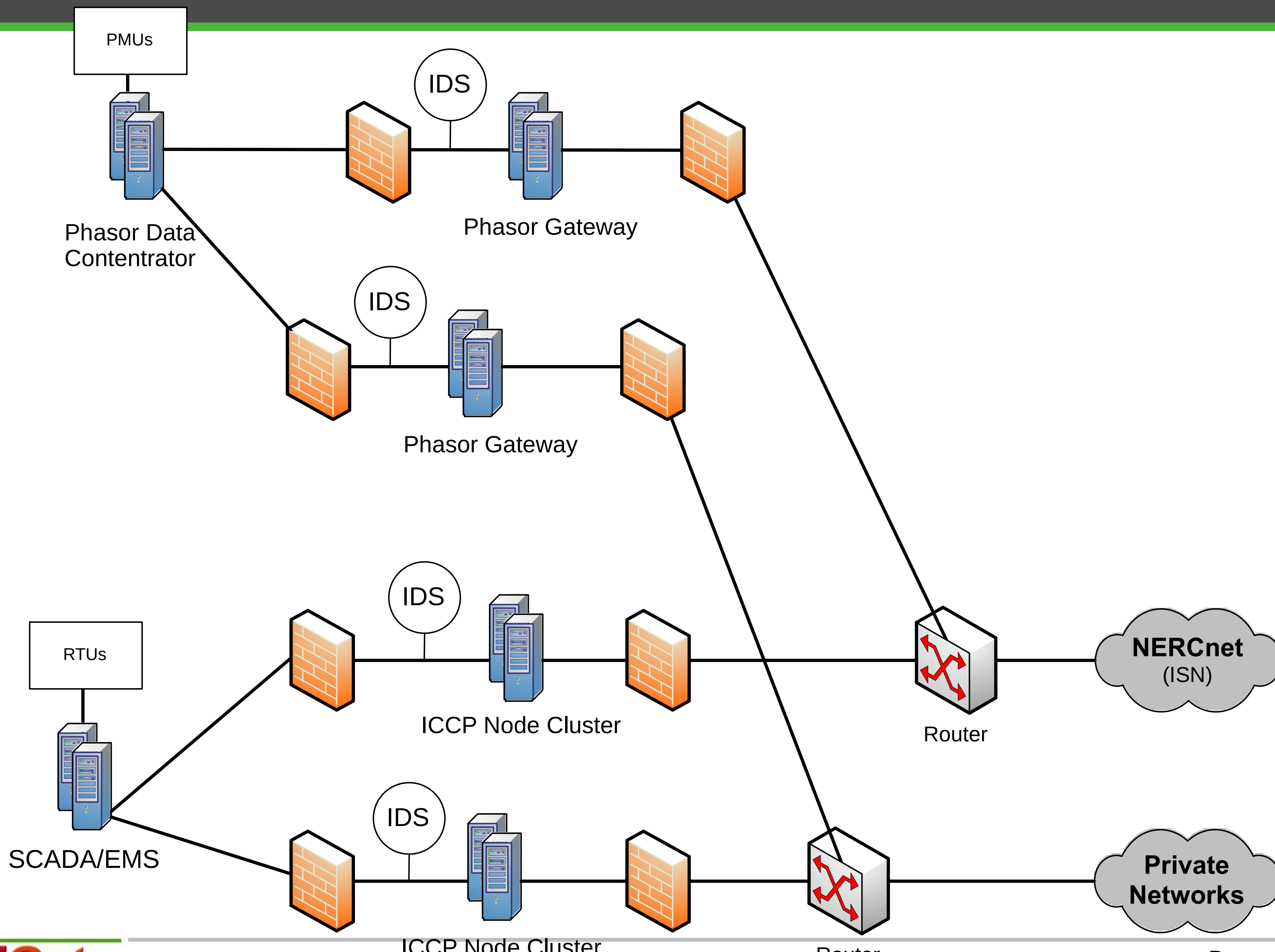
Legacy SCADA Architecture



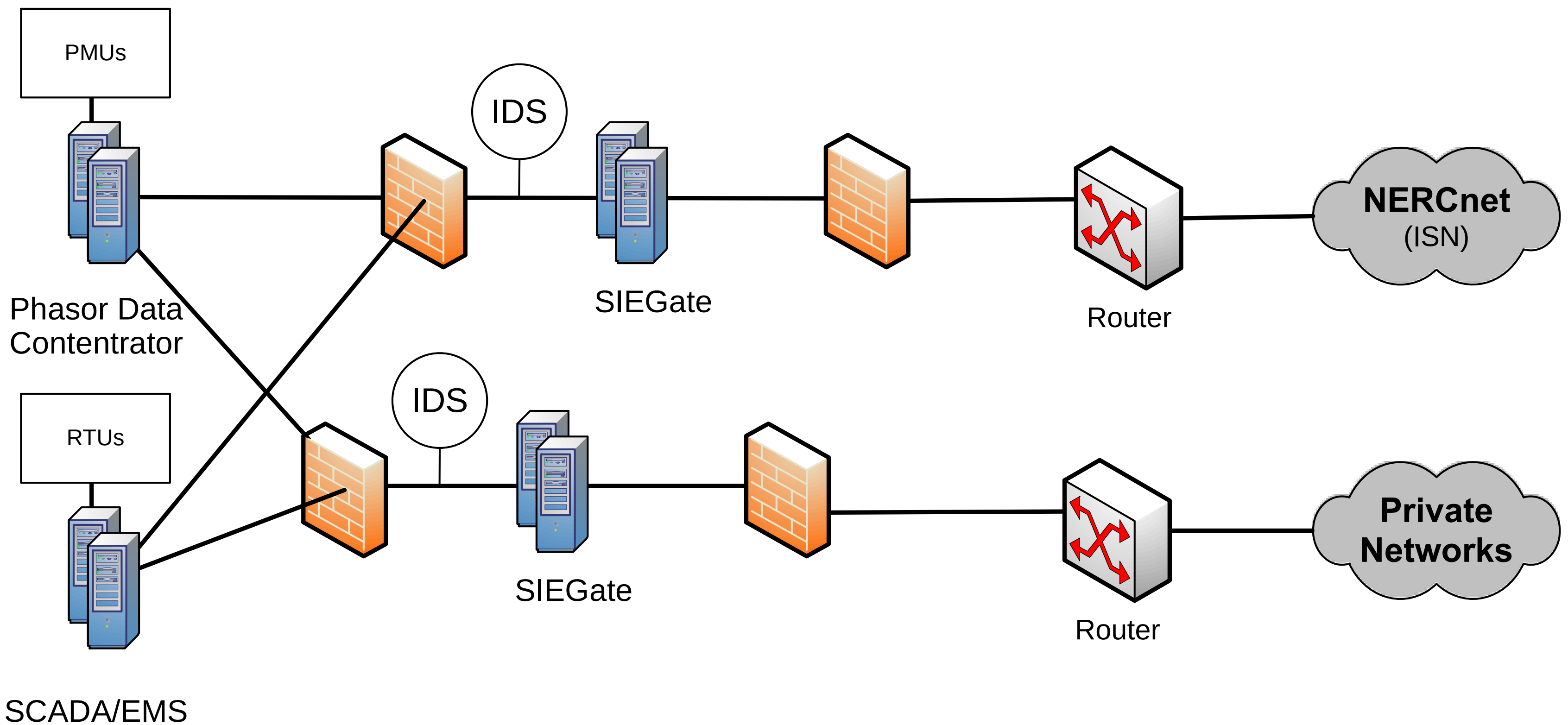
Modern SCADA Architecture



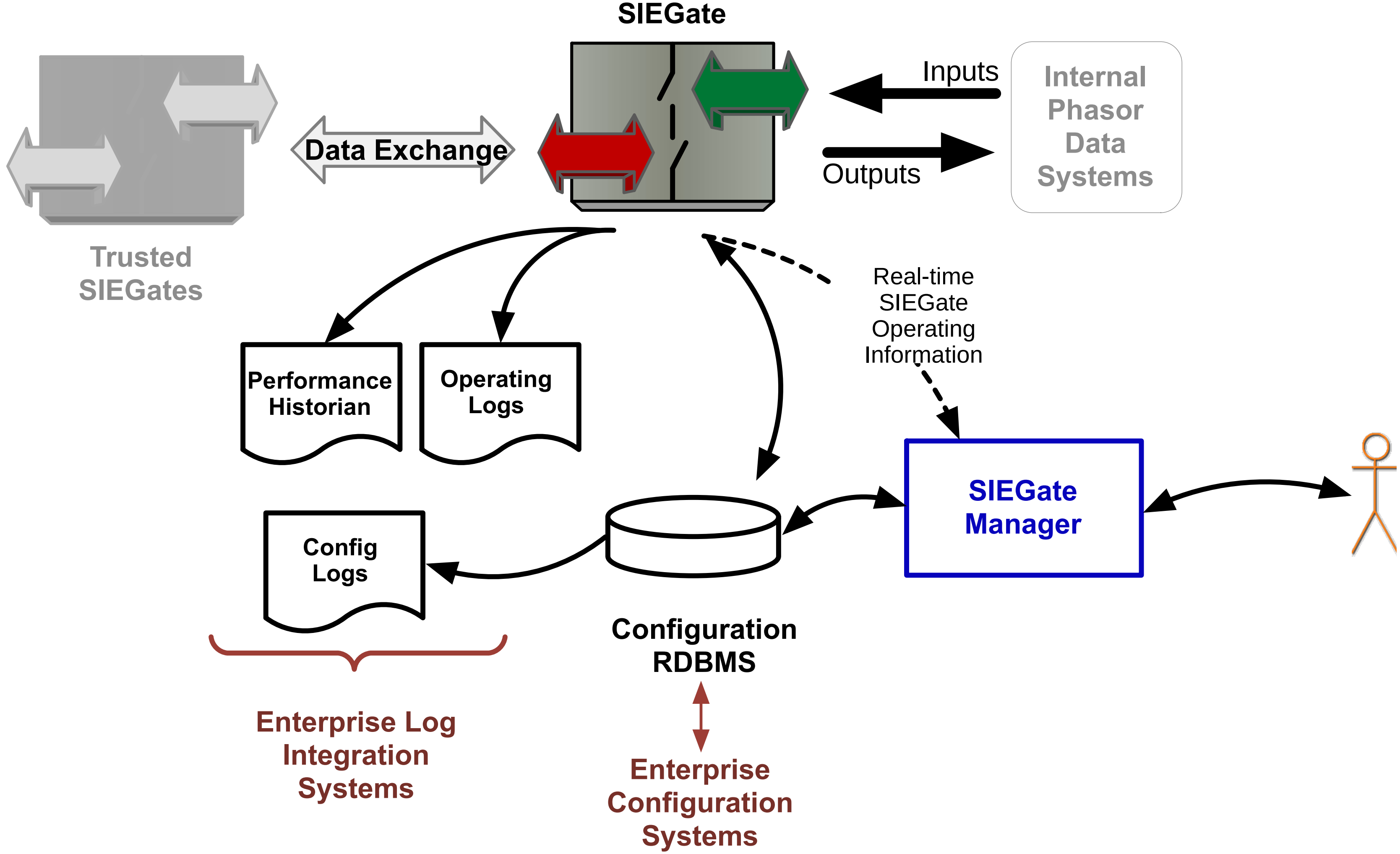
Incorporating Phasors



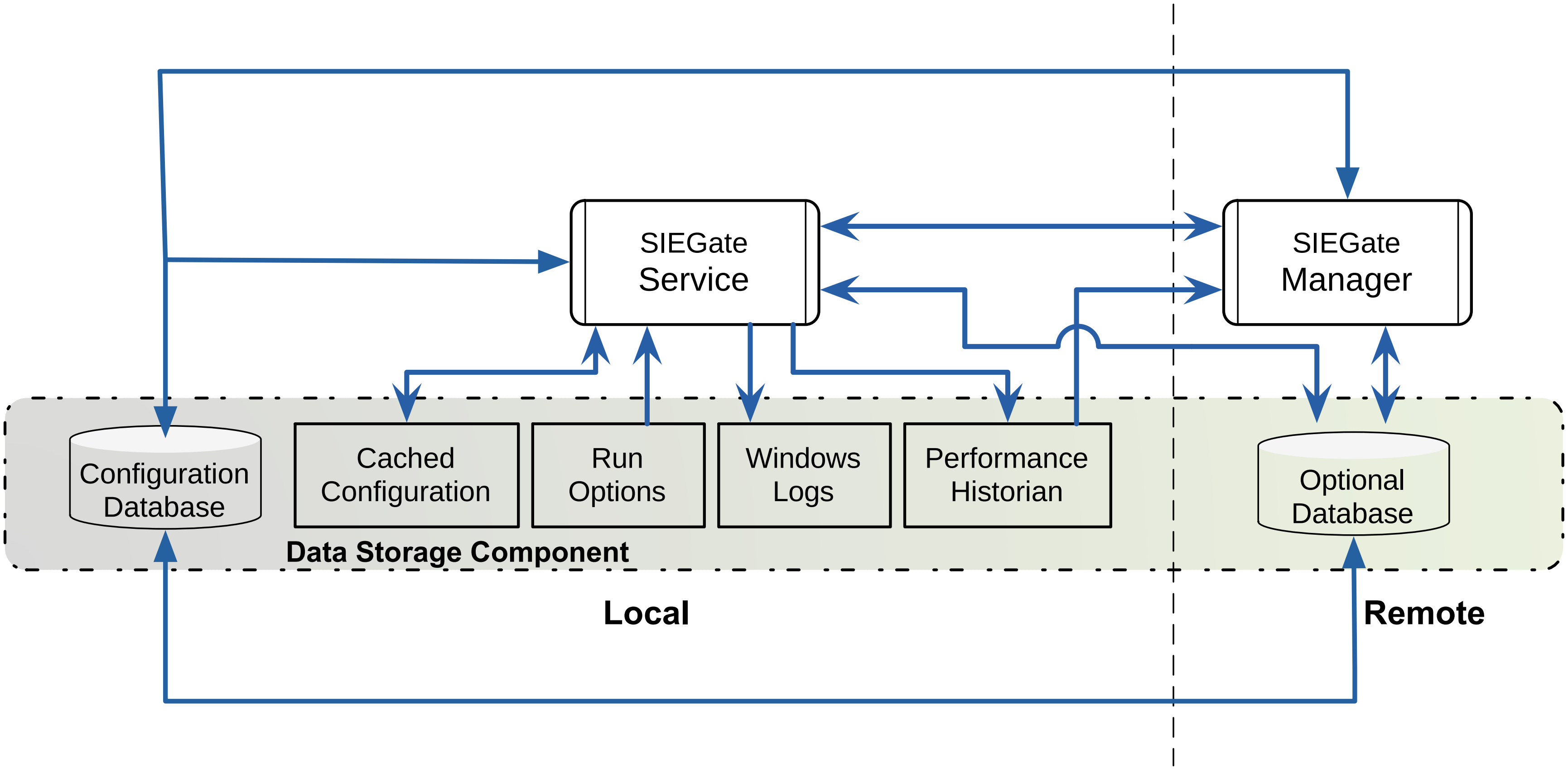
Simplifying with SIEGate



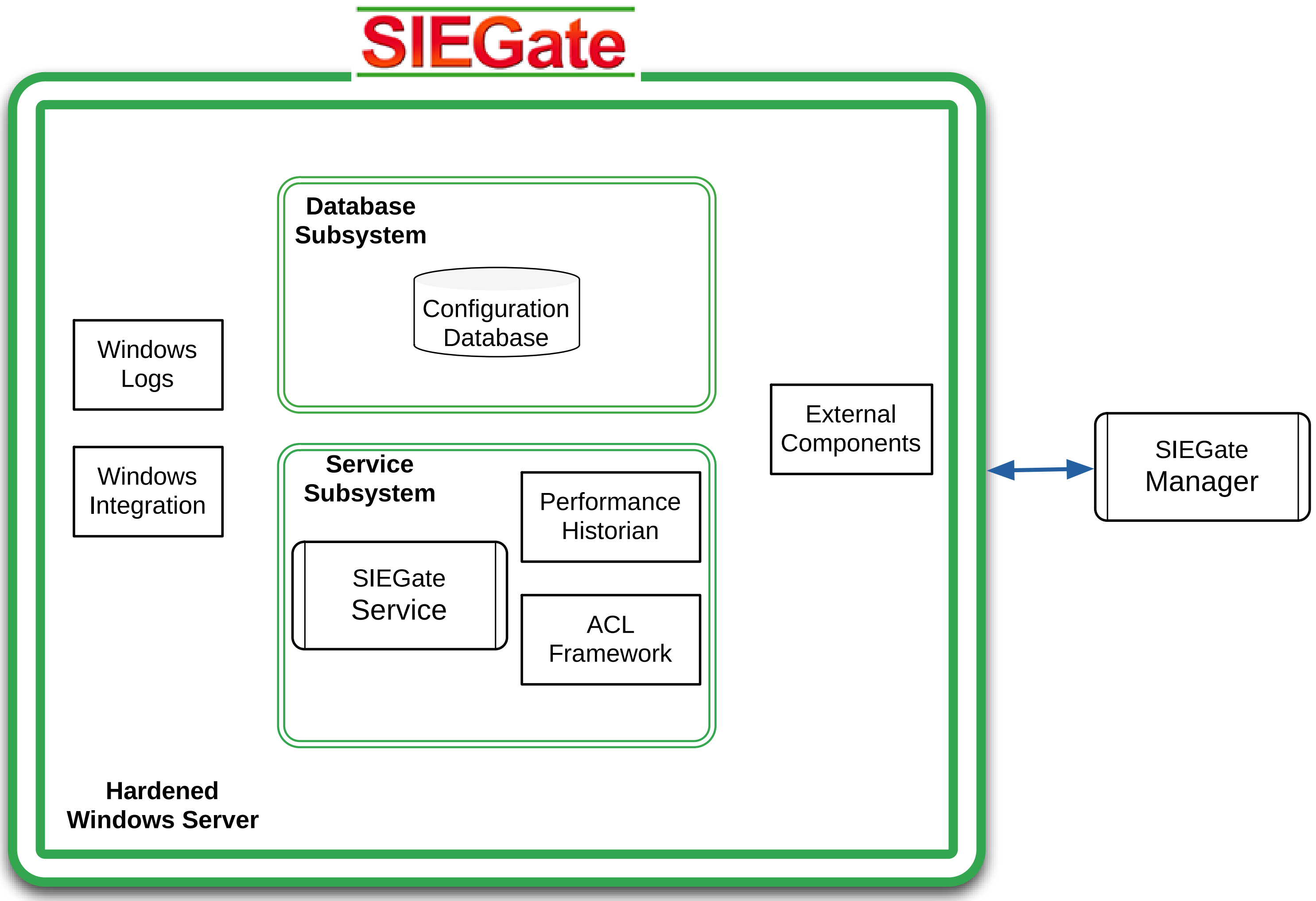
Basic Overview



System Breakdown

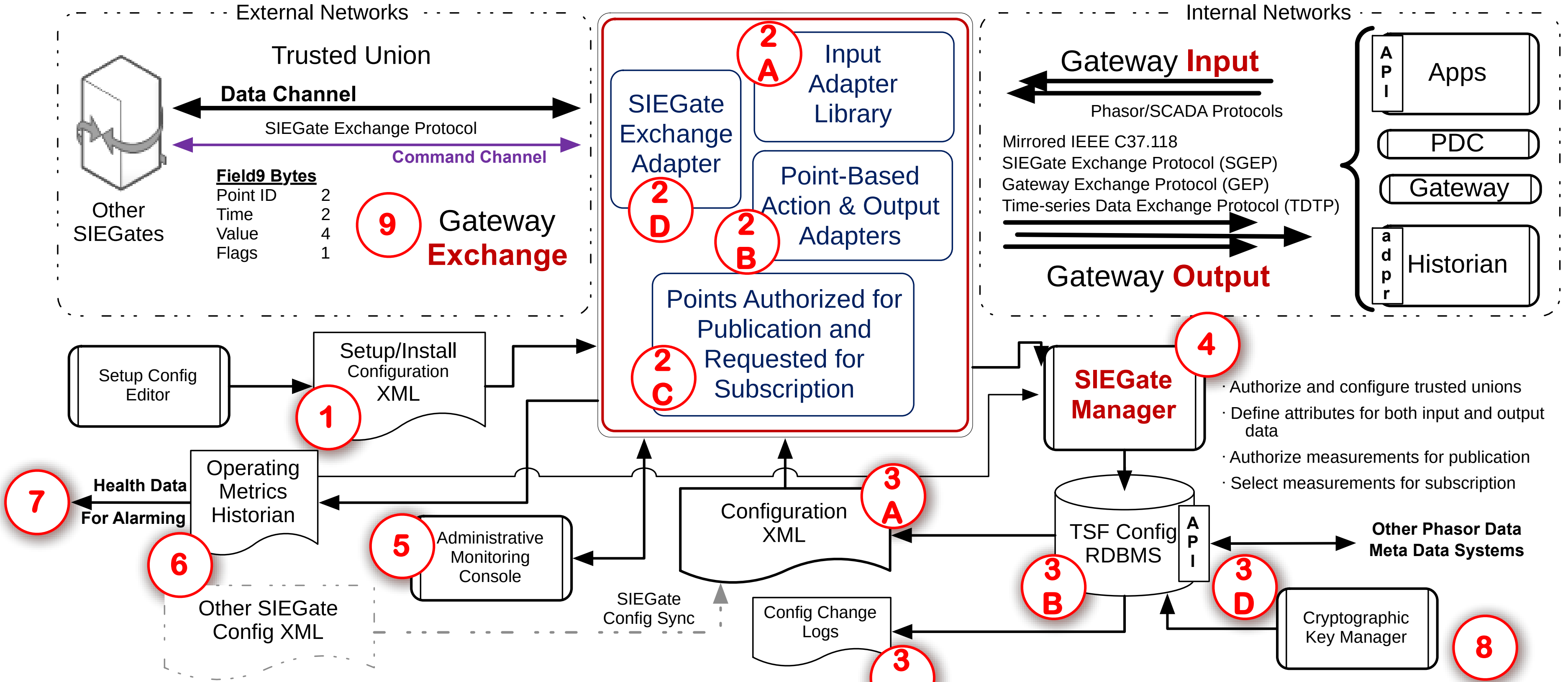


SIEGate Appliance



Detailed Architecture

SIEGate Service



SIEGate

Deployment Improvements

- **OpenPG**
  - Application-based install
  - Normal user privs.
  - Monolithic architecture
- **SIEGate**
  - Hardened OS
  - Composible components
  - User account restrictions
  - Application and Appliance installations
  - Windows Server 2008 R2 (standard and core)
  - Security policies for enterprise integration and audit

Future: Best Practice Guides

- **Coming soon…**
  - More granular security policies
  - Representative typical architectures
  - Guide on security related settings and what they mean for system security

… more (stay tuned)!