Pacific Northwest
NATIONAL LABORATORY

PNNL-32679

Enabling Data Exchange and Data Integration with the Common Information Model

An Introduction for Power Systems Engineers and Application Developers

March 2022

Alexander A. Anderson
Eric G. Stephan
Thomas E. McDermott

U.S. DEPARTMENT OF ENERGY

Prepared for the U.S. Department of Energy
under Contract DE-AC05-76RL01830

PACIFIC NORTHWEST NATIONAL LABORATORY
operated by
BATTELLE
for the
UNITED STATES DEPARTMENT OF ENERGY
under Contract DE-AC05-76RL01830

Printed in the United States of America

Available to DOE and DOE contractors from the
Office of Scientific and Technical Information,
P.O. Box 62, Oak Ridge, TN 37831-0062;
ph: (865) 576-8401
fax: (865) 576-5728
email: reports@adonis.osti.gov

Available to the public from the National Technical Information Service
5301 Shawnee Rd., Alexandria, VA 22312
ph: (800) 553-NTIS (6847)
email: orders@ntis.gov <https://www.ntis.gov/about>
Online ordering: http://www.ntis.gov

PNNL-32679

Enabling Data Exchange and Data Integration with the Common Information Model

An Introduction for Power Systems Engineers and Application Developers

March 2022

Alexander A. Anderson
Eric G. Stephan
Thomas E. McDermott

Prepared for
the U.S. Department of Energy
under Contract DE-AC05-76RL01830

Pacific Northwest National Laboratory
Richland, Washington 99354

Executive Summary

The electric grid is being reshaped rapidly by decarbonization efforts, smart grid technology adoption, and projects to improve the reliability, resiliency, and robustness of both the bulk electric power system and distribution networks. Each year brings significant increases in the number of distributed energy resources, electric vehicles, controllable customer assets, power electronics, and intelligent grid-edge devices. Integration of these new devices and associated operational paradigms are driving the need for more streamlined data integration workflows and data exchange between advanced applications and between utilities.

This report introduces key concepts related to the adoption of the Common Information Model (CIM) to enable data integration and data exchange, looking towards the creation of the next generation of advanced power applications that will be needed to ensure reliable, resilient, robust, safe, and economical operations. This new generation of applications will need to operate in a data-rich environment, leveraging multiple sources of data not available to current energy management system (EMS) and advanced distribution management system (ADMS) applications. The CIM is introduced in the context of the need for standardized representation of power system networks, assets, and data exchange messages. After discussing some of the drivers for creating data-rich environments, the report discusses the issues with legacy application integration techniques (which typically build custom data adapters between each application and each proprietary data format used within a utility). After discussing some of the historic challenges to data integration (notably the lack of consistent naming of equipment types, properties, and individual assets), five use cases for CIM modeling with increasing levels of data integration are presented. These use cases range from manual power system model export/import using CIM/XML files to complete enterprise-level data integration where all planning, operations, and business applications are integrated across a single enterprise message bus and use a consistent set of object, attributes, and unique identifiers across the utility.

The CIM is an information model for describing power system networks, asset information, operational procedures, and market data in a consistent manner. More specifically, the CIM provides an ontology (i.e. a consensus-based vocabulary) for describing the semantics of what information is meaningful for which types of power system equipment. CIM does not directly specify how those objects should be used in a database or data structure. Rather, it simply defines the vocabulary of how these objects should be described and what is the relationship between them. The full CIM provides an enormous amount of semantic breadth and depth for power system objects, likely more than what is required by any single application or platform. As a result, it is useful to have a procedure to down-select to only the minimum set of object and attributes needed (CIM profile). This process is detailed in Section 5 along with procedures needed to export that profile to an empty structure (data profile) and populate the power system network data into the data profile.

The CIM is freely available to use and extend. The CIM is maintained by the UCAiug (informally known as the CIM User's Group) under an Apache 2.0 license. The CIM Users Group collaborates with the IEC and other standards communities for the development of technical and informative specifications. Although portions of the information model are referred to by the corresponding IEC standards naming, it is not necessary to purchase any of the IEC standards to use the CIM information model.

Acronyms and Abbreviations

ADMS	Advanced Distribution Management System
AGC	Automatic Generation Control
AMI	Advanced Metering Infrastructure
AOR	Area of Responsibility
API	Application Programming Interface
BTM	Behind the Meter
CGOC	Compliance, Governance, and Oversight Council
CIM	Common Information Model
DDL	Data Definition Language
DER	Distributed Energy Resource
DR	Demand Response
DSO	Distribution System Operator
EPRI	Electric Power Research Institute
ESB	Enterprise Service Bus
EMS	Energy Management System
GIS	Geographic Information System
ICCP	Inter-control Center Communication Protocol
IEC	International Electrotechnical Commission
IoT	Internet of Things
IT	Information Technology
JSON	JavaScript Object Notation
mRID	master Resource Identifier
NDR	Naming Design Rule
NIEM	National Information Exchange
OMS	Outage Management System
OT	Operational Technology
PMU	Phasor Measurement Units
PNNL	Pacific Northwest National Laboratory
RDF	Resource Description Framework
SCADA	System Control and Data Acquisition
TSO	Transmission System Operator
UML	Unified Modeling Language
UUID	Universally Unique Identifier
XSD	eXtensible Schema Definition
XML	eXtensible Markup Language

Contents

1.0 Introduction

The Common Information Model (CIM) is an open-source information model that is used to model an electrical network and the various equipment used on the network. CIM is widely used for data exchange of bulk transmission power systems and is finding increasing use for distribution systems. Use of a non-proprietary information model (such as CIM) that has been agreed upon and adopted by numerous utilities, vendors, and researchers allows significant reduction in the effort and cost of data integration. Likewise, adoption of open data platforms built around the CIM increases available functionalities for managing and optimizing the smart grid of the future.

This report is intended as an introduction to CIM for utility engineers, power systems researchers, and application developers, providing a broad view of the CIM and how particular profiles can be adapted for various use cases. Unlike most other CIM introduction documents and the International Electrotechnical Commission (IEC) standards (which are mostly targeted to an audience of data scientists, enterprise database managers, and platform developers), this report is intended for users of traditional power systems analysis software and other readers without any prior experience with canonical information models, data profiles, or UML modeling. An example use case is used throughout this report to support a practical understanding of the CIM. This use case maps distribution-side distributed energy resources (DERs) to feeder breakers and individual transmission / sub-transmission substations for DER coordination and management.

This report is intended to be read as the first of three documents that cover the why, what, and how of using the CIM for model exchange, data exchange, and data integration, as illustrated in Figure 1. This introductory report outlines *why* an agreed-upon information model is needed. The reader will be introduced to key concepts around data integration, model exchange, and the difference between information models, data profiles, and network model data. The second document [1]

Figure 1: Recommended guides to read as first introduction. This document explains *why* an information model is needed. The second document [1] explains *what* is specified by CIM to represent power system networks and assets. The third document [2] explains *how* to build a custom profile and use some of the available modeling tools.

outlines the *what*, specifically what set classes and attributes are used by the CIM to specify common power system components, such as lines, transformers, and switches. It also provides a summary of available tools and a first introduction to reading class diagrams and schemas used by CIM to specify how power system components should be modeled. The third document is the EPRI Common Information Model Primer [2], which explains *how* to create a CIM profile and provides a much more detailed and technical description of schemas, data structures, and implementation techniques.

This remainder of this report is organized as follows: Section 2 introduces several issues related to data integration and common challenges in power system interoperability. It also introduces the concept of an information model as well as other fundamental concepts, such as syntactical vs semantical understandings of messages and data. Section 3 introduces the Common Information Model and what is and is not defined by the CIM. It also outlines the ongoing efforts by the UCAiug (informally known as the CIM Users Group) to evolve and expand the CIM as the power grid continues to evolve. Section 4 introduces five different use cases for adoption of the CIM with increasing depth of interoperability, ranging from simple model exchange to enterprise-level data integration. Section 5 presents the set of steps that would be taken by a project to select a CIM data profile and implement for that data integration effort.

The companion to this report, *A Power Application Developer's Guide to the Common Information Model* [1], introduces the key technical concepts needed for power system engineers and power application developers to start working with the CIM. It systematically describes the set of classes, attributes, and associations used by CIM to model lines, transformers, switches, generators, loads, and distributed energy resources. It also covers many of the technical concepts that are required to understand, including how to read UML diagrams, how to create a CIM profile, and how to create a data profile.

2.0 Enabling the Next Generation Electric Grid through Data Integration

The electric grid is being reshaped rapidly by decarbonization efforts, smart grid technology adoption, and projects to improve the reliability, resiliency, and robustness of both the bulk electric power system and distribution networks. Each year brings significant increases in the number of distributed energy resources, electric vehicles, controllable customer assets, power electronics, and intelligent grid-edge devices. At the feeder level, customer rooftop solar, distributed generation, bi-directional power flows, and shifting load profiles are causing significant impacts to distribution operations. Simultaneously, many utilities anticipate that the displacement of conventional thermal power plants by intermittent renewables (at both the bulk transmission and distribution level) will disrupt current operational paradigms based on load-following automatic generation control (AGC) and traditional approaches to maintain operating reserves, frequency balancing, voltage control, interchange scheduling, and black-start capability [3]. Furthermore, increasing penetration levels of large-scale renewables, customer rooftop solar, electric vehicles, battery storage, and other distributed energy resources will soon blur traditional silos of operations between generation, transmission, and distribution [4]. Furthermore, the increasing frequency and severity of weather events, coupled with the growing threat of cyber-physical attacks, are putting additional stress on the ability of utilities to operate the grid in a safe, secure, and economical manner. As a result, a new generation of advanced power applications will be needed to ensure reliable, resilient, robust, safe, and economical operations. This new generation of applications will need to operate in a data-rich environment, leveraging multiple sources of data not available to current energy management system (EMS) and advanced distribution management system (ADMS) applications.

2.1 Data-Rich Environments

Forthcoming data-rich environments will need to support a combination of existing data streams (such as traditional system control and data acquisition (SCADA) data from substation remote terminal units and poletop equipment), existing data that is currently not aggregated in real time (such as advanced metering infrastructure (AMI) smart meter readings), and emerging data feeds from internet-of-things (IoT) devices, smart inverters, and distribution-side micro-phasor measurement units (PMUs). However, integration and aggregation of these disparate data sources into a data-rich environment is not trivial.

Development of data-rich environments has long been recognized as a difficult task with many challenges [5], [6]. One of the largest is creation of a common vocabulary to describe the same piece of equipment across different software systems and applications. Consider the simple case of an overhead line. Among common power systems tools, it will be referred to as “Line” in a GE eTerra EMS/ADMS, “AC Line” in Siemens PSSE, “Line Section” in a Survalent ADMS, and “Branch Device Type: Line” in PowerWorld. Likewise, among different various tools, the impedance may be specified on a per unit basis, in ohms, or on a per-length basis. To aggregate data from disparate sources, the same vocabulary must be used consistently to describe a particular type of object. This shared vocabulary is then recognized by all applications and users working within the new data-rich environment.

Development of such an agreed-upon vocabulary requires a large user community to come together and develop as set of “living” documents that define the terminology and attributes of various things that are important to the community’s focus area. Users and software developers within the community collectively adopt and use the same consensus-based vocabulary within their own software platforms and data environments. To reflect evolving needs and new data types, these documents are then maintained and updated by the community on an ongoing basis.

One of the most successful examples of such a community and shared model is the National Information Exchange Model (NIEM), which was originally developed for the Department of Homeland Security and Department of Justice for information exchange across national, state, and local governments [7]. NIEM was considered critical to overcome gaps between information silos in government identified by the 9/11 Commission Report [8]. Over the last two decades, the NIEM information model has been expanded to cover 15 domains, including agriculture, immigration, infrastructure protection, international trade, maritime shipping, biometrics, cyber security, emergency management and surface transportation. To ensure naming consistency for data structures derived from NIEM, NIEM like many community standard models employs Naming Design Rules (NDR). The NDRs are syntax specifications created to ensure the transformation from information model to XML schemas are consistent, thus insuring data interoperability.

Within the electric power industry, the Common Information Model has served a similar role for enabling power system model exchange between utilities and vendors, data integration from multiple sources, and creation of data-rich application environments. Appendix B of the EPRI CIM Primer [2] lists some notable success stories from multiple utilities (such as Southern California Edison, Sempra Energy, and Idaho Power Company) using the CIM to solve complex integration problems involving asset management, operations visibility, customer programs. It has also been used as the foundational information model for multiple data-rich environments, such as the Oracle Utility Data Model and the Pacific Northwest National Laboratory GridAPPS-D platform [9], [10]. Like the NIEM, the CIM offers Naming Design Rules for its information model to data format transformations in a variety of formats including eXtensible Schema Definition (XSD) [11] and JavaScript Object Notation (JSON) schema [12].

2.2 Data Management and Interoperability in Electric Utilities

In a typical electric utility, there are hundreds and even in some cases thousands of software solutions and applications that are managed by the information technology (IT) department. These applications are used and operated independently by the various groups, departments, and organizations within the utility. Whenever a business process requires data from one system or application to be transferred to another system or application, the data needs to be manually extracted from the first database and then converted to the format of the other application's database.

Three strategies (shown in Figure 2 below) exist for dealing with the extreme level of effort needed to manage, update, export, convert, and import data formats between different applications and databases:

1) Reduce the number of databases by purchasing a large software suite from a single vendor using a single proprietary data format that is internally-integrated and compatible with all the applications needed by utility

2) Build custom application adapters between each set of software that need to exchange data

3) Adopt a common data integration platform that allows external integration between multiple software packages using a shared data format

The first approach is quite effective within a single organization (especially large vertically-integrated utilities) to exchange real-time data and models, even across the transmission-distribution boundary. This approach also requires little IT management or data integration effort on the part of the utility as all applications work seamlessly with that vendor's application programming interface (API) and database structures. The vendor is also generally responsible for all maintenance of the applications, databases, and interfaces.

The second approach is acceptable when integrating a handful of applications from different vendors or a new custom application with an existing software suite. This approach builds custom adapters between individual applications. Each adapter is responsible for converting the data types, message structures, and API calls of one application into a format understandable to the other application. This type of application integration is extremely time-consuming and requires a deep understanding of all the data models used by both applications. It is also very expensive to maintain as any changes to either application can break the custom adapter. Furthermore, the difficulty of integration increases nonlinearly with the number of applications that need to exchange data.

The third approach reduces the number of directions in which data needs to be exchanged by introducing a shared platform and set of shared services that are utilized by all the integrated applications. In this approach, interfaces do not need to be created between each induvial application and every other application. Instead, each application only needs a single interface with the standards-based platform. Additionally, each application only needs to understand set of object attributes laid out in the consensus-based vocabulary used by the selected information model.

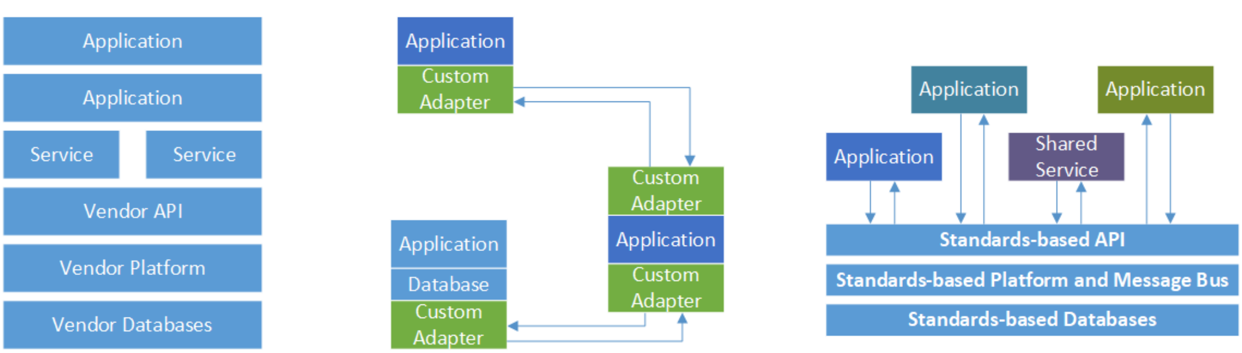

Figure 2: Conceptual representation of three data management and interoperability strategies. At left is a single vendor system that is fully integrated with that vendor's API. Center depicts the set of custom adapters that are typically required to exchange data between applications from different vendors. At right is a standards-based platform with a unified API and set of shared databases and services that reference an information model (such as CIM) to describe all objects and attributes in consistent terms.

While the first two approaches are acceptable within a single utility or collection of utilities, the complexity of building custom adapters increases nonlinearly with the number of applications that need to exchange information. This problem quickly becomes unmanageable when multiple utilities need to exchange information, models, and measurements from system control and data acquisition (SCADA) systems.

One of the first large-scale efforts to create interoperability between power applications suites was the creation of inter-control center communication protocol (ICCP) [13]. ICCP was developed in an effort led by the Electric Power Research Institute (EPRI) to exchange real-time SCADA data among multiple transmission and distribution control centers running energy management systems (EMS) and distribution management systems (DMS) from multiple vendors. The ability to exchange real-time data using ICCP was first demonstrated in 1995 with a systematic interoperability test between EMS software platforms from four different vendors [14]. ICCP became widely used to share SCADA data between various regional transmission system operators (TSO). Some distribution system operators (DSO) even use ICCP to pass SCADA data to their outage management systems (OMS). However, most switching orders and other control commands between control centers are still exchanged verbally between human operators over a telephone call due to the need for data tracking and lack of protocol support for equipment tags and work clearances.

Similarly, EPRI led the development of an open standard for representing power system components, which became the Common Information Model and is now maintained by the UCAiug and published as a set of standards by the International Electrotechnical Commission (IEC) Technical Committee 57.

2.3 The Naming Problem in Power System Interoperability

One of the largest issues in creating interoperability between different advanced power applications is the lack of standardization of naming of equipment and their attributes. Consider the example of mapping a set of distribution-side distributed energy resources (DERs) to feeder breakers and individual transmission / sub-transmission substations.

Information detailing the various physical assets and power system network models will be located across multiple databases from multiple software systems. Data regarding the capacity, location, and owner of DER will typically reside in the system and interconnection planning applications and/or asset management database, while metering and generation output data will be contained in the customer billing applications [15]. At the same time, electrical connectivity of the DER through a step-up transformer to rest of the distribution feeder will be contained within the DMS and geographic information system (GIS) databases, with the data formats (for example shape files) used by ArcGIS and GE SmallWorld the most common in the electric power industry [4].

Conversely, information about transmission network and substations will be contained within the set of bus-branch and node-breaker models used by EMS databases and analysis tools, with the most common model formats those used by Siemens PSSE, GE Grid PSLF, GE Grid e-terra EMS, and ABB Spider EMS. It should be noted that most of these data formats do not contain the set of objects and attributes needed to describe distribution-side resources.

To accomplish the task of mapping DERs to substations without a standard representation of power system components, a series of data tables would need to be created for each network model from each utility. Each application will likely use the same “human-readable” name for a particular piece of equipment, but the exact equipment name, description, and set of properties modeled will vary by application. To create a mapping between the equipment descriptions obtained from two different applications, three databases are now required (one for each application and one for mapping). This type of custom mapping between datasets still remains a very common approach, but it is extremely time-consuming to build and costly to maintain.

2.4 Introduction to Data Integration

Within an electric utility planning and operations context, there are two aspects to data integration. The first aspect is assembling and aggregating data from disparate sources into a common point of access. Historically, one of the most common approaches was by feeding all the data streams into a data historian, such as an AVEVA / OSIsoft PI Historian. The PI Server contains numerous adapters that allow it to ingest and interpret data using various communication protocols, such as DNP3, Modbus, MQTT, etc. [16]. Users and applications are then able to access all the relevant data from the data historian through a variety of means including a streaming data service, web API, a server client, or a full software development kit (SDK).

The second aspect of data integration relates correlating data by creating proper relationships between devices, measurements, and control settings. A simple example of this would be establishing a relationship between two different measurements of different data types taken from adjacent poletop equipment. Good data integration will identify that these two measurements have locational proximity and should be correlated.

Within the context of creating a data-rich environment for power systems operations, the primary focus of data integration efforts is to ensure that all available and relevant data is made available to drive advanced power applications and help power system operators make more informed decisions. The particular approach used should be selected based on the functional objectives of the applications and services that will use the integrated data. An example of such an analysis is provided by [17], which lays out the data integration strategy for the GridAPPS-D reference implementation, in which multiple data streams and measurement types are aggregated and published onto a single message bus to which advanced applications can subscribe and receive all real-time SCADA data, AMI measurements, and forecasts from a single source [9].

If multiple applications are to be integrated without development of numerous custom adapters and model converters, a key prerequisite is an agreed-upon information model that provides a consensus-based vocabulary for describing all aspects of the power system network, physical assets, and measurement data in consistent manner. Without such an information model, it is extremely difficult to aggregate and correlate data, and significant amounts of effort will be required to manually format, validate, and correlate asset data, measurements, and network model information.

2.5 The Role of Information Models

Generally, an *information model* is a representation of the concepts, relationships, constraints, rules, and operations to specify data semantics. *Semantics* refer to whether a particular message or data file is understandable and interpretable, as opposed to *syntax*, which refers to whether the message was structured correctly. The sentence "My transformer has a blue shirt" is perfectly correct, syntactically. However, it makes no sense at all semantically because substation assets do not wear clothes. The Common Information Model presents a semantic view of the power system in that it defines a consensus-based vocabulary for what attributes makes sense (or not) for various types of devices and assets. In other words, the CIM sets out that the sentence "My transformer has a high-side winding with a rated voltage of 115 kV" makes sense semantically.

The CIM is also a *canonical* information model in that it is a superset of multiple information models and is intended to reduce the costs associated with data integration by providing standardized data definitions for integration enterprise software systems. The CIM contains a far larger set of classes and attributes than could possibly be needed by a single application or project. It is anticipated that any adopters of the CIM will select only a small subset of classes covering the relevant aspects of the power system model. This subset is known as a *CIM Profile*.

One of the most confusing aspects of the CIM to power systems engineers and power application developers is that CIM is not a data structure or the power system network model. Rather, the CIM itself is an abstract information model that only provides a consensus-based vocabulary and set of attributes for describing power system networks, assets, and data exchanges.

Such an information model is needed not only to help solve the naming issue described earlier, but also to provide context about the meaning and units of a particular numerical value. In any operational technology (OT) environment, and particularly in power systems operations, discrete facts and recognizable data are needed to inform real-time decisions and control actions. Both human operators and advanced power applications cannot spend extensive amounts of time or resources trying to decipher the meaning of a set of data or the units in which it was given.

Without a semantic or syntactic context, the string "7647-14-5" could be interpreted as circuit breaker number 5 in bay 14 of substation 7647, the date "May 14, 7647", or the Chemical Abstracts Service (CAS) number for sodium chloride. Likewise, SCADA measurements and other observational data have similar potential ambiguities; for example, an overhead line property with an analog value of "0.1019" does not provide sufficient context about the physical property it represents. Even if it is deduced that it is a line impedance, it still cannot be properly used and understood without the semantic units of measure context "0.1019 ohms" as opposed to "0.1019 per unit". Even with the units specified, that value does not provide enough context to be useful to the operational technology (OT) system – the physical quantity associated with that value must also be specified, such as "the individual phase reactance is 0.1019 ohms". That value then must be associated with correct overhead line segment and the correct phase conductor of that line. The association between an AC line segment, a phase conductor, and the reactance of that conductor need to be explicitly specified in the information model for users and applications to interpret the data correctly. Lastly, network parameters and associated values need to be managed in a consistent way. For example, the information model needs to differentiate the relationships between various objects (such as AC lines, buses, and breakers) from the relationships between those objects and the values related to individual pieces of equipment (such as the AC line's MW value or the asset information associated with the conductor's diameter and geometry).

Usage of an agreed-upon information model, such as the CIM, provides the consensus-based vocabulary and relationships needed to define a power system model in manner such that any CIM-compliant application, service, or software platform can interpret the model without any custom adapters or data translation. The CIM provides explicit definitions of the most common equipment types, model parameters, and control settings. It also defines the relationships between different objects and equipment types in unambiguous terms.

From this information model, it is then possible to create a custom data profile, which provides a structure with an empty set of fields for specifying power system data in the selected format (XML, JSON, etc.). This empty data profile can then be populated with the actual power system network model, equipment command messages, or whatever information is needed for the data integration process. In turn, this CIM-compliant data file can then be imported into the data-rich environment for usage by all applications, services, and user-interfaces within that environment.

3.0 The Common Information Model

The Common Information Model is a canonical information model that provides a formal representation of objects, their attributes, the relationships between them, and the operations that can be performed on them. It is *not* a database structure or physical data store. It is a technology-agnostic model for describing the properties of physical power system equipment, power flow data, and messages that can be exchanged between various platforms and applications.

3.1 What CIM Provides

The CIM acts as an *ontology* of the electric utility business domain in that it presents the set of concepts within a domain, identifies the relationships between those concepts, and defines a formal naming scheme for discussing those concepts using a consensus-based vocabulary. It serves a semantic reference model describing the majority of utility business objects, power system network model components, physical asset data, operations procedures, and transactive market mechanisms.

Using CIM to exchange power system models and operational messages largely solves the data mapping problem described in Section 2.3. Regardless of the original data format and property naming used by the source application, the naming of object types and attributes in an exported CIM model will always remain the same. Any overhead line will always be modeled as an `ACLineSegment` object, with a set of standard attributes from `ACLineSegment` that specify the line impedance, line geometry, phasing, conductor length, etc. Subsequently, any application compliant with the CIM profile used will be able to parse the new network model data without any modification to the application or any need for custom mapping between data formats. The generic, consistent representation of power system network components provided by CIM greatly simplifies the effort required to exchange models and data between applications. Rather than building custom mapping tables and data converters between every proprietary data format, each application only needs to create a single converter that can import and export CIM XML data.

Unlike many power system data formats that use comma-delineated or space-delineated text files (e.g. Siemens `*.raw` or GE `*.epc` files), power system model files specified using CIM typically use a data profile based on eXtensible Markup Language (XML) to specify the network as a set of unique objects with various attributes. Thus, instead of specifying a particular overhead line from node 1 to node 2 with certain geometry and spacing as a set of columns in a single text row, CIM will define several objects, including

- `ACLineSegment` (the particular line),
- `ACLineSegmentPhase` (the particular phase conductor),
- `OverheadWireInfo` (the conductor radius and material),
- `WireSpacingInfo` (the conductor count and spacing),
- `ConnectivityNode` (the from/to nodes),
- `Terminal` (the end terminals on either side of the line).

Each of these objects have a standard set of attributes, such as `WireInfo.ratedCurrent`, `WireSpacingInfo.phaseWireSpacing`, and `ACDCTerminal.sequenceNumber`. As a result, the

CIM XML file for a power system model tends to be much larger in size than for the same model using a space-delineated format.

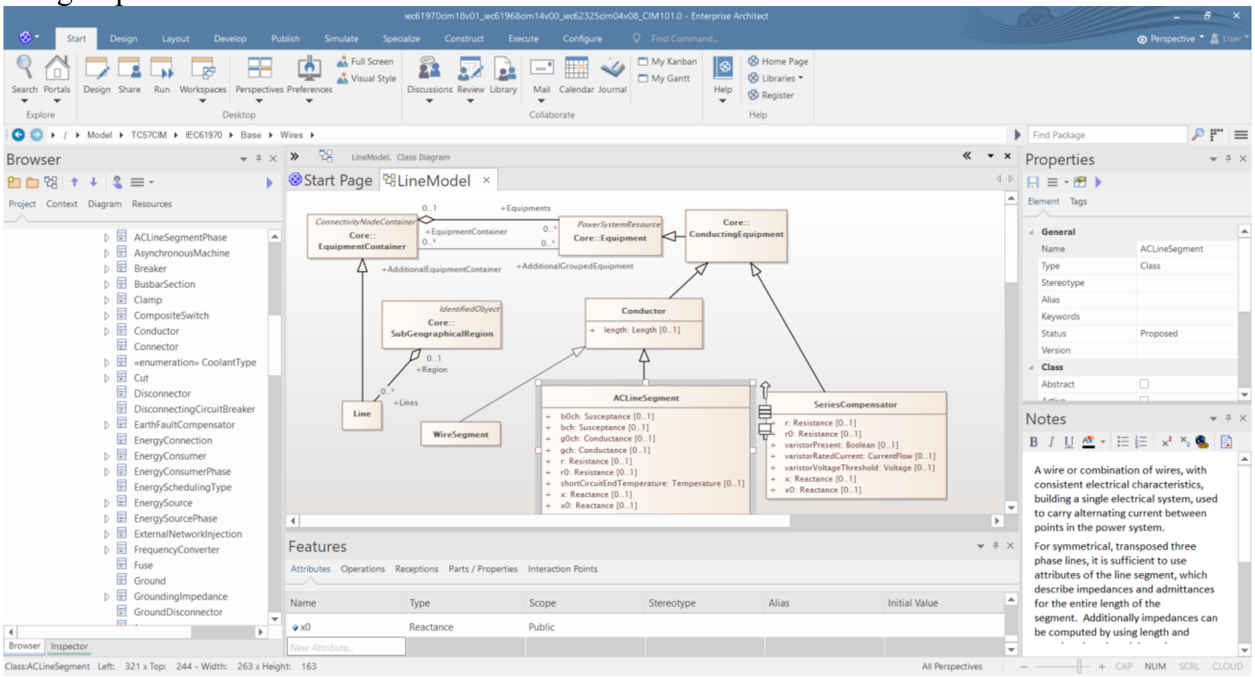

Figure 3: A UML class diagram for `ACLineSegment` and related classes viewed in Enterprise Architect. Each class is described in detail, with the ability to view technical descriptions of each attribute and the set of links to other classes throughout the entire information model.

As an information model, CIM does not directly specify how those objects should be used in a database or data structure. Rather, it simply defines the vocabulary of how these objects should be described and what is the relationship between them. The CIM extensively uses Unified Modeling Language (UML) class diagrams to describe each class of objects, their attributes, and their relationships with other classes. An example of a UML class diagram for classes and attributes related to `ACLineSegment` as viewed in Enterprise Architect (EA) is shown in Figure 3 above. Each class and each attribute contain detailed descriptions of its usage, enumerations, and relationships with other classes. This report will not discuss how to read UML diagrams or discuss any of the classes and attributes used for power system modeling. These aspects of information modeling and the CIM are explored in detail in the companion to this report [1]

3.2 Resolution of the Naming Problem using CIM

The Common Information Model resolves the naming problem of Section 2.3 through the use of the master resource identifier (mRID). The mRID does not have to be human-readable and is generally not intended to be displayed to end-users of advanced power applications. This 3-18 character identifier is intended to be used by software systems for consistent identification of network data and asset information. Global uniqueness of mRIDs can be easily achieved by using a Universally Unique Identifier (UUID), also known as a Globally Unique Identifier (GUID), for the mRID of every object in the power system network model. In CIM-based applications, the mRID is used as the unique pointer or dictionary key that allows all the properties of a particular piece of equipment to be called from memory or queried from a database. For user interface support and other display uses, the CIM uses the attributes `name`, `description`, `aliasName`, and `pathName` for providing identifiers that are human-readable.

It is common for names of objects within a utility to be non-unique due to historical naming conventions, the results of mergers and acquisitions, and the inability of other software systems to manage uniqueness. For these reasons, there are no constraints on these names requiring them to be unique. The uniqueness of the equipment assets and properties is instead managed through the equipment mRID.

For example, it would not matter if a switch was named “NORTHSUB-115KV-BAY01-031” in one application but “NSB115031” in another application since both would reference the switch using its unique machine-readable mRID `"_68B7C1EF-AC11-48E2-8D85-43BB0EF41B58"`. Note again that the mRID is not intended to be displayed to the user and is only used internally by the application code and databases.

When re-exporting models, a master list of mRIDs is typically kept serving as a mapping between the equipment naming within a particular application and the unique mRID that is referenced and used by all applications working from the same CIM XML power system model. It is important that this master list of mRIDs is maintained in the same way that the power system network model is maintained such that a particular device is always referred to by the same mRID even as the power system network model is edited and updated over time.

3.3 CIM Standards and Working Groups

The Common Information Model is freely available to use and extend. The CIM is maintained by the UCAiug (informally known as the CIM User’s Group) under an Apache 2.0 license. The CIM Users Group collaborates with the IEC and other standards communities for the development of technical and informative specifications. Although portions of the information model are referred to by the corresponding IEC standards naming, it is not necessary to purchase any of the IEC standards to use the CIM information model.

The CIM is divided into three separate information models that cover different aspects, centered around a single unified model for the entire power system network. Each portion of the information model is aligned with the series of technical standards published by IEC:

- The IEC 61970 model focuses on sharing of the power system network model between systems and applications for both planning and real-time operations, as well exchanging information between different participants in grid operations, such as the transmission system operator (TSO), distribution system operator (DSO), microgrid controller, individual generators, and consumers. This standard specifies the schema for representing the power system network model, including generation, loads, topology, overhead lines, underground cables, transformers, voltage control equipment, etc.

- The IEC 61968 model focuses on business functions, such as asset management, work management, system planning, and customer metering/billing. This standard specifies the schema for representing equipment health, switching operations, dispatch of DERs, loads that can be controlled for demand response, and electric operating data other than the network model.

- The IEC 62325 model focuses on electricity market processes, such as transmission capacity allocation, load forecasting, generation bidding, and clearing of market auctions.

The CIM is maintained and expanded by several working groups under UCAiug and is shared with IEC Technical Committee 57 working groups that develop the IEC 61970, 61968, and 62325 specifications, each responsible for a different focus area:

- WG13 maintains the core CIM model and specifications for power system network data, primarily from the perspective of the transmission system operator and applications for real-time operations and offline analysis of power system model.

- WG14 focuses on extending the CIM data schemas to cover distribution systems and enterprise integration. This working group has expanded the CIM information model from its original EMS focus to include unbalanced medium-voltage distribution networks, low-voltage customer-side equipment, DERs, and advanced metering infrastructure (AMI). It also created numerous schemas to support the range of planned and unplanned work activities common in distribution operations, including network extension planning, construction and maintenance, outage response, switching clearance, real-time network changes (e.g. cuts and jumpers), customer billing, asset management, and operations planning.

- WG16 focuses on deregulated energy market communications and extensions to the CIM to represent European style and US style markets. These extensions do not model the structure of a particular market, but rather the types of data that need to be exchanged between various participants to enable market clearing and provide the necessary input data for security constrained unit commitment and security constrained economic dispatch applications.

Additionally, within the CIM User's Group, there exists the CIM Interop Working Group, which evaluates the interoperability of various third-party vendor software tools based on CIM and verifies compliance with the set of standard classes and attributes specified by CIM. A set of test models are used to demonstrate the exchange of power system data between different software and demonstrate data integration through an actual field implementation project using the IEC 61970 standards.

Versions of the CIM are managed under configuration control through the UCAiug. All changes are documented and managed by the model managers designated by each working group. The CIM information model has evolved since its initial release, just as the power grid has evolved. Major releases can reflect breaking changes, such as deprecation of obsolete classes or the introduction of new concepts. Minor releases may reflect minor spelling errors or slight corrections to the model that occurred during publication. The version of a particular CIM release is given in the format iec61970cim17v38, which specifies that the particular information model contains the IEC 61970 portion (power system network classes), on the 17th major release, and 38th minor release.

CIM provides open-source comparison tools so that end users can understand the impact to their profiles by a new release. It is important to note that when new releases occur, adoption of the new release is up to the end user. Some utilities and vendors may choose to adopt a new release because it contains extensions for new equipment types (such as DERs and distribution assets). Others may choose to use a single CIM release for many years (for example, ERCOT is still using CIM 10 for its network models to avoid any changes). However, it is recommended that custom applications comply with original CIM standards to ensure they are interoperable and able to interpret data from other CIM-compliant apps.

Additionally, if the CIM is selected as the desired information model within a utility, vendor, or project, it is not necessary to use the CIM standard exactly as it is given from the UCAiug UML files or IEC standards. Users can modify the profile, add custom attributes, or create entirely new extensions (such as representation of house appliances). For example, the GE eTerra EMS/DMS Model Export tool uses a custom CIMDNOM profile derived from the base CIM 15 information model [18]. Likewise, the PNNL GridAPPS-D platform uses a set of custom extensions for residential houses and IEEE 1547-compliant DERs derived from the base CIM 17 information model [9].

Finally, it is important to note that the CIM information model is separate from the IEC standards, although they cross-reference each other. The current, past, and draft versions of the CIM information model are available freely through the UCAiug website [19] as set of Enterprise Architect `.eap` files and CIMTool `.XMI` files. It is not necessary to purchase any of the IEC standards documents to use the CIM information model, develop a custom profile, or derive a data profile for containing power system network model data.

4.0 Common Information Model Use Cases

This section introduces several common uses for the Common Information Model, with increasing levels of data integration and interoperability between applications, databases, and interfaces. The interoperability discussion will reference the GridWise Interoperability Framework layers [20], which is reproduced in Figure 4 below.

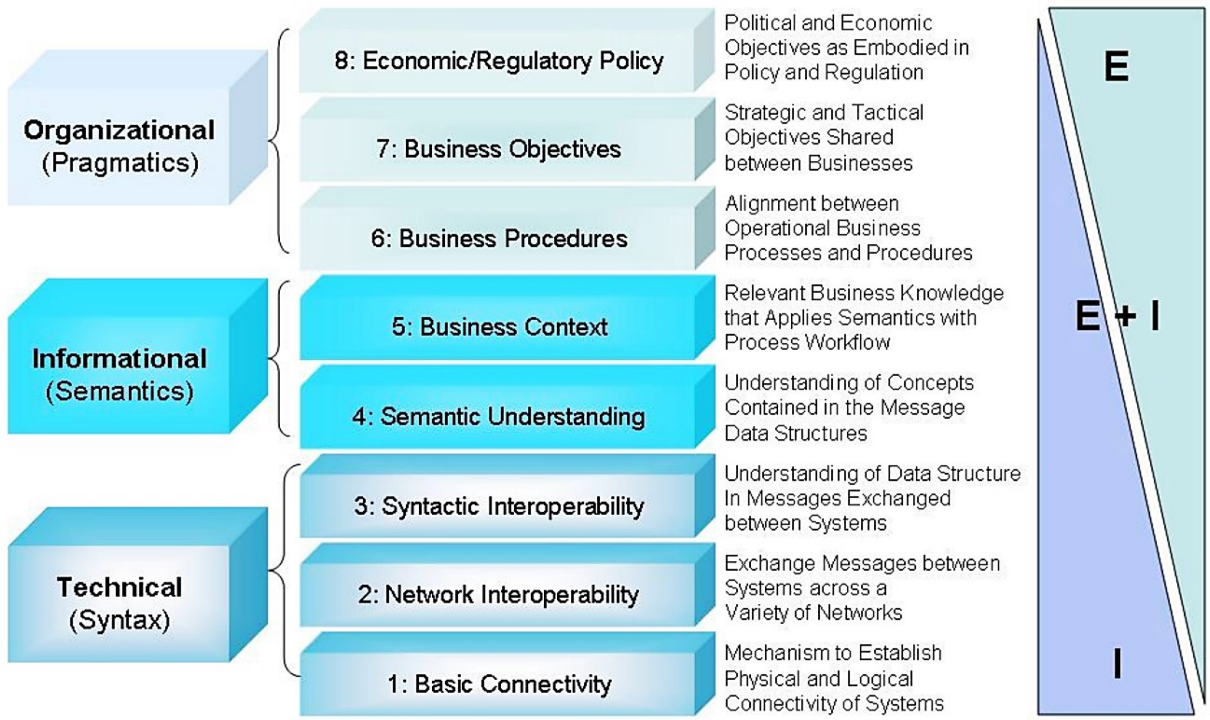

Figure 4: GridWise Interoperability Framework Interoperability Layers, taken from [20].

The lower three layers in Figure 4 (basic connectivity, network interoperability, and syntactic interoperability) refer to methods by which information is exchanged and deal with physical communications network (e.g. ethernet), communications protocols (e.g. TCP/IP), and syntax of data structures (e.g. HTML and XML). The two middle layers (semantic understanding and business context) are of the most interest for this discussion and refer to the semantics of how to interpret what is contained within a message. The CIM plays a key role at this level by providing a consensus-based approach to understanding what is contained within a particular model file or message. The three highest layers (business procedures, business objectives, and economic/regulatory policy) refer to organizational procedures, objectives, and policies. While the CIM is not directly referenced at these layers, it still can play a key role by enabling data exchange between the power applications and enterprise software suites.

4.1 Exchange of Power System Network Models

The most basic use case for the CIM is for exchange of power system network model data between utilities and vendors without the need to develop custom importers for proprietary vendor data formats. In this use case, the IEC61970 and IEC61968 portions of CIM are used just as an information model for specifying the power system model in a format that is understandable to the application suites of both vendors. The model export-import process is typically performed manually by the utility and/or vendor application engineers on an as-needed basis, depending on the frequency of model updates. There is little to no information exchange between application suites, and each suite uses its own set of internal databases with local copies of the power system network model in their own proprietary data formats. Figure 5 below illustrates the typical workflow of model exchange between independent software suites within a single utility. Note that this use case typically requires direct involvement of human application engineers to perform manual file export and import of the power system model files. Additionally, this file export/import process needs to be performed every time there are significant changes to the power system model. For these reasons, model exchange in this fashion is arguably not a true example of data integration or application interoperability, but it is currently one of the most common uses of CIM.

Likewise, power system models can be exchanged using CIM between utilities using different vendor model formats for a range of studies, planning, and operations planning activities that require detailed models of the electrical networks of both utilities. In this case, both utilities would use the CIM export feature of their software packages to export their own models. These models could then be shared with other neighboring utilities or imported into a different set of application tools and combined into a unified network model.

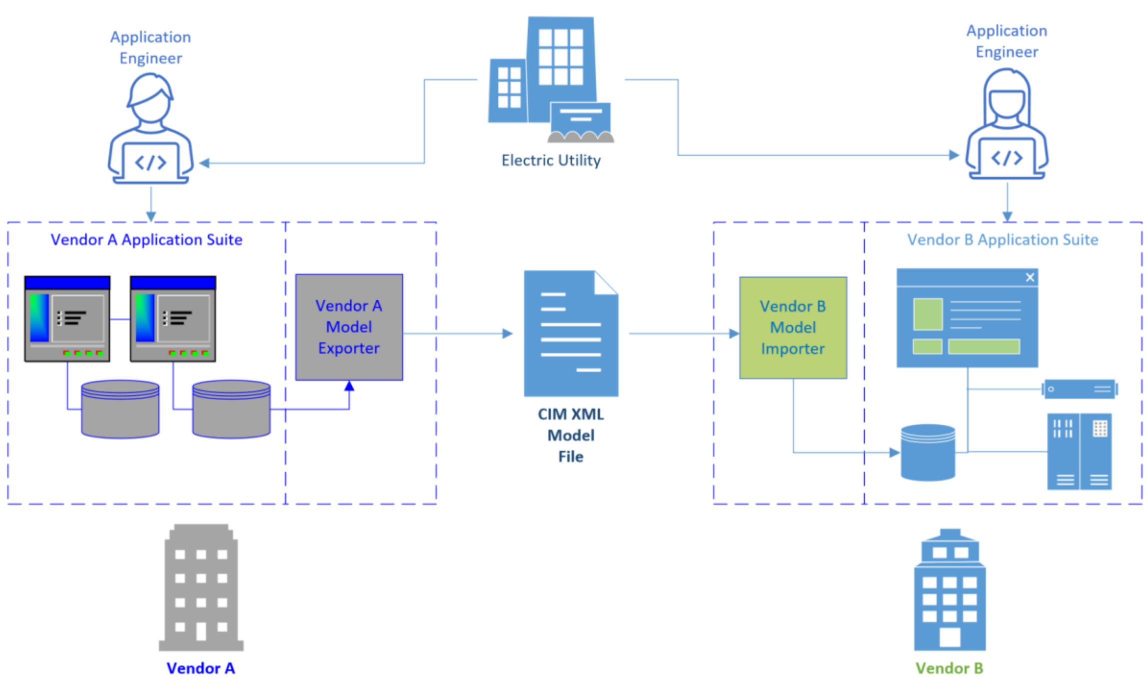

Figure 5: Exchange of Power System Network Models using CIM

4.2 Exchange of Operational Data and Messages

A related use case is exchange of messages and operational data between power applications and equipment from different software ecosystems. Common uses include energy markets, dispatch of DERs, demand response, and customer participation programs. IEC 61968-Part 100 defines a set of standardized message formats referencing the classes and attributes defined in the CIM information model. These CIM-compliant messages can be used in a variety of contexts. Some of these messaging use cases include:

- *Simple request/reply between a client and a server* – A client makes a request to a server using the commands get/create/change/delete etc. The server then responds with a set of objects reflecting the instructions contained in the client query.

- *Request/reply using an intermediate Enterprise Service Bus (ESB)* – An intermediate service bus can be inserted so that the client can send all requests to the service bus, and the service bus will then pass those requests to the correct server.

- *Publish/subscribe using an ESB* – The service bus can also be used in a publish/subscribe context, which is extremely useful for aggregating real-time data streams and making them available in real time so that applications can subscribe to a single message bus. This context can also be used for publishing customer incentive signals, smart inverter control settings, or other control commands to a large number of subscribing equipment assets.

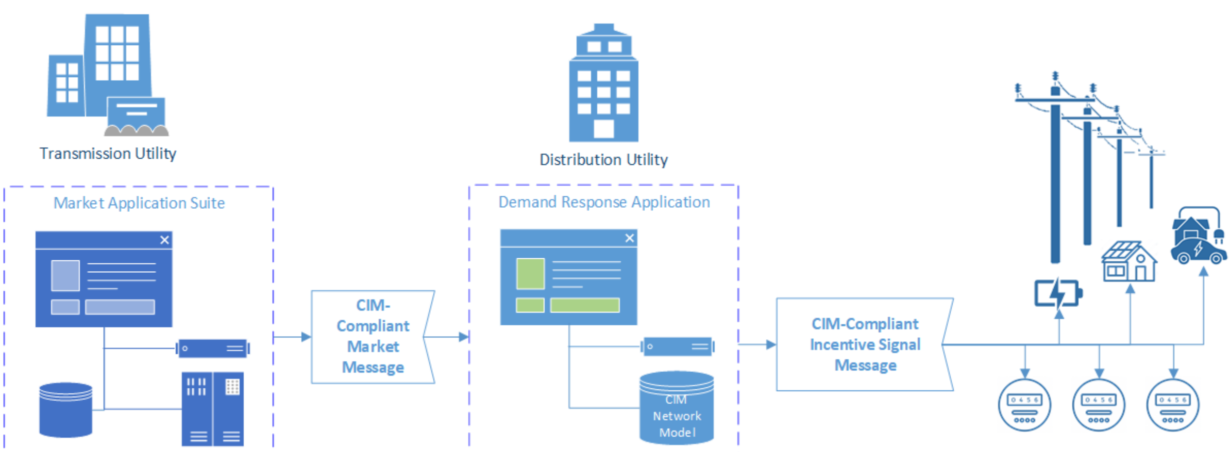

Figure 6: Conceptual representation of exchange of CIM-compliant incentive signal messages

Figure 6 above depicts a conceptual representation of how CIM-compliant messages can be used for incentive signals in a deregulated market context. In this case, one or message buses (not shown for simplicity) are used as a bridge between applications using the publish/subscribe paradigm just described. The transmission utility then publishes a CIM-compliant message requesting a load reduction. The distribution utility's demand response application is subscribed to the message bus and receives the market message. Receipt of that message then triggers the demand response application to run its own internal set of algorithms. The application then publishes a set of incentive signal messages over the AMI network. The controller object classes would use the CIM object classes and attributes, but residential controllers only need to understand the objects specific to demand response, and not any of the larger classes related to load models and market operations. If desired, some of the IEEE 1547-2018 and house load model extensions to CIM could be adopted as well to provide control of smart inverters and other distributed resources.

4.3 Project-based Data Integration

The next common use for the CIM is to provide the information model for project-level data integration. This approach to CIM adoption is often driven by a particular business case that involves a complex data exchange problem between multiple existing applications and a new custom application. In these situations, it is desirable to standardize the way that data is formatted, messages are exchanged, and control commands are issued. However, for these single projects, there does not yet exist a sufficiently strong business case to move the entire utility to CIM-based enterprise-level data integration.

Without a shared information model for specifying objects and their attributes, the burden of building custom application-to-application bridges and model converters becomes unacceptable. As the number of involved applications and data streams to be integrated increases, usage of CIM for the new application becomes more compelling. With implementation of CIM, only a single data format needs to be used for ingesting the power system model and exchanging messages between applications. Using for CIM-based information exchange and application integration becomes even more compelling when dealing with power system network models and operational data exchange between multiple utilities. Each utility can export the network model for their own area of responsibility (AOR) into a CIM model file, which is then loaded into a single application database containing all the network models of interest. The only manual data mapping issue will be related to ensuring that the same set of mRIDs are used consistently to refer to equipment and physical assets that appear in multiple models (e.g. substation transformers serving multiple circuits or located in substations where the high-side and low-side buses lie in the AOR of different utilities).

The example data mapping problem of tracing DERs to substation breakers (first discussed in the introduction of this paper) could likely fall in this category. On the power system modeling side, it will likely be prohibitive to build custom data converters from each of the databases and application suites containing information about substation layouts, feeder network topology, asset descriptions, and customer DER locations. However, if the various aspects of the power system network from each data source are exported into a consistent data format (XML, JSON, etc.) using the same CIM profile with a common set of device master resource identifiers (mRIDs), the data can be aggregated and synthesized relatively easily.

Likewise, at this project level of CIM-based data integration, operational messages exchanged between applications can be formatted to be CIM-compliant messages between the various applications, as shown in Figure 7 above. For example, a load shed directive issued by the independent system operator (ISO) could be formatted as CIM-compliant message that is published on a shared message bus. The DER coordination application would subscribe to these messages and identify the affected areas. It then would pass a set of CIM-compliant queries to its

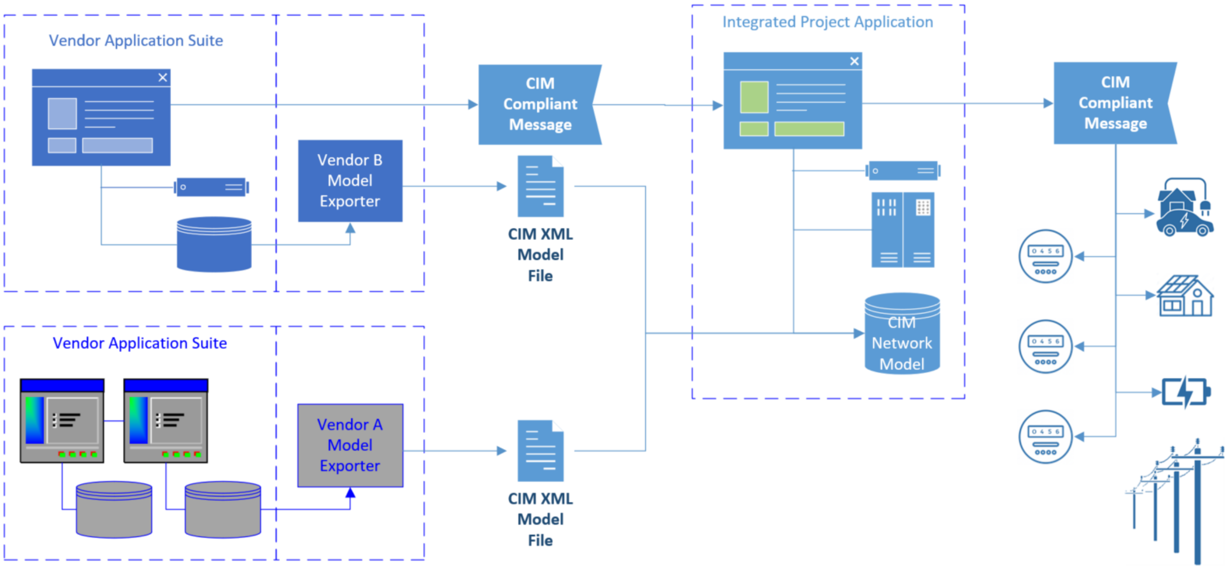

Figure 7: Conceptual representation of a single custom application using CIM to receive power system model data and operational messages from existing applications and model databases.

network model database(s) containing a set of all relevant network models using IEC 61970 objects to describe the electrical network and IEC 61968 objects to describe the physical assets. The database(s) would respond with a set of CIM-compliant messages containing the power system network model and topology relating which DERs were currently configured to which substations. The DER coordination application would then generate a list of suitable switching actions for load shedding and publish that recommendation as a CIM-compliant message that would be displayed by a visualization application that mapped the equipment mRIDs to human-readable names.

4.4 Development of Standards-Based Platforms

The next level of interoperability and data integration combines multiple CIM-compliant applications and application services through a common CIM-based message bus and CIM power system network models. This type of platform serves as the basis for creating a data-rich environment aggregating and correlating many types of data and sensor feeds from disparate sources, as discussed in Section 2.1.

Such a CIM-based platform enables multiple applications to work within a single data-rich environment, working from a single set of databases that all use the CIM to represent the power system model with consistent use of the same classes, attributes, and mRIDs. All applications compliant with the CIM standard will be able to interpret any network model in the platform with no code customization and will also be able to use consistent mRIDs to reference equipment across multiple databases, data streams, and even "study-mode" simulations run with offline simulators.

A conceptual representation of a standards-based platform is depicted in Figure 8. Data from multiple disparate sources (such as SCADA measurements, weather info, and smart meter data) are aggregated and published to a CIM-based message bus. These data streams are mapped to the

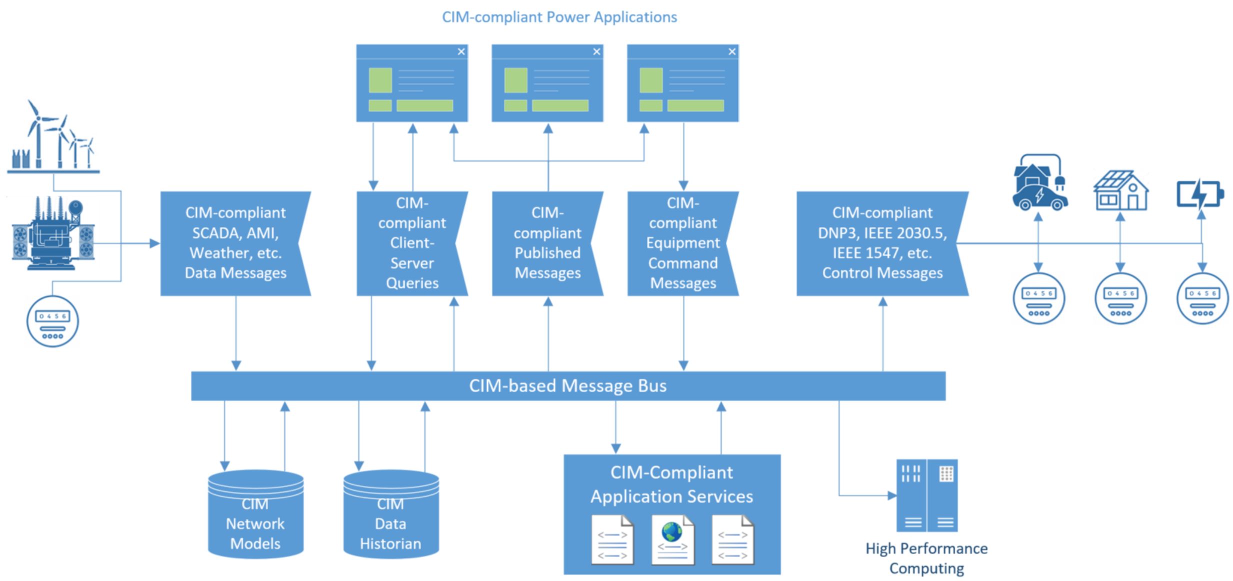

Figure 8: Conceptual model of a CIM-based platform for creating a data-rich application environment.

terminals of associated power system equipment through a set of unique measurement mRIDs defined for each measurement ingested by the platform. The ingested measurements, power system network model, and associations are contained in a set of databases structured around the unique mRIDs and directional relationships between the set of CIM classes and attributes used. CIM-compliant advanced power applications can then pass request/reply queries to the CIM-based data stores and subscribe to real-time streaming data through the message bus using CIM-compliant messages, as introduced in Section 4.2. Power applications within the platform can publish CIM-compliant equipment control commands through the message bus. These messages can then be converted to the format of any specific communications protocol (such as DNP3, IEC61850, IEEE 1547, etc.) and then sent to physical equipment controllers.

Usage of the message bus also reduces the number of interfaces and code duplication between applications and services. A single application service, such as topology processor, can run in real-time and publish the current topology of the power system to the message bus. Then, all power applications that require the current topology can simply subscribe to the service output through the CIM-based message bus, rather than run their own internal topology processing script. Coordination between applications is also simplified because all equipment control commands are published through the shared message bus, and any application can subscribe and recognize if a different application is already issuing control commands to a particular asset through its unique mRID.

An example of such a standards-based platform and data-rich environment is the application development ecosystem developed by the GridAPPS-D program [9] [10]. Multiple CIM-compliant applications can subscribe to aggregated data streams, query static and timeseries databases, publish equipment control commands, leverage code block and application service repositories all using a single CIM profile. The GridAPPS-D platform architecture is shown in Figure 9.

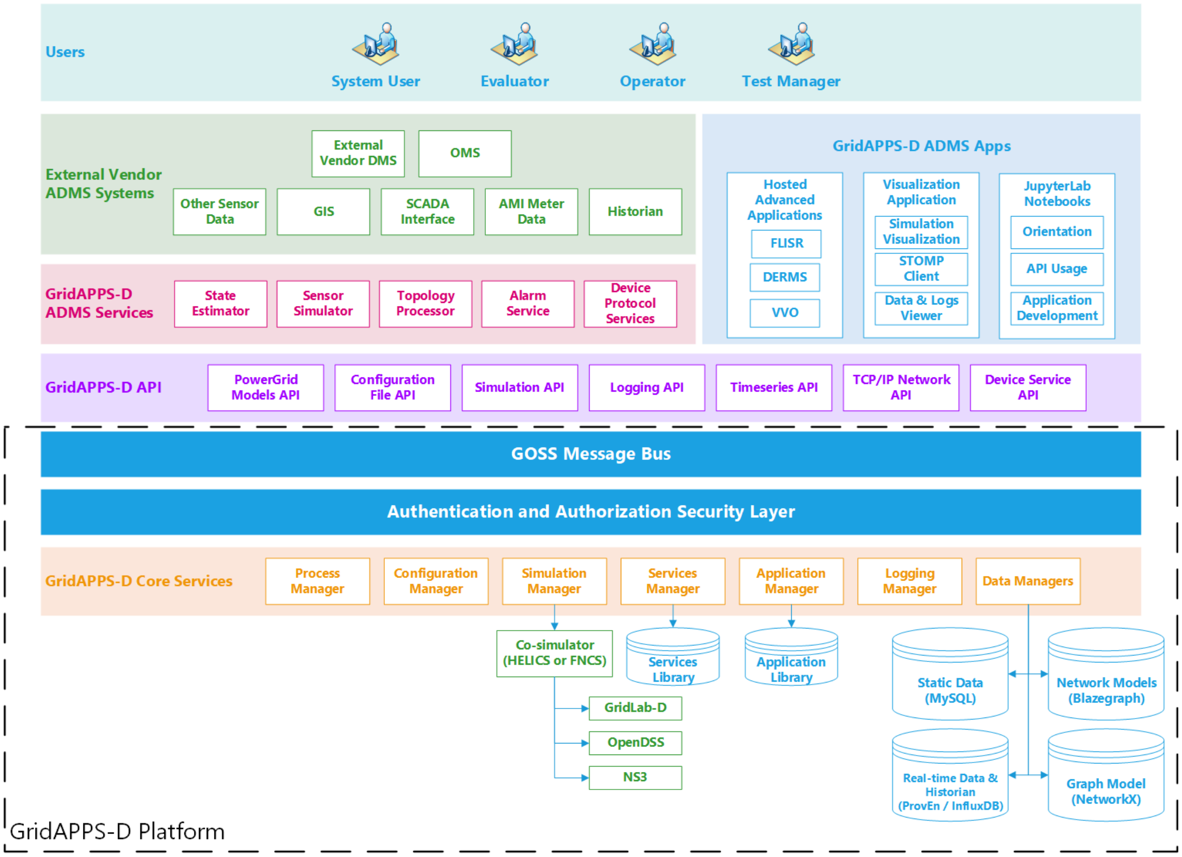

Figure 9: Architecture of the GridAPPS-D platform, a representative example of a CIM-based data-rich environment for application development, testing, and deployment.

4.5 Enterprise-level Data Integration

The highest level of data integration using the Common Information Model is at the enterprise level. Unlike the previous examples, the decision to adopt CIM is not at the project or platform level but is organization-wide. Also, the approach to data integration is not bottom-up (as in aggregating data streams into a single source), but top-down (i.e., all organizational entities work from a single set of models, databases, and middleware). This level of integration is not a trivial effort but can provide significant benefits for large utilities and organizational entities with large amounts of disparate data from siloed business systems.

Canonical information models, such as CIM (for power systems) and NIEM (for national security), address the domain space from an enterprise perspective. The Common Information Model considers not only the technical aspects of power system modeling, but also the business aspects of operations, asset management, customer billing, and energy markets.

The IEC 61968-1 standard [21] defines a set of standard interfaces between the different business functions of a utility that are integrated across a single set of CIM-compliant middleware services, as shown in Figure 10. The IEC 61968-1 standard does not attempt to specify which applications a utility needs to purchase or that a vendor needs to produce. Rather, it provides a reference model for how the CIM can serve as the basis for creating a unified software solution for the entire utility business entity, independent of any vendor-specific systems. Within this model, a common set of

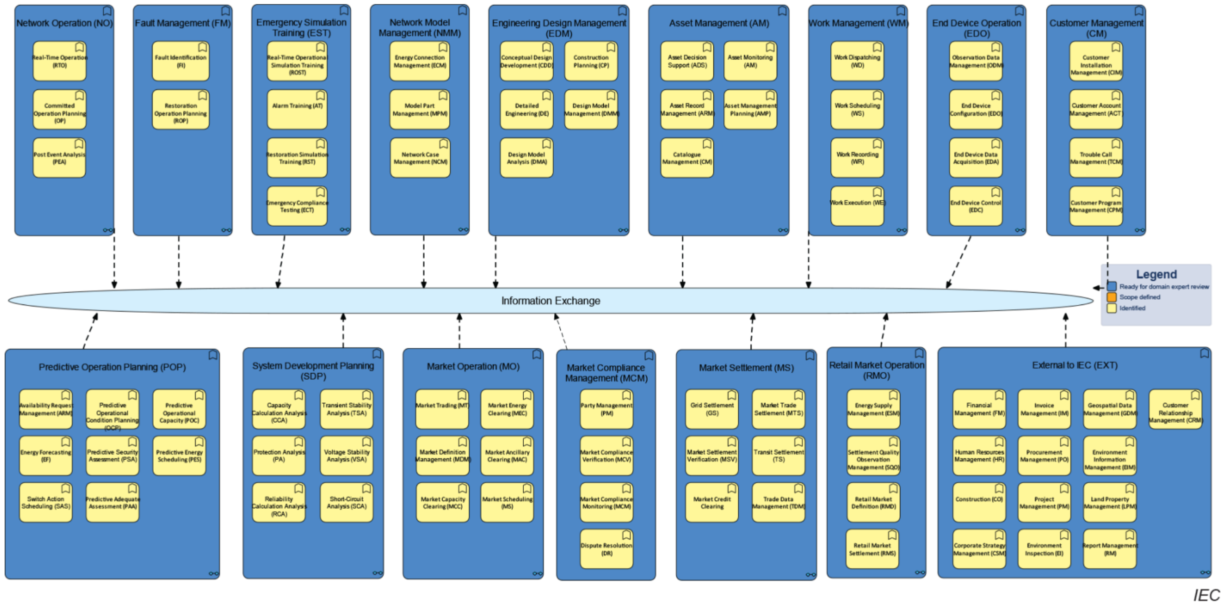

Figure 10: IEC 61968-1 Interface Reference Model (reproduced from [21]) illustrating the interfaces between business and management functions in a utility using enterprise-level CIM integration.

application services and network models are used throughout the entire utility. For example, a given transformer will be referred to by a consistent mRID throughout the set of network operations, asset management, and system planning applications. Likewise, a fundamental application service, such as a power flow solver, will only need to be implemented once and exchange its solutions across the CIM-compliant middleware service (rather than each individual application containing its own power flow solver).

Within this enterprise-level data integration reference model, the IEC 61968-1 standard also identifies a set of layers within a particular interface profile, shown in Figure 11. At the highest level are abstract components, which represent CIM-compliant advanced power applications and

Figure 11: Interface layers defined in the IEC 61968-1 Standard [21]

other business function applications. At the next level is component adapter, which serves as a translator to enable non-compliant applications compatible with the CIM. This adapter also serves as a link between the vendor-specific API used by the application to the generic interface specification of the CIM. The IEC 61968 interface specification at the third level contains all the attributes needed for all data exchanges through the particular interface. The interface specification separates the logical interface and its implementation. It also is agnostic of any middleware or programming language. The fourth level is the middleware adapter between a single application and the set of middleware services common to a certain vendor software suite. The role of the middleware adapter is to provide a gateway for IEC 61968 message exchanges, especially for services that are not CIM-compliant. In theory, middleware adapters should be reusable for multiple interfaces to the same set of middleware services running in the same computing environment. At the fifth level are the set of middleware services that enable information exchange across multiple applications (both locally and remotely) across one or more message buses. Both advanced power applications and business function tools communicate through the middleware services through a combination of request/response, publish/subscribe, and other messaging approaches. Delivery of these messages is enabled by the sixth layer, communication services, which should be able to deliver messages regardless of the underlying platform, programming language, or technologies used. Finally, the platform environment is the set of underlying hardware and software that hosts the enterprise-wide set of applications and services used by all the organizational functions within an electric utility.

4.6 Increasing Data Maturity

A discussion of data integration use cases would not be complete without mention of the importance of data maturity. Data maturity models provide a way of measuring characteristics of data products used within an enterprise. Data maturity models can assist organizations in developing roadmaps to evolve proprietary and siloed data to reusable products that are interoperable throughout the enterprise.

A recurring issue with custom power applications and data integration projects is the difficulty of software maintenance and upgrades without involvement of the original software development team. Direct application-to-application adapters (created in the manner described in Section 2.2) often rely on custom data-mappings and data structures to convert between the model and message formats of individual vendor applications. These types of adapters are very susceptible to breaking changes from software upgrades and rely on the availability of an application engineer to make manual software patches needed to support evolving business functions.

Adopting an agreed-upon information model, such as CIM, can help utilities, vendors, and application developers take a major step forward in increasing the consistency of data representations and increase the data maturity of their software systems. Adoption of the CIM not only helps eliminate proprietary processes that depend on a single application engineer or vendor representative, but also provide helps provide a data maturity target to aim for. Use of CIM consistently across an organization further helps identify applications and services using ad-hoc or inconsistent data structures.

There are many data maturity models in existence, with one of the most popular ones developed by the Compliance, Governance, and Oversight Council (CGOC) [22], shown in Figure 12 below. The CGOC data maturity model defines four tiers associated with increasing maturity, decreasing risk, and increasing transparency.

At the lowest tier of data maturity, data is a standalone product using a format that is possibly proprietary, rigid, or useful to a single tool. Software rely on "folkonomies" (local definitions) to describe data, and the data is useless without the producer of the data explaining how to interpret it. The low data maturity level presents not only significant risk to the enterprise (because of its high reliance on the data owner), but also its lack of transparency makes it less trustworthy to others (especially auditors).

The second level represents data that are understood by a local group of domain experts, shared across a project, and use open formats such as spreadsheets where the data can be combined with other tools. Although the second level data maturity is significantly better than proprietary data, its lifespan is usually limited to the lifespan of a project and its impact is limited to the projects it supports.

Third maturity level is usually domain-based, adopting standards such as the CIM to represent metadata names. The concepts of integration, reuse, and interoperability are built into the development of data products so that they can be combined with other databases or standards to expand their usefulness. In this level, efficiencies are achieved through automated or semi-automated tools, which are based on data models that establish well-understood, configuration-controlled interfaces. These interfaces and common data structures are considered "instrumented" because they represent stable products that new software tools can be built against. CIM Profiles

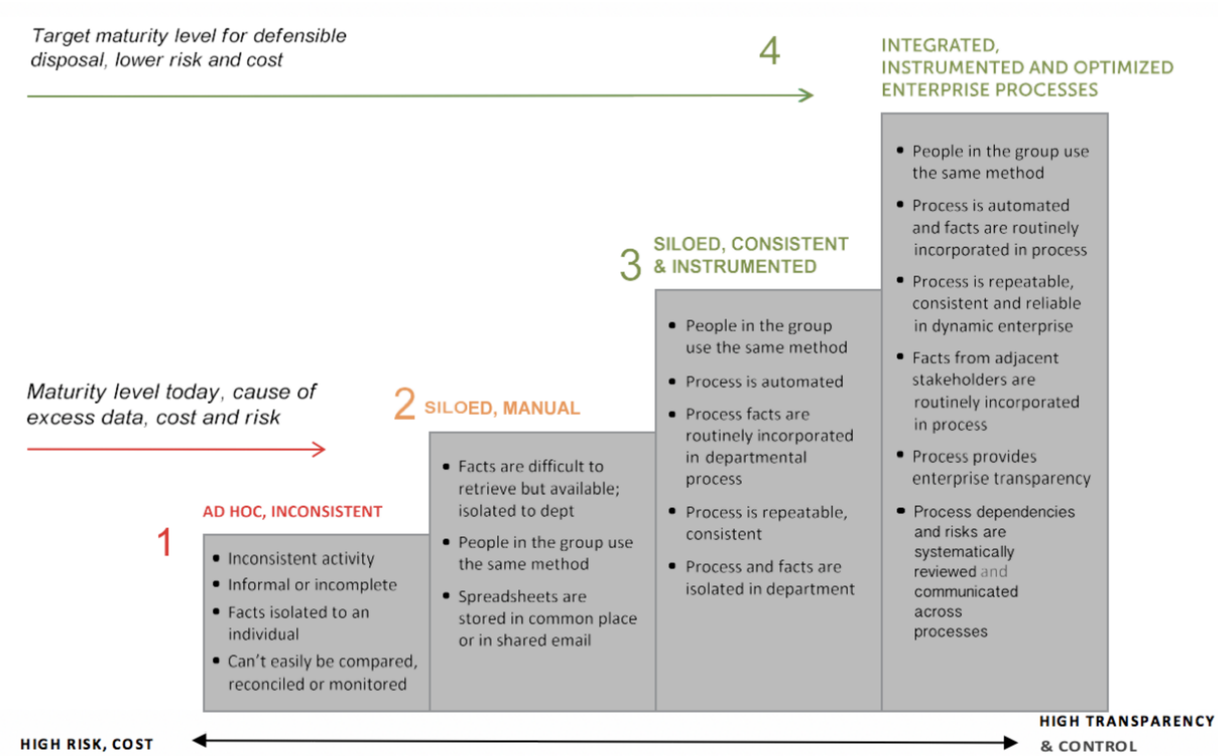

Figure 12: The CGOC Information Governance Process Maturity Model, reproduced from [22]

(introduced in the next section) are built to achieve this level of data maturity and represent the building blocks of specific themes (e.g. an EMS equipment profile) or more complex themes (e.g. equipment profile + topology profile + geography profile = complete feeder model).

The fourth maturity level of data are products that typically fulfill business functions and that represent stakeholders between or across organizations such as operational technology (OT) data sharing between utilities. These products rely on well understood stakeholder agreements, policies, and protocols that agree on the use and understanding of data.

5.0 Next Steps for CIM Adoption

One of the common misconceptions regarding the CIM is that the entire information model must be adopted and implemented. As an enterprise-focused canonical information model, the full CIM covers an extremely broad range of aspects of modeling, operations, billing, asset management, and energy markets. Any given application or even an entire data-rich environment will only need a small subset of the full common information model.

When choosing CIM for a particular project, application, and data integration effort, it is important to consider the particular use case, what functional objectives must be met, and what is the minimum number of attributes that must be modeled to meet the identified requirements. This section will briefly discuss some of the consideration involved in selecting a CIM profile, building a data profile, populating that profile with power system data, and then selecting a database structure to contain that data. This section will also provide concise definitions of some of the difficult aspects of canonical information modeling. The data mapping problem of tracing DERs to substation breakers will be used a running example to illustrate the concepts presented.

5.1 Understand Business, Planning, and Operations Processes

Adoption of the CIM cannot occur inside a vacuum and should consider the set of operational procedures and business processes that will be impacted and involved in the data integration and data exchange process. This process draws heavily on the systems engineering domain, and involves an organizational understanding of units or groups involved, existing operational procedures, formats currently used for data storage, legacy application interfaces, data streams, etc. This understanding should then be translated into a set of functional objectives, user requirements, and performance specifications.

If the data integration project will involve operations or operations planning recommendations decisions, then it is important to ensure that the data is integrated and presented in a meaningful manner to power system operators and dispatchers. Ensuring proper data presentation and access will require appropriate human factors to overcome the fact that many of the components of the CIM are not human-readable or understandable in the context of traditional operations procedures. Care must be taken at the initial concept development stage to ensure that completed data integration product does not add to the mental demand, temporal demand, effort, frustration, or other aspects of cognitive workload. If the output of a new CIM-based application is not understandable or creates a burdensome workload for operators in the high-risk, high-stress, high-tasking environment of control rooms, then that tool is likely to be rejected by operators and dispatchers.

Returning to the DER coordination and mapping example, there are a range of business and operations processes that must be considered. Systems planning, interconnection studies, customer billing, and asset management activities are all directly involved in determining the location and electrical connection of a given DER. Its connection to an energizing substation at any time will be determined by the current switching configuration of the feeder, ongoing maintenance activities (which may have placed temporary cuts and jumpers), and a combination of planned and unplanned outages. Reporting of the DER's location to avoid inadvertent disconnection during execution of an emergency load shed directive requires knowledge of not only current network topology and unique equipment mRIDs used by the CIM-compliant application, but also the human readable names of substation switches and breakers as they are displayed to the power system operator (which are often not contained in planning bus-branch models, but only in the EMS/ADMS operational network databases). Finally, the information presented to the operator must be reported in a concise format that supports real-time decision-making. In this case, a graphic display of feeder breakers sorted by “safe to shed” (no DERs or critical load) and “do not drop” (large DERs or critical loads) is far more useful than a table mapping the locations of DERs to their energizing substations.

The functional requirements, data interfaces, usability considerations, and anticipated role of the new application in current business / operations processes should be captured in detail and used as the basis for a concept of operations (ConOps) document. A well-defined ConOps document helps lay the foundation for the data integration effort and avoid numerous project pitfalls. It also provides an understanding of what information needs to be modeled, which can then be translated to the exact set of objects and attributes that need to select a CIM profile

5.2 Identify the Use Case for CIM

The next step is identifying the particular use case for adopting the CIM and amount of data integration that needs to be performed. This decision is informed by the list of interfaces between existing application, databases, and measurement streams identified in the previous section. It is then possible to determine how the CIM will be employed. Possible paths could include strictly using CIM to exchange base power system models between applications, integrate applications through a common message bus with CIM-compliant messaging and models, or create an integrated data-rich environment and modeling platform that uses CIM classes, attributes, and message formats to the greatest extent possible. Again, this decision will be informed by the particular use case and the types of data that need to be exchanged.

For the DER example, the use case for CIM could be on the project level or platform level, depending on the exact operational needs. If DER coordination is to be performed at the system planning or operations planning (i.e. day-ahead) horizon, then the exchange of CIM models and messages is likely to be sufficient. However, a platform-based approach will be more useful if the topology mapping must be performed in real-time, synthesizing incoming SCADA data, DER dispatch commands, and switch positions as they are received. In the latter case, a CIM-based message bus will be a more practical solution and will enable the involved applications, services, and data managers to publish and subscribe via a single method (instead of creating numerous point-to-point links between different participants).

5.3 Identify the Required Information to be Modeled

Although it is possible to use the entire CIM to represent every single object and attribute in the power system, it is more practical to reduce the modeling scope to the minimum information required to accomplish the functional objectives originally identified. This requires careful consideration of what data needs to be exchanged, whether a full power flow solution is needed, and level of modeling accuracy that is needed.

For the DER mapping example, the minimum required information is simply a topology model of the power system network, identifying the connectivity between lines, transformers, switches, and DERs. Detailed modeling of line geometry, transformer impedances, tap settings, and reactive control modes is not necessary, as none of these attributes impact the connectivity of a DER to its associated energizing substation and feeder breaker.

5.4 Identify the Required Classes and Attributes

Once the minimum required set of information has been identified, it is possible to reduce the CIM information model to a much smaller profile with only the set of classes and attributes that are needed. This custom subset of classes, attributes, and associations is known as a *CIM Profile*. A CIM Profile can also add custom extensions for classes and attributes that are not included in the base CIM information (such as home appliances or DER control attributes) using the workflow outlined in the EPRI CIM Primer [2]. If the data integration effort needs extensions to the CIM, it is recommended to involve a domain expert with a strong background in both power systems and information modeling to ensure that all extensions are properly defined and constructed. It is possible to build one's own custom CIM profile using open-source software, such as CIMTool [23], or adopt an existing CIM profile that has already been developed and tested.

For the DER mapping example, an extremely small CIM profile could be used to accomplish the topology mapping task. Only a handful of classes (including ACLineSegment, LoadBreakSwitch, PowerTransformerEnd, SynchronousMachine, PowerElectronicsConnection, ConnectivityNode, and ACDCTerminal) would be needed to specify the entire power system network. If the DER coordination application does not need to solve a full power flow solution, numerous other classes (such as PhaseImpedanceData, TransformerMeshImpedance, and RatioTapChanger) can be omitted from the CIM profile without impacting the application.

Before introducing physical models, a brief introduction to profiles is necessary. *Profiles* are defined as a secondary model that is derived from the information model. Profiles must be based on classes, attributes, and associations contained in the information model. They can never introduce anything new. Their purpose is to identify a subset of the information model to meet a particular need. Profiles can constrain cardinality, remove unwanted attributes, and classes to represent structures required for a given application. After completing the logical model UML, profiles can be derived to depict structures that support specific topics such as equipment, geological, single state hypotheses, and topology. Profiles serialized to a specific physical data structure or system (repository or stream) are known as a *data profile*.

5.5 Create a Data Profile

Once a CIM Profile has been selected or created, the next step is to create a data profile to contain the power system network model. It is important to note that the data profile is not the power system data. A data profile is an empty file structure that specifies how the network model data should be organized in a file.

Data profiles may be derived directly from the UML information model or a UML profile. Examples of data profile formats may be JavaScript Object Notation (JSON) schema, eXtensible Schema Definition (XSD) schema, relational data definition language (DDL), and Resource

Description Framework (RDF) serialization. The selection of a particular data profile schema is not related to adoption of the CIM, but rather the purview of the software development team. XSD schema are very popular because they be used to validate the XML files commonly used for power system model files. JSON schema are identical in structure to Python dictionary objects and easily parsed by simple Python scripts.

Data profiles can be used as building blocks from which to create layers of the power system network in a manner similar to layers of a geographic information system (GIS) model. One of the most common examples of layers within a data profile has topology profile at the lowest layer, which describes how all buses, switches, and branches are connected to each other and how their connectivity should be described using the consensus-based vocabulary established by CIM. However, the topology layer has only limited usefulness. Most advanced power applications also need an equipment profile that contains nameplate info, physical characteristics, line impedances, etc. This is another layer that be stacked on top of the topology layer for additional info. Then, a geographical profile can be layered on top of the other data profiles to provide the vocabulary for specifying geospatial information about the location of assets in the distribution feeder or transmission network.

5.6 Populate the Data Profile with Network Model Data

The next step is populating the selected data profile with the actual power system network model data. This is usually done by creating a custom script that interprets the empty data profile and uses the specified structure to build an XML file that contains the actual power system network model data. This step typically requires the involvement of an experienced software developer familiar with UML modeling and interpretation of XSD or JSON schema.

XML files are the most popular format for exporting CIM-compliant power system network model data for two reasons. The first is XML files are easily interpreted by web-based tools. The second reason is the availability of tools to validate the network model file for any syntactic or semantic errors. If the XML file is "well-formed," an XML parser will be able to validate that the network model data file is valid and does not contain any syntactical errors. If XSD was used for the data profile in the previous step, the XML network model file can be validated semantically to ensure that all objects and attributes are defined correctly. In a similar manner, JSON-schema data profiles can be used by custom scripts in multiple programming languages to ensure the XML network model file conforms to the data profile. Finally, semantics specifications such as the Resource Description Framework Schema (RDFS) and Web Ontology Language (OWL) provide ways to ensure that the network model file follows the consensus-based vocabulary outlined in selected CIM profile.

To populate data profile with network model data, the software developer needs to understand the elements of the data profile, which classes and attributes are being used, and how those attributes correspond to the power system elements contained in the source data. The UML class diagrams of the full CIM information model and the subset of classes selected for the CIM profile are the readable and understandable format for a human software developer. However, it is critical that the software developer systematically compare the data profile and schema definition to the available network model data to understand the information that needs to be written to the XML model file in the power system model conversion script.

To date, the authors have not found any convenient power system modeling tools for building distribution feeders or transmission networks in CIM XML format natively. It is still much more effective to build network models using existing commercial tools (such as PSLF) or open-source tools (such as OpenDSS) and then convert the power system network data to a CIM XML file.

One of the critical steps in the model conversion process is establishing a persistent set of Universally Unique Identifiers (UUIDs) to be used as the Master Resource Identifiers (mRIDs) for all network parameters, asset characteristics, and available SCADA measurements. This is important for multiple reasons. The first is all applications within the data integration effort will need to able to refer to a particular piece of equipment by the same mRID. If applications are not able to use a consistent mRID across multiple databases and middleware services, it may be necessary to create mapping tables of inconsistent mRIDs between application (which is contrary to the objective of CIM-based data integration). The second reason is that any changes to the list of mRIDs are likely to affect other data mappings, such as the correspondence of SCADA measurement points to asset data. Keeping a persistent set of mRIDs helps ensure that any edits to the power system network model do not affect any other data mappings within the software platform.

When exchanging information between utilities, additional care must be taken to maintain the mRIDs of equipment included in the network model data of both utilities. Common examples of such equipment are substation transformers at the transmission-distribution boundary and large transfer paths running through a particular utility's area of responsibility. Two approaches can be taken to managing the lists of unique identifiers of equipment included in multiple utility models. The first is to use the same set of mRIDs across both utilities. This requires communication, coordination, and sharing of the master list of mRIDs between the planning and operations departments of both utilities. It is important to refer to the master list of mRIDs to avoid possible conflicts or non-unique identifiers while adding new assets and objects to the network model, whether within a single utility or across utilities. The second approach is to create a reference table converting the set of mRIDs used by one utility to those used by the neighboring utilities with which it needs to exchange data.

The final step in populating the network model data into the data profile is generating a list of measurement mRIDs for all SCADA, AMI, and other measurements that are to be aggregated and made available to advanced applications. CIM does not provide any specification of the format in which measurements are to be reported. Instead, the only requirement is that every measurement or data source be assigned a unique mRID and be associated with one terminal of a piece of equipment. For example, the voltage and current measurements from the potential transformer (PT) and current transformer (CT) on a transformer bushing would be defined as two separate measurement objects (each with a unique mRID) associated with the Terminal object of that transformer winding. If those measurements are then used to calculate MW, MVAr, and MVA power flow values, those calculated measurements would also be defined as new measurements objects (each with a unique mRID) associated with the transformer Terminal object.

Although this approach of defining mRIDs for every measurement may seem unnecessarily complex at first, it greatly simplifies the format for publishing measurements in real-time and storing them in a historian database. All the information about the measurement location, units, geospatial location, and equipment association are included in the network model data. As a result, the actual measurement data can be published in the simple format of `"meas1-mrid-1234": value` without the need to include any other information in the data message.

5.7 Select a Database Format

Once all power system network data and measurement data have been converted in a format specified by the data profile, the database format and structure must be selected. CIM does not specify any requirements for how the network data should be stored, but some formats are better suited than others for handling the directional relationships between CIM objects and attributes. However, it must be understood clearly that each database platform offers various advantages and disadvantages, as shown in Table 1 below. There is no perfect solution for all problem spaces.

Table 1: Comparison of Database Platform Advantages & Disadvantages for Storing CIM Data [9]

Requirement	**Relational**	**Triple-Store**	**Graph**
Administrative-free schema maintenance	No	Yes	Yes
Ability to represent CIM objects intuitively	No	Yes	Partial
Polymorphism	No	Yes	Partial
Directed graphs	No	Yes	No
Schema evolution free solution	No	No	Yes
Value type and referential integrity checking	Yes	Partial	No
Indexing	Yes	Yes	Yes
Data management language support	Yes	Yes	Yes
Ease of technology transfer	Yes	Yes	No
Community involvement	Yes	Yes	Yes
Product maturity	Yes	Yes	Yes

The relational database became exceptionally popular in the early 1990s with the advent of American National Standards Institute (ANSI) standard interfaces to databases, which finally made portability achievable. On the client-side, standardized APIs such as Open Database Connectivity (ODBC) and Java Database Connectivity (JDBC) provided non-proprietary solutions so that one client could support many relational database providers. Relational database technologies require the schema to be defined by a system administrator. Developers can then populate data in the predefined schema. This approach works well in situations where the data structures are stable and can be predefined. Most vendor EMS and DMS software suites still use relational databases.

The relational database offers a very stable ability to create database tables, evolve the schema, and upload bulk data. It also offers column data-types, referential integrity between inter-related tables and indexing can optimize the table access. The data definition language (DDL) is based on ANSI SQL. The database schema can be managed through administrative tools, a command line client, or through ODBC / JDBC APIs. For CIM-based projects, an CIM Enterprise Architect

Project (EAP) can be imported into the Eclipse-based CIMTool software to create CIM profiles and data files that can be exported as a DDL specification.

The key strength of the relational database is that it offers a proven, standardized, state-of-the-art way of managing the database and structures. Due to its reliance on ANSI standardized approaches, the DDL and queries can be easily ported to other relational databases. Its weaknesses lie in requiring a designated database administrator for specifying database structures. In turn, this also requires all the structures to be well thought-out. Changes in profiles require a process called schema evolution to port older records into new evolving concepts. This can be difficult if concepts are changing, which requires existing data to be ported to new structures. For OT systems, incorporating new database changes could require downtime to support new concepts.

However, one of the largest problems with using relational databases for CIM network model data is the lack of support for polymorphism (the ability of a single column to contain data of different types) and artificial data structures. Introducing artificial data structures can lead to highly complex and unintuitive queries to support this level of complexity. Introducing new data structures and complex queries schema require rigorous configuration control methodologies, downtime, and managed rollouts of new queries. This process is particularly burdensome for data structures and queries that are constantly evolving along with the evolution of the power system.

Database technology has evolved over the decades since the advent of the relational database. During the 1990s, several data formats began emerging to provide structured data outside databases, including delimited format files, Javascript Object Notation (JSON), and the eXtensible Markup Language (XML). These newer formats differed significantly from the relational database in that data structures could be defined independent of whether a schema existed or not. These data formats began appearing in standalone files and are embedded as data structures in Web pages as well as in next generation databases. In the early 2000s, NoSQL (Not Only Structured Query Language) or schemaless databases emerged as an alternative to the traditional database. A few examples include graph databases (property and semantic), time-series stores, and document-based stores (JSON and XML). In contrast to relational database technology (based on related, predefined tabular structures), NoSQL can support introduction of irregularities, such as new data elements at the data record level. Depending on the technology, these platforms offer new ways to pattern-match, aggregate, and traverse topologies using advanced query languages and filtering algorithms.

The triple-store database offers a semantic solution for data management. Unlike the relational database solution (which requires a DDL database schema), the triple-store database structure is comprised of resource descriptive framework (RDF) statements. These RDF statements take the form of subject (node), predicate (relation), object (node) that can be dynamically generated to form inter-related complex class structures. RDF Schema (RDFS) supports polymorphism concepts, provides a means to create graph constructs, and supports the concept of subgraphs so that partitioning of the graph space is possible. RDF can be specified in a variety of ways including XML, TTL, and JSON-LD. Other languages, such as the Web Ontology Language (OWL), can be used to offer more sophisticated constraints and additional constructs for reasoning and inferencing.

The RDF statement structure intuitively corresponds to the structure of object-attribute specifications used in CIM (e.g. ACLineSegment (subject) has an attribute (predicate) of length (object)). The CIM supports translation of data structures directly into RDF with the ability to automate correlation of data. The CIMTool software supports translation of a CIM profile into RDFS and OWL. After ingesting the power system network model data and the RDFS / OWL structures from the data profile, the user can use a triple-store databased to correlate well-defined CIM structures with newly generated data automatically.

The main advantage of using a triple-store database is that it directly supports CIM development. It is highly agile and supports dynamic data structures generated by multiple developers. It is also supported by a standardized, mature language for specifying directed graph and hierarchical class structures as well as a standardized way to manage RDF data. RDF supports both type-checking and the Shapes Constraints Language (SHACL) for structure validation. The main disadvantage of triple-store is the risk for garbage-in-garbage-out (GIGO), meaning that without a well-thought-out strategy to support data management activities, maintenance can be difficult. Likewise, without a certain rigor for data contributors, dangling references or conflicts can appear in the database.

Another alternative is a graph database offering a bare-metal solution for data management. For the purposes of this report, comments are shared based on exploration using the Neo4J database. Graph databases each offer their own language and can support directed, undirected, or both kinds of graph structures. Graph database structures are comprised of nodes, edges, and descriptive properties that can used to provide scalar values describing each node or relationship. The Neo4J Cypher query language is used to create, update, and delete the structures at the time of data creation. Because of the simplicity of its data constructs, there is no concept of subgraphs, or inheritance. The nodes and relationships use integer identifiers that are not globally unique but appear unique to a given database. Nodes, relationship, and property names do not allow many special characters without being surrounded by back tick special characters. These data structures are very helpful for calculating paths, traversing topologies, searching for patterns. Neo4J does support a notion of indexing by labels. Property graph databases offer highly agile, dynamic data structures that could be generated by multiple developers, supporting a fairly mature language for managing basic graph structures.

Some of the weaknesses of graph databases are inherent to the highly dynamic data structures used. It can be difficult to reverse engineer more complex structures, such as the CIM. Likewise, deriving schema may be more difficult to perform, as the graph grows larger. No referential integrity or type checking are performed, so it can be very difficult to find data errors. Ill-defined or obscure graphs could also lead to confusing results or meaningless query results. However, these weaknesses should lead graph databases to be discounted, as the lack of pre-defined data structures mean that the data structures can evolve with the research and types of data that need to be included.

5.8 Implement Data Integration Workflows

After a database structure is selected and the CIM power system network data are ingested, the remaining data integration workflows can be implemented. The complexity of this step largely depends on the number of data streams, models, shared services, and applications that need to be integrated. The core of this step is defining the manner in which measurements will be received,

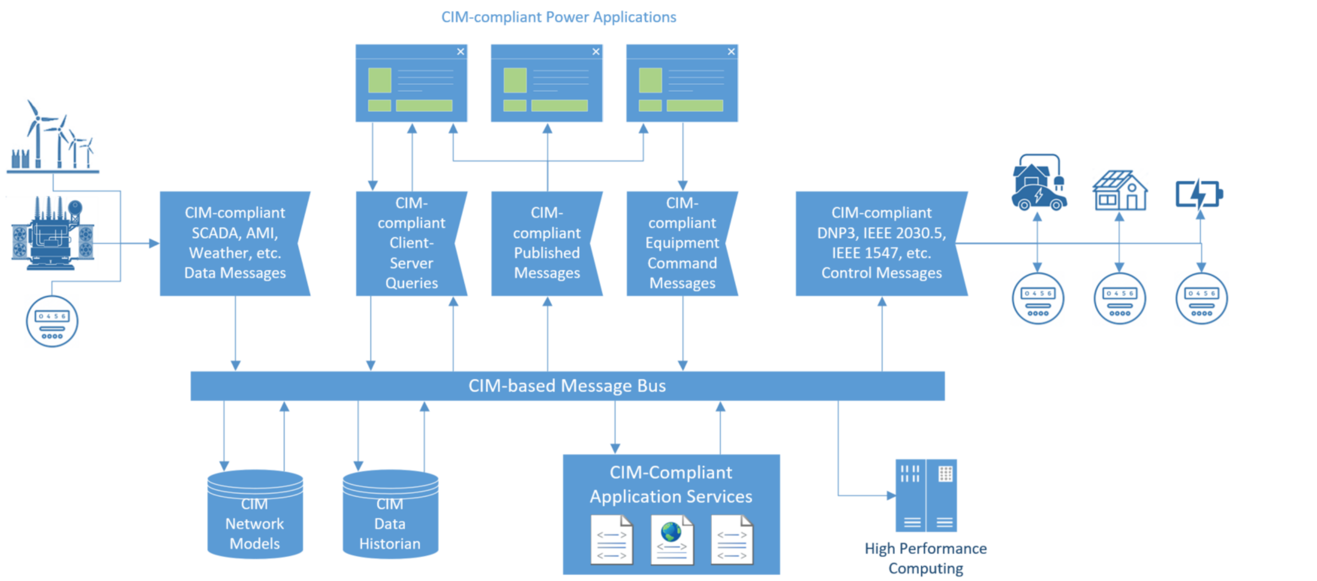

Figure 13: Conceptual model of a CIM-based platform for creating a data-rich application environment. (repeated for convenience from Section 4.4)

database queries will be made, and equipment control commands will be published. Figure 13 (repeated from Section 4.4) illustrates some of the workflows that need to be established. Although there are numerous mappings that need to be made, most can be accomplished by a power applications engineer with just fundamental knowledge of the consensus-based vocabulary used by the CIM information model.

An example data integration workflow is focused on managing and mapping incoming measurements (e.g., AMI measurements and SCADA data) to previously defined measurement mRIDs, and then associating measurement mRIDs with equipment mRIDs. This enables multiple data streams and measurement source to be aggregated using a single set of consistent identifiers. The synthesized data streams can then be configured to be published on the CIM message bus using the simple `mRID: value` format first discussed in Section 5.6. All shared services and advanced application within the data integration environment can subscribe to streaming data to obtain whatever values (e.g. bus voltages) or triggers (switching actions) are needed to complete their own functional objectives.

Another example workflow is setting up database queries and reporting network model data to CIM-based power applications. Database queries can be executed directly or packaged into a user-friendly API. These workflows can be scripts or ETL (extract, translate, load) tools similar to those used in relational databases.

In creating data integration workflows, an important concept is that of *tool chains*. Frequently, a single script is not sufficient to complete the necessary set of data integration tasks. Multiple steps are often required to convert model files, ingest network data, generate measurement mRIDS, etc. These related tasks can be accomplished by chaining together multiple tools that are executed in a sequential manner by the user. An example of a tool chain consists of a simple custom script that converts one row of a PSSE .raw file and exports the data to a CIM XML file. The script can be

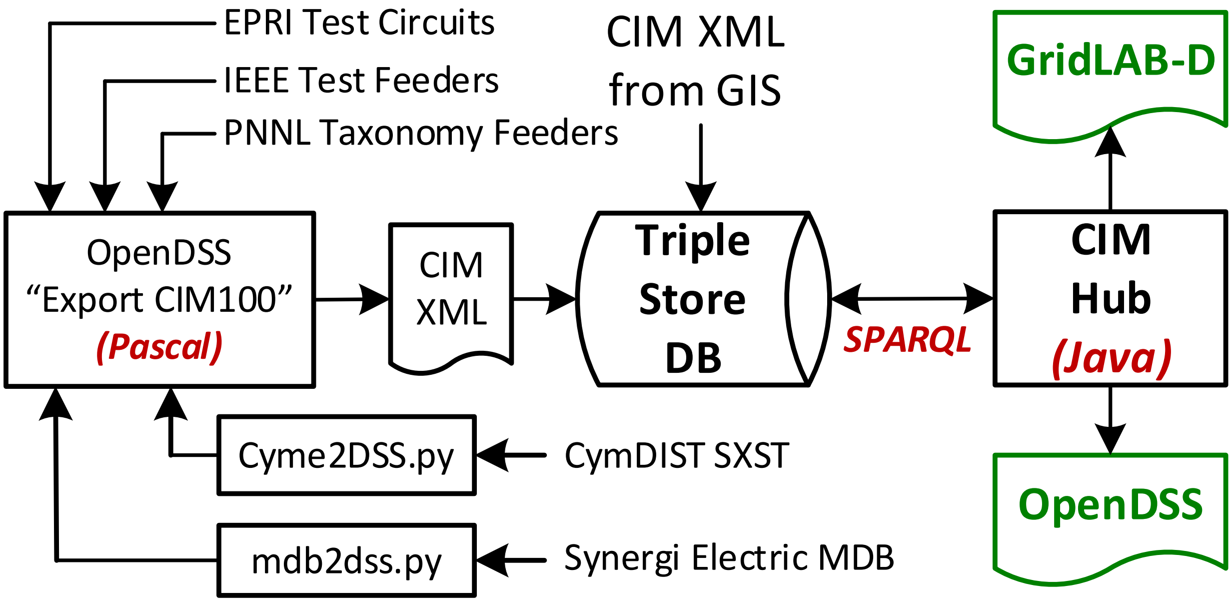

Figure 1411: CIMHub [24] is an example of data integration tool chain for extracting, translating, and loading distribution feeder data using common open-source data formats.

chained to a Python script that reads each row of file and specifies the destination directory to which the CIM XML data should be written. Then, the third script in the tool chain imports the CIM XML file and loads it into the selected database structure. Then, a fourth script parses the list of measurements and maps those measurement mRIDs to the equipment mRIDs ingested into the database, and so on. It is important that the tools developed are robust enough that workflow scripts can be chained together.
An example of a CIM tool chain for open-source distribution data formats is CIMHub [24], which is illustrated schematically in Figure 14. CIM Hub is combination of all three types of ETL tools (extract, translate, load) and can be used to convert distribution feeder models, generate mRIDs for available measurements, and then load the combined network/measurement model into a triple-store database.

5.9 Test and Deploy

The final step is to test and deploy the data integration solution and CIM-compliant advanced applications. This section will not attempt to cover the vast set of considerations involved in software testing, but rather share some recommendations and best practices related to CIM modeling.

It is recommended that the advanced application or data integration platform include a simple set of unit tests that execute validation scripts that run common queries for power system equipment and measurements. These unit tests help ensure that any changes to applications or databases do not cause breaking changes to the CIM profile, data profile, CIM data ingest, or network model data. These unit tests can also ensure that equipment attribute values are within their expected range by comparison to defined minimum / maximum values or logical checks based on knowledge of physical characteristics of the power system (e.g. no negatives values allowed for conductor lengths).

An example of a more detailed testing tool for ensuring the validity of CIM power system data is the Model Validator App [25], [26] developed by the GridAPPS-D program and depicted graphically in Figure 15. The application focuses on providing an automated method for detecting modeling errors that cannot be detected syntactically or semantically but are due to errors in the actual data that will cause a power flow solution to diverge. The application systematically parses each piece of power system equipment for device-level inconsistencies. It then computes individual device-level primitive admittance matrices from CIM model parameters, which are validated against a full system admittance matrix.

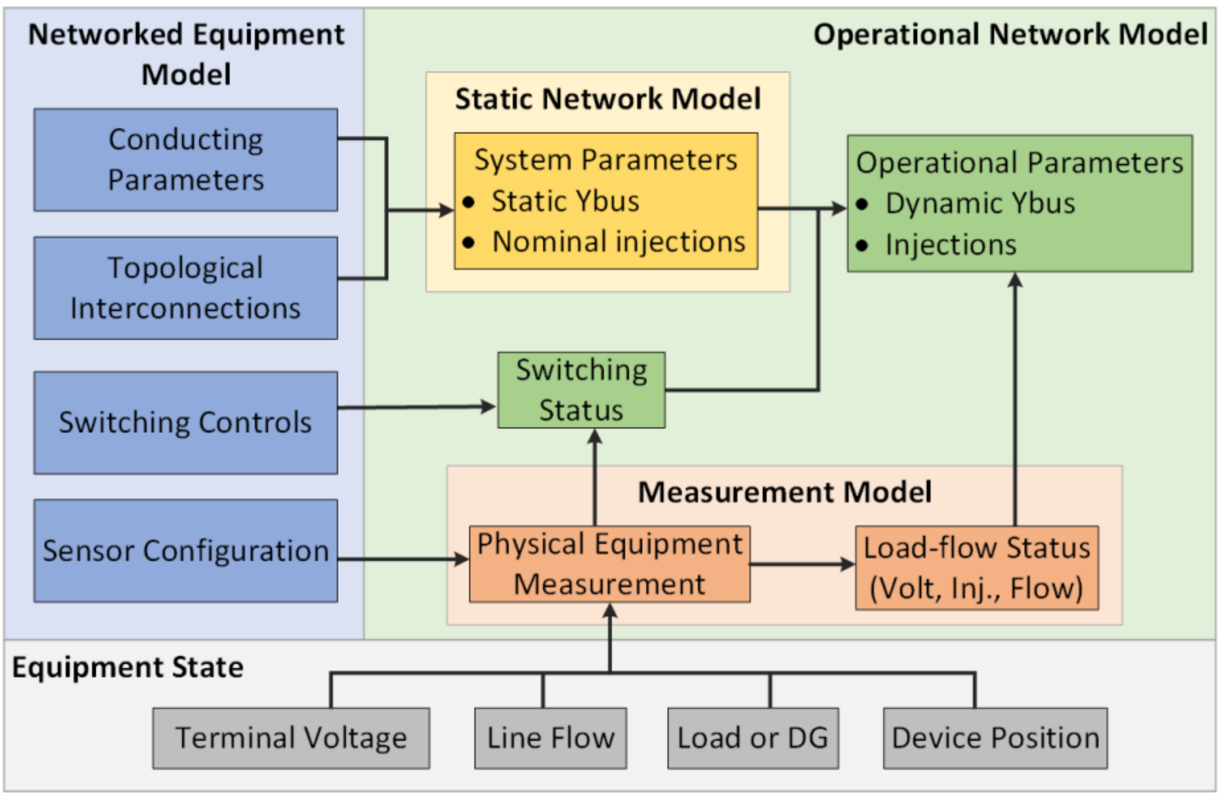

Figure 15: Structure of the Model Validator App for verifying the numerical integrity and validity of CIM distribution feeder model data, taken from [25].

6.0 Conclusion

This report introduced the set of key concepts related to adoption of the CIM to enable data integration and data exchange, looking towards creation of the next generation of advanced power applications that will be needed to ensure reliable, resilient, robust, safe, and economical operations.

The Common Information Model was discussed in the context of the need for standardization of representation of power system networks, assets, and data exchange messages. Three different approaches to management and exchange of utility data were presented, shifting from vertically-integrated proprietary application stacks to horizontally-integrated applications and shared services communicating across a shared enterprise message bus. Agreed-upon information models, such as CIM for power systems and NIEM for homeland security, were introduced as proven approaches to avoiding the need for custom data adapters between each application and each proprietary data format used within a utility. Key concepts related to information modeling, including semantics, syntactics, canonical models, data profiles, data structures, and database schemes were introduced using a running example of mapping distributed energy resources to substation breakers for improved coordination.

The CIM is an agreed-upon canonical information model for describing power system networks, asset information, operational procedures, and market data in a consistent manner. More specifically, the CIM provides a consensus-based vocabulary for describing the semantics of what information is meaningful for which types of power system equipment. As a canonical model, the full CIM provides far more detail across a far broader range of power system objects than could possibly needed by a single application or platform. As a result, it is typically necessary to follow a sequence of steps to create a CIM profile from the minimum set of object and attributes, export that CIM profile to an empty structure (known as a data profile), build custom converter script(s) to convert the power system network data to populate the selected data profile with CIM XML network model data, select a database structure, implement the selected data integration workflows, and create unit tests to ensure the syntactic, semantic, and numerical/topological validity of the CIM power system model.

Power systems engineers and application developers seeking to use the CIM are encouraged to download the CIM information model (available freely under an Apache 2.0 license) from the CIM Users Group website [19]. The companion to this report, *A Power Application Developer's Guide to the Common Information Model* [1], provides a detailed explanation of all technical concepts needed to understand the set of classes and attributes within consensus-based vocabulary provided by CIM. The combination of both reports should give most power application developers a sufficient technical background for developing CIM-compliant applications and services. For those interested in developing custom data converters between CIM and other power system model formats, the EPRI CIM primer [2] provides a highly detailed discussion of the process of deriving CIM profiles, data profiles, and database schema (which were introduced at a conceptual level in Section 5 of this report).

References

[1] A. A. Anderson, T. E. McDermott and E. G. Stephan, "A power application developer's guide to the Common Information Model," Pacific Northwest National Laboratory, 2022.

[2] Electric Power Research Institute, "Common Information Model Primer 7th Edition," Electric Power Research Institute, Palo Alto, CA, 2021.

[3] NY-ISO, "Reliability and market considerations for a grid in transition," 2019.

[4] S. Vadari, Electric system operations: Evolving to the modern grid, Artech House, 2020.

[5] M. Kezunovic and T. Popovic, "Substation data integration for automated data analysis systems," in *IEEE PES General Meeting*, Tampa, FL, USA, 2007.

[6] A. W. McMorran, G. W. Ault and J. R. McDonald, "Solving data integration challenges for web-based geographical power system data visualization using CIM," in *IEEE PES General Meeting*, Pittsburgh, PA, USA, 2008.

[7] NIEM, "About the National Information Exchange Model," [Online]. Available: https://www.niem.gov/about-niem. [Accessed 20 01 2022].

[8] T. Kean and L. Hamilton, "The 9/11 commission report: Final report of the national commission on terrorist attacks upon the United States. Vol. 3.," Government Printing Office, 2004.

[9] R. Melton, K. Schneider, E. Lightner, T. McDermott, P. Sharma, Z. Y. F. Ding, S. Vadari, R. Podmore, A. Dubey and R. Wies, "Leveraging standards to create an open platform for the development of advanced distribution applications," *IEEE Access,* vol. 6, pp. 37361 - 37370, 2018.

[10] R. B. Melton, K. P. Schneider and S. Vadari, "GridAPPS-D a distribution management platform to develop application fo rural electric utilities," in *IEEE Rural Electric Power Conference*, 2019.

[11] International Electrotechnical Commission, "IEC 62361-100: Power systems management and associated information exchange - Interoperability in the long term - Part 100: CIM profiles to XML schema mapping," June 2016. [Online]. Available: https://webstore.iec.ch/publication/25114. [Accessed January 2022].

[12] International Electrotechnical Commission, "IEC 62361-104 (draft) CIM Profiles to JSON schema Mapping," July 2021. [Online]. Available: https://cimug.ucaiug.org/WG19/IEC%2062361-104%20CIM%20Profiles%20to%20JSON%20schema%20Mapping%20-%20DRAFT/57-62361-104-IS-CIM%20Profiles%20to%20JSON%20Schema%20Mapping_rev01v20.pdf. [Accessed January 2022].

[13] Electric Power Research Institute, "Inter-control center communication protocol (ICCP) user's guide (TR-107176)," December 1996. [Online]. Available: https://www.epri.com/research/products/TR-107176. [Accessed December 2021].

[14] Electric Power Research Institute, "Inter-control center communications protocol interoperability test version 5.1 (TR-105552)," 1995. [Online]. Available: https://www.epri.com/research/products/TR-105552. [Accessed 2021 December].

[15] S. Vadari, Smart grid redefined: Transformation of the electric utility, Boston: Artech House, 2018.

[16] AVEVA / OSIsoft, "OSIsoft Documentation," 2021. [Online]. Available: https://docs.osisoft.com/. [Accessed January 2022].

[17] R. Melton, K. Schneider, T. McDermott and S. Vadari, "GridAPPS-D Conceptual Design," May 2017. [Online]. Available: https://www.pnnl.gov/main/publications/external/technical_reports/PNNL-26340.pdf. [Accessed January 2022].

[18] GE Grid Solutions, "Model Export Software Installation and Maintenance Guide," GE Grid Solutions, 2017.

[19] CIM Users Group, "Current CIM Model Drafts," December 2021. [Online]. Available: https://cimug.ucaiug.org/CIM%20Model%20Releases/Forms/AllItems.aspx. [Accessed January 2022].

[20] GridWise Architecture Council, "GridWise interoperability context-setting framework," March 2008. [Online]. Available: https://www.gridwiseac.org/pdfs/interopframework_v1_1.pdf. [Accessed January 2022].

[21] International Electrotechnical Commission, "IEC 61968-1:2020 Application integration at electric utilities - System interfaces for distribution management - Part 1: Interface architecture and general recommendations," April 2020. [Online]. Available: https://webstore.iec.ch/publication/32542. [Accessed January 2022].

[22] Compliance Governance and Oversight Council, "Information Governance Process Maturity Model," July 2018. [Online]. Available: https://community.ibm.com/HigherLogic/System/DownloadDocumentFile.ashx?DocumentFileKey=fa0b2e30-817b-b948-1833-c097d6eb651e&forceDialog=0. [Accessed January 2022].

[23] CIMug-org, "CIMTool Eclipse platform for the Common Information Model," May 2020. [Online]. Available: https://cimug-org.github.io/CIMTool/ . [Accessed Jan 2022].

[24] M. Mukherjee, E. Lee, A. Bose, J. Gibson and T. E. McDermott, "A CIM based data integration framework for distribution utilities," in *Proc. IEEE Power & Energy Society General Meeting*, 2020.

[25] S. Poudel, G. D. Black and A. P. Reiman, "An application for validation of power distribution system models in an ADMS environment," in *IEEE PES General Meeting*, 2021.

[26] S. Poudel, G. D. Black, E. G. Stephan and A. P. Reiman, "Admittance matrix validation for power distribution system models using a network equipment model framework," *IEEE Access,* vol. 10, pp. 9108 - 9123, 2022.

[27] Object Management Group, "Object Constraint Language specification version 2.4," Feb 2014. [Online]. Available: https://www.omg.org/spec/OCL/. [Accessed Jan 2022].

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99354
1-888-375-PNNL (7665)

www.pnnl.gov