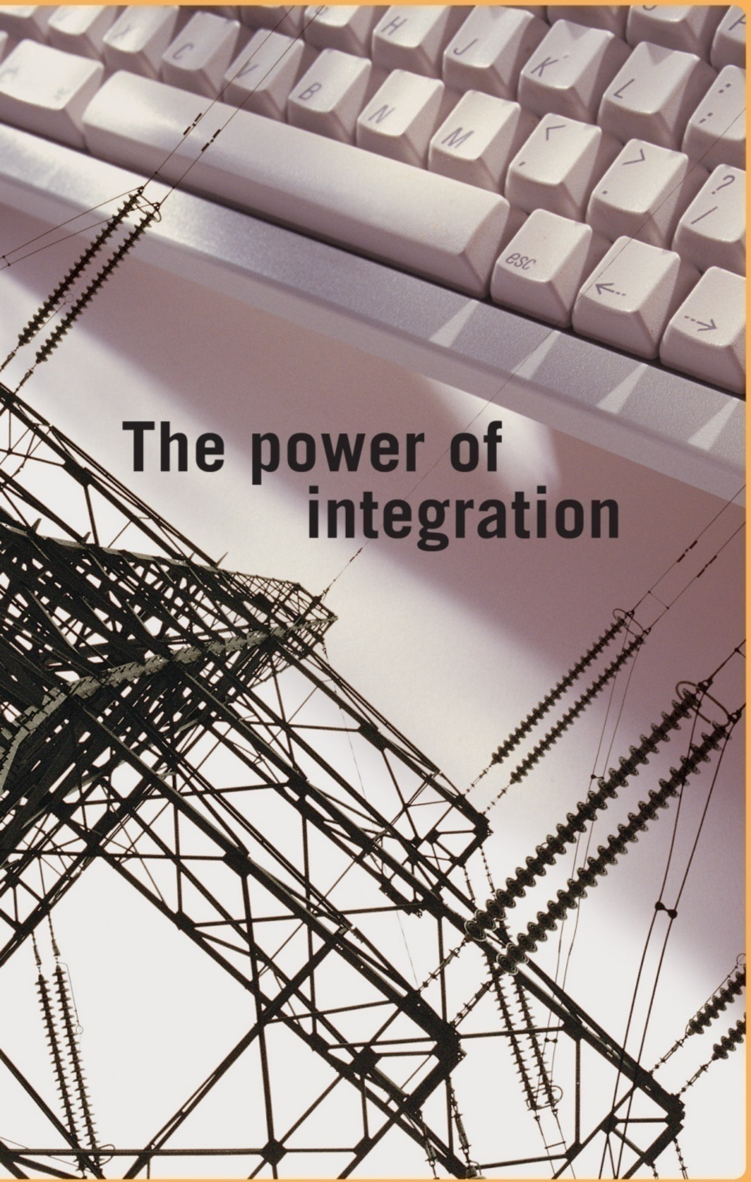


UISOL
Utility Integration Solutions, Inc.

CIM Interoperability Challenges

Terry Nielsen, UISOL
Scott Neumann, UISOL

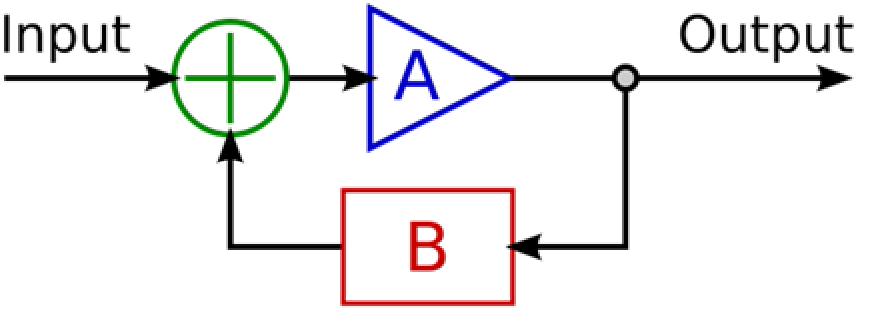
July 26, 2010

Introduction

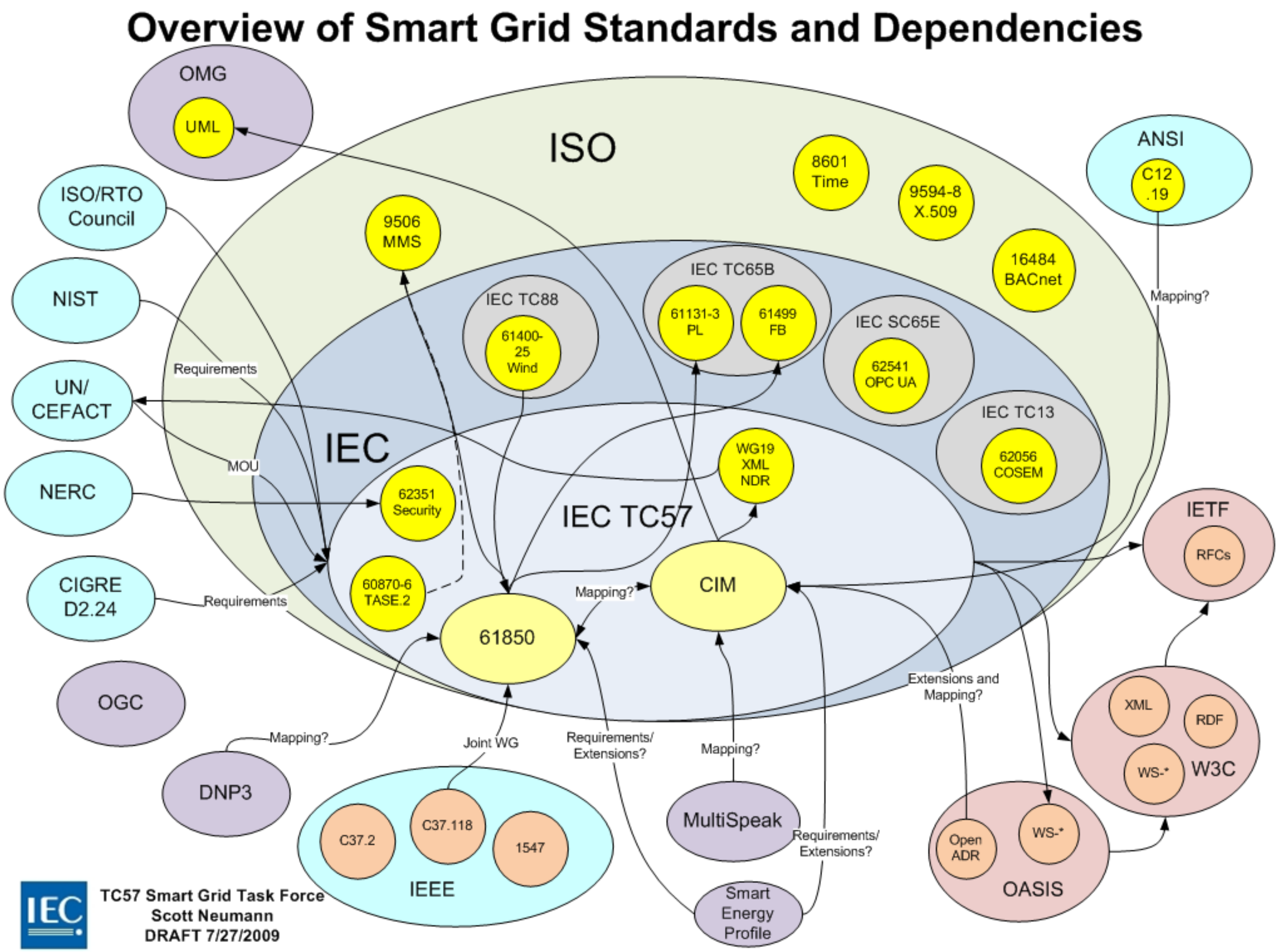
- IEC Common Information Model (CIM) as defined by the IEC 61970/61968 standard is a domain model central to many existing and future Smart Grid standards
- The original and ongoing intent of the CIM is to provide a basis for the definition of interfaces that improve interoperability
- The purpose of this presentation is to describe some of the interoperability challenges that are often encountered when using the CIM (or any common domain model) as the basis for the definition of interoperable interfaces

Intended Audience

- The intended audience of this presentation is for engineers with a basic understanding of systems integration, because engineers understand and recognize the importance of the concept of negative feedback
- We need to be aware of issues that need to be recognized and addressed if we are going to improve the state of interoperability within our industry



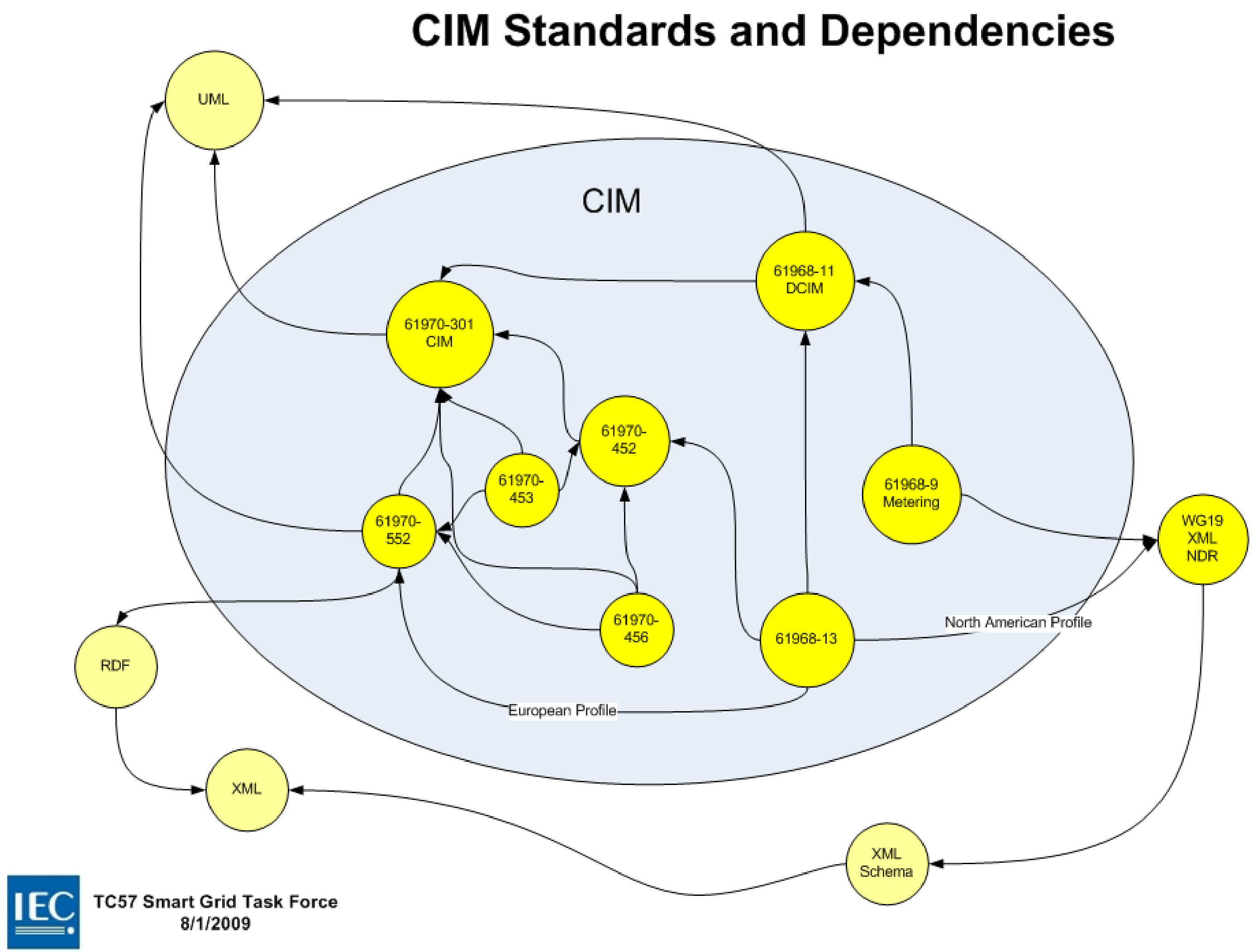
CIM in a World of Standards



Key IEC CIM Standards

IEC

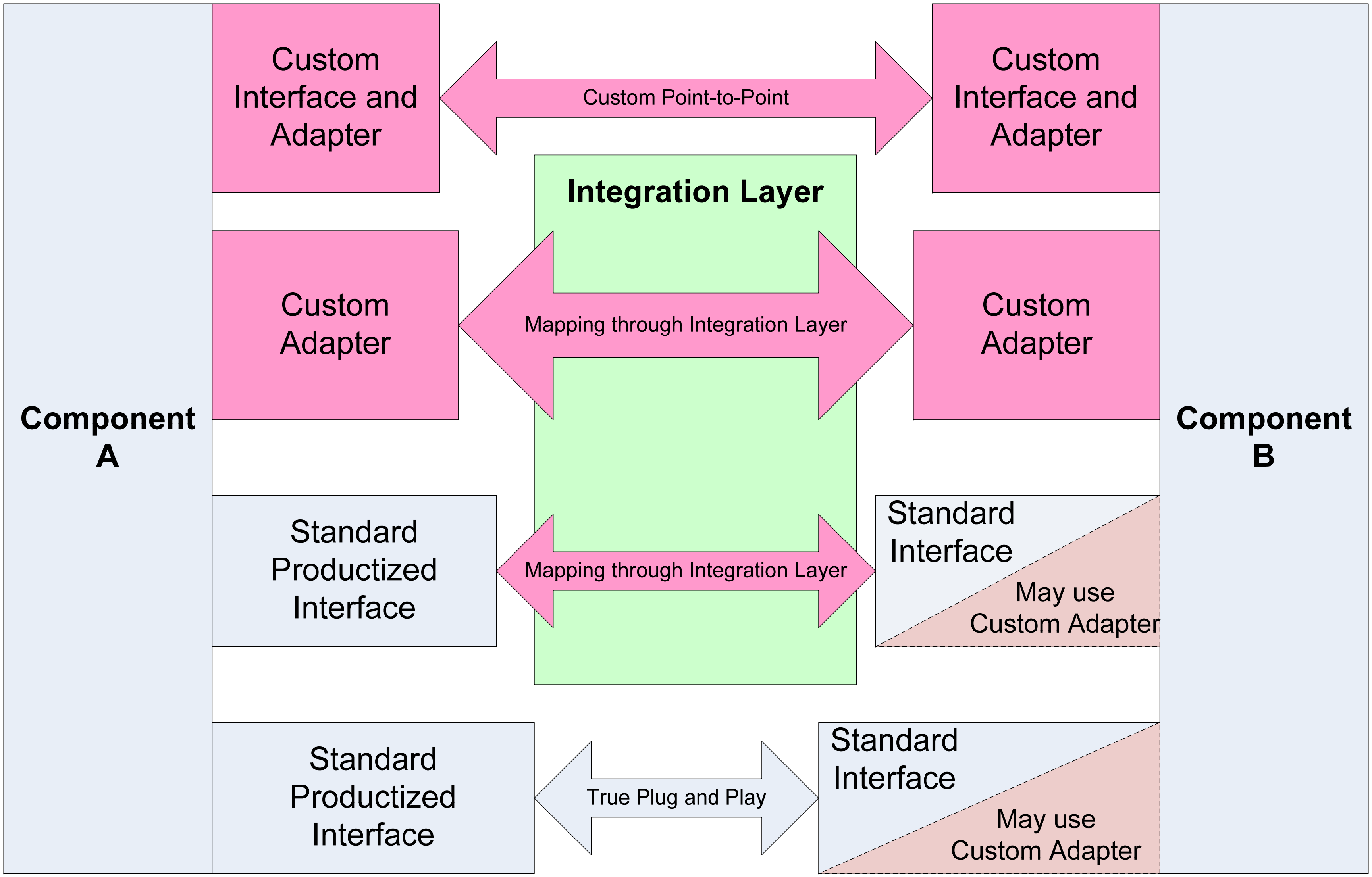
- 61970
- 61968
- 62325



Common Misconceptions

- All standards provide or imply interoperability
- ‘CIM compliant’ is a well defined term
- A UML model will describe all of the semantics of a model
- There is only one way to generate an XML Schema (XSD) from a UML model
- There is only one way to define an interface using a WSDL, and a WSDL guarantees interoperability
- All XML information exchanges are defined using XSDs
- XSD validation can be used to completely validate message content
- All SOAs use web services, and use them the same way
- Web services can be readily support all integration patterns

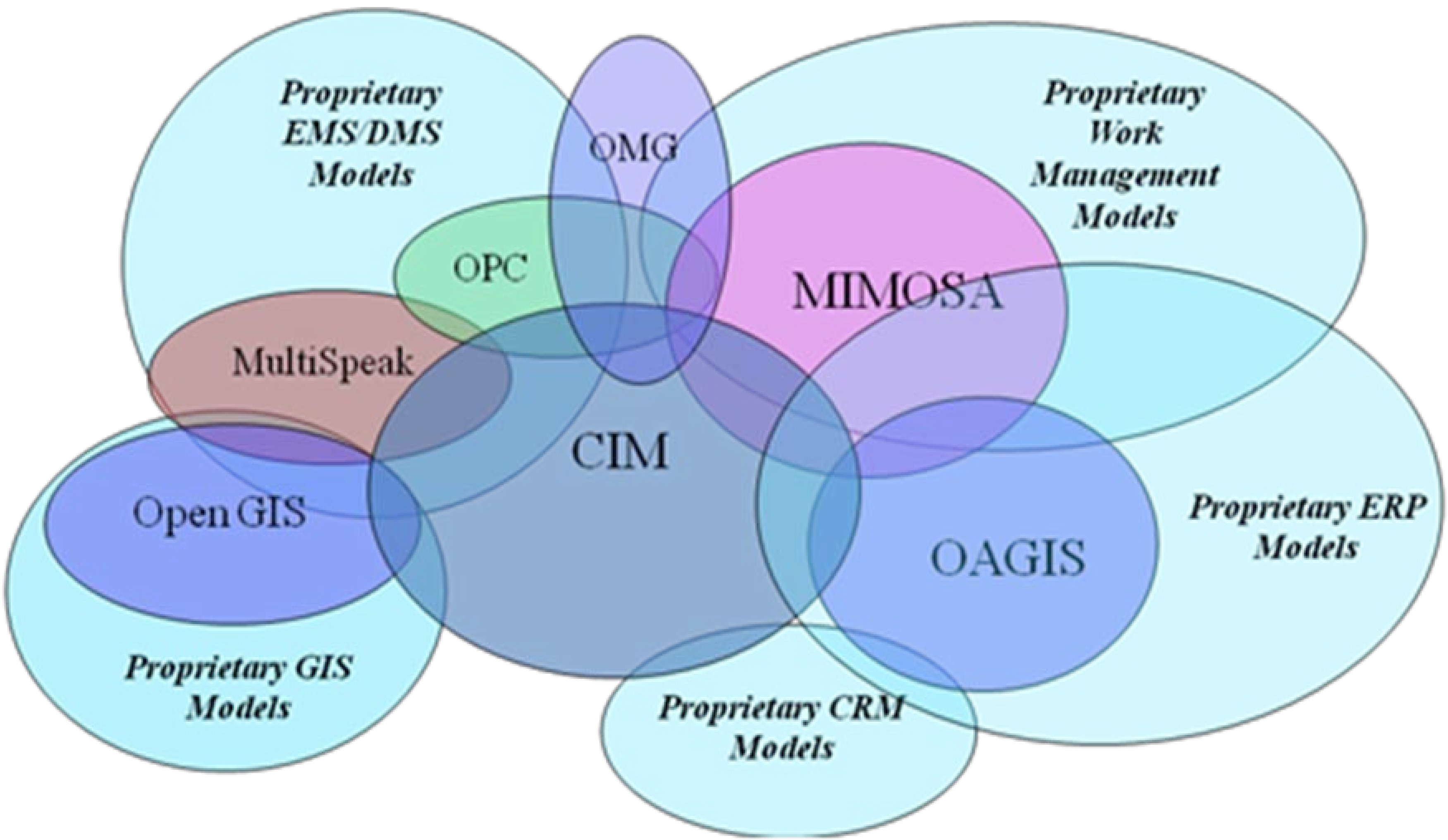
Integration - Standard to Custom



Federation of Ontologies

- The IEC CIM is but one ontology that may be encountered when defining standards for the Smart Grid:
  - A domain may sometimes have overlapping ontologies that are in use
  - Products may also be used within the domain that are also used across other domains

Overlapping Ontology Definitions



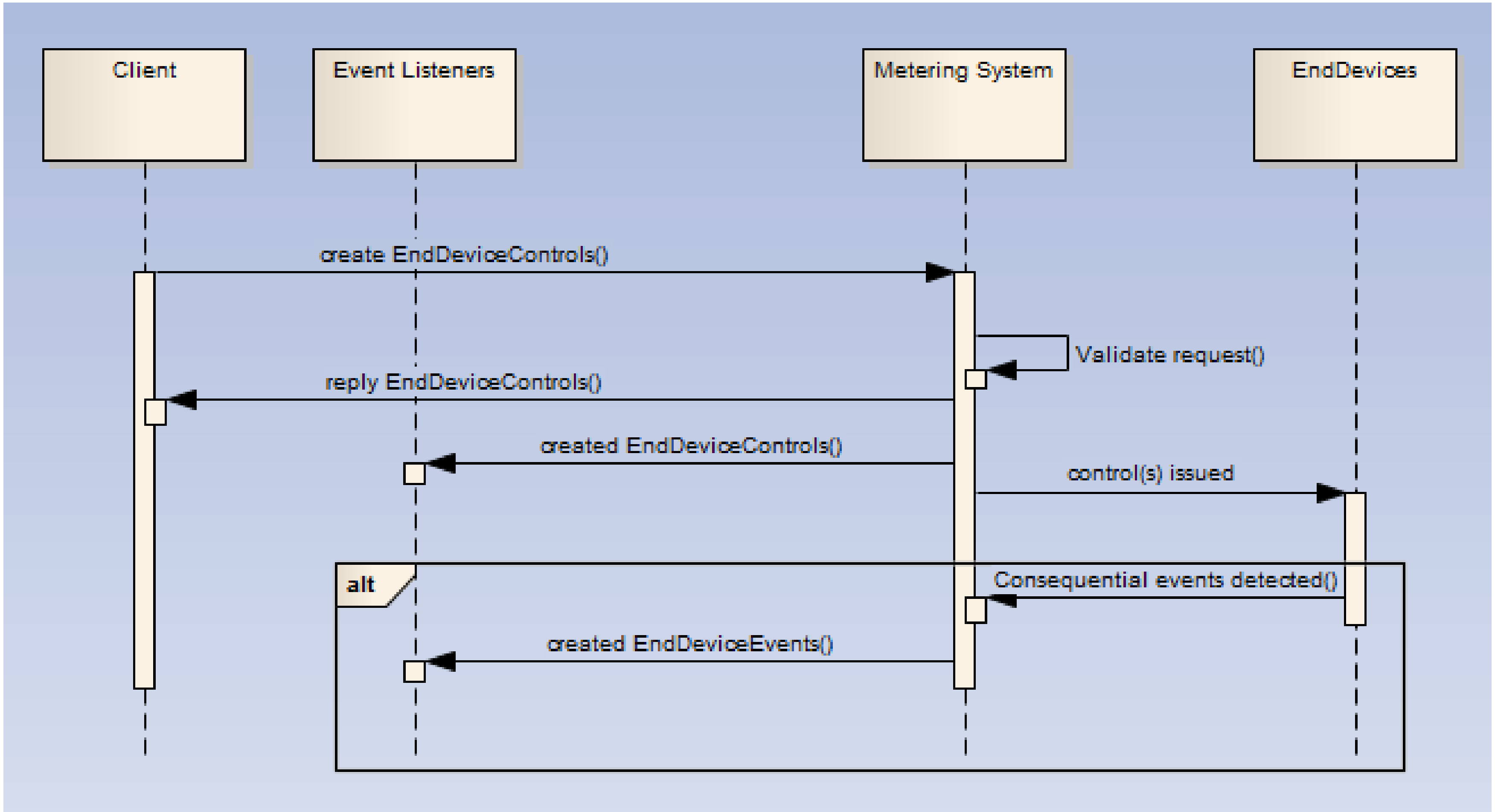
Management of Ontologies

- There are a variety of means that can be used to realize, manage and/or document each of these ontologies. Some examples include:
  - Unified Modeling Language (UML)
  - XML Schema
  - Entity-Relationship Model (ERM)
  - RDF Schema
  - Web Ontology Language (OWL)
  - Domain-specific XML dialect

Why not an ‘uber model’?

- **IF** all of the products in the universe were to adopt a common semantic model
  - There would still be a significant distance to interoperability
- Instead:
  - Focus on mapping where needed
  - Benefit of not overly stifling needed innovation

Example: Integration of Metering Systems

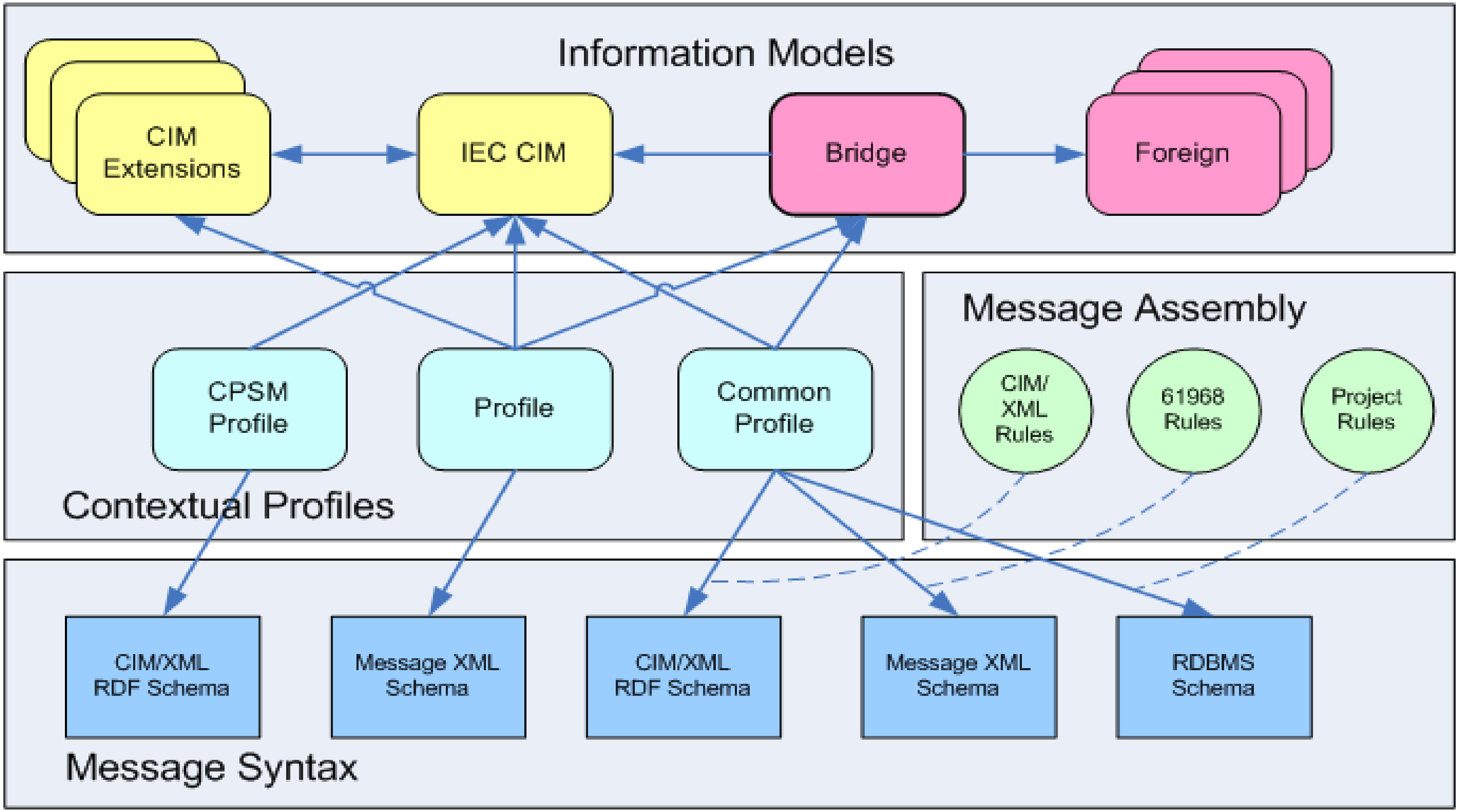


Sequence Diagram Based on IEC 61968-9 – Transactional Message Resulting in Subsequent Event Messages

Contextual Profiles

- Information conveyed through an information exchange is a subset of:
  - Classes
  - Attributes
  - Relationships
- Defined by a domain model
- Within IEC TC57, this is known as a ‘contextual profile’
- Contextual profiles are often standardized themselves

Models, Profiles and Messages



XML: XSD ‘vs.’ RDF

- It is not really XSD vs. RDF
- XML is very flexible, affording room for may ‘dialects’
- RDF used for model exchanges such as CPSM and CDPSM, a.k.a. CIM/XML where data has many interrelationships
- XSD is most commonly used for defining small hierarchical messages for enterprise integration, with IEC 61968-9 being a CIM-based example
- WSDL-based web services essentially require XSDs
- RDF is very flexible and promotes adaptable interfaces
- XML based on an XSD can be almost as flexible, but XSDs are often constrained by the limitations of naming and design rules (e.g. UN/CEFACT XML NDR) and development tools
- IEC 61968-9 may use RDF for model exchanges, such as customer-meter data sets
- *Note that we are avoiding discussions of dynamic XML, DTDs and other XML dialects*

XML Content

- Within an information exchange, there will be many code values that both sides need to understand, often defined as enumerated values:
  - Status codes
  - Control codes
  - Quality codes
  - Event codes
  - Error codes
  - … many others
- IEC standards define use of SI units
- Time values should be represented using ISO 8601
- Object identifiers need to be understood by both sides:
  - mRID
  - Name class defined names

Namespaces

- Used within XML to provide for traceability and versioning
- A change of namespace can break an implementation of an interface … so they should be changed deliberately and with care
- Namespaces are also used to distinguish extended elements from the standard elements
- W3C SAWSDL can be used to trace an XML element back to its source model within an XSD
- Should consider different namespaces for different parts of the CIM

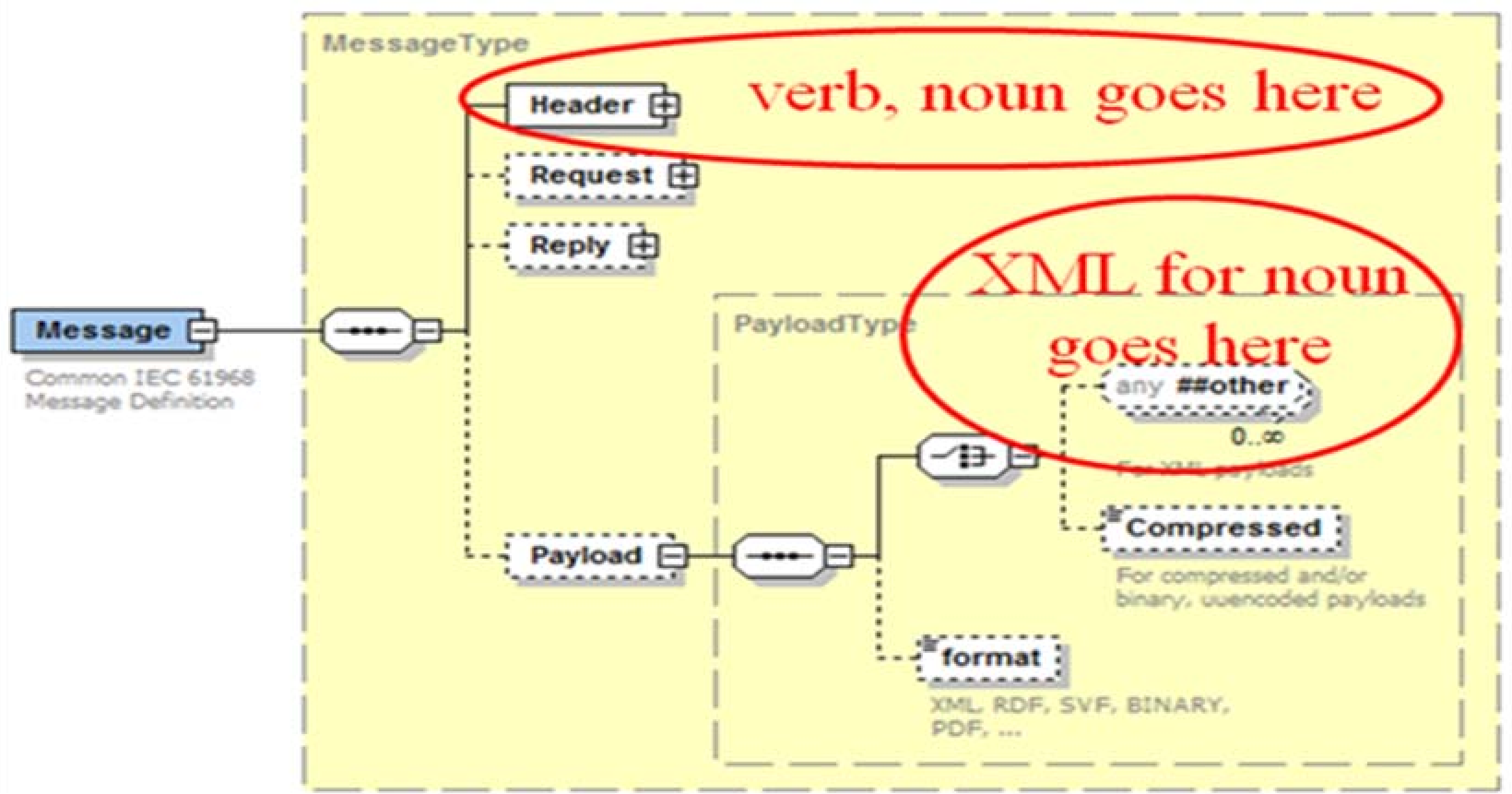
Still Not Interopable!

- Even using the same:
  - Domain Model
  - Approach (XML)
  - Naming Rules (NDL)
- Still possible (and likely) for two independent parties to define messages that are not interopable

Interface Design

- Interfaces need to be designed, not autogenerated
- Need to identify integration patterns:
  - Synchronous Request/Reply
  - Asynchronous Request/Reply
  - Publish/Subscribe
  - Claim Check
  - … many others have been defined
- Variety of implementation options:
  - Language-dependent APIs
  - Web services
  - REST services
  - JMS messages
  - Binary protocols
  - File transfers
- Many other design considerations exist, including security

IEC 61968-1 Implementation Example

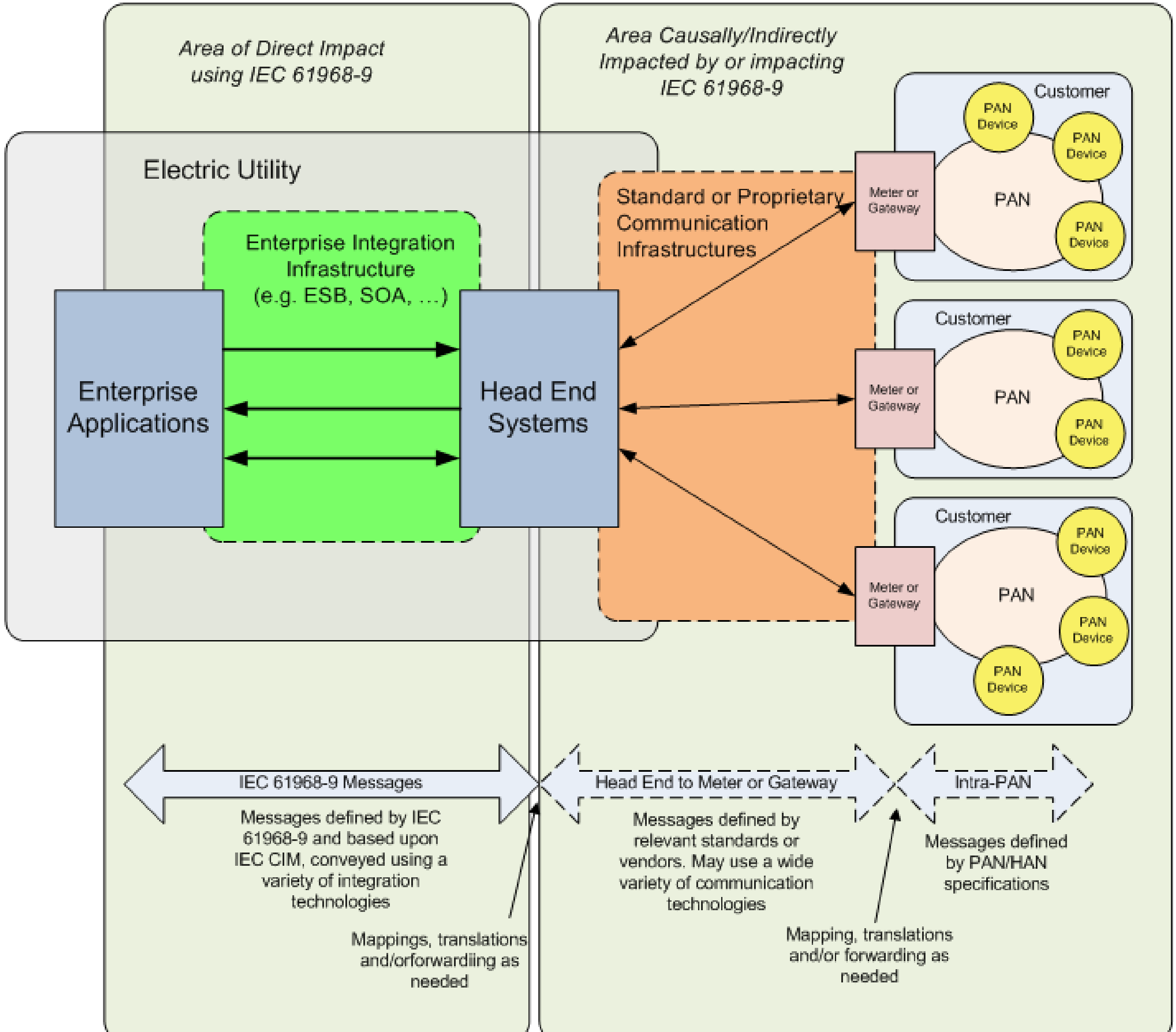


one implementation approach for IEC 61968, where an XML schema-based message envelope is used to convey a verb, noun and a payload

Shallow Integration

- ... is a principle of the Smart Grid, along with loose coupling and layered systems
- Shallow integration dictates that a client using an interface provided by a service need only have minimal knowledge of the internal models and working of that service in order to interact
- As integrations will generally evolve to include a more diverse set of services, the complexities of ‘deep integration’ would exponentially increase the complexities of integration

Example: IEC 61968-9 Integration



Extensibility and Evolution

- The CIM will evolve, mainly through adoption of extensions
- Examples of evolution:
  - Model extensions, including new classes, attributes and relationships to reflect extensions to the standard or extended product capabilities
  - New elements to be included in message definitions
  - New enumerations, to define types, codes, status values, etc.
  - Optional message elements become required
- New integration technologies and standards will emerge

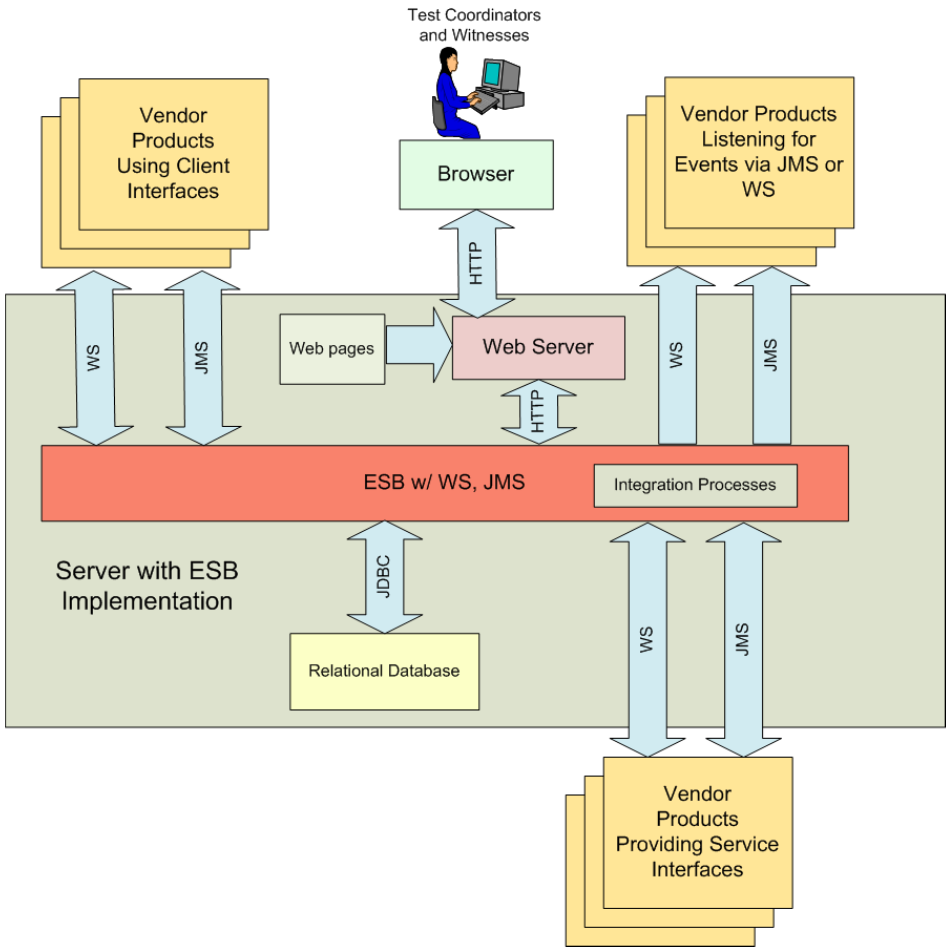
Interoperability Testing

- Often standards are developed without any plan for interoperability testing
- Ideally implementations can be done against a draft of a standard
- Lessons learned, corrections and elimination of ambiguities can be factored into the final draft of a standard

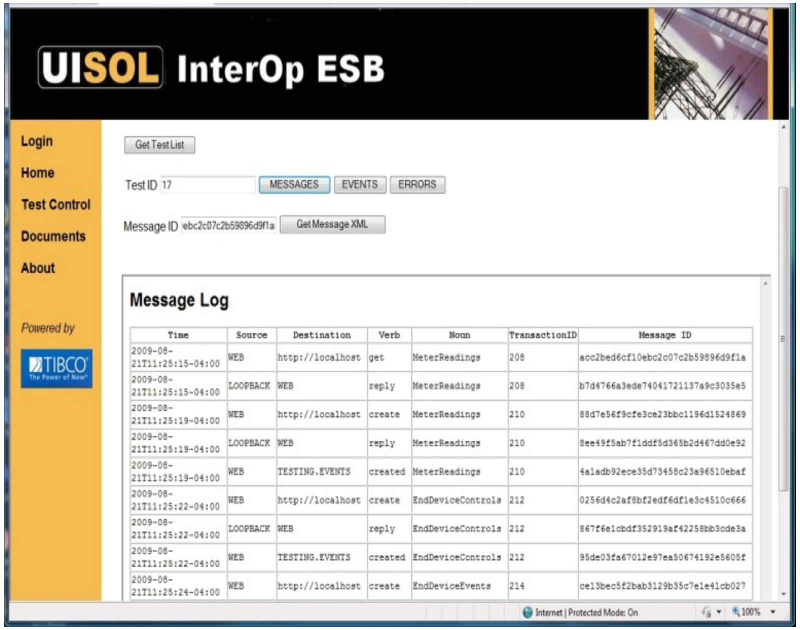
Interoperability Testing

- CIM interoperability tests have been conducted for:
  - Transmission network model exchanges using IEC 61970-452 (CPSM)
  - Distribution model exchanges using IEC 61968-13 (CDPSM)
  - Metering systems using IEC 61968-9
- Historically EPRI has provided some support for these tests
- More and expanded testing is needed, but it is not currently clear how it will happen
- As profiles get more complex, interoperability testing gets more complicated

61968-9 Interopability Testing



Web Interface Based Interop Tests



Virtual Interoperability testing :

- All participants are in remote locations.
- Elimination of travel requirements
-Easier to perform more interoperability testing more often

The web interface used by a test witness -
each message exchanged between two (or more) processes for a given test can be tracked, retrieved and analyzed.

Other Observations …

- The goal of defining standard interfaces is to minimize the needs for custom integration and reduce integration and customization costs
- Well designed interfaces that promote reuse should be strongly favored over methodologies that result in a multiplicity of fine-grained, limited purpose interfaces
- Standards development and resulting interoperability is most successful when product vendors that plan to implement are actively involved
- Look for implementation approaches that are adaptable, rather than rigid and correspondingly brittle

Conclusion

- The notion of ‘CIM compliance’ still leaves a large distance to achieve interoperability
- This paper certainly did not attempt to provide a comprehensive treatment of this topic
- Interoperability testing will be extremely important to insure that standards development efforts achieve the interoperability goals of the ‘Smart Grid’