PNNL-28207

Universal Utility Data Exchange (UUDEX) Functional Design Requirements

Cybersecurity of Energy Delivery Systems Research and Development

November 2018

SR Mix
MA Rice
CM Schmidt
S Neumann
S Sridhar
SV Singh
C Gonzalez-Perez
ML Cohen
C Peloquin

Prepared for the U.S. Department of Energy
under Contract DE-OE0000190

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor Battelle Memorial Institute, nor any of their employees, makes **any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights**. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or Battelle Memorial Institute. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

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,
U.S. Department of Commerce, 5285 Port Royal Rd., Springfield, VA 22161
ph: (800) 553-6847
fax: (703) 605-6900
email: orders@ntis.fedworld.gov
online ordering: http://www.ntis.gov/ordering.htm**

This document was printed on recycled paper.
(9/2003)

Universal Utility Data Exchange (UUDEX) Functional Design Requirements

SR Mix
MA Rice
CM Schmidt
S Neumann
S Sridhar
SV Singh
C Gonzalez-Perez
ML Cohen
C Peloquin

November 2018

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

Pacific Northwest National Laboratory
Richland, Washington 99352

Revision History

Revision	Date	Deliverable (Reason for Change)	Release #
0	11/2018	Initial Release	PNNL-28207

Summary

This document contains the set of functional and architectural requirements for building the Universal Utility Data Exchange (UUDEX) Framework.

Section 1 contains an introduction to UUDEX, including a description of the project scope, a discussion of how UUDEX will support existing and emerging utility communications and infrastructures, an overview of the proposed UUDEX architecture, and a high-level overview of the lifecycle of UUDEX data.

Section 2 contains 12 use cases, based primarily in electric entity interactions, that document how UUDEX could be used to provide communications structure for both operational data (such as ICCP), large data files (like power system model updates), incident and event reporting (such as OE-417 reports, or information sharing with the E-ISAC), mass alert notifications (such as NERC alerts), and temporary ad-hoc connections with first responders (such as FEMA during a hurricane response).

Section 3 contains functional descriptions of operational and cyber security data are described showing the breadth of data UUDEX is capable of communicating. Operational data types include ICCP, RCIS, power system model updates, synchrophasor, disturbance files, operations planning, and asset management. Cybersecurity data includes incident reporting, indicator of compromise sharing, guidance (e.g., firewall rule sharing), conformance reports (e.g., verification that patches have been installed) patch availability notification, vulnerability disclosure reporting, and threat notification.

Section 4 introduces the specific functional requirements, including a taxonomy of roles and terms is used to provide functional descriptions of the interactions between various UUDEX participants, as well as a discussion of data exchange architectures and requirements, including information flow, identity, communications tunnel, data storage, testing, and data lifecycle.

Section 5 introduces the UUDEX message and data exchange formats and describes them at a functional level.

Section 6 describes how UUDEX is proposed to be a hybrid publish-subscribe and query-response architecture supporting real-time and near-real-time notification of data elements available to data subscribers (following the current ICCP model), as well as the capability for querying a database of stored information (such as available software patch updates).

Section 7 contains an overview of security threats and mitigations planned for inclusion in the final product, including information disclosure, information corruption, denial of service, identity spoofing, and trust relationships, all with functional descriptions of how UUDEX will mitigate the threats or address the issues.

Section 8 contains a list of reference documents considered for use by UUDEX.

Appendix A contains a set of notional data elements characteristics considered during the development of the functional specification.

The functional specification addresses a number of areas that were raised in discussions with industry, both during an information gathering phases early in the project, as well as later conversations with the project's Industrial Advisory Board.

These functional design requirements are intended to be used in future more detailed design documents related to this project.

Acronyms and Abbreviations

ACL	access control list
API	application programming interface
BA	Balancing Authority (NERC term)
BES	Bulk Electric System (NERC term)
CIM	Common Information Model, as defined by IEC 61970 and IEC 61968 series of standards
DHS	U. S. Department of Homeland Security
DOE	U. S. Department of Energy
E-ISAC	Electricity Information Sharing and Analysis Center
EMS	energy management system
ESB	enterprise service bus
FERC	Federal Energy Regulatory Commission
GPS	Global Positioning System
ICCP	Inter-Control Center Communications Protocol, as defined by IEC 60870-6
ICS-CERT	Industrial Control Systems Cyber Emergency Response Team
IDS	intrusion detection system
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IoC	indicator of compromise
IPS	intrusion prevention system
IROL	Interconnected Reliability Operating Limit
ISAC	Information Sharing and Analysis Center
IT	information technology
NERC	North American Electric Reliability Corporation
NIST	U. S. National institute of Standards and Technology
OMS	outage management system
OT	operations technology
PCAP	packet capture
PMU	phasor measurement unit (also known as a Synchrophasor)
QOS	quality of service
RC	Reliability Coordinator (NERC term)
RCIS	Reliability Coordinator Information System
SCADA	supervisory control and data acquisition
SOL	System Operating Limit
STTP	Streaming Telemetry Transport Protocol
TOP	Transmission Operator (NERC term)

UUDEX	Universal Utility Data Exchange
UUID	universally unique identifier, as defined by IETF RFC 4122
XML	Extensible Markup Language, as defined by the W3C

Contents

Figures

1.0 Introduction

The purpose of this document is to provide a functional specification for the Universal Utility Data Exchange (UUDEX).

1.1 Project Scope

UUDEX describes a communications architecture and protocol suite that allows organizations to exchange data and information. It does this by defining relations between a set of client nodes that share data via a set of server nodes.

The primary focus of UUDEX is to facilitate communications between control centers, operations centers, and other trusted organization. This involves the communication mechanisms necessary for reliable and secure operations of an energy delivery system. The most common usage would be for the conveyance of measurements, calculations, and schedules between entities and applications that are responsible for management of the electrical grid at both the transmission and distribution levels. UUDEX can be used for communications between utility organizations and non-utility organizations, including government organizations and commercial enterprises, for example between a utility organization and its Information Sharing and Analysis Center (ISAC), the E-ISAC for the electricity sub-sector. UUDEX could also be used to send event and outage information to the U.S. Department of Energy (DOE) following requirements of the OE-417 reporting criteria. It could be used to coordinate information dissemination about line outage and restoration information between a utility and local first responders or the Federal Emergency Management Administration. Or, UUDEX could be used to provide near-real-time alerts from security service providers pertaining to vulnerability disclosure or patch availability.

UUDEX could also be used by an organization to coordinate internal data communications and some aspects of application integration. For example, it could be used as an intermediary for exchange of network models between systems such as a network model manager, geographic information system, energy management system (EMS), distribution management system, or outage management system (OMS); to transmit current operations information from a control system to a market system; or to pass data from a protected enclave at a control center to a server on a business network for non-real-time use.

UUDEX could also be leveraged to facilitate communications between a market operator and market participants. Although each market often establishes its own data communications protocols, market information could be exchanged using the same interfaces as for other UUDEX data exchanges.

The focus of UUDEX is an information exchange mechanism, not a mechanism for issuing control commands. Therefore, UUDEX is not designed to communicate control instructions to field devices (e.g., distributed energy resources) or other locations (e.g., peer control centers). In general, the use of UUDEX for direct communications to end devices in the field is beyond the scope of this document.

For organizations in the electricity sector, Figure 1 shows a high-level overview of the potential uses of UUDEX between electric sector organizations described by the North American

Electrical Reliability Corporation (NERC) functional model.[1] In the figure, the solid lines show logical communications that take place between the control centers or centralized control systems of various functional organizations that are within scope of UUDEX, while the dashed lines show communications between some functional organization's control centers and field devices that are out of scope for UUDEX. Not all communications interactions are shown in the figure, but it is clear that a significant number of existing communications interactions could make use of UUDEX. Ultimately, UUDEX could be applied to communications within other energy sectors such as oil and gas delivery systems in a similar manner, but initially UUDEX is focused on the electric sector.

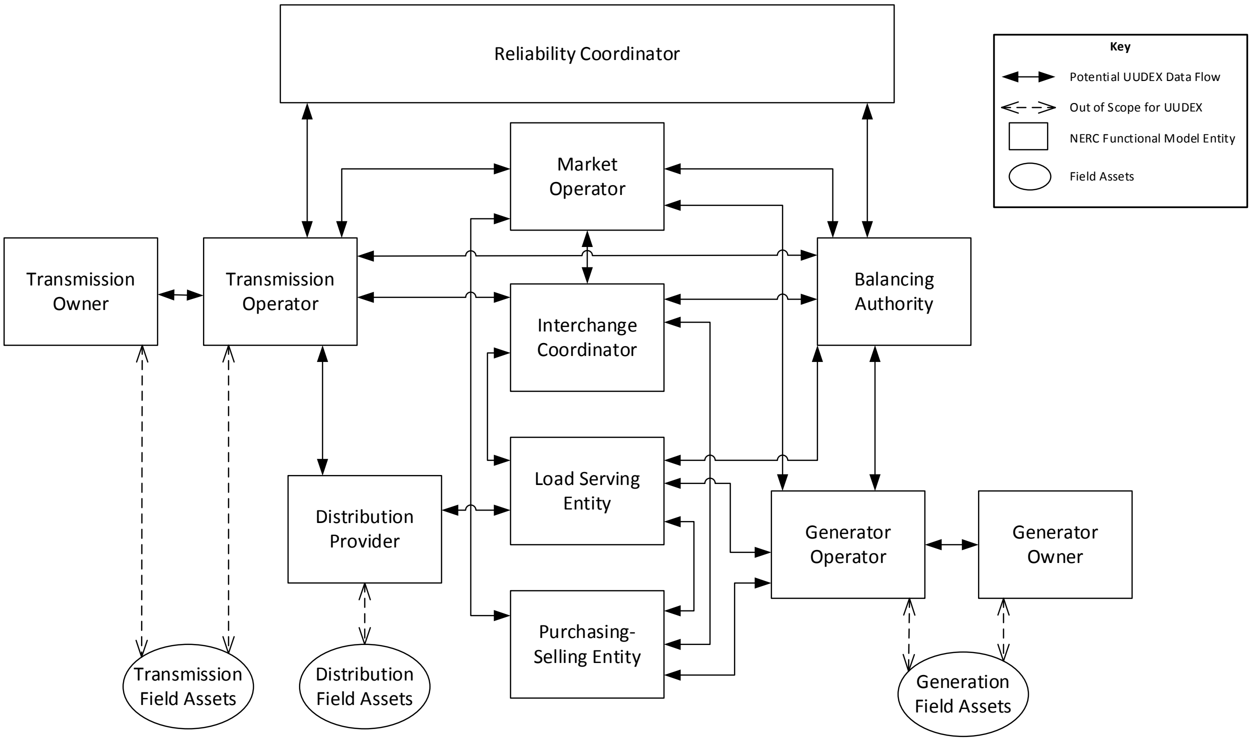

Figure 1: NERC Functional Model Notional Relationships

UUDEX is designed to support the transfer of most any type of data, but especially data that use formats commonly exchanged by control centers. There are some types of information exchanged by control centers that do not have a standardized data format or use a format that is inherently tied to the network protocol used to deliver it. In these cases, UUDEX may define a data model that can convey this information. However, UUDEX itself is not tied to the use of any particular data models and can facilitate the exchange of data that are structured in any way. This is necessary to ensure use of UUDEX will be supportive of future needs to exchange new types of data or by the evolution of how existing data are expressed.

Another key aspect of UUDEX is that communications will be "secure by design." A compliant UUDEX implementation will be deployed with security enabled by default, while still allowing the communications and data exchanges to be diagnosed in the event of errors or data mismatches. The security will provide for integrity and confidentiality of data transmissions. All

[1] See https://www.nerc.com/pa/Stand/Pages/FunctionalModel.aspx

this will be done with minimal disruption to the availability of the links and general flow of information.

In addition to the security of the data in transit, UUDEX will provide access security of the data within the system, allowing UUDEX Clients to specify or agree to how the data can be accessed. UUDEX will provide an extensible set of roles that can be specified in a role-based access control system to grant access to data.

UUDEX is designed to be “transport agnostic,” in that the underlying physical communications infrastructure is largely irrelevant to how it works. UUDEX can be implemented using utility-owned infrastructure, communications leased from a common carrier, the public internet, or any combination. UUDEX’s secure-by-design philosophy allows a common implementation to simultaneously use any combination of physical infrastructure, subject to the risk appetite and security requirements of the utility organization.

UUDEX is also designed to ensure that necessary communications remain possible even in conditions of network congestion or connectivity loss. UUDEX supports message prioritization schemes that help ensure high-priority data are given preference over lower priority data in the event that congestion precludes transmission of all data. In addition, UUDEX includes support for server and communications redundancy, which can make implementations more resilient. Operators are not required to deploy such redundancy (since redundancy can be costly and not all UUDEX instances will necessarily have the same criticality), but components are required to be able to support redundancy should operators wish to make use of it.

If they wish, organizations can use a single UUDEX instance for all their data exchanges or can segment the UUDEX network so that different instances handle different data. Regardless of this choice, the standardized interfaces and roles of UUDEX will mean that all instances can be administered by common management tools and techniques.

1.2 Coordination with Existing Communications and Initiatives

The UUDEX project will support a number of existing and emerging utility communications infrastructures and technologies, including but not limited to the following:

1. Inter-Control Center Communications Protocol (ICCP) – the ICCP is well established, tracing its roots to the desire for standardized control center communications in the mid-1990s. ICCP uses a subset of the Manufacturing Message Specification protocol to provide messaging and allow control commands to be passed from one organization to another. ICCP has been augmented to include a secure communications option, but has limited use of the secure options, at least in the U.S.
2. DOE Electric Emergency Incident and Disturbance Report Form OE-417 – the DOE OE-417 report form is required to be submitted to DOE following the occurrence of a specific disturbance or incident as outlined on the form. There are currently 24 categories of incidents that trigger submission on the web version of the form and 12 categories on the PDF version. Currently, the form can be submitted either by filling out a web form (preferred) or completing a PDF form and emailing it to DOE.
3. NERC Standard CIP-008 revisions – The Federal Energy Regulatory Commission (FERC) has recently released a directive to revise the reporting requirements in NERC Standard CIP-008 to include specific required fields when reporting cybersecurity incidents.

4. Cybersecurity notifications – many formats already exist for the exchange of cybersecurity-related information, whether it represents vulnerabilities, mitigations, alerts, patches, or firewall ruleset updates.
5. Reliability Coordinator Information System (RCIS) – the RCIS tool is a bulletin-board-like system used by Reliability Coordinators (RCs) to exchange operational and reliability-based information. Currently, RCs maintain a database of postings with limited search capability.
6. Phasor measurement unit (PMU) data – Institute of Electrical and Electronics Engineers (IEEE) standard C37.118.2™ or IEEE Std P2664™ – Raw PMU data represent a challenge for data communications. The streaming data from a PMU consists of samples taken at 30, 60, or 120 samples per second for certain power system variables, contain no logical end to the data stream, and should be delivered with no buffering or latency in order for the data to be used. The current protocol for transporting PMU data is IEEE Std C37.118.2; a new standard for Streaming Telemetry Transport Protocol (STTP) is under development as IEEE Std P2664. Both of these protocols are used to gather data from PMUs in the field (which is not in scope for UUDEX), as well as transfer PMU data between phasor data concentrator nodes at control centers. UUDEX is not designed to support delivery of streaming data, so would not support delivery of real-time PMU data. However, UUDEX could be used to distribute PMU snapshots and aggregated values for use across control systems.
7. Market participants – Since most market operators and many market participants will likely have other uses for a UUDEX communication server, using it for market data communications would appear to make sense.

1.3 UUDEX Architecture Overview

UUDEX is an architecture for the management and distribution of UUDEX Data Elements within a closed community. A UUDEX Data Element can be any data object, including but not limited to documents, images, data sets (e.g., spreadsheets or database snapshots), and other forms of both structured and unstructured data. UUDEX differs from other forms of content dissemination systems in that it is built to be highly secure, with all content confidentiality and integrity protected and access to data closely controlled. UUDEX is also optimized for the types of data exchanged by the energy sector, ensuring that all content is transported with the metadata necessary for it to be understood within that context.

UUDEX employs a client-server architecture for data distribution. This architecture supports both query-response and publish-subscribe interactions. UUDEX Servers store information, publish information to subscribing UUDEX Clients, and respond to requests from UUDEX Clients. UUDEX Servers are intended to be very simple and have strictly limited interfaces, which, in turn, makes them much easier to secure.

UUDEX Clients interact with UUDEX Servers. Specifically, UUDEX Clients can manage subscriptions, publish data, delete data, query data, or replace data on a UUDEX Server. Subscriptions can be defined that either allow for alerts of new data objects or automatic forwarding of new data objects. All of these actions are subject to security policies based on the identity associated with the UUDEX Client requesting the action.

The UUDEX Server will also provide an application programming interface (API) that allows UUDEX Clients to be collocated on the same physical device as the UUDEX Server and interact with it programmatically, rather than through network communications. Data collected by these UUDEX Clients can then be processed, potentially creating new derived UUDEX Data Elements the UUDEX Client can then post on the UUDEX Server. Examples of such processing include down-sampling or expunging unneeded or outdated data. The advantage of the API is to bypass the overhead of using the network UUDEX Client and subscribe interface. The same functionality could be performed using the standard UUDEX Client interface and could reside anywhere in the network. The client API could be leveraged by software products, custom adapters, custom applications, or an Enterprise Service Bus (ESB) integration layer that could integrate a variety of applications.

1.4 UUDEX Data and Data Lifecycle

UUDEX supports a wide variety of data object types, where categorizations can include but are not limited to operational status information, incident and other reporting, security alerts and materials, and even general communications and messaging. A data object conveyed using UUDEX is called a UUDEX Data Element.

The following sequence is descriptive of the lifecycle of UUDEX Data Elements:

1. Some entity creates the UUDEX Data Element. UUDEX Data Elements might be automatically generated by sensors or other tools, or they might be manually created by parties filling out forms or writing messages. The entity that creates the data is its Producer.
2. The Producer uses their UUDEX Client to send the UUDEX Data Element to one or more UUDEX Servers. In addition to the data, the UUDEX Client will add a UUDEX Data Envelope that includes handling instructions and other metadata. Among other things, the data envelope specifies who is allowed to view the data and how the data are to be organized on the UUDEX Server.
3. The UUDEX Server receives the message from the UUDEX Client and verifies that the UUDEX Client is allowed to add the given data to a UUDEX Repository. The UUDEX Server then responds to the UUDEX Client, noting that the data were added or rejected.
4. Assuming the data are accepted, the data are added to a UUDEX Repository on the UUDEX Server. A UUDEX Repository is simply a collection of data that is treated similarly. A UUDEX Server might host many UUDEX Repositories. The UUDEX Server will continue to store the data until instructed to delete them.
5. At the time at which the data are added, the UUDEX Server will compare the data to the patterns provided in the subscriptions that UUDEX Clients have established on that UUDEX Server. For each matching subscription, the UUDEX Server will queue a copy of the UUDEX Data Element for delivery to the UUDEX Client that established that subscription.
6. Sometime later, a different UUDEX Client might query the UUDEX Server to learn the contents of the UUDEX Data Element’s UUDEX Repository. In this case, the UUDEX Server will send a data manifest, with metadata taken from the data envelope the UUDEX Data Element's Producers provided. The manifest will only include entries for UUDEX Data Elements for which the requesting UUDEX Client has the appropriate access to read a manifest entry.
7. At some point, a UUDEX Client might query the UUDEX Server for data. It could do this either by requesting the specific UUDEX Data Element using its unique identifier (as might be discovered in a manifest) or it could submit a pattern the UUDEX Server compares to its UUDEX Repository entries. Assuming the requesting UUDEX Client has read access to the UUDEX Data Element, the UUDEX Server will send the UUDEX Data Element to the UUDEX Client in response to its request.

8. At some point, the UUDEX Data Element’s Producer (or other authorized party) might use a UUDEX Client request to submit a modification of the UUDEX Data Element to the UUDEX Server. In this case, the original UUDEX Data Element will be deleted and the new UUDEX Data Element placed in the UUDEX Repository.
9. Eventually, the UUDEX Data Element’s Producer (or other authorized party) might use a UUDEX Client request to instruct the UUDEX Server to delete the UUDEX Data Element from its UUDEX Repository.

2.0 Use Cases Supported by UUDEX

This section describes a set of representative use cases that are applicable to UUDEX. This in no way implies specific limitations on the use of UUDEX. Where the primary purpose of a use case is to answer the question ‘WHO does WHAT to WHO, WHEN, and WHY do they do it?’ It is not the intent to describe ‘HOW’ this is achieved, as this would be described by a design that depicts the underlying technical infrastructure. In the use case discussion, the term “Electric Entity” is used in a general case to represent a generic UUDEX Client or Server organization that plays the role of an actor in the use cases. There is no explicit limitation or implication on the set of allowable actors.

In a real UUDEX environment the UUDEX Servers would be implemented in a redundant mode, with multiple instances of a given UUDEX Server and replicated data between them. In UUDEX Servers with a high volume of use, this would allow load balancing and greater efficiency. In any case, such redundancy would reduce the impact of server failure, as backup servers could immediately take over if the primary server became unavailable. However, to make the use case drawings and descriptions easily understood, none of these redundant considerations are included in the discussions in this document. In other words, the diagrams provided are intended to depict key roles, rather than a deployment architecture.

Similarly, the diagrams show separate UUDEX Producers and UUDEX Consumers. These are different behaviors of the UUDEX Client role and in most cases would be performed by the same piece of software. However, the diagrams show these behaviors as separate entities to make the data flows clearer.

The UUDEX Server only performs actions in response to UUDEX Client direction and only the specific actions UUDEX Clients are allowed to request. This means that UUDEX Servers are not allowed to perform any additional processing of received data, beyond what is necessary to store it in a UUDEX Repository and any processing necessary to efficiently serve queries and subscription requests. For example, UUDEX Servers might sort and index UUDEX Data Elements; they will not, however, alter these elements. If there is a desire to create derived UUDEX Data Elements from UUDEX Data Elements on a UUDEX Server (e.g., to down-sample certain UUDEX Data Elements, or create aggregated UUDEX Data Elements based on multiple other elements), then this will need to be done by having a UUDEX Client download the relevant data from the Server, perform the necessary derivation, and upload the data back to the UUDEX Server. An API is provided to reduce the overhead of using the normal network interface provided by the UUDEX Client. UUDEX requires that UUDEX Servers support minimal processing to keep them simple and easier to secure.

Figure 2 shows the symbols used in the use case drawings.

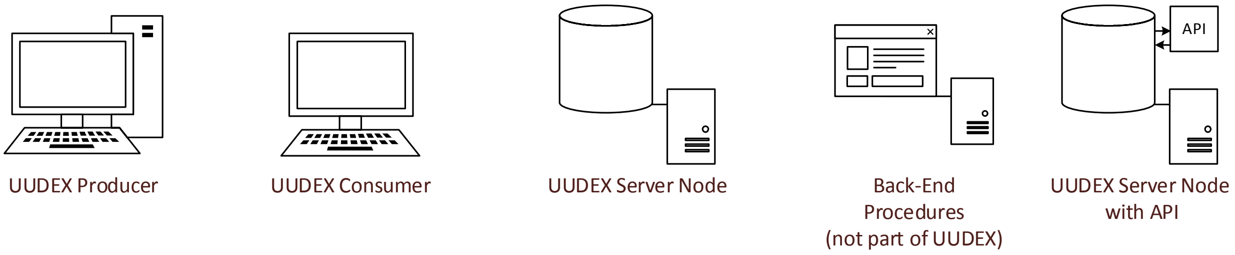
API
UUDEX Producer
UUDEX Consumer
UUDEX Server Node
Back-End Procedures (not part of UUDEX)
UUDEX Server Node with API

Figure 2: Symbols used in Use Case Drawings

2.1 Operational Data Shared between an Electric Entity and a Reliability Coordinator

The use case drawn in Figure 3 represents sharing of information to a third party who is implicitly trusted to protect the data from inadvertent disclosure.

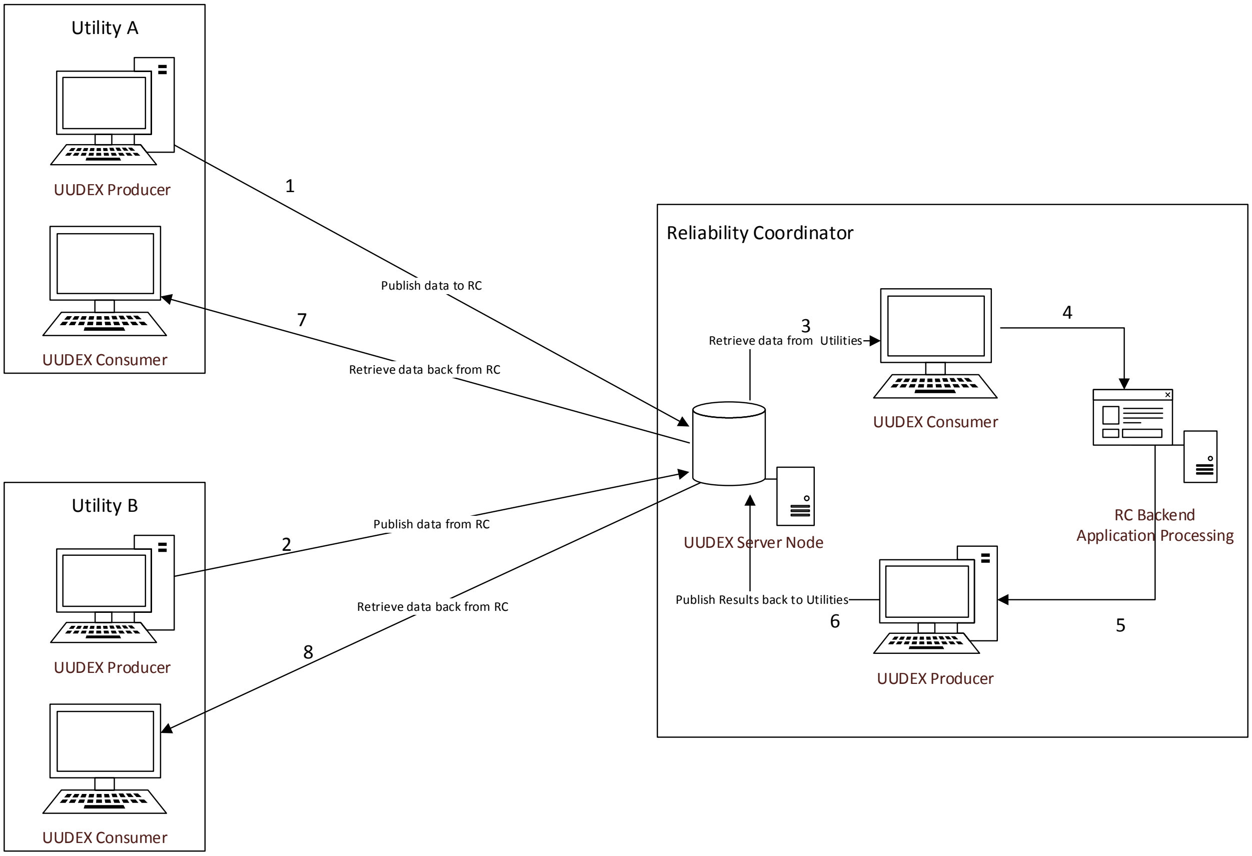

Figure 3: Sharing Data with a Reliability Coordinator

This use case involves the following activities:

- The RC will establish a UUDEX Server with sufficient storage and communications bandwidth to handle connections (primary and backup) from all of its Electric Entity constituents (for example, a Transmission Operator [TOP] or Balancing Authority [BA]), noted in the drawing as “Utility A” and “Utility B.”
- Electric Entities will periodically provide data to the UUDEX Server using UUDEX Client to send data objects to the UUDEX Server where they will be stored while being requested (1, 2).
- The RC’s UUDEX Client will retrieve the data from the UUDEX Server (likely a single request to gather data from all Electric Entities) (3).
- The RC will process the data and prepare new or modified UUDEX Data Elements based on the provided submissions to be sent back to the Electric Entities (4, 5).
- The RC’s UUDEX Client will send data to the UUDEX Server for publication to the Electric Entities (6).

- The Electric Entity Client will request published data from the UUDEX Server, obtaining only the data that the Client is authorized to receive (7, 8).

2.2 Operational Data Shared Between Two Electric Entities

The use case drawn in Figure 4 represents sharing of data between peers where limited trust exists between those peers, particular with regard to how data might be redistributed or used.

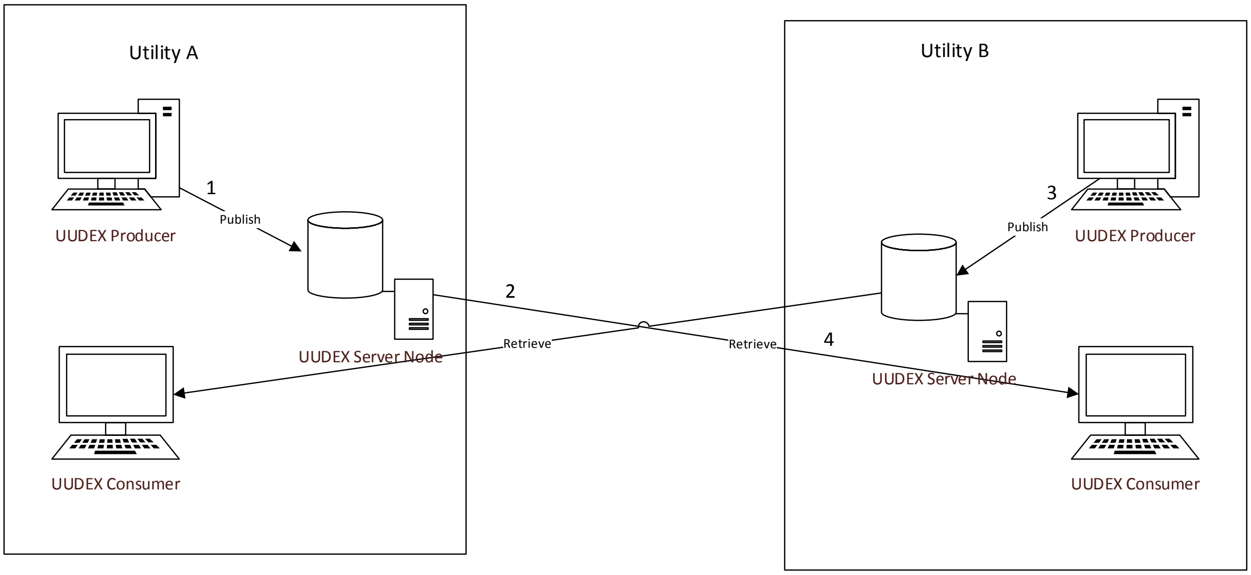

Figure 4: Data Sharing Between Two Electric Entities

In this case, each Electric Entity has control over what UUDEX Data Elements other Electric Entities can discover or read. This is done through administrative configurations and access controls associated with the Data Elements on the UUDEX Server node.

- Each Electric Entity (noted in the drawing as “Utility A” and “Utility B”), for example two transmission operators, will establish a UUDEX Server that will contain data available for sharing to other Electric Entities.
- When one Electric Entity (henceforth, “the originator”) wishes to convey information to other Electric Entities (henceforth, “the recipients”), the originator’s UUDEX Client will publish data to its own UUDEX Server using the UUDEX Client (1, 3).
- The UUDEX Client at each recipient will request data from the originator’s UUDEX Server using a UUDEX Client, only receiving data that it is authorized to receive, and store it in a local database (2, 4). Storing the data in a local database eliminates the need to go back to the data publisher's UUDEX Server every time the data are needed in the future.
- Multiple logical point-to-point (bilateral) links from different Electric Entities can be established to each UUDEX Server.

2.3 Operational Data Shared between Two Control Centers of the Same Electric Entity

The use case drawn in Figure 5 represents sharing between locations within the same organization, for example, the primary control center and backup control center of a transmission operator.

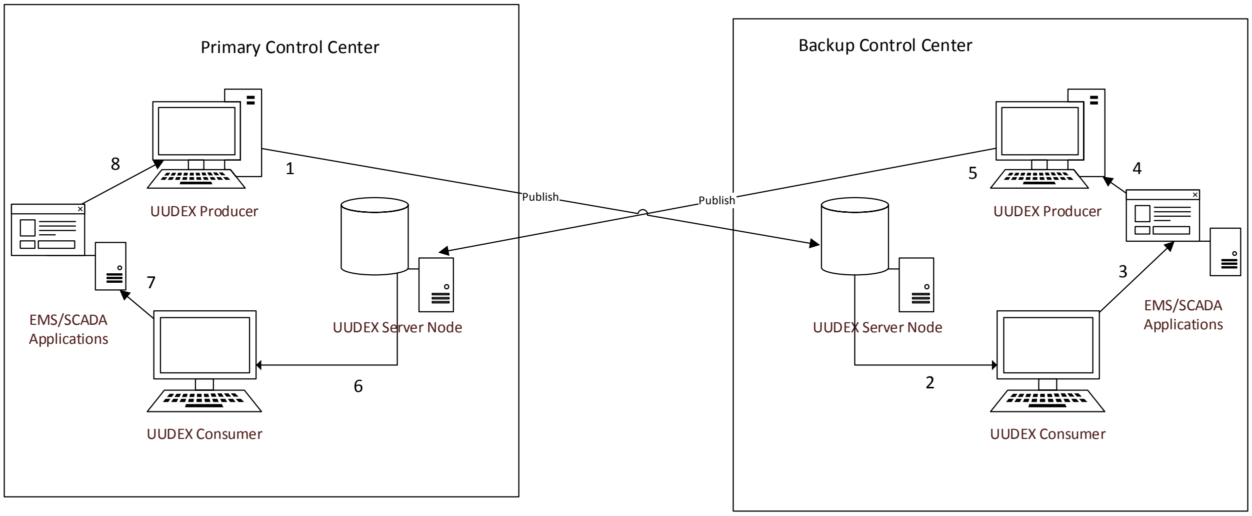

Figure 5: Data Sharing Between a Primary and Backup Control Center

This case looks very similar to the one for two different Electricity Entities; however, in this case, the publish client sends data directly to the “other” UUDEX Server node rather than waiting for the other center to pull data from the primary control center's UUDEX Server. This results in more immediate replication of data between control centers.

- Each control center will establish a UUDEX Server to contain data that will be shared or replicated to the other control centers.
- The operational primary control center will extract data from its local database (8) and use the UUDEX Client to publish data to the UUDEX Server at the backup (1).
- The operational backup will use the UUDEX Client to request the data from the UUDEX Server (2) and store it in its local database (3).

When the backup control center needs to send data to the primary the steps are the same:

- The backup control center (now acting as a primary) will extract data from its local database (4) and use the UUDEX Client to publish data to the UUDEX Server at the primary control center (now acting as a backup) (5).
- The old primary will use the UUDEX Client to request the data from the UUDEX Server (6) and store it in its local database (7).

2.4 Simple Data Manipulation Using the UUDEX API

The use case drawn in Figure 6 represents the functionality of using the API, allowing simple calculations to be performed on the UUDEX Server outside of the normal processing provided by the UUDEX Server functionality.

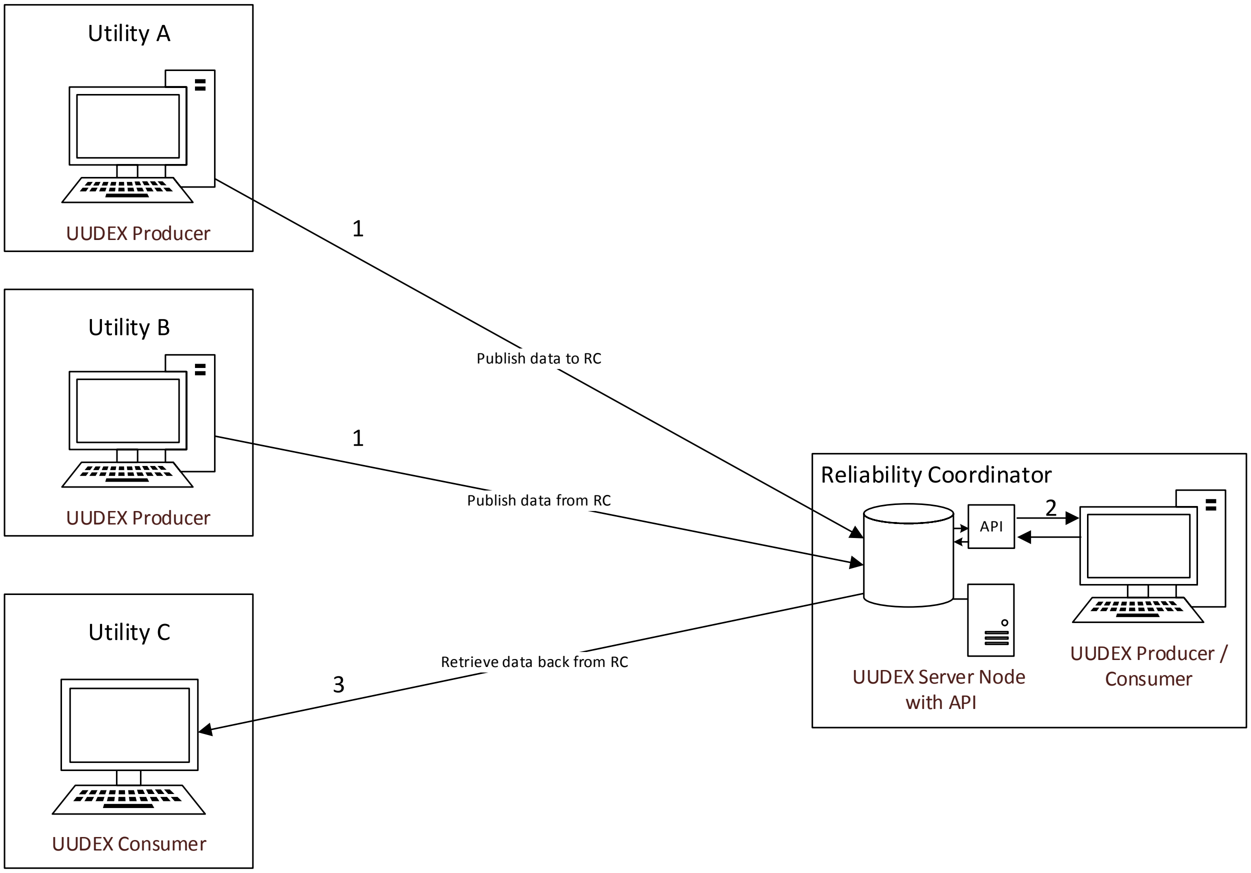

Figure 6: Simple Data Manipulation using the API

In this example, two Electric Entities (noted as “Utility A” and “Utility B” in the drawing) provide data to the RC, while a third (noted as “Utility C” in the drawing) receives the augmented data. These augmented data are produced by the RC. Specifically, the RC has a UUDEX Client that collects information published to its UUDEX Server (specifically, the data published by Utility A and B), uses these data to created augmented content, and then publishes the data back to its UUDEX Server where Utility C can collect it. While this could be done with a separate UUDEX Client over network protocols, the RC has installed the UUDEX Client on the same device as its UUDEX Server and interfaces with the UUDEX Server using an API rather than using network connections.

An example use of this would be for the RC to receive individual generation output values from generator operators and make a “total generation” value available without revealing the individual values for each generator. Other uses could be calculating a running 5-minute average of an analog value, or down-sampling PMU or supervisory control and data acquisition (SCADA).

- The RC will establish a UUDEX Server to receive data from Electric Entities.
- The RC has installed a UUDEX Client on the same device as the UUDEX Server. This UUDEX Client is attached to a custom application that takes the data retrieved by the UUDEX Client, performs calculations on them (such as summing the information), and then uses the UUDEX Client to publish the data back to the UUDEX Server. Because both the UUDEX Server and Client are on the same devices, the interactions between them are handled by an API rather than network communications.

- Individual Electric Entities use their UUDEX Client to publish data to the RC's UUDEX Server (1).
- The UUDEX Client application at the RC periodically uses the API to retrieve data from the UUDEX Server’s Repository, perform calculations on the data, and uses the API to publish the data back as a different UUDEX Data Element to the UUDEX Server (2). Execution of the custom application could be based on an internal timer or be event driven based on the UUDEX Client creating a subscription to be alerted whenever data are published to the UUDEX Server. As both the UUDEX Client and UUDEX Server share the same device, all of this interaction occurs without external network communications.
- Electric Entities can retrieve the calculated data from the Reliability Coordinator's UUDEX Server (3).

There are multiple ways the API capabilities of the UUDEX Server might be used:

- A commercial software product that uses the UUDEX Client API directly
- A custom application or adapter that uses the UUDEX Client API
- An ESB integration layer used to integrate one or more applications (e.g., EMS, OMS, distribution management system, distributed energy resource management system) where the ESB leverages the UUDEX Client API for communication with UUDEX Server(s).

2.5 Security Event Data Shared between an Electric Entity and the E-ISAC

The use case drawn in Figure 7 represents sharing of security event data from an Electric Entity to the E-ISAC and from the E-ISAC back to one or more Electric Entities noted in the drawing as “Utility.”

In addition to sharing the data with the E-ISAC, since data sent to the E-ISAC might be sensitive, there is a need to minimize the amount of time that the data are potentially exposed for inadvertent access on the E-ISAC’s UUDEX Server.

- The E-ISAC will establish a UUDEX Server to receive security event data.
- The E-ISAC UUDEX Client will issue a subscribe request to be notified when security event data are published to the UUDEX Server (3).
- An Electric Entity participant will issue a subscribe request to be notified when the E-ISAC publishes a security notice to the UUDEX Server (1).
- An Electric Entity participant publishes a security event to the E-ISAC’s UUDEX Server (2).

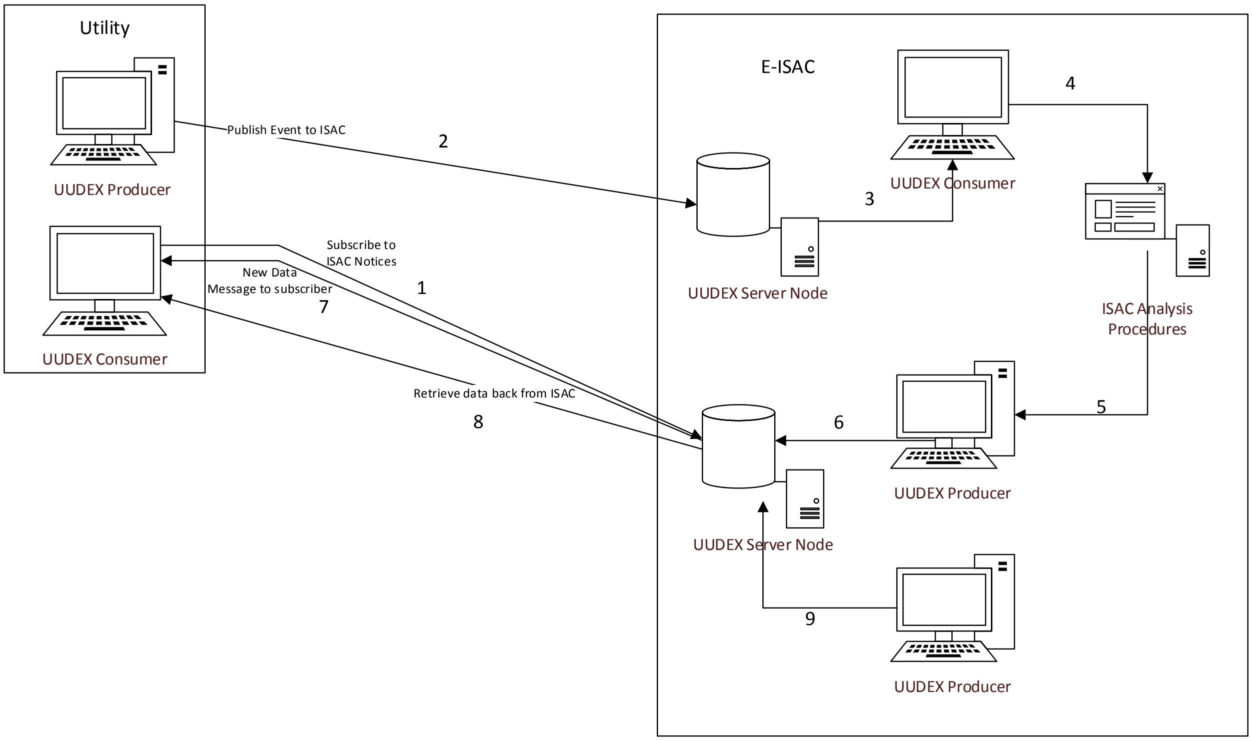

Figure 7: Data Sharing with the E-ISAC

- The E-ISAC UUDEX Server sends a notification to the E-ISAC UUDEX Client that new data have been added (3).
- The E-ISAC UUDEX Client requests the security event from its UUDEX Server and stores it locally (3, 4).
- The E-ISAC UUDEX Client sends a delete data request to remove the security event from the UUDEX Server's Repository. This minimizes the time that the security information is exposed in a network-visible device (3).
- The E-ISAC develops an industry alert and uses its UUDEX Client to publish the alert to its UUDEX Server (5, 6, 9).
- The E-ISAC UUDEX Server sends a notification to each subscribing Electric Entity that an alert has been published (7).
- Each utility UUDEX Client requests the alert if is relevant to their operations (8).

2.6 Power System Model Updates Published by an RC

The use case drawn in Figure 8 represents how large data sets (files) can be shared using an update notification that does not contain all the data to be shared.

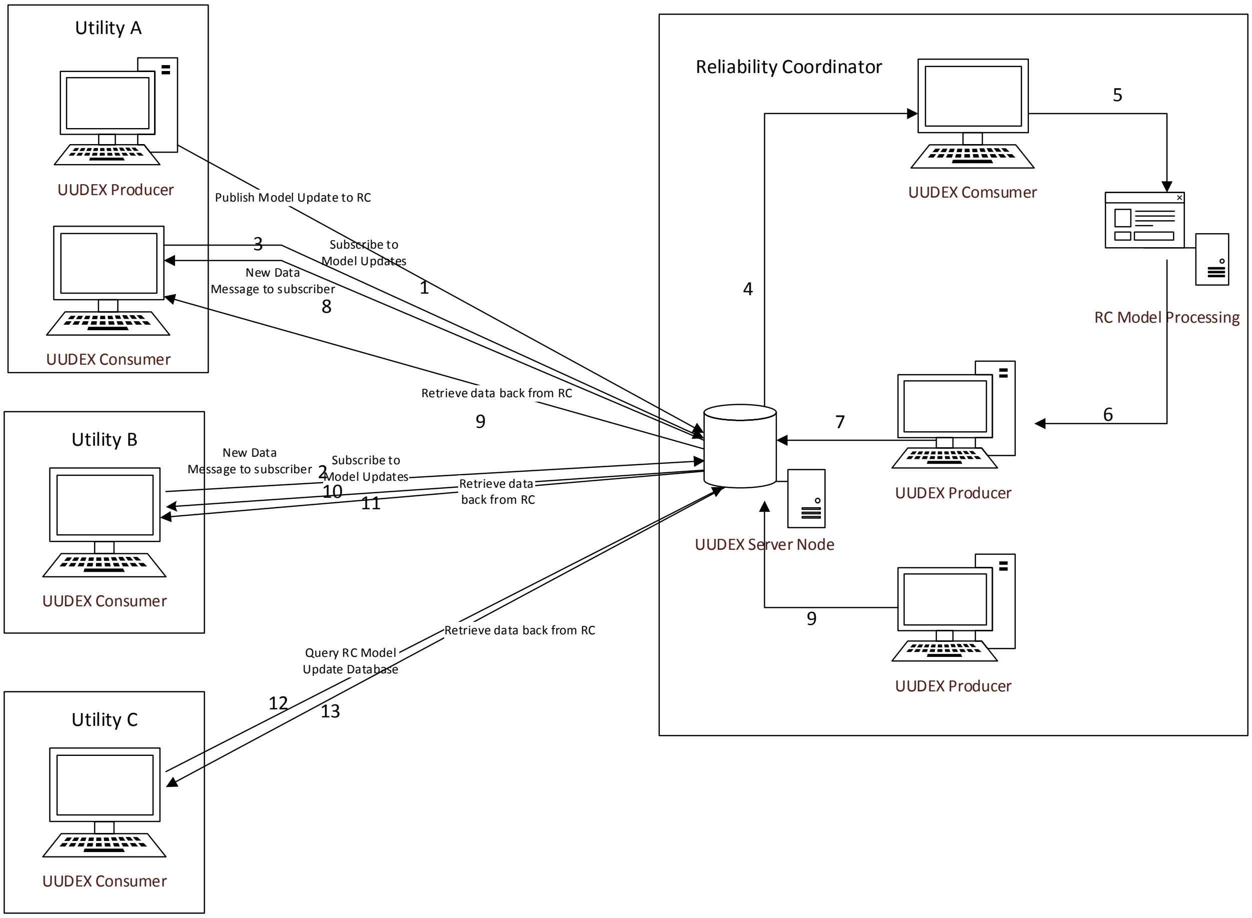
Utility A
UUDEX Producer
UUDEX Consumer
Publish Model Update to RC
3
Subscribe to Model Updates
New Data Message to subscriber
8
1
Retrieve data back from RC
9
Utility B
UUDEX Consumer
New Data Message to subscriber
2
Subscribe to Model Updates
10
11
Retrieve data back from RC
Utility C
UUDEX Consumer
Query RC Model Update Database
Retrieve data back from RC
12
13
Reliability Coordinator
UUDEX Comsumer
5
4
RC Model Processing
6
7
UUDEX Producer
UUDEX Server Node
9
UUDEX Producer

Figure 8: Power System Model Update Processing

The use case also discusses how a directory listing of available large files could be made accessible in the event that obtaining a particular file is necessary.

- A central Electricity Entity (for example, an RC) will establish a UUDEX Server for sharing power system model updates (multiple power system model update versions will be supported).
- Electric Entities may subscribe to power system model updates it is interested in (1, 2).
- An Electric Entity will send power system model updates to the RC UUDEX Server (3).
- The RC's UUDEX Client will retrieve the power system model updates and process them to produce an updated power system model (4, 5, 6).
- The RC's UUDEX Client will publish power system model updates back to its UUDEX Server (7).
- The RC's UUDEX Server will send an “update notification” message to each Electric Entity UUDEX Client that has established a matching subscription (8, 10). The subscription response contains some identifier that uniquely identifies the model file that was just published.
- Subscribing Electric Entities will receive the subscription responses and determine if the power system model update should be requested.

- Electric Entities will use their UUDEX Clients to download power system model files applicable to them (9, 11).

Alternately:

- Any time an Electric Entity wishes to receive an updated power system model, the Electric Entity may query the RC's UUDEX Server to see what power system model updates are available (12).
- The Electric Entity may use their UUDEX Client to retrieve power system model updates of interest to them (13).
- In this case, the retrieving party collects new model information based on their own timeline, rather than responding to notifications from a central UUDEX Server.

2.7 Patch Updates Published by the E-ISAC

The use case drawn in Figure 9 shows how UUDEX could be used by the E-ISAC, or another organization, to disseminate software patch notifications.

This use case is similar to the model updates but patches may contain additional fields available for query (e.g., equipment type, version, patch metadata) that may also need to be communicated in the notification data. Such fields would be necessary for recipients to determine whether the patch is relevant to their local systems (i.e., are they running the software that the patch fixes?). Fields available for query will need to be agreed upon by all participants, as would the appropriate values for those fields. The latter would be necessary to avoid otherwise synonymous values producing different results (e.g., “Windows 10” vs. “Win10”).

- The E-ISAC will establish a UUDEX Server.
  - The UUDEX Server’s Repository will be configured to contain records for multiple types of patches.
  - The UUDEX Server’s Repository will be able to be queried by UUDEX Consumers.
- Organizations can use UUDEX to request notification whenever a new patch matching specified criteria is added to the E-ISAC’s UUDEX Server (1, 2, 3).
  - Utility A indicates specifically which patches it is interested in (1).
  - Utility B and Utility C do not indicate which patches they are specifically interested in but request notification when any patch for any software is added (2,3). (While this will result in many alerts about irrelevant patches, this might be desired because it means the software the utilities use is not exposed to the server in their subscriptions.)
  - Utility D does not request to be notified when patches are made available.
- The E-ISAC will use the UUDEX Client to publish patch data to a UUDEX Repository on the UUDEX Server (4) that supports query requests.
- The E-ISAC’s UUDEX Server will send notifications to organizations that have subscriptions that match the newly added patch data.
  - Utility A receives the subscription notification (5) and requests the patch file be downloaded (6).

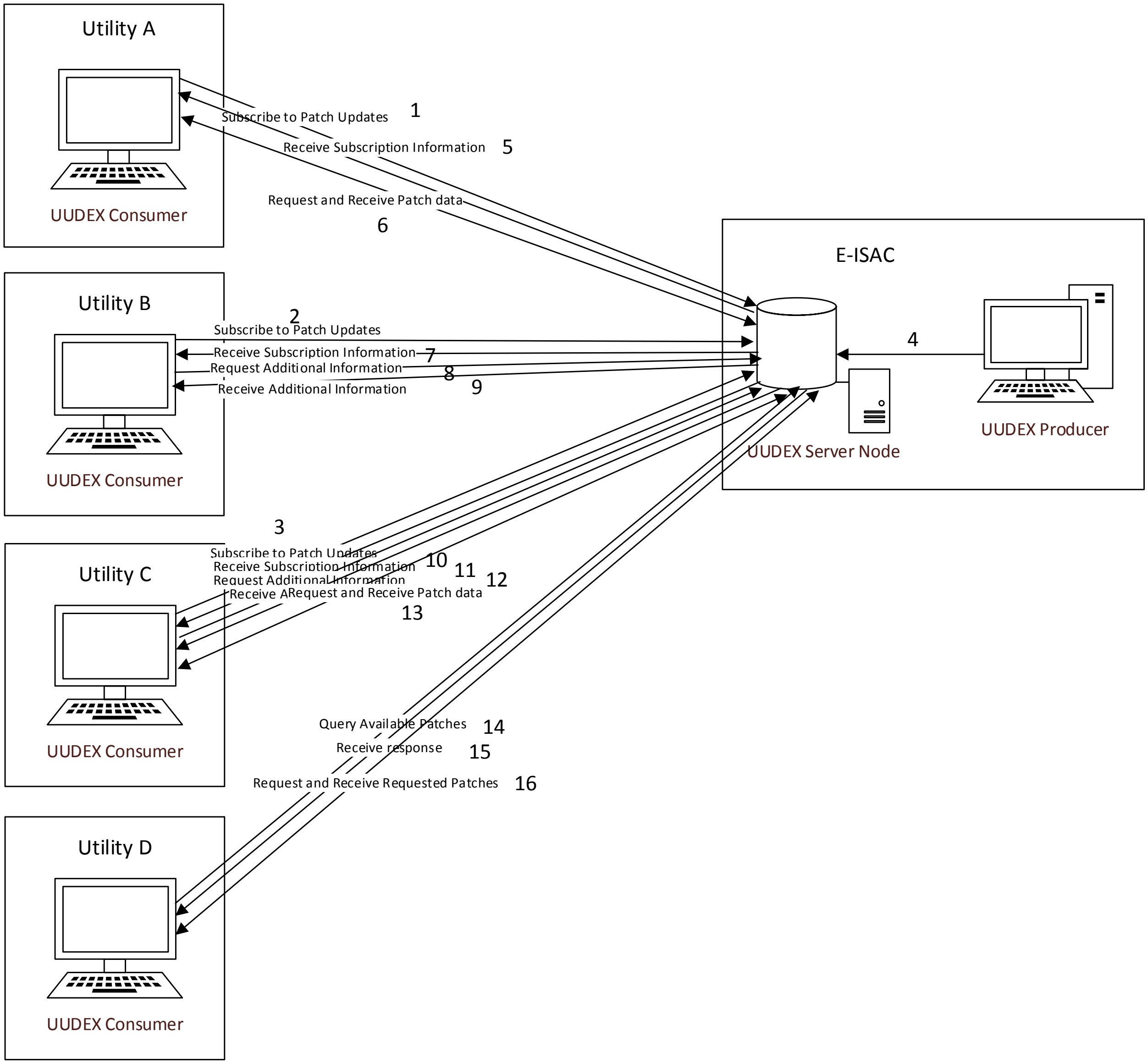
Utility A
UUDEX Consumer
Subscribe to Patch Updates
1
Receive Subscription Information
5
Request and Receive Patch data
6
E-ISAC
4
UUDEX Server Node
UUDEX Producer
Utility B
UUDEX Consumer
2
Subscribe to Patch Updates
Receive Subscription Information
7
Request Additional Information
8
Receive Additional Information
9
3
Subscribe to Patch Updates
Receive Subscription Information
Request Additional Information
Receive A
Request and Receive Patch data
10
11
12
13
Utility C
UUDEX Consumer
Query Available Patches
14
Receive response
15
Request and Receive Requested Patches
16
Utility D
UUDEX Consumer

Figure 9: Patch Updates Published by the E-ISAC

- Utility B and C receive the subscription notification (7) and request the patch file, understanding that many might be discarded as irrelevant. However, neither the server nor any party observing their communications would be able to determine what software Utility B or Utility C were using.
- Utility D does not subscribe to patch update notifications. Rather, it periodically queries the UUDEX Server's UUDEX Repository for a list of available patches (14). The UUDEX Server responds with the detailed list of patches available (15). Utility D determines which patches are necessary, and requests and downloads the desired patches (16).

2.8 RCIS Messaging

The use case drawn in Figure 10 is for a remote database update or query capability as used by the RCIS.

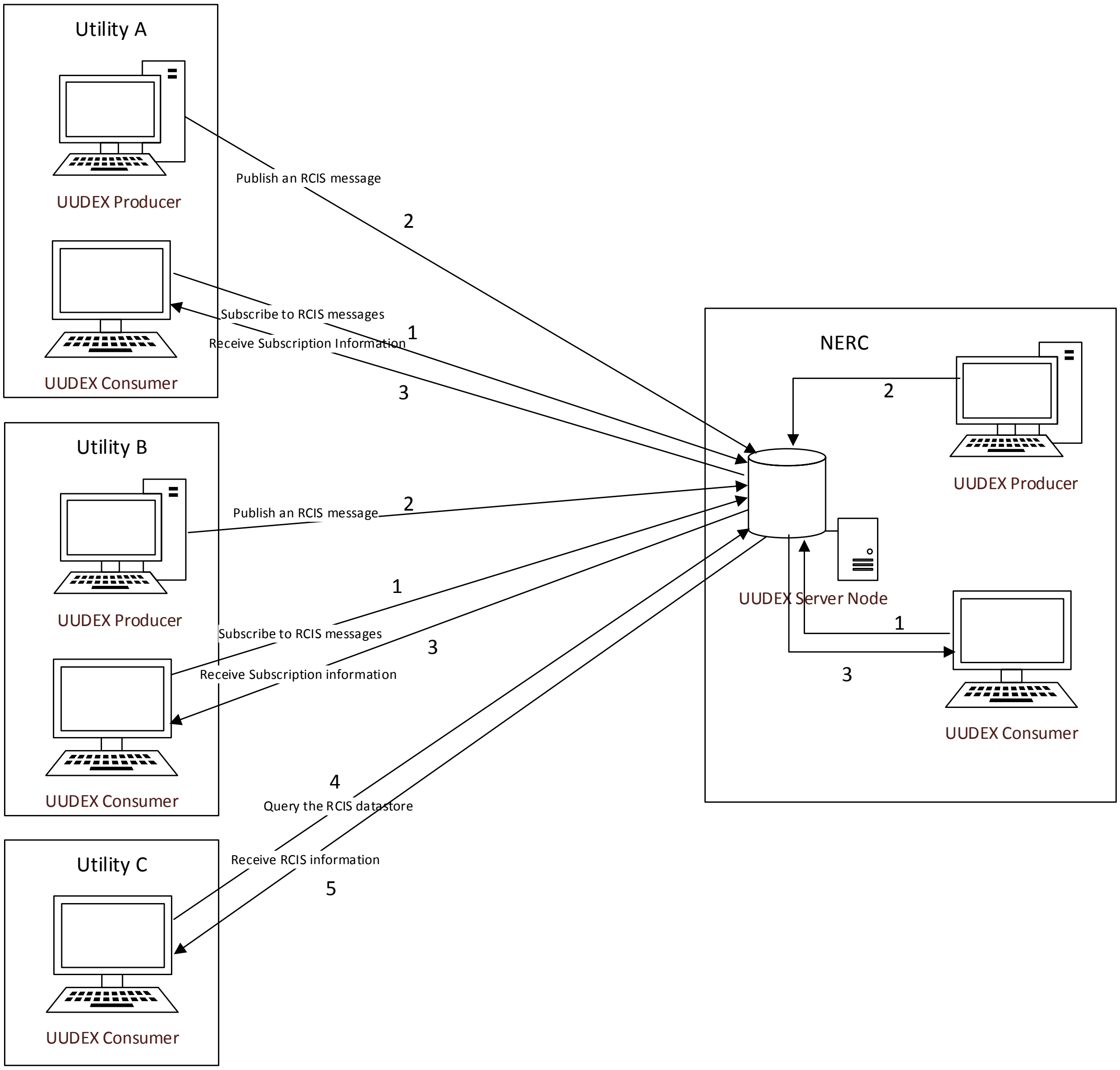

Figure 10: RCIS Messages

- An organization (like NERC or the Eastern Interconnection Data Sharing Network) will establish a UUDEX Server. The UUDEX Server will contain a database of messages available for query (referred to as the UUDEX Server's Repository).
- Electric Entities or NERC will register a subscription request for RCIS message types they are interested in (1).
- Electric Entities or NERC will use the UUDEX Client to publish data to the UUDEX Server and store it in the UUDEX Server's Repository (2).
- The UUDEX Server will send the published data to subscribers who have requested updates (3).

- It is conceivable that the UUDEX Producer and UUDEX Consumer could be the same program at the utility for some RCIS messages, while they could be separate programs for other RCIS messages.

Alternately:

- Organizations can use the UUDEX Client to query for information from the UUDEX Server's RCIS Repository on an ad-hoc basis (4).
- The UUDEX Server will respond with the requested data from the RCIS UUDEX Repository and return it to the requesting UUDEX Client (assuming the UUDEX Client is granted access to that data) (5).

2.9 DOE OE-417 Reporting

The use case drawn in Figure 11 represents a mechanism for sending the same information (in this case incident reports) to multiple organizations.

Different organizations will host separate UUDEX Servers for political and jurisdictional reasons. The organizations receiving reports are noted as "Report Recipient Organizations" (and are the DOE, E-ISAC, NERC Bulk Power System Awareness, and an RC in this example). In this use case, the Electricity Entity (noted as "Utility" in the drawing) will need to establish a separate link to the UUDEX Server at each Report Receiving Organization.

- Each Report Recipient Organization will establish a UUDEX Server. These parties will instruct organizations that need to file reports and other information with them to use these UUDEX Servers to submit these data.
- Electric Entities will generate OE-417 report data and use their UUDEX Client to publish the data to the UUDEX Server at each Report Recipient Organization.
- Each organization will have their own UUDEX Client that is subscribed to their UUDEX Server. Whenever new information (i.e., a new OE-417 report) is posted to their server, the Client is automatically notified and pulls the report down. Once retrieved by the Client, the report can be processed as necessary. In the case where the OE-417 data were expressed using a standardized data model, the first step in this processing will likely to be to generate an actual OE-417 form from the data.

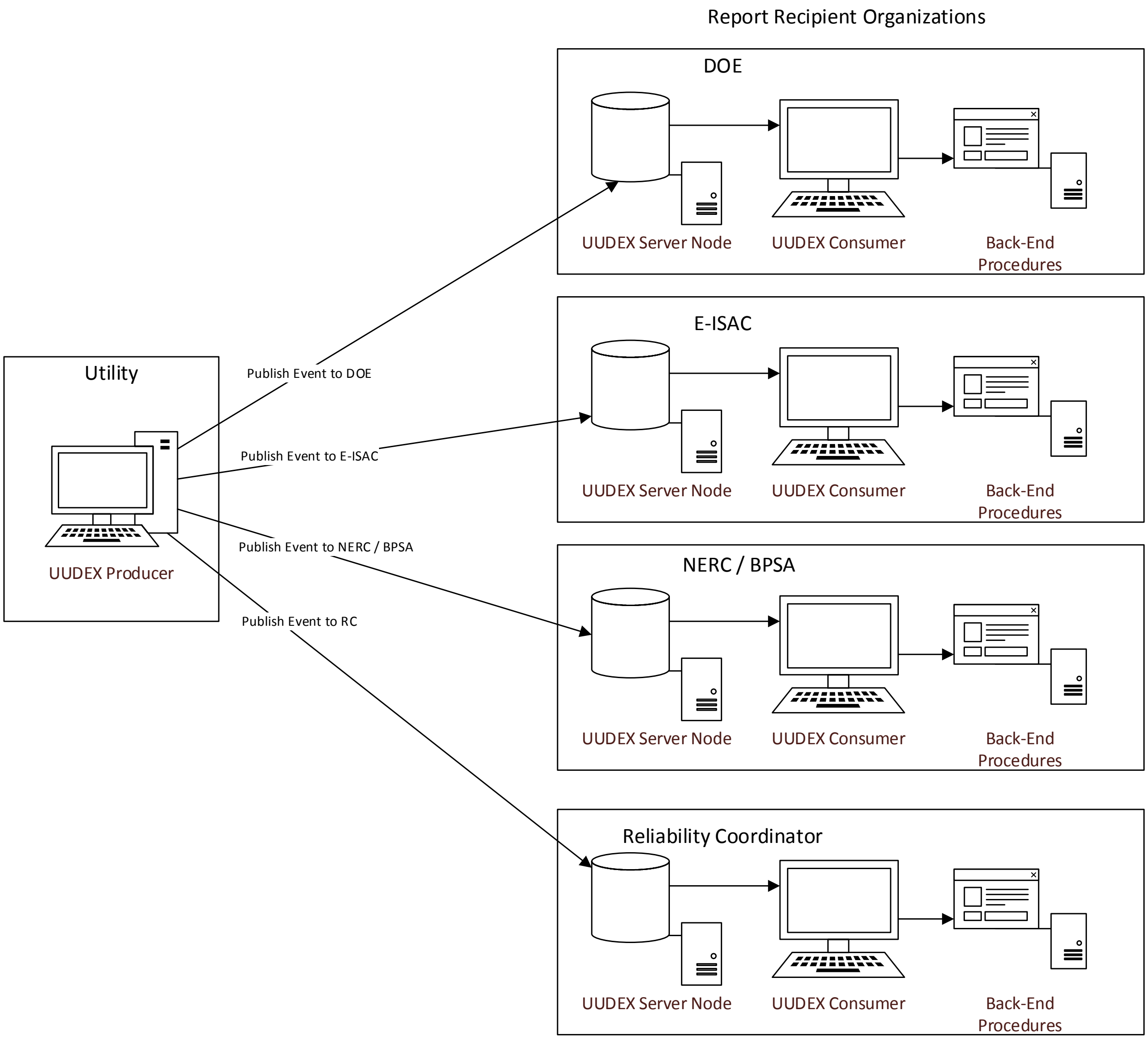

Figure 11: DOE OE-417 Reporting

2.10 Emergency Responder Information

The use case drawn in Figure 12 represents how an existing UUDEX Server could rapidly be configured to provide specific information to a new type of organization.

The prime example of such a goal is the need to respond to a natural disaster or other crisis. In such a case, emergency responders (state, federal, local, or non-governmental organization) might be granted specific, limited information about the current and anticipated state of the power grid to plan their efforts.

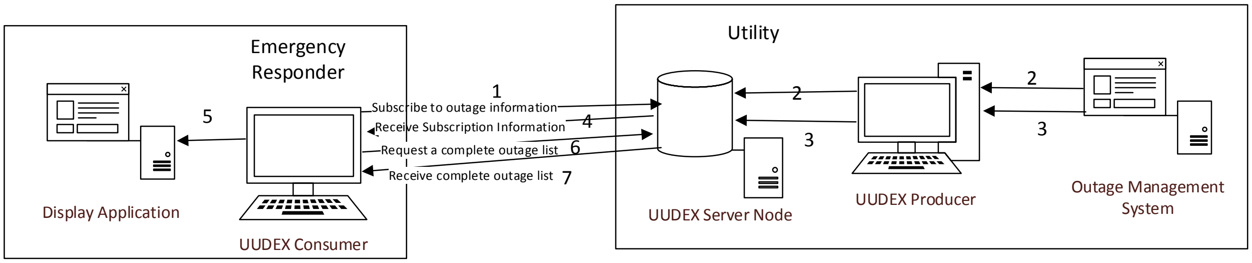

Figure 12: Emergency Responder

For example, following a natural disaster, police, fire, and government response organizations need to know which portions of the electric grid are energized, which are de-energized, and the approximate order of restoration. Traditionally this information is available by telephone or email correspondence on an infrequent basis. By using a UUDEX Server that can be queried for current outage information and updated in near real time by an Electricity Entity's OMS, the information can be provided in near real time to the first responder with no processing or impact to the Electricity Entity other than establishing the initial link for the emergency responder.

NOTE: the software used by the emergency responders to display the outage information is beyond the scope of the UUDEX project.

- An Electric Entity (for example a Transmission operator or distribution operator) will establish a UUDEX Server. Alternately, they might grant access to outage information already present on an existing UUDEX Server.
- Emergency responders can use a UUDEX Client to subscribe to updates to the outage information (1).
- The Electric Entity will extract information from its OMS and publish outage information (in standardized format) to the UUDEX Server (2).
- The Electric Entity will extract information from its OMS and update outage information as outages are restored to the UUDEX Server (2).
- After a period of time, the Electric Entity may delete the outage information from the UUDEX Server (3).
- The UUDEX Server will send updates to emergency responders in fulfillment of their subscriptions (4). Once downloaded, an application program at the emergency responder location can process and display the outage information to facilitate emergency operations (5).
- Emergency responders may use the UUDEX Client to request information for which they have not previously established subscriptions as necessary (6, 7).

2.11 Mass Alert

The use case drawn in Figure 13 represents the situation where a UUDEX Server's data include information that needs to be distributed to a broad audience quickly.

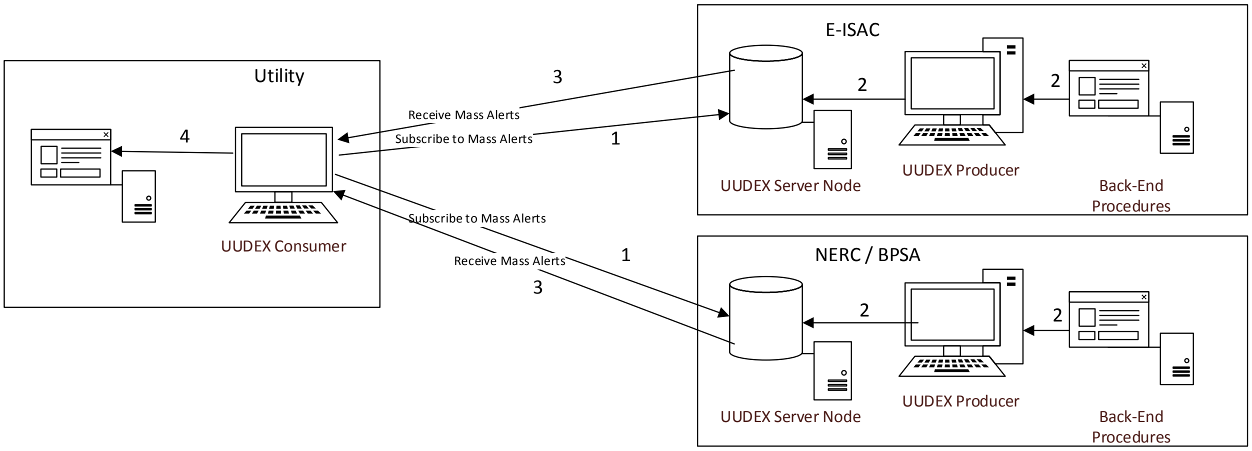

Figure 13: Mass Alert

Examples of this could include alerts from the E-ISAC regarding an active cyberattack campaign for which a broad group of UUDEX participants should take action. While the E-ISAC and NERC Bulk Power System Awareness are shown in this example, other organizations, such as DOE, U.S. Department of Homeland Security (DHS), or an RC, could use the same process to send alert information to a large number of organizations.

- The party originating the alerts will stand up a UUDEX Server.
- All parties that should receive the alert will establish subscriptions with the originator's UUDEX Server (1). These subscriptions will request that UUDEX Data Elements of the alert's type and priority be delivered immediately.
- When an alert is generated (2), it is immediately added to the appropriate UUDEX Repository on the originator's UUDEX Server.
- When the alert is added to the UUDEX Server's Repository, it is immediately compared against subscription criteria. This will match all subscriptions for immediate delivery of the alert, leading to fulfillment of the subscriptions (3). Each utility receiving an alert would process it (4).
- When the alert is delivered, UUDEX's reliable message delivery mechanisms will allow the sender to know that the given Client received the message.
- Alert messages could remain in the UUDEX Server's Repository. This would provide archiving of alerts and would allow parties to query for old alerts at a later date.

Variation:

- It might be desirable to have a hierarchical distribution chain, where the originator sends messages to other entities acting as relays, who then pass the message on to other parties. There might be multiple levels in this hierarchy. While this could slightly increase latency, it would simplify subscription management by reducing the number of subscriptions a given party needed to manage and allowing clients to be organized into natural groupings for management.
- In this case, each level of the distribution hierarchy would have their own UUDEX Client and UUDEX Server. Each distribution node's UUDEX Client would subscribe for updates from

the UUDEX Server of the next entity above them in the distribution tree. When alerts are received by this Client, they are immediately posted to that entity's UUDEX Server.

- All children in the distribution tree would have UUDEX Clients who had established subscriptions to this UUDEX Server, which would immediately be fulfilled when the new data are added. This could be repeated through any number of levels in the distribution hierarchy.

2.12 Mass Alert with Required Response

The use case drawn in Figure 14 is a variation of the mass alert use case in section 2.11, where a response from the recipient of an alert is required.

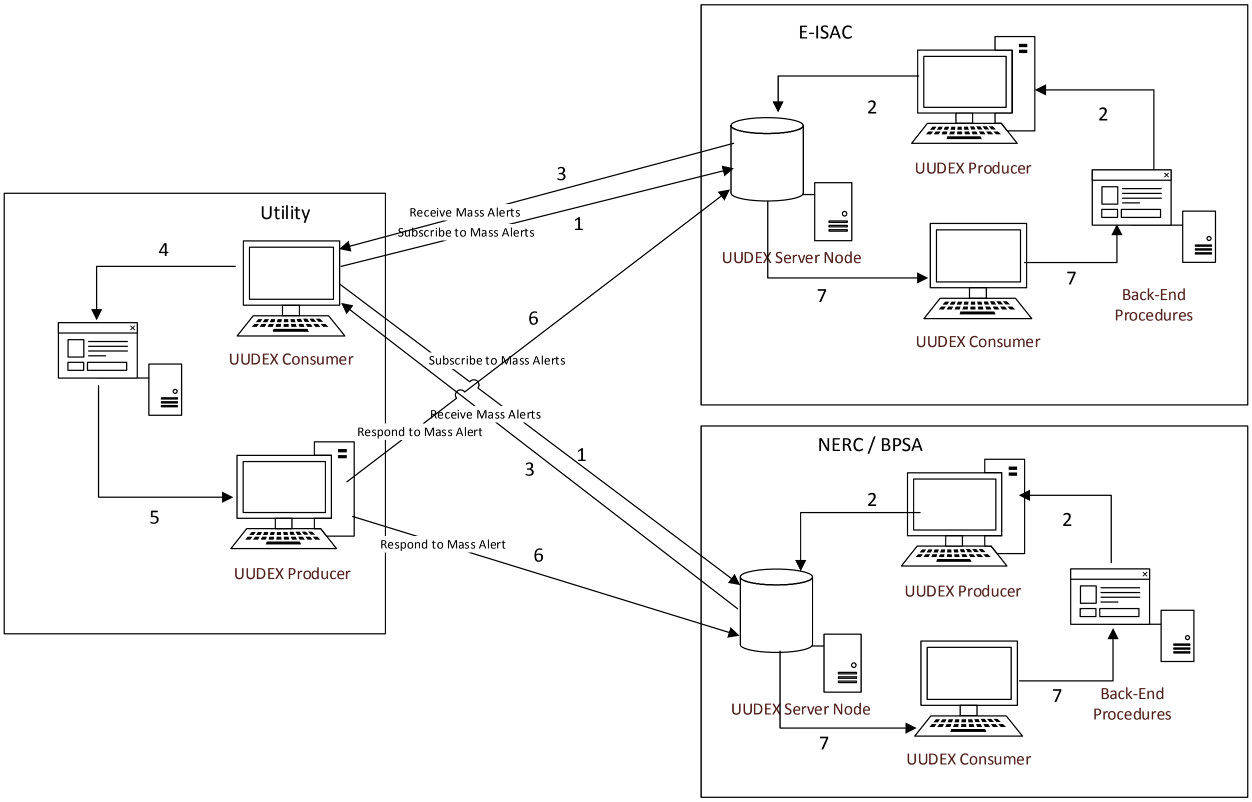

Figure 14: Mass Alert with Required Response

In this use case, a high-priority alert that requires an asynchronous (i.e., after a period of data gathering) response is distributed.

- The first part of this use case would be identical to the mass alert use case.
  - An alert originator stands up a UUDEX Server.
  - All parties that should receive the alert establish subscriptions for those alerts on the originator’s UUDEX Server (1).
  - Generated alerts (2) are automatically added to the UUDEX Server Repository.
  - Adding to the UUDEX Repository triggers comparisons against subscription patterns.
  - The alert is sent to all subscribed clients in fulfillment of the subscriptions (3).

  - When the alert is delivered, UUDEX's reliable message delivery mechanisms allows the sender to know that the given client received the message.

- The data envelope of the message will have the "Response Required" metadata field set, which will flag the message as requiring follow up. The data envelope for the alert will contain a universally unique identifier (UUID) to verify and track submitted responses.

- Each recipient's UUDEX Client will hand off the message to the organization's internal processes. They will handle identifying what response is necessary and crafting that response (4).

- The nature of the response (5) will depend on the characteristics of the alert itself.

  - The alert might dictate a specific response mechanism external to UUDEX (e.g., phone call, email to a given address). In this case, UUDEX will not be employed in the response and the organization's internal processes will be responsible for ensuring the response occurs.

  - The alert might allow for responses to be delivered via UUDEX (6). In this case, the alert content would indicate (directly or by reference) the appropriate UUDEX recipient's UUDEX Server. Responses would be sent directly to the named UUDEX Server, which might not be the same server that delivered the alert.

  - A UUDEX Client at the party originating the alert will gather and process responses sent by the utility to the UUDEX server (7).

- As the deadline approaches, the originator could compare the list of response providers to the expected list of respondents and send reminders to those whose response is missing.

Variation:

- As with the mass alert use case, a hierarchical distribution model could be used to send out the alerts.

- In this case, the relays that delivered messages to clients would need to be the parties that collected delivery confirmation and sent out reminders to those late in responding. In the latter case, this would require coordination between the relays and the UUDEX Server to which responses need to be sent, since these might not be the same parties.

3.0 Data Exchanges supported by UUDEX

This section identifies the initial set of data exchanges the UUDEX tool will support. Two criteria were used to create this shortlist. First, based on findings from the first set of industry interviews, data exchanges that were essential for day-to-day operation were included. Presently, these exchanges are largely performed using individual mechanisms that have several disadvantages ranging from cumbersome link setup to lack of uniform data representation. Second, data exchanges that we identify as critical to secure grid operation in the future will also be supported. These primarily include cybersecurity-specific data exchanges that will support the dissemination of threat, vulnerability, and security upgrades to enhance the overall security posture of the stakeholder.

The list below is included as examples of some of the different types of data that UUDEX can convey. Some of the types of data below might be revised and undergo changes in structure, possibly in such a way that both old and new structures are employed simultaneously within a given environment. Ultimately, UUDEX's ability to exchange data is agnostic with regard to the nature and format of that data, so UUDEX will be able to handle exchange of new or updated forms of data. Thus, the following sections are intended to call out some of the ways UUDEX might fulfill current data exchange needs, but with the recognition that UUDEX's capabilities go beyond these examples.

3.1 Grid Operational Data

The primary data exchanged by UUDEX are grid operational data. These include analog and status values telemetered by transmission operators from substations and plants and shared with neighboring transmission operators, RCs, and others to allow them to analyze grid conditions, and perform required functions to ensure grid reliability and maintain situational awareness.

3.1.1 ICCP Data

ICCP is used extensively by utility organizations to exchange grid operational data over wide-area communication networks. The types of information supported by ICCP include analog values, binary status data, control signals, and schedules. When analog values are conveyed, each is typically represented as a real or integer value along with data quality flags that provide further context about the data. These include flags to highlight data source (options: telemetered, calculated, estimated, and manually inputted), flags to indicate data normalcy (options: normal or abnormal), and flags to specify data validity (options: valid, not valid, held, and suspect). Typically, SCADA applications use a combination of these flags to sufficiently describe the conveyed data during a wide range of scenarios, including normal operation, telemetry failure, stale data detection, data out of range, and data conversion errors. The protocol also defines the capability to include a timestamp for the data value at the source. In the case of data values calculated from telemetry at the source, the timestamp to be included is left to the implementation of the application. Binary status data are represented by bitmasks that support per-phase representation. The data quality and timestamp attributes are used in this case to provide additional context. The control capability supported by ICCP includes device switching, raise/lower commands, set-point specification, and device turn on/off. ICCP also supports the ability to exchange scheduling information (e.g., generator schedules, interchange schedules, pricing information) between a client and server.

Each data value communicated using ICCP has a tag used by the application software at the recipient to associate the value with a specific measurement object in an application data model. It is the responsibility of applications on each end of the data link to establish any mappings that are necessary as well as data conversions to get the values to a representation where the scaling and units on both sides are well understood.

The data exchanged using ICCP are organized into “conformance blocks” representing fundamental types of service supported by the standard. These include:

Block 1 – Periodic Power System Data

The data objects categorized under this block are used for periodic exchange of field device status, analog measurements, and accumulator values between a client and server. As introduced earlier, these exchanges could be accompanied by suitable data quality flags that provide additional context about the exchanged data.

Block 2 – Extended Data Set Condition Monitoring

Also referred to as "report by exception," this block provides a client the capability to configure report generation and transmission by exception. Examples of exceptions include cases where the value or quality code of a particular data point has changed an operator-initiated push of value from server to client.

Block 3 – Block Data Transfer

This block defines an optimized method of transferring data described in Blocks 1 and 2 as groups of data, rather than individual enumerated values. The optimization is achieved by combining individual data values into blocks, removing the tags associated with each data value for mapping to suitable point at the recipient, and also removing the length fields associated each data value. In place of the variable name-based tags used for mapping in Blocks 1 and 2, this block employs an index-based tagging mechanism.

Block 4 – Informational Messages

The exchange of ASCII text and binary files is supported by this block. It is typically used for the exchange of complex information that cannot be conveyed using the other blocks (e.g., text file describing an emergency situation or system restoration summary).

Block 5 – Device Control

The device control block provides a client the capability to request control of remote device to the server. As introduced earlier, the controls that can be implemented using this block range from switching actions to set-point specifications on the remote device. In addition, ICCP supports both interlocked operation, wherein select-before-operate confirmation is required, and non-interlocked operation.

Block 6 – Program Control

Block 6 introduces the capability for an ICCP client to control a program on the remote server.

Block 7 – Event Reporting

This block provides the capability for ICCP clients to receive event-specific information that the client has subscribed to.

Block 8 – Additional User Objects

This block provides the utility with the means to exchange scheduling and accounting information; examples include generator schedules, interchange schedules, and pricing information.

Block 9 – Time Series Data

This block adds the capability to exchange time series data between an ICCP client and server. This is useful to applications that do not need data sampled at a very high rate in real time (e.g., post-disturbance voltage data recorded in millisecond intervals for analysis).

The third edition of the ICCP specification (IEC 60870-6-503:2014), released in 2014, made Blocks 6, 7, 8, and 9 out of scope. UUDEX will provide a mechanism to transport data described in Blocks 1, 2, 3, and 4. It will not provide device control as specified in Block 5 or program control as described in Block 6. UUDEX will investigate inclusion of data from Blocks 7, 8, and 9.

3.1.2 Reliability Coordinator Information System

RCIS is used primarily by RCs to post information concerning reliable operations of the Bulk Electric System (BES). The current RCIS is a web-based system that functions much like a message bulletin board, allowing users to post messages to the system and providing access for other users to view posted messages. Some messages are generated and reported autonomously by software at the RC. Users can either monitor the web interface for new activity or can sign up to receive emails when new messages are posted. Most RCIS messages are free-form text messages, although some message types have a limited message structure.

Current RCIS message types include:[2]

- **Critical Infrastructure Protection - DHS** has been supplanted by reporting through the E-ISAC portal.
- **Critical Infrastructure Protection - Free Form** has also been supplanted by reporting through the E-ISAC portal, but it is still used to alert control room staff of issues that would otherwise not be made known to them in a timely manner, especially if access to the E-ISAC portal is not provided to the control room operator 24x7.
- **Energy Emergency Alert reporting** by RCs is required by NERC Standard EOP-002 from the time such an alert is issued to the time the alert has been cancelled.
- **Frequency** is used for communicating system events that have or could result in a rapid change in frequency that significantly impacts system operation; also used to report changes in frequency for which the cause is unknown.
- **Geomagnetic Disturbance** information originates from the National Oceanic and Atmospheric Administration's Space Weather Prediction Center and is made available by designated RCs to receive and disseminate notifications of possible geomagnetic disturbances to RCs, BAs, and TOPs.

[2] Much of the information was extracted from minutes of the NERC Reliability Coordinator Working Group of May 3, 2011, "Exhibit C" located at: https://www.nerc.com/comm/OC/RCWG%20Agendas%20Highlights%20and%20Minutes%20DL/Agendas,%20Highlights,%20and%20Minutes%20-%202011/RCWG_Minutes_3May11.pdf

- **System Emergency** – used to provide notification when an RC foresees transmission problems (such as a system operating limit [SOL] or interconnected reliability operating limit [IROL] violation, loss of reactive reserves, etc.), when results of operational studies for the current day or the next day indicate that there is potential for SOL or IROL violations, or when an interconnected system separation, system islanding, or blackout has occurred.
- **Transmission Outage** – messages relating to transmission line outages for facilities greater than 230 kV, automatically generated and sent to the System Data Exchange database and posted on the RCIS.
- **Generation Outage** – messages relating to generation facility outages greater than 300 MW, automatically generated and sent to the System Data Exchange database and posted on the RCIS.
- **Time Correction** – indication of the start and end of time error correction within an interconnection.
- **Transmission Line Loading Relief** – messages relating to transmission loading relief following NERC Standard IRO-006, automatically generated by the Interchange Distribution Calculator and posted to the RCIS.
- **Weather Advisory** – notifications of approaching or existing severe or extreme weather conditions that have the potential to affect system reliability. These conditions could include severe heat or cold, insulator ice bridging, large thermal generation limitations (due to fuel restrictions), tower damage (due to tornado, hurricane, or flooding), extensive ice storms, galloping on transmission circuits, forest fires, etc.
- **Free Form** – designed to capture situations that an RC determines to be appropriate to communicate with other RCs, Bas, or TOPs regarding an issue that is not directly related to any of the other message board categories available on RCIS. A number of free-form message classes have come into use, even though there is not a specific RCIS message class for them. UUDEX should consider specifically creating message classes including but not limited to the following:
  - **Test & Maintenance** – messages indicating testing of the RCIS system, such as test notification, testing of successful message submission, or maintenance of the RCIS.
  - **Drills & Exercises** – notification of operational drills, such as testing of backup control center locations, testing of new communications facilities, (parallel) testing of new platforms and applications, etc. (Note: actual evacuation should be reported as an Emergency).
  - **Software Issue** – notification that the RC is experiencing EMS or software issues, such as State Estimator not converging, or installation or testing of major software updates.
  - **Timing Integrity Issue** – notification that an RC, BA, or TOP has detected a timing integrity issue such as that caused by spoofing the time from a Global Positioning System (GPS) receiver. The issue must be of sufficient magnitude to affect the timing alignment of SCADA, PMU, or other time-sensitive data across an RC, BA, or TOP geographic area.
  - **Theft, Burglary, Vandalism** – reports of (mostly) nuisance events that do not have a direct impact on operations, like copper theft, substation break in, and bomb threats.
  - **Non-Transmission Emergency** – similar to System Emergency, but not for transmission issues. Examples include control room evacuations due to fire or bomb threat, or physical damage to a transmission station (like fire or flood) that does not

necessarily induce an IROL violation, islanding, or cascading (therefore not qualifying as a System Emergency).

3.1.3 Power System Models

Power system models that enable real-time and study simulation of the electricity grid are commonly exchanged between utilities and grid operators. These models describe the electrical connectivity between objects such as transformers, breakers, generators, meters, transmission lines, and distribution feeders, and then expand to include aspects such as associated measurements, electrical impedance characteristics, asset information, etc. Given the many uses of these models, profiles are defined to identify the information (e.g., classes, attributes, and relationships) needed for a given use. The models may be exchanged in forms that are suited toward a given usage. This is a realization that there is diversity in the set of information needed for use by applications such as power flow, state estimation, distribution outage management, metering, or planning.

One form used is Common Information Model/Extensible Markup Language (CIM/XML) as defined by the International Electrotechnical Commission (IEC) 61970 and IEC 61968 series of standards, where files can represent full or incremental updates to the model. These files convey the electrical characteristics of power system resources and their connectivity relationships.

There is no implicit restriction as to whether models exchanged relate to transmission, distribution, substations, or even microgrids. Models exchanged may (or may not) carry graphical relationships, which may be based on geographic (e.g., GPS) or schematic coordinate systems.

3.1.4 Phasor Measurement Unit Data

PMU data can be characterized as fined-grained, time-series voltage, current, and frequency data that are time-stamped at the source (the PMU) against GPS time. Timestamping using a GPS enables the synchronization of measurements from geographically dispersed PMUs, providing an accurate representation of the power system state at a given time. The advent of PMU technology has ushered in the development of novel control applications that offer significant benefits to improving power system reliability. These applications can be broadly classified into the following three categories: automated closed-loop control, human-in-the-loop control, and offline analyses.

Automated closed-loop controls operate very quickly from the sensing of an event or disturbance to the execution of a control action. This type of control application typically leaves the human out of the loop, although there could be use cases demanding instantaneous intervention from a system operator. Applications that fit this description include protection, fast-reactive switching, damping controls, resource integration support, and alarming and operating limit monitoring among others.

Human-in-the-loop controls involve the analysis of processed phasor measurements from an event or disturbance by a system operator, who then proceeds to implement specific control actions, if deemed necessary. These applications, implemented with the objective of providing system operators with situational awareness, have a more lenient expectation on communication latency. Applications of this type include tools for wide-area situational awareness, voltage monitoring and trending, dynamic line ratings calculation, outage restoration, and operations planning.

Offline analyses carried out using historical PMU data have the potential to enhance overall reliability of the grid. System planners benefit significantly from the additional insight received by simultaneously studying synchronized measurements from multiple locations in the grid, enabling them to improve operational strategies. Applications that fall under this category include power system baselining to support predictive tools, post-event analysis for event reconstruction and reoccurrence prevention, static and dynamic model calibration, load characterization and modeling, and design and testing of special protection systems (or remedial action schemes).

As stated earlier, UUDEX is not intended for control purposes, eliminating the need for extremely low-latency communications and the requirement to support PMU-based automated closed-loop controls. However, UUDEX is capable of supporting human-in-the-loop controls and offline analyses. In both these cases, UUDEX merely functions as a conduit for time-stamped measurements from the source to the system operator or planner for processing. It will not support the execution of any control actions identified by the applications.

UUDEX will consider protocols under development such as STTP as well as techniques that provide for down-sampling of the information in a form that might be useful to a wide array of applications. The initial version of UUDEX will include a time-bounded UUDEX Data Element that can encompass down-sampled PMU data associated with a disturbance event. For example, a bounded UUDEX Data Element for a disturbance of any type lasting 1 minute might include 2 minutes of PMU data sampled at 60 times a second before, during, and after the disturbance for post-event analysis purposes.

3.1.5 Industry Incident Reports

The electricity industry has several required formats to report incidents that affect reliable operations to DOE, NERC, or the E-ISAC. Additionally, NERC developed a guideline in 2008 for Threat and Incident Reporting[3] that provides guidance for voluntary reporting.

The DOE report form (OE-417) and the NERC Standard CIP-008 refer to the reported items as “incidents,” while the NERC Standard EOP-004 refers to them as “events.” For purposes of UUDEX, incidents and events refer to the same thing.

The NERC guideline document discusses topics that should be reported, as well as suggested reporting timeframes, but does not specify a particular format or data fields that should be included in the reports.

3.1.5.1 DOE OE-417 Electric Emergency Incident and Disturbance Report

The DOE OE-417 Electric Emergency Incident and Disturbance Report is used to report certain electrical, operational, cybersecurity, and physical security incidents to the DOE, with optional copies to NERC and the E-ISAC. The reports must be filed within 1 hour (emergency reports), 6 hours (normal reports), and 24 hours (system reports). The report form includes sections for:

[3] Document marked as “under revision”; see https://www.nerc.com/files/Incident-Reporting.pdf

1. Alert criteria, including whether “Emergency,” “Normal,” or “System,” and providing a report status, along with information about who is filing the report (organization name and address).
2. Information about where and when the disturbance or incident occurred and whether the incident involved load or customer outage.
3. Information about the type of emergency, including its cause, impact, and any actions taken.
4. Free-text information, including contact information for the person making the report, a free-form block to describe the incident, estimated restoration time, names of any assets (electrical) that were impacted, and indication of whether the information should be shared with NERC or the E-ISAC.

For implementation within UUDEX, in addition to transporting the PDF or Word OE-417 file itself, the field identifiers and values can be extracted from the form and transported using a UUDEX-defined data model. Those field values can be reimported into its respective document format (PDF or Word) at the destination. This ability notwithstanding, it will always be possible to transport the whole file itself as a UUDEX Data Element.

There are certain sub-documents of DOE OE-417 reports and NERC Standard CIP-008-6 reports[4] that are assigned a 1-hour time limit by OE-417 and CIP-008-6 as the upper bound on reporting. The reason for the time limit is that it enables national authorities who receive these reports (DOE, E-ISAC, and the Industrial Control Systems Cyber Emergency Response Team [ICS-CERT]) to determine whether there is a coordinated cyberattack underway against the BES and to issue security guidance as quickly as possible to mitigate its further impact and geographic spread.

While a 1-hour time limit normally is not a message latency requirement that is difficult to meet, this requirement must also be met under network congestion conditions. To ensure the time limit is always met, the operators should consider prioritizing delivery of UUDEX Data Elements for OE-417 and CIP-008-6 reports. As discussed later, UUDEX supports a way to prioritize message delivery so higher priority messages have a better chance of being delivered despite network congestion.

- OE-417 Reports: All sub-documents under the Emergency Alert heading, namely items 1-8, to include item 2, “Cyber event that causes interruption of electrical system operation.”
- CIP-008-6[5] Reports: Part 4.3, “One hour from the determining of a Reportable Cyber Security Incident” that affects “High Impact BES Cyber Systems.”

3.1.5.2 NERC Standard EOP-004

NERC Standard EOP-004 requires that NERC Responsible Entities must file electrical disturbance reports to NERC within 24 hours following the disturbance. NERC will accept an OE-417 report or information following a sample form provided in the standard. Copies are also to be sent to the appropriate Regional Entity, company personnel, the Responsible Entity’s RC, law enforcement, or the Applicable Governmental Authority, as described in Requirement R1 of the standard.

[4] NERC Standard CIP-008-6 is under development and pending approval by NERC/FERC as of October 8, 2018

[5] Proposed language as of October 8, 2018

Information required to be submitted includes:

1. Information pertaining to the reporting entity
2. Date and time of the event
3. An indication of whether the event originated within the organization's part of the electrical system
4. Event identification and description, including selection boxes and a free-text field.

All EOP-004 field values can be mapped into OE-417 field values. For example, the EOP-004 field value "3: An indication of whether the event originated within the organizations part of the electrical system" can be mapped into the OE-417 field "13: Damage or destruction of a Facility within its Reliability Coordinator Area, Balancing Authority Area or Transmission Operator Area that results in action(s) to avoid a Bulk Electric System Emergency."

Like the OE-417 form, UUDEX will define a data model so that fields and values can be extracted from the EOP-004 form. These values can be transported over UUDEX and ultimately reimported into a destination form. This ability notwithstanding, it will always be possible to transport the whole file as a UUDEX Data Element.

3.1.5.3 NERC Standard CIP-008

NERC Standard CIP-008 requires that NERC Responsible Entities file cyber incident reports with the E-ISAC for specific kinds of "Reportable Cyber Security Incidents" within 1 hour of determining a report should be made. The standard does not prescribe any specific format or data fields that should be included in the report.

On July 19, 2018, FERC issued a final rule requiring that NERC develop modifications to CIP-008 to increase the scope of reportable incidents, specify that reports should be made to the ICS-CERT, and provide the following minimum information to be include in the report:

1. Functional impact that the Cyber Security Incident achieved or attempted to achieve
2. Attack vector that was used to achieve or attempt to achieve the Cyber Security Incident
3. Level of intrusion that was achieved or attempted as a result of the Cyber Security Incident.

FERC indicated that NERC may wish to add data fields to this list when developing the standards modification.

UUDEX can develop a data model for fields associated with a CIP-008 report that encompasses the minimum information required by FERC (items 1, 2, and 3 above). If information for an OE-417 report are present, then item 1 will be satisfied by the "impact" information associated with the OE-417 reported fields.

As noted under the discussion of OE-417 reports, some CIP-008 reports might have time constraints on their delivery. As such, parties submitting reports may wish to ensure their systems are configured to prioritize the delivery of these reports.

3.1.6 Files (COMTRADE [IEEE Std C37.111^{TM}] and Others)

Evolution of business processes and the regulatory environment can cause development of new types of information. To support this evolution, there needs to be the ability to exchange these new types of information without the necessity of software changes to the communication infrastructure or associated interfaces. This necessitates the ability to exchange files independent of type or data format.

To accommodate this, there needs to be:

- A way to indicate the type of file transmitted (such as by providing a media type, as identified in the Internet Assigned Numbers Authority Media Types registry[6]). There also needs to be a way to describe new file types on an ad-hoc basis or refine a named type to indicate specific use of a broader media type.
- A set of data models that may be used to define the structure of some documents as a way to improve interoperability.
- UUDEX Data Envelope fields that provide metadata for each file that might include, but not be limited to:
  - File type (as may relate to a specific application or defined schema)
  - File name (need not be unique, may be hierarchical)
  - File ID (unique key, e.g., a UUID)
  - File source (organization that is owner or creator of the file)
  - Created by (optional, person within the organization that created the file)
  - File format (e.g., CSV, XML, JSON, PDF, text, image, binary)
  - Schema reference (for structured documents, optional)
  - Signature (e.g., MD5 hash)
  - File creation date (ISO 8601 timestamp, set by submitter)
  - File submission date (ISO 8601 timestamp, set by UUDEX)
  - File expiration date (after which file is no longer valid)
  - Abstract (short description of file contents)
  - Keywords (that may be useful for searches)
  - Priority
  - Status (default = ACTIVE)
  - Version (default = 1)
  - Obsoletes (optional, UUID of file version that this replaces)
  - ObsoletedBy (optional, UUID of file version that this is replaced by).

3.1.7 Operations Planning Data

In order to operate the BES reliably, RCs), BAs, and TOPs perform day-ahead and future hour power system studies to mimic future operating condition.

[6] Located at https://www.iana.org/assignments/media-types/media-types.xhtml

These studies have a dual purpose. First, they try to predict and anticipate potential operating transmission limit violations, both SOL and IROL, as well as voltage stability limits, generation shortages, or other operating condition that would threaten the reliability of the BES. In addition, they provide preventive actions to mitigate unsecure operating conditions.

To perform these studies, operating authorities (RCs, BAs, and TOPs) exchange power system models as previously indicated and it is of paramount importance to share information on load forecast or neighboring systems, generating units' operating plans, and power transfer schedules.

3.1.7.1 Load Forecast

The load forecast is normally calculated by BAs and the information that is exchanged with other neighboring entities include the following information:

- BA reporting the load
- RC the BA belongs to
- Time zone in which the load is being reported
- Period for which the load is being provided that depends on the time horizon (could be 5 minutes, 15 minutes, hourly, daily, weekly, etc.)
- Actual load forecast for the specified period.

3.1.7.2 Interchange Schedules

Information on energy transfers from one balancing area to another is commonly shared between RCs, BAs, and TOPs on a periodic basis. These entities can then perform power flow and advance application studies using schedules and interchange with neighboring areas.

The schedule information that is normally exchanged for each schedule could include the following:

- Reporting Entity
- Source Point
- Source Balancing Area
- Sink Point
- Sink Balancing Area
- Schedule Start Time
- Schedule End time of the transaction
- Schedule Energy profile
- E-Tag Interchange Transaction reference, if applicable.

3.1.7.3 Current Operating Plan

It is common for Load Serving Entities and Generator Operators to share the current hour, current day, and extended days operating plan for load and generating resources with their BAs, RCs, and TOPS. The information shared could be very extensive, but in most cases would include as the minimum the following:

- Delivery Date and Time
- Resource Name
- Resource Status (can indicate if the unit is on, off but available, out of service and unavailable, etc.)
- Resource Limits:
  - High Sustained Limit
  - Low Sustainable Limit
  - High Emergency Limit
  - Low Emergency Limit
- Ancillary Services that they are providing:
  - Regulation Up
  - Regulation Down
  - Responsive Reserve
  - Non-Spinning Reserve

This type of information is well established and varies in wholesale markets.

3.1.8 Asset Management

Beyond the needs to exchange power system models as described previously, the ability to exchange asset information will be a growing need over time. Information that is found in an asset model over what is required for simulation includes the following examples:

- Identification of the individual physical, serialized assets that comprise a power system resource
- Location of an asset, in terms of GPS coordinates or physical address
- Manufacturer, model, and version of a given asset
- Attributes of the asset beyond electrical characteristics, such as size, weight, height, volume, supporting structures, etc.
- References to related specifications, data sheets, etc.
- Ownership and value
- Lifecycle history of the asset.

This information could be conveyed in a variety of ways, such as CIM/XML files or Shapefiles.

Other asset management data could include:

- Spare Equipment Database
- Cyber Asset Management (National institute of Standards and Technology [NIST] document under development)

- Domain Management Task Force

3.2 Cybersecurity Data

The information technology (IT) and IT-connected assets of the energy sector are increasingly targets of malicious activity, including criminal attacks to steal or extort money, vandalism and public reputation attacks, and state-sponsored attacks, in addition to potentially being victims of target-of-opportunity attacks, such as having assets compromised for use in botnets. For these reasons, IT operators need to ensure adequate protection of their IT assets. Key to this protection is the receipt and sharing of cybersecurity data. This includes guidance and information regarding current threats, proposed courses of action to address vulnerabilities, and sharing incident data both for reporting requirements and to use experience gained in the incident to help protect other parties. This section looks at some of the types of cybersecurity data UUDEX is able to convey.

In the case of operational cybersecurity data (i.e., data the recipient might use to alter the operations of their cyber assets), the issue of trust relationships needs to be carefully considered. Specifically, the data recipient needs to trust that the information received is accurate and provides the asserted benefits. For example, most parties trust patches released by the vendor of the product to which the patch applies. Very few would be willing to install a patch created by some unknown third party. As such, careful documentation of the provenance of cybersecurity data or the identities of parties who vouch for the accuracy of the data, is necessary. This is especially true in cases where the party that produced the data is not the same party from which the data is received. This will be a frequent occurrence in UUDEX since UUDEX Servers are likely to be reposting UUDEX Data Elements provided by other parties. To address this, it will be important to identify the original authors of cybersecurity data or (in the case where the author wishes to remain anonymous, such as a party reporting sensitive data breach information about their enterprise) that the data be endorsed by an authority the recipient trusts. It will be up to each organization to determine which cybersecurity data authors they trust to provide accurate information, but that decision hinges on the authors (or vouching authority) being identified.

3.2.1 Cyber Incident Reporting

Intrusion detection system (IDS), intrusion prevention system (IPS), and packet capture (PCAP) data types cover logs from cybersecurity tools as well as other network and endpoint monitoring tools. The data are primarily used for forensics and might also be required by certain parties as part of an incident report. The goal of sharing this type of information is to help provide comprehensive context with regard to network or other activities within a given period in time. Log files can be quite large, so will likely only be sent in response to specific incidents or needs.

Data elements of this type would need to include:

- Organization sending the log.
- Specific times associated with the log collections.
- Specific tool (including vendor, model, and version number of software or firmware) of the tool that generated the log.

- Information about the scope of the collection. For example, identification of subnets monitored, any filters applied to the data log, or other information. Because this information could vary widely in nature, it will likely need to be provided using descriptive free text.
- References to other UUDEX Data Elements relevant to the log. This could include such things as a formal incident report submission or free-text analysis by a local operator offering their thoughts on the provided logs.

Log information is likely to be highly sensitive as it will reveal not only the identity of the security tools in use that produced the log, but often include a wealth of information about the sender's IT infrastructure. For this reason, logs will require confidentiality and integrity protection, and access to them will generally be granted to only a very small number of parties.

3.2.2 Operations Technology Cyber Incident Reporting

Due to the increasing number and types of cyberattacks on operations technology (OT) within electric power and other critical infrastructure, there is a need for better technical characterization of incidents involving these types of attacks. Leaning forward, UUDEX will support the exchange of incident reports relating to OT equipment. Fields could include the impact of an incident (both on function and on data), how long it took to recover from the attack, identities (make and model) of impacted OT equipment, vector by which the system was attack, and how the attack was detected. Standards for encapsulating such information are under development and UUDEX will be able to support their transportation.

Data elements of this type would need to include:

- Organization sending the report.
- References to other UUDEX Data Elements relevant to the report. This could include such things as data logs surrounding the event, change of system state information resulting from the event, or even a formal description of the indicators of compromise (IoCs).

Report information is likely to be highly sensitive as it will reveal the identity of the compromised party as well as specific OT equipment they use and how they were compromised. For this reason, reports will require confidentiality and integrity protection, and access to them will generally be granted to only a very small number of parties.

3.2.3 Indicator of Compromise Sharing

An IoC is a concise expression of network traffic, endpoint behavior, or other patterns used to help guide the recipient in the detection of likely attacks or compromises. The goal of such UUDEX Data Elements is that the recipient will be able to use the information to alter monitoring tools (e.g., IDS, IPS) or perform other scans to detect if the described behavior is present on their own networks. Ideally, IoCs are expressed in a way that can be directly ingested by security tools. IoCs often include additional information about the nature of the compromise being detected so that IT operators can better understand the implications of a positive or negative detection.

Data elements of this type would need to include (although some IoC formats will already include):

- Author of the IoC.

- If the IoC can be ingested by certain security tools, the specific tools that can ingest the IoC.
- When the IoC was authored. There might also be an expiration date when the IoC will no longer be applicable.
- Severity level of the described compromise, indicating how urgently the recipient should check for the indicators.
- References to other UUDEX Data Elements relevant to the IoC. These could include reports on certain cyber threats associated with the indicator or instructions to the recipient regarding their use of the indicator.

Some IoCs are public information and can be shared with anyone, even outside the energy sector. Others represent proprietary information and can only be distributed to parties that have purchased a license or have joined certain organizations. In other cases, it might be necessary to control release of an IoC because adversaries could be tipped off that their activities have been detected, prompting them to evolve their procedures to better avoid that detection. In all cases, IoCs need to be integrity protected so the indicator patterns cannot be corrupted, which would render the IoC useless. This is especially true of IoCs that are intended to be automatically ingested by tools.

3.2.4 Guidance

Guidance refers to any material intended to provide instruction with regard to how an enterprise is configured. Examples include Center for Internet Security Benchmarks[7] or material from NIST's National Checklist Repository.[8] Guidance comes in a variety of forms, from structured content that can be automatically ingested and used by security tools to prose descriptions of best practices. Guidance can represent general recommendations for best practices or could come with requirements to adopt the described practices as issued by a suitable authority such as NERC.

Data elements of this type would need to include:

- Author of the guidance.
- Date the guidance was authored.
- Applicability of the guidance (e.g., the specific operating system or software application to which the guidance applies).
- Format of the guidance. For guidance that can be automatically ingested by tools, this would identify the tools that could ingest this guidance.
- Criticality of the guidance. This could include whether some authority was mandating its adoption.

Some guidance material is public information; others might only be releasable to parties with a certain license from the guidance author. In either case, guidance needs to be integrity protected against corruption. This is especially true of guidance that is intended to be automatically ingested by tools. Similarly, even if public guidance is being disseminated, there

[7] Available at https://www.cisecurity.org/cis-benchmarks/
[8] Available at https://nvd.nist.gov/ncp/repository

might be a desire not to expose the nature of the guidance being issued as it could reveal desired configurations. For this reason, confidentiality of guidance is also recommended.

3.2.5 Conformance Reports

Conformance reports describe the state of enterprise assets, usually with regard to some specific piece of guidance or patch notification. Often conformance reports are used by parties to inform an authority whether or to what extent their enterprise conforms to certain configurations, patch levels, or other standards. For example, a party might issue guidance and then expect the recipients to report where they conform or deviate from that guidance. Conformance reports are useful to gather a focused snapshot regarding certain important properties of an enterprise.

Data elements of this type would need to include:

- Source of the conformance report.
- Tool used to generate the report, if any.
- Reference to any guidance, patch notification, or other material that guided the generation of the conformance report.
- Date the report was generated.
- Format of the report (structured or free text).
- Any additional contextual information regarding the report (e.g., whether the report covers the whole enterprise or just specific assets).

Conformance reports will often reveal sensitive information about the party that generated the report. This could include information about software used in the enterprise, the presence of unpatched software vulnerabilities, and other network infrastructure information. For this reason, it is important that both the confidentiality and integrity of the report be protected and that the list of parties authorized to view the report be carefully controlled.

3.2.6 Patch Notification

Patch notifications are issued by software vendors to correct flaws in their software products. Sometimes patches simply add features or correct undesired user interactions with the software. Other times they are issued to fix security vulnerabilities associated with a software product. Patch notifications will either include an executable patch file that can be used to fix the described software or will include a reference (often a Uniform Resource Identifier) that can be used to retrieve this patch file from a remote source. In most cases, recipients use a patch notification to deploy the patch to applications in their enterprise, usually after first confirming the patch does not have any disruptive side effects through testing in a lab environment.

Data elements of this type would need to include:

- Applicability of the patch (e.g., the specific operating system or software application to which the patch applies.)
- Severity of the issue the patch corrects.
- Date the patch was released (or approved)

- References to additional UUDEX Data Elements as necessary. For example, the patch might be associated with some guidance that requires its adoption, or with a vulnerability or threat notification that the patch corrects.

In general, patches are public information; however, the integrity of patch notifications is critical as corruption of patches, or of references to patches, could at best lead to corruption of the patched software and at worst be used to introduce malware into an environment.

3.2.7 Vulnerability Notification

Vulnerability notifications are informative materials disseminated to make recipients aware of flaws in one or more software products that might be exploitable by adversaries. A notification itself is informative and intended to recommend increased vigilance against the described vulnerability. It might be accompanied by a patch notification that closes that vulnerability. In some cases patches are not available and the recipient might respond to the notification by altering the configuration of the vulnerable software to mitigate the vulnerability (if possible), reduce access to vulnerable software to decrease exposure, or even uninstall the vulnerable software if the threat of exploitation exceeds the software’s utility.

Data elements of this type would need to include:

- Author of the vulnerability notification.
- Applicability of the vulnerability notification (e.g., the specific operating system or software application to which the vulnerability notification applies).
- Date the vulnerability notification was authored.
- Severity of the described vulnerability. This could include whether the vulnerability is being actively exploited by malware.
- References to additional UUDEX Data Elements as necessary. For example, the vulnerability notification might be associated with some guidance to mitigate the vulnerability or (ideally) with a patch that closes the vulnerability.

Many vulnerability notifications are public information. Some vulnerability notifications are non-public, often because the vendor has not yet developed a patch but the notification author is sharing the vulnerability report selectively so certain parties can deploy mitigations. In the latter case, the confidentiality of the vulnerability notification is extremely important since exposure of this information to malicious parties could provide them with the information necessary to develop malware that exploits the vulnerability. Similarly, recipients of non-public vulnerability notifications need to be carefully limited to reduce the chance of exposure to malicious parties. Finally, the integrity of vulnerability notifications is important since corruption of these notifications could lead to incorrect or incomplete mitigations by the recipient.

3.2.8 Threat Notification

Threat notifications are warnings of activity by cyber adversaries, including identification of specific tools, target groups, or cyberattack campaigns. A notification itself is informative and intended to recommend increased vigilance against the described threat. A threat notification might be accompanied by an IoC to help detect the described threat. Alternately, a threat notification might simply recommend additional vigilance given general behavior observed.

Data elements of this type would need to include:

- Author of the threat notification.
- Date the threat notification was authored.
- Severity of the threat.
- References to additional UUDEX Data Elements as necessary. For example, the threat notification might be associated with guidance to mitigate the threat or with IoCs to detect the describe threat.

Some threat notifications are public information. Others are proprietary and only distributable to parties with the necessary license or group membership. Still others might be sensitive due to a desire not to tip off adversaries that their activities are being monitored. Threat notifications that have limited distribution need to be kept confidential and limited to certain recipients. All threat notifications need to be integrity protected since corruption of these notifications could mislead the recipient regarding the described threat.

4.0 Functional Requirements for Data Exchange

UUDEX will support the exchange of diverse data types that demand specific functional requirements primarily based on the nature of the data exchanged and the application it supports. This section introduces performance features that will be used to characterize the data exchange, the outcome of which is a key input to UUDEX development.

4.1 UUDEX Roles and Definitions

The following terms and associated definitions are used in the descriptions of UUDEX and related functionality.

UUDEX Framework
The overall communications framework defined by UUDEX, inclusive of its roles, connections, and procedures.

UUDEX Client
An entity that produces or consumes UUDEX Data Elements through interactions with a UUDEX Server.

UUDEX Producer
A UUDEX Client that publishes UUDEX Data Elements to a UUDEX Server.

UUDEX Consumer
A UUDEX Client the retrieves UUDEX Data Elements from a UUDEX Server.

UUDEX Server
An entity that brokers delivery of UUDEX Data Elements to UUDEX Clients as data consumers by supporting queries and managing or servicing subscriptions by UUDEX Consumers. UUDEX Servers also support UUDEX Clients as data producers by receiving, filing, and controlling access to UUDEX Data Elements submitted by a UUDEX Producer.

UUDEX Repository
A collection of data hosted by a UUDEX Server that is exposed to UUDEX Clients. May be organized by topic, access level, or other factors. A UUDEX Server will host one or more UUDEX Repository.

UUDEX Identity Authority
An entity that creates, certifies, manages, and revokes UUDEX Identity Objects. In effect, it serves as an identity authority within a UUDEX Instance.

UUDEX Roles
Any of UUDEX Client, UUDEX Producer, UUDEX Server, or UUDEX Identity Authority.

UUDEX Participant
An entities that acts in conformance with one or more UUDEX Role.

UUDEX Instance
A collection of connected UUDEX Participants that is closed with regard to the exchange of one or more type of UUDEX Data Elements. In other words, if, for a given collection of UUDEX Participants, certain UUDEX Data Elements that are exchanged between those UUDEX Participants will not be exchanged with UUDEX Participants not in that set, then the set forms a UUDEX Instance. Note that separate UUDEX Instances may have

overlapping UUDEX Participants, but UUDEX Data Elements do not cross from one UUDEX Instance to another via those UUDEX Participants.

More intuitively, a UUDEX Instance is created when a group of entities deploys the components of a UUDEX Framework for the sharing of certain information within their own community.

UUDEX Tunnel

A communication channel between two UUDEX Participants that conforms to all UUDEX requirements (e.g., security, performance).

UUDEX Exchanges

A communication between two or more UUDEX Participants where all communicants are acting as elements of a UUDEX Framework (i.e., excludes out-of-band exchanges between UUDEX Participants). The UUDEX Exchange will involve one or more UUDEX Tunnels and communications will only occur over UUDEX Tunnels.

UUDEX Data Element

Any data collection conveyed between UUDEX Participants over UUDEX Exchanges.

UUDEX Configuration Objects

A type of UUDEX Data Element that is created by UUDEX Participants to facilitate UUDEX behaviors. Examples include UUDEX Access Control List (ACL) Objects (UUDEX Identity Objects) and UUDEX Data Manifests. UUDEX Configuration Objects are exchanged between UUDEX Participants just as any other UUDEX Data Element. The difference is that UUDEX Configuration Elements are used to manage the operation of a UUDEX Instance.

UUDEX ACL Object

A UUDEX Configuration Object that specifies access rights to a given UUDEX Data Element or UUDEX Repository.

UUDEX Identity Objects

A type of UUDEX Configuration Object that contains information necessary to authenticate the identity of a UUDEX Participant.

UUDEX Data Manifests

A UUDEX Configuration Object that describes the information within a UUDEX Repository, often filtered by specific search criteria, that is available to a given UUDEX Client.

UUDEX Data Envelope

A collection of metadata that is attached to a UUDEX Data Element. A UUDEX Data Envelope is connected to a UUDEX Data Element at all times, including at rest and in transit. UUDEX Data Envelopes support access control, indexing, and other data services.

UUDEX Message Envelope

A collection of metadata that is attached to a UUDEX Data Element while the UUDEX Data Element is in transit between UUDEX Participants. The UUDEX Message Envelope controls behaviors associated with the delivery of the UUDEX Data Element. The UUDEX Message Envelope is discarded when the UUDEX Data Element arrives at its destination.

4.2 Information Flow Requirements

UUDEX is responsible for facilitating the exchange of information between UUDEX Clients via persistent UUDEX Servers. This section outlines the requirements that surround the general framework of data exchange supported by UUDEX.

FLOW-1 UUDEX Servers MUST have the ability to receive requests from UUDEX Clients to post UUDEX Data Elements.

FLOW-2 When a UUDEX Server receives UUDEX Data Elements from a UUDEX Client, it MUST be able to store that UUDEX Data Element and its UUDEX Data Envelope as a record in a UUDEX Repository.

FLOW-3 UUDEX Servers MUST have the ability to search their stored UUDEX Data Elements and UUDEX Data Envelopes for matches against search criteria. The UUDEX Server MUST be able to do this efficiently so that searches complete in a reasonable amount of time. Facilitating this might require indexing data as it is added or other optimizations.

FLOW-4 UUDEX Servers MUST be able to receive requests by UUDEX Clients to retrieve UUDEX Data Elements. These requests can either specify the UUDEX Data Element’s UUID (see requirement DAT-1) or search parameters.

FLOW-5 UUDEX Servers MUST be able to receive requests from a UUDEX Client to replace a UUDEX Data Element or its UUDEX Data Envelope, in part or in full, with different information provided by the UUDEX Client.

FLOW-6 UUDEX Servers MUST be able to receive requests to delete a UUDEX Data Element and its UUDEX Data Envelope from a UUDEX Client.

FLOW-7 UUDEX Servers MUST be able to receive requests from a UUDEX Client to establish subscriptions on that UUDEX Server. These subscriptions specify search parameters that could match elements of UUDEX Data Elements or UUDEX Data Envelopes, similar to UUDEX Client data retrieval requests.

FLOW-7.1 UUDEX Servers MUST be able to store and serve subscription requests until instructed to do otherwise by a UUDEX Client.

FLOW-7.2 UUDEX Servers MUST be able to receive requests to delete, pause, resume, or replace existing subscriptions.

FLOW-7.3 When a subscription is active, every time a new UUDEX Data Element is added to a UUDEX Repository on the UUDEX Server, those data MUST be processed against the subscription. Processing involves first comparing the access rights on that UUDEX Data Element to see whether the subscription holder has rights to read or view the data; if so, the UUDEX Data Element is compared against the search parameters in the subscription. If the UUDEX Data Element matches these parameters, this is called a “match” against the subscription.

FLOW-7.4 Upon detecting a subscription match, the UUDEX Server MUST immediately queue a response message to the subscription holder. Depending on parameters in the subscription, this response message might be a UUDEX Notification, mentioning match and providing limited metadata about the UUDEX Data Element, at least including its UUID. Alternately, if the subscription is so configured, the queued message might contain the whole UUDEX Data Element and UUDEX Data Envelope. Queued messages are to be delivered to the subscriber in accordance with the subscription parameters and UUDEX Server configuration. Prioritization (ARCH-2), supported fulfillment models (FLOW-7.6), and other configuration choices might impact the details of this delivery.

FLOW-7.5 UUDEX Clients MUST have the ability to signal the UUDEX Server to pause and resume delivery of queued messages. UUDEX Clients MUST have the ability to signal the UUDEX Server to purge messages in their delivery queue. Both of these can help the UUDEX Client recover from situations where the queue of messages to deliver is so large that it is overwhelming the UUDEX Client's ability to receive them.

FLOW-7.6 UUDEX Servers MUST have the ability to store messages queued for delivery in response to a subscription until such time that the relevant UUDEX Clients contact the UUDEX Server and request delivery of this message queue. This is called "deferred subscription fulfillment." In addition, UUDEX Servers MUST have the ability to immediately contact the subscribing party to deliver the queued messages. This is called "immediate subscription fulfillment." Deployments of UUDEX can decide whether a given UUDEX Server will support deferred or immediate subscription fulfillment or both.

FLOW-8 All described actions MUST be taken only after authentication of the UUDEX Client requesting the action and validation that the action is permitted by access controls associated with the relevant UUDEX Data Repositories or UUDEX Data Elements.

4.3 Identity Requirements

UUDEX Identity Authorities are standalone systems within UUDEX that perform identity proofing and provide authentication services (such as creating and delivering UUDEX Identity Objects) to a UUDEX Instance. As part of the role of the Identity Authorities the system must provide UUDEX with a mechanism for accessing either attribute-based or role-based information for making access control decisions.

UUDEX Identity Objects are data structures that associate the contact information (e.g., name, email address, physical address, and phone number) associated with an entity with a cryptographic puzzle that only the identified entity should be able to solve. Thus, proof that some party is able to solve the cryptographic puzzle can serve as evidence that that party is the named entity in the identity object. Public key certificates are one example of such an identity object.

ID-1 UUDEX MUST support a distributed management model for identities used in the UUDEX Framework. UUDEX Instances MUST be able to avoid dependency on a single UUDEX Identity Authority.

ID-1.1 All UUDEX Instances must have at least one UUDEX Identity Authority.

ID-2 All UUDEX Identity Objects MUST be validated by a UUDEX Identity Authority prior to that authority adding them to the collection of UUDEX Identity Objects used by the UUDEX Instance.

ID-2.1 Identity proofing goal 1: determine to a reasonable level of certainty that the entity identified in the UUDEX Identity Object is the entity it claims to be (i.e., detect and prevent cases where party A attempts to register a UUDEX Identity Object in party B's name).

ID-2.2 Identity proofing goal 2: validate the identity attributes associated with the UUDEX Identity Object. Identity attributes will identify things like role and might be used for access control decisions, so need to be reliable.

ID-3 UUDEX Identity Objects must conform to a standard format.

ID-3.1 All UUDEX Identity Objects will identify the UUDEX Identity Authority that validated them.

4.4 Communications Tunnel Requirements

The following requirements are related to communications.

COM-1 All UUDEX Exchanges MUST be encrypted using algorithms deemed sufficient for protecting Sensitive But Unclassified information.

COM-2 All UUDEX Exchanges MUST be integrity protected using algorithms deemed sufficient for protecting Sensitive But Unclassified information.

COM-3 All UUDEX Exchanges MUST be mutually authenticated between the endpoints of the communication.

COM-4 UUDEX MUST allow UUDEX Participants to establish a UUDEX Tunnel prior to the need to communicate data and be able to keep the UUDEX Tunnel open for as long as these UUDEX Data Elements might be exchanged. Establishing the UUDEX Tunnel might include activities such as contacting, encryption or integrity algorithm negotiation, agreement on encryption keys, and mutual authentication of parties.

COM-5 UUDEX MUST allow any UUDEX Participant to measure the bandwidth and latency between itself and any other UUDEX Participant with which it has authorization to send or receive information.

COM-6 UUDEX MUST support a UUDEX Data Element delivery prioritization scheme. This scheme must include support for UUDEX Data Elements that must be delivered immediately regardless of what other elements are preempted, elements that should only be delivered when the communications channel between UUDEX Participants is otherwise idle, and one or more levels of relative priority that exists between these extremes (e.g., where level 1 preempts level 2 UUDEX Data Elements, and level 2 preempts level 3 UUDEX Data Elements).

COM-6.1 Prioritization is tied to individual UUDEX Data Elements rather than to UUDEX Data Element types. Some UUDEX Data Elements types might always have the same priority, but other UUDEX Data Element types might have elements with different priorities. UUDEX needs to be able to support the latter. That said, messages that control UUDEX behaviors (e.g., UUDEX Configuration Objects) SHOULD be assigned a priority higher than all other data messages, since their successful delivery will impact all other UUDEX behaviors including delivery of all priority-levels of UUDEX Data Elements.

COM-7 All UUDEX Tunnels must use reliable message delivery. This means that the sender of a message will always know if the recipient did not receive a given message, allowing the sender to attempt to resend.

4.5 Data Storage Requirements for UUDEX Servers

The following are data storage requirements for UUDEX Servers:

STO-1 All data stored by UUDEX Servers, including UUDEX Repositories but potentially also including other information such as subscription information, MUST be encrypted while at rest.

STO-2 All data stored by UUDEX Servers MUST include mechanisms to detect data corruption (e.g., checksums or other integrity protections).

STO-3 UUDEX Servers MUST support a mechanism for associating UUDEX ACL Objects with UUDEX Data Elements. UUDEX ACL Objects MUST be protected against data corruption or inadvertent modification.

STO-4 All UUDEX Repositories MUST support a mechanism for storing UUDEX Data Envelopes, including but not limited to timestamp, source, type, retention period, and sensitivity of the data, in conjunction the their associated UUDEX Data Elements. These UUDEX Data Envelopes MUST be protected against data corruption or inadvertent modification.

STO-5 All UUDEX Servers MUST support a mechanism for performing data backup or archiving of UUDEX Repositories. All UUDEX Servers MUST include mechanisms to restore UUDEX Repositories from backup data.

STO-6 UUDEX Servers MUST support a mechanism for controlling access to data to prevent unauthorized parties from modifying the data or metadata in the UUDEX Repository.

4.6 Requirements for Testing

The following requirements relate to the capability of UUDEX to perform testing of UUDEX services without disrupting current operational UUDEX services. While testing new or existing UUDEX services the UUDEX exchange must have the ability to continue normal operation without compromising the capabilities or features of UUDEX components. In this section the conditions in which UUDEX will operate during testing are defined.

UUDEX must allow for a test mode.

TM-1 In test mode, verbose logging messages will report any anomalies or exceptions.

TM-2 Test mode will allow for the transportation and storage of dummy data to test the full capabilities of an operational UUDEX Instance.

TM-3 In test mode, a UUDEX Client, UUDEX Server, UUDEX Consumer, or UUDEX Producer will be monitored to ensure it will not adversely affect the UUDEX Instance.

TM-4 In test mode, a separate, isolated capability will be available to allow UUDEX Exchanges to test functionality separately from production UUDEX Exchanges.

TM-4.1 All data transmitted by parties in an isolated test mode will not be transmitted or received from parties that are not in test mode.

TM-5 All messages will be flagged as a test message from the UUDEX Participants that are in test mode.

TM5.1 It is recommended that UUDEX Clients and UUDEX Servers receiving a test message respond with a message indicating whether expected conditions were met from that test message.

UUDEX must support verbose software logging.

SD-1 Software verbose logging can be enabled while the UUDEX Role is being used operationally or in test mode.

SD-1.2 All logs must be written locally. Logs MAY also be copied to remote locations.

SD-2 When performing verbose logging, each UUDEX Client or UUDEX Server must report on any anomalies detected for each software component.

UUDEX must support verbose UUDEX Data Element logging.

DD-1 Data element verbose logging can be enabled while the UUDEX Role is being used operationally or in test mode.

DD-1.2 All logs must be written locally. Logs MAY also be copied to remote locations.

DD-2 When performing verbose logging, each UUDEX Client or UUDEX Server must report on any anomalous structure or inappropriate values detected for each UUDEX Data Element.

UUDEX must support verbose message logging.

MD-1 Message verbose logging can be enabled while the UUDEX Role is being used operationally or in test mode.

MD-1.2 All logs must be written locally. Logs MAY also be copied to remote locations.

MD-2 When performing verbose logging, each UUDEX Client or UUDEX Server must report on any anomalous structure or inappropriate values detected for each UUDEX Exchange.

4.7 Architectural Requirements

These requirements deal with architectural aspects of UUDEX that must be supported in the design. They focus on capabilities that UUDEX elements must be able to support; actual utilization of many of these features within deployed environments would remain at the discretion of operators.

ARCH-1 UUDEX Servers MUST be capable of supporting redundant deployment, where multiple UUDEX Server implementations (i.e., nodes hosting UUDEX Servers) support a single UUDEX Server role. These redundant UUSEX Servers will all be capable of serving the same UUDEX Client requests and the same UUDEX Repositories will be replicated between them. This provides both service redundancy (allowing other, redundant UUDEX Servers to field requests if one UUDEX Server becomes unavailable) and data redundancy (ensuring that other UUDEX Servers are effectively providing data backup in the case of corruption of one UUDEX Server’s data).

ARCH-1.1 UUDEX Servers SHOULD support distributed redundant deployment, where a set of redundant UUDEX Servers do not need to be geographically co-located.

ARCH-2 All UUDEX Producers MUST be able to add an annotation to the UUDEX Data Envelope with a priority value. This priority value is used to prioritize activities within a UUDEX Instance in the case where demand for UUDEX capabilities is outstripping the UUDEX Instance’s ability to supply those capabilities.

ARCH-2.1 UUDEX Servers MUST be configurable with a policy that compares the assigned priority to other fields of a UUDEX Data Element or UUDEX Data Element Envelope and rejects or downgrades the priorities of UUDEX Data Elements that have been assigned an inappropriate priority. For example, a UUDEX Server could define a policy where certain classes of routine messages are not allowed to be given a high priority, and automatically reject or reduce the priority of such routine messages whose priority has been set too high by their UUDEX Producer. Those who deploy UUDEX Instances are not required to make use of this capability, but UUDEX Server implementations must be capable of letting operators define and enforce such a policy. (This document refers to priorities as high, medium, or low for the sake of examples, but any number or type of levels of prioritization might be employed so long as they are strictly ordered.)

ARCH-2.2 UUDEX Server MUST be able to assign priorities to UUDEX Data Elements based on fields of a UUDEX Data Element or UUDEX Data Element Envelope in the case that the priority is not assigned by the UUDEX Producer. This allows UUDEX Instances to delegate assignment of UUDEX Data Element priorities to UUDEX Servers if the operators wish to do so.

ARCH-3 All UUDEX Roles MUST have the ability to detect network connectivity status of a UUDEX Tunnel between them and any other UUDEX Role to which they might directly communicate.

ARCH-3.1 All UUDEX Roles MUST be able to detect and alarm when they are not able to establish a UUDEX Tunnel between any two UUDEX Participants who are attempting to communicate with each other (e.g., detecting channel break or other disruption detection).

ARCH-3.2 All UUDEX Roles MUST be able to detect and alarm whenever the bandwidth of the UUDEX Tunnel is less than the bandwidth required to deliver the data within the latency requirements of that data (e.g., channel congestion detection).

ARCH-4 All UUDEX Servers MUST be configurable with a prioritization policy. The prioritization policy MUST allow the UUDEX Server to limit query responses, subscription fulfillment, and possibly other client interactions at certain times based on the UUDEX Data Element’s priority level. Specifically, the UUDEX Server can choose to only send UUDEX Data Elements with higher priorities when the policy is executed. This policy MAY contain more than one priority level, enabling to scale the UUDEX Server’s restrictions service use based on appropriate factors (e.g., a minor network constrain might cause the UUDEX Server to only deliver medium or high priority UUDEX Data Elements, while a more significant network constrain might cause the UUDEX Server to only deliver high priority UUDEX Data Elements).

ARCH-4.1 UUDEX Servers MUST be able to execute their prioritization policy on a UUDEX Tunnel by UUDEX Tunnel basis. Thus, at any given time, some UUDEX Tunnels might be constrained by the prioritization policy while others might not.

ARCH-4.2 UUDEX Servers MAY include the ability to automatically engage their prioritization policy based on the state of network connectivity or other factors. Whether or not a UUDEX Server has the ability to automatically set the prioritization level, all UUDEX Servers MUST have the ability to manually set the prioritization level, and the manual level MUST take precedence over the automatic level.

ARCH-5 UUDEX Clients SHOULD have the ability to constrain their behavior based on a prioritization policy. This could include deferring or dropping data publication or requests for data that would be beneath the priority threshold in effect between that UUDEX Client and a given UUDEX Server.

ARCH-6 UUDEX Implementations MUST be able to support fielding multiple simultaneous UUDEX Tunnels between individual UUDEX Participants in order to provide redundancy in the communications fabric. It must be possible for these UUDEX Tunnels to be over different communications media, including mixes of TCP/IP and non-TCP/IP networks.

ARCH-6.1 UUDEX Participants MUST have the ability to automatically switch between different UUDEX Tunnels based on the connectivity status of any link.

ARCH-6.2 UUDEX Participants MUST have the ability to manually switch between different UUDEX Tunnels.

4.8 Data Lifecycle Requirements

All data that are exchanged using UUDEX undergo a common lifecycle of creation, distribution, modification (potentially), and ultimately deletion. The following requirements govern the treatment of data throughout this lifecycle.

DAT-1 All UUDEX Data Elements that are added to a UUDEX Server MUST be assigned a UUID.

DAT-1.1 UUID structure MUST conform to the format and creation rules outlined in RFC 4122 for “name-based” UUIDs.

DAT-1.2 UUIDs are only necessary when data are added to a UUDEX Server. It is possible that the UUDEX Data Element itself might have been created significantly before being added to a UUDEX Server, in which case the UUID is only required to be added at the time the UUDEX Data Element is added to the UUDEX Server. That said, for processing reasons, content creators MAY assign UUIDs to UUDEX Data Elements before they are added to a UUDEX Server, even if there is no guarantee that the UUDEX Data Element would ultimately be added to a UUDEX Server.

DAT-1.3 UUIDs MUST NOT be reused. Even if a UUDEX Data Element is deleted, the UUID associated with the deleted UUDEX Data Element remains forever bound to that UUDEX Data Element and cannot be reassigned.

DAT-1.4 As far as UUIDs are concerned, modification of a UUDEX Data Element creates a new UUDEX Data Element that must be assigned a new UUID.

DAT-1.5 In some cases, the producer of a UUDEX Data Element might wish not to be associated with the UUDEX Data Element. For example, entities wish to submit cyber threat intelligence data regarding a detected intrusion anonymously so they do not reveal they were able to detect the attack. For this reason, the UUID associated with the UUDEX Data Elements does not need to come from the party that produced it. One option to accomplish this includes having a service to which UUDEX Data Element producers can submit data, which will assign a UUID to that UUDEX Data Element and submit it to a UUDEX Server on behalf of the original UUDEX Producer without assigning attribution to the original producer. Another option would be to set up a service by which a UUDEX Producer could request a UUID generated by a third party. Either of these would allow the UUDEX Data Element to enter a UUDEX Server with a UUID that is not associated with the original data producer.

DAT-2 When a UUDEX Data Element is modified, it is equivalent to the creation of a new UUDEX Data Element on the same UUDEX Server that supersedes the original UUDEX Data Element. The UUDEX Server holding these UUDEX Data Elements MUST delete the original UUDEX Data Element, but MUST retain both its UUID and create an association between the original UUDEX Data Element’s UUID and the UUID of the new UUDEX Data Element.

DAT-2.1 Modification of the UUDEX Data Envelope associated with a UUDEX Data Element, including details such as the UUDEX Data Element’s UUDEX ACL Object, do not constitute a change and do not create a new UUDEX Data Element with a new UUID. Thus, a party with appropriate rights can alter the access permissions associated with a UUDEX Data Element without creating a changed UUDEX Data Element.

DAT-2.2 When a UUDEX Client requests a UUDEX Data Element by UUID, and that UUID is associated with a UUDEX Data Element that has been changed (i.e., it was the original UUDEX Data Element that was deleted and replaced by a new, derived UUDEX Data Element), the UUDEX Server MUST treat this as a request for the UUID of the changed UUDEX Data Element (i.e., the new, derived UUDEX Data Element) and resolve the request as such. If multiple changes have occurred, the UUDEX Server might end up traversing multiple UUIDs before finally coming to the latest version of the UUDEX Data Element. In resolving the access control surrounding the request, only the access control in the UUDEX Data Envelope of the latest version of the UUDEX Data Element is considered.

DAT-3 When a UUDEX Client requests a UUDEX Data Element, modification of a UUDEX Data Element, or deletion of a UUDEX Data Element from a UUDEX Server, the UUDEX Server MUST first verify that the UUDEX Identity Element associated with this UUDEX Client request is valid. The UUDEX Server must then compare the identity, roles, or other attributes of the UUDEX Identity Element against the relevant UUDEX Data Element’s access controls to determine whether the action is allowed. Only if the action is permitted in the UUDEX ACL Object will the action proceed.

DAT-3.1 If a UUDEX Client action request (e.g., read, modify, delete, or post) is denied, the message returned by the UUDEX Server MUST NOT leak information to the UUDEX Client. For example, if a UUDEX Data Element’s access controls prevent a particular UUDEX Client from knowing of the existence of that UUDEX Data Element, and the UUDEX Client requests that UUDEX Data Element, the UUDEX Server’s response must be as if the UUDEX Data Element did not exist, rather than reporting that access to the UUDEX Data Element was denied.

DAT-4 UUDEX Servers MAY establish access controls over UUDEX Repositories. This is in addition to access controls that individual UUDEX Data Elements in a UUDEX Repository might include. UUDEX Repository access controls could limit which entities were permitted to post data to that UUDEX Repository. For other actions (e.g., read, modify, delete), a UUDEX Client would need to be granted access both to perform the given action by the UUDEX Repository and by the UUDEX Data Element for the action to proceed.

DAT-5 If a UUDEX Data Element is deleted from a UUDEX Server, the UUDEX Server MUST immediately make the UUDEX Data Element unavailable to all UUDEX Clients and treat any requests for the UUDEX Data Element as being made to non-existent data. The UUDEX Server MUST remove the UUDEX Data Element, including any chain of UUIDs associated with change events, from memory at its earliest convenience. The latter might be later than the former due to needs to ensure the deletion is replicated across all UUDEX Servers in a redundant deployment.

DAT-5.1 UUDEX Servers MAY employ backup systems to protect against data loss in the case of server failure or storage corruption. If such a backup exists, deleted UUDEX Data Elements SHOULD be purged from the backup if and when doing so is possible.

DAT-5.2 It is not the intended role of UUDEX Servers to provide a long-term, historical archive of UUDEX Data Elements. A historical archive might retain deleted UUDEX Data Elements for historical reasons, but that would violate the previous requirements about removing deleted UUDEX Data Elements from memory. If an organization wishes to maintain a historical archive of UUDEX Data Elements on a UUDEX Server, this needs to be done by downloading the UUDEX Data Element from the UUDEX Server to a UUDEX Client, which can then use the downloaded data to create and maintain such an archive.

DAT-6 When a UUDEX Data Element is published to a UUDEX Server, the UUDEX Server MUST store it in the appropriate UUDEX Repository and perform any preprocessing necessary to make the UUDEX Data Element available to the appropriate UUDEX Client requests. This might include indexing the UUDEX Data Element, if possible and permitted by the UUDEX Data Element's access controls, to enable the UUDEX Data Element to be matched against search and UUDEX Subscription requests.

DAT-6.1 All UUDEX Data Elements published to a UUDEX Server MUST (implicitly or explicitly) grant the UUDEX Server the rights to be aware of the UUDEX Data Elements. The UUDEX Server MUST reject requests to publish UUDEX Data Elements that do not grant this right.

DAT-6.2 A UUDEX Server MAY receive UUDEX Data Elements to which it is not granted read access. In this case, the server MUST only use the UUDEX Data Element in the UUDEX Data Envelope for the purpose of storing the UUDEX Data Element in the appropriate UUDEX Repository and for matching against search and UUDEX Subscription requests. If the UUDEX Data Envelope does not have the necessary information to allow the UUDEX Server to do these tasks, the UUDEX Server MUST reject the request to publish the UUDEX Data Element.

DAT-6.3 A UUDEX Server does not require access rights to a UUDEX Data Element regarding a particular action in order to undertake the action as instructed by an authorized UUDEX Client. For example, if an authorized UUDEX Client requests the deletion of a UUDEX Data Element, the UUDEX Server can fulfill that request even if the UUDEX Server does not have access rights to delete the UUDEX Data Element. Thus, with regard to the UUDEX Server and access control of UUDEX Data Elements, those controls describe UUDEX Server processes rather than strictly enforced controls. The UUDEX Server must be trusted not to violate the terms of the access controls, despite the fact that, in practice, it will have the ability to do so.

5.0 UUDEX Messages and Data

The purpose of this section is to provide a very high-level overview of the metadata and message structure used by UUDEX. The intent is to describe the interfaces and infrastructure, while remaining largely agnostic with respect to the data models of UUDEX Data Elements being exchanged. The overall UUDEX message structure is shown in Figure 15.

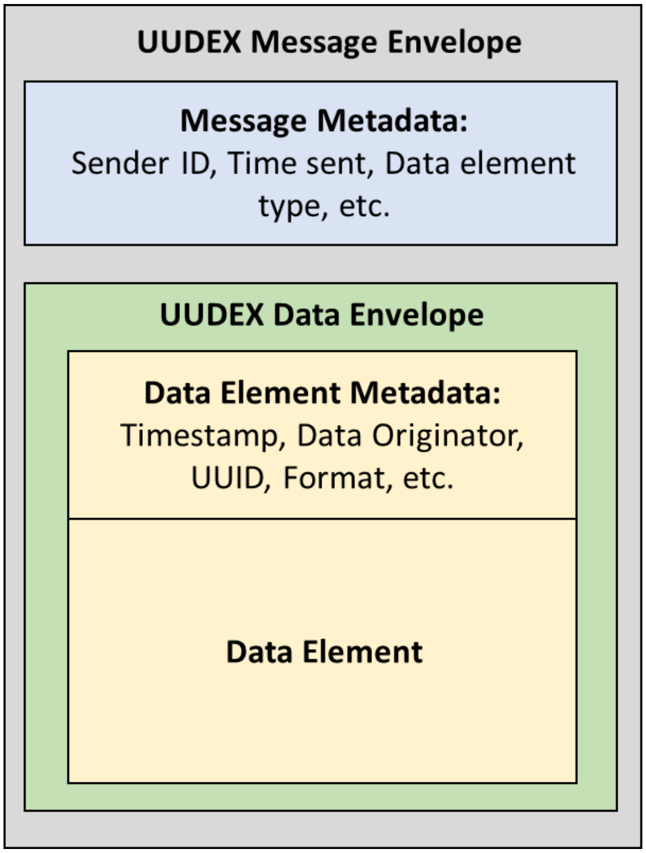

Figure 15: UUDEX Message Structure

5.1 The UUDEX Message Envelope

The UUDEX Message Envelope shown in Figure 15 is used to wrap a UUDEX Data Element and associated UUDEX Data Envelope while they are in transit between UUDEX Participants. Once the message has been received and processed, the UUDEX Message Envelope is discarded (although some fields might be copied into the UUDEX Data Envelope).

The UUDEX Message Envelope is used to assist with the delivery and processing of the UUDEX Date Element and Data Envelope. It contains the message metadata that is related to the flow of the message.

The message metadata minimally includes:

- Sender identifier
- Time sent

- Data element type

Other fields might be included in the UUDEX Message Envelope, such as:

- Special handling instructions (e.g., acknowledgement required)
- Additional security related information
- Additional information used by the UUDEX Server for routing the UUDEX Data Element to the appropriate UUDEX Repository.

The UUDEX Message Envelope is designed to be easily extensible, so additional fields might be specific to a given UUDEX Instance or even a specific UUDEX Tunnel between two UUDEX Participants.

5.2 UUDEX Data Envelope

All UUDEX Data Elements stored in the UUDEX system must be associated with a UUDEX Data Envelope as shown in Figure 15. This UUDEX Data Envelope contains key information needed to support handling of the UUDEX Data Element.

5.2.1 Data Element Metadata

Much of the metadata can be derived by the publish processing on the UUDEX Server (e.g., the storage timestamp can be determined when data are stored into the UUDEX Repository, or the identity of the data publisher can be established when the communication session is established). The minimum metadata includes:

- Timestamp of storage of data into the UUDEX Server Repository.
- The UUDEX Producer is the ultimate source of the data into the UUDEX Server Repository. This field ties the data to a particular organization. Note that the UUDEX Producer that creates the data might be a party other than the UUDEX Producer that submitted the UUDEX Data Element if the source needs to be anonymous.
- A UUDEX ACL Object for the UUDEX Data Element. This UUDEX ACL Object might be assigned by the UUDEX Producer or the UUDEX Server. The UUDEX ACL Object is used by the UUDEX Server when responding to UUDEX query, deletion, or change requests to determine if the action is permitted.
- The UUID associated with the UUDEX Data Element. In most cases this will be assigned by the UUDEX Producer of the UUDEX Data Element, but it might be assigned by a different party.

In addition to the required fields, some of the following metadata might be relevant. Note that additional metadata may be included in the future.

- Sender identifier to indicate the immediate source (i.e., the UUDEX Producer) of a UUDEX Data Element, which might be different from the UUDEX Producer that created the UUDEX Data Element
- UUDEX Data Element type
- Format (e.g., JSON, CSV, XML, PDF, binary)
- Expiration time

- Other security related information
- Data priority, as assigned by the UUDEX Producer or the UUDEX Server

Other metadata fields can be added to the UUDEX Data Envelope. The UUDEX Data Envelope is intended to be extensible, so additional fields might be specific to a UUDEX Instance, specific to certain UUDEX Data Element types, or to individual UUDEX Producers or UUDEX Servers.

5.2.2 UUDEX Data Elements

UUDEX Exchanges involve the use of messages. There are two categories of messages:

1. UUDEX Configuration Object messages for the management of end points and subscriptions
2. Messages that convey all other types of UUDEX Data Elements. Details on the types of UUDEX Data Elements are provided below.

UUDEX can be configured to enable the exchange of an extensible set of UUDEX Data Element types. The types of UUDEX Data Elements would fall into one of a number of categories:

- Time series data, as are commonly conveyed using ICCP where data points are defined for the capture of values and changes over time.
- Structured documents that convey data using a common format (e.g., JSON, XML, CSV) based on some information model such as (but not limited to) the IEC Common Information Model that may be parsed by applications.
- Unstructured documents, which are conveyed using formats such as PDF, JPEG, or binary executable and are not typically parsed by applications.
- Power system network and asset models, which represents the descriptions and relationships of objects that comprise the portions of or changes to the electricity grid.

Given that UUDEX tries to be largely agnostic to the data models used by UUDEX Data Elements, when applicable, different UUDEX Data Element types could be based on different logical information models. The IEC CIM is one example of this.

At the same time there are cases where an underlying data model is very important. This can also be true if the UUDEX Server is expected to perform any processing of UUDEX Data Elements. Down-sampling of time series data is one example of this.

There is also the issue of granularity. A simple example is where a UUDEX Producer may publish a set of generator measurements. It will be possible to impose access controls via a UUDEX ACL Object such that a UUDEX Client may subscribe to or access only a specific subset of those generator measurements that convey the granularity the UUDEX Client requires.

The diagram shown in Figure 16 is a high-level class (type) hierarchy for UUDEX Data Elements. This is used to categorize the different types of UUDEX Data Elements. This hierarchy would be extended by adding types such as OE-417, RCIS, COMTRADE (IEEE Std C37.111), etc. as needed for UUDEX Exchanges. Note that this hierarchy is intended to be informative and is not complete, especially at the lower layers.

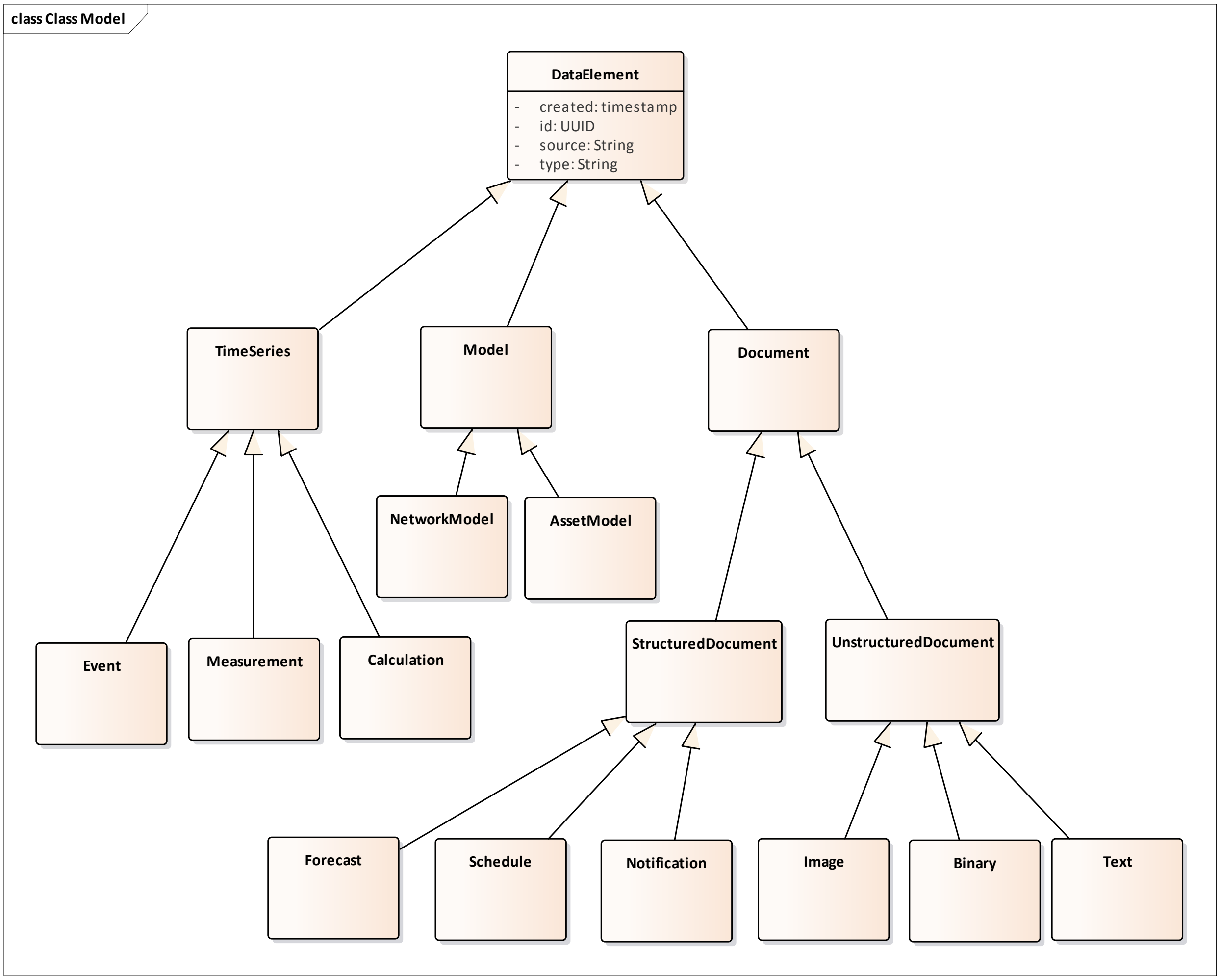

Figure 16: Data Element Hierarchy

6.0 Data Exchange Architectures

This section describes data exchange architectures and data flows.

6.1 Pub-Store-Forward

Figure 17 illustrates the basic sequence of a UUDEX Publish – UUDEX Subscribe information exchange.

In this example, UUDEX Subscriber 1 and UUDEX Subscriber 2 have already established subscription on the UUDEX Server. The UUDEX Data Element published by the UUDEX Producer matches the criteria associated with both subscriptions and the UUDEX ACL Objects associated with the UUDEX Data Element and UUDEX Repository in which it is stored allow both subscribers to retrieve the data.

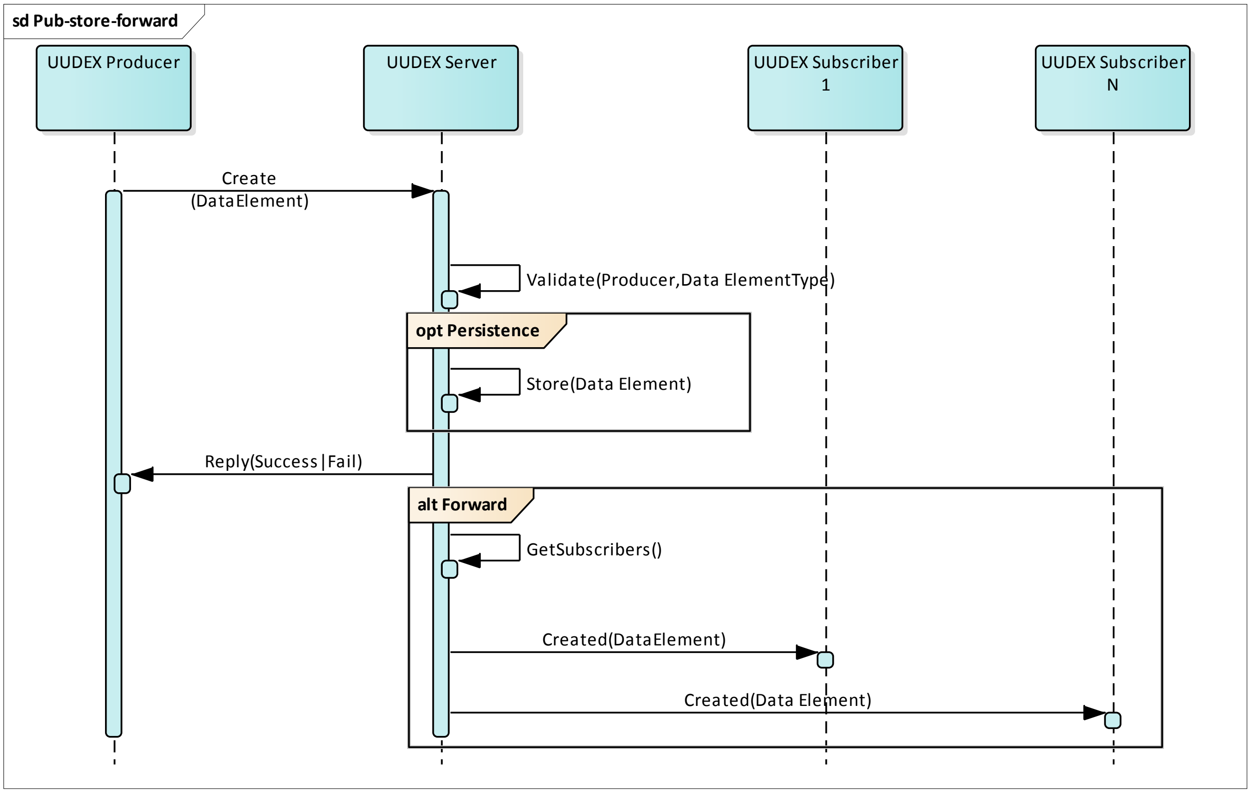

Figure 17: PUB-SUB Store-forward

6.2 Pub-Store-Notify

Figure 18 describes the sequence of a UUDEX Publish and notify information exchange. In this pattern, the published data are stored on a UUDEX Server and a notification is issued to potentially interested UUDEX Consumers. Upon receipt of the notification the UUDEX Consumer can then decide to retrieve the information from the UUDEX Server at a convenient time.

In this example, both UUDEX Subscriber 1 and UUDEX Subscriber 2 have established subscriptions on the UUDEX Server. As before, the UUDEX Data Element published by the UUDEX Producer matches the criteria of these subscriptions and is accessible to them. However, instead of sending the UUDEX Data Element itself in fulfillment of the matching subscriptions, the UUDEX Server instead sends a UUDEX Manifest that identifies the new UUDEX Data Element to both UUDEX Subscribers. Such UUDEX Manifests are smaller than the UUDEX Data Elements they identify. In this case, both UUDEX Subscribers receive the UUDEX Manifest alerting them to the new UUDEX Data Element. UUDEX Subscriber 1 decides not to retrieve the UUDEX Data Element. UUDEX Subscriber 2 decides to retrieve the UUDEX Data Element and sends a query to the UUDEX Server that contains the UUID for that UUDEX Data Element as specified in the UUDEX Manifest it received. The UUDEX Server processes this request and returns the requested UUDEX Data Element.

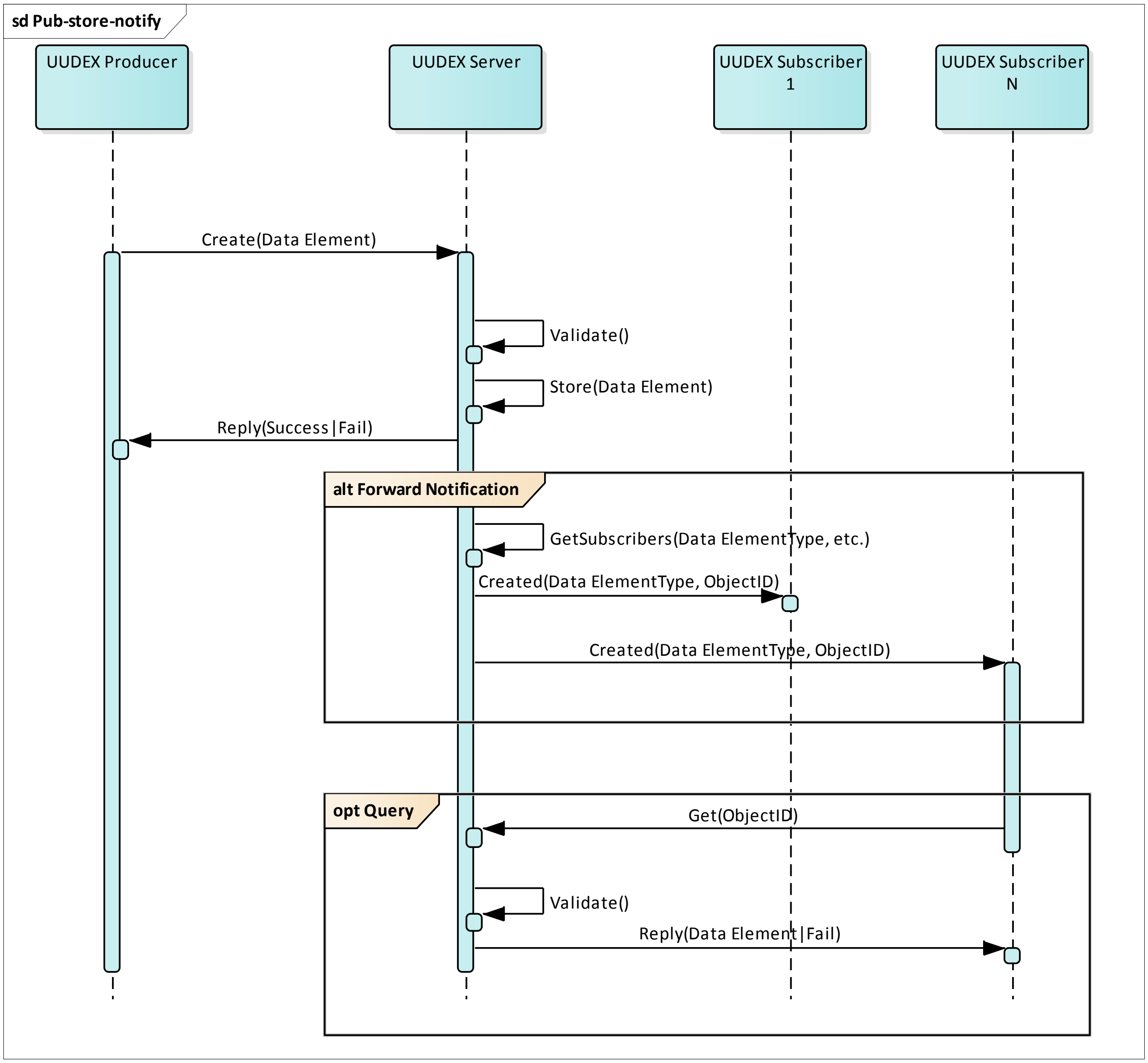

Figure 18: PUB Store Notify

6.3 Query

The sequence diagram shown in Figure 19 describes a UUDEX Client querying the UUDEX Server for information. Provided that the UUDEX Client is authorized for the specific type of information as defined by the UUDEX ACL Object, the query can be honored by the UUDEX Server.

In this case, the UUDEX Client sends a query to the UUDEX Server asking for UUDEX Data Elements that match a certain set of criteria. Criteria could include data type, time the data were submitted, source of the UUDEX Data Element, etc. The request is validated and, if correct, the collection of matching UUDEX Data Elements where the associated UUDEX ACL Objects allow the UUDEX Client to retrieve them are compiled and returned to the UUDEX Client.

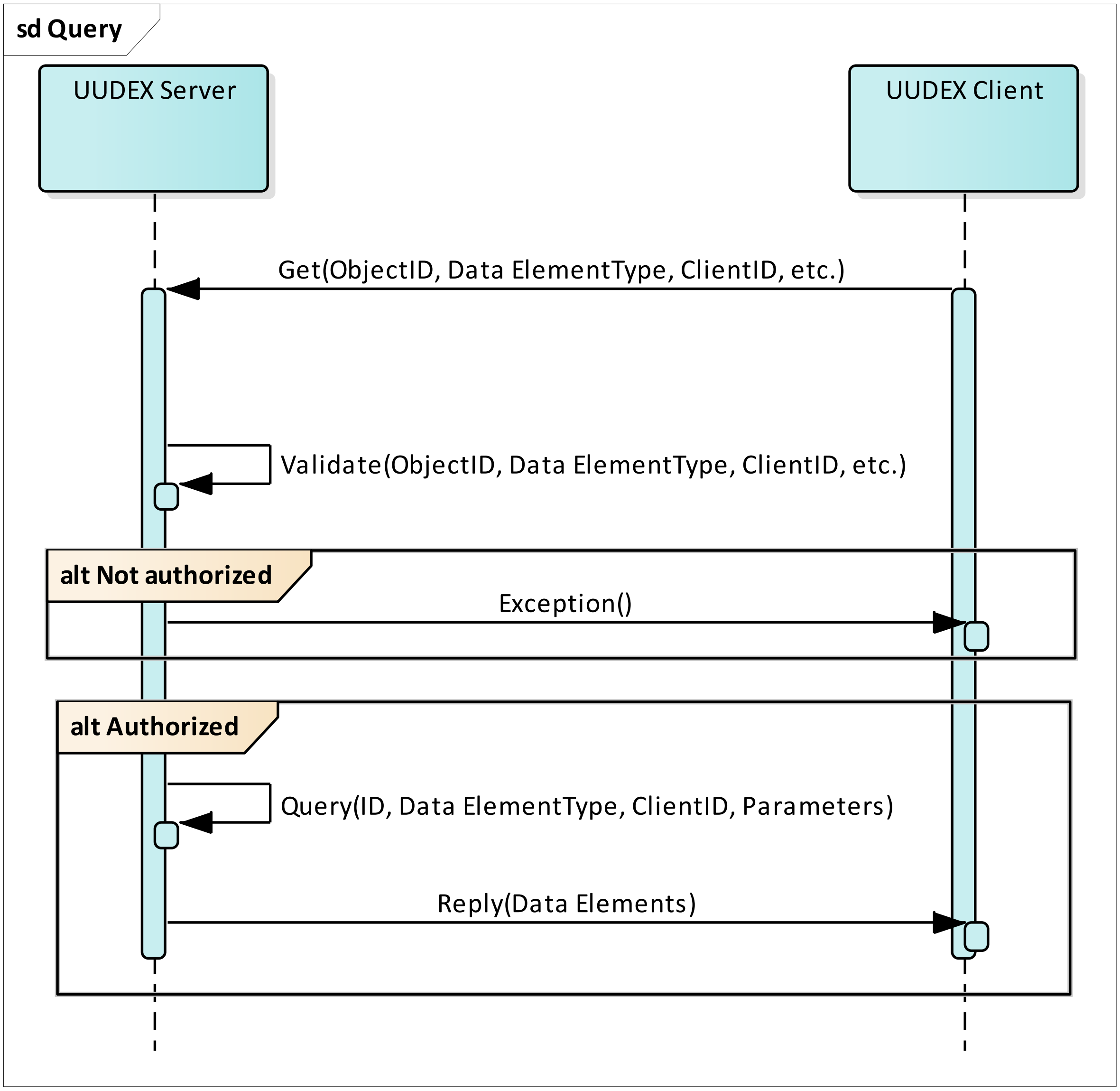
sd Query
UUDEX Server
UUDEX Client
Get(ObjectID, Data ElementType, ClientID, etc.)
Validate(ObjectID, Data ElementType, ClientID, etc.)
alt Not authorized
Exception()
alt Authorized
Query(ID, Data ElementType, ClientID, Parameters)
Reply(Data Elements)

Figure 19: Query

6.4 Set ACL

The sequence diagram in Figure 20 shows a producer of information creating, updating or deleting a UUDEX ACL Object for a given type of information. The UUDEX ACL Object is persisted by the UUDEX Server and used to validate UUDEX Consumer subscriptions and UUDEX Consumer query requests. The UUDEX ACL Objects can be defined to permit or prohibit access to certain types of UUDEX Data Elements by different UUDEX Clients, with consideration given to both the UUDEX Producer and potential UUDEX Consumer of a given UUDEX Data Element type.

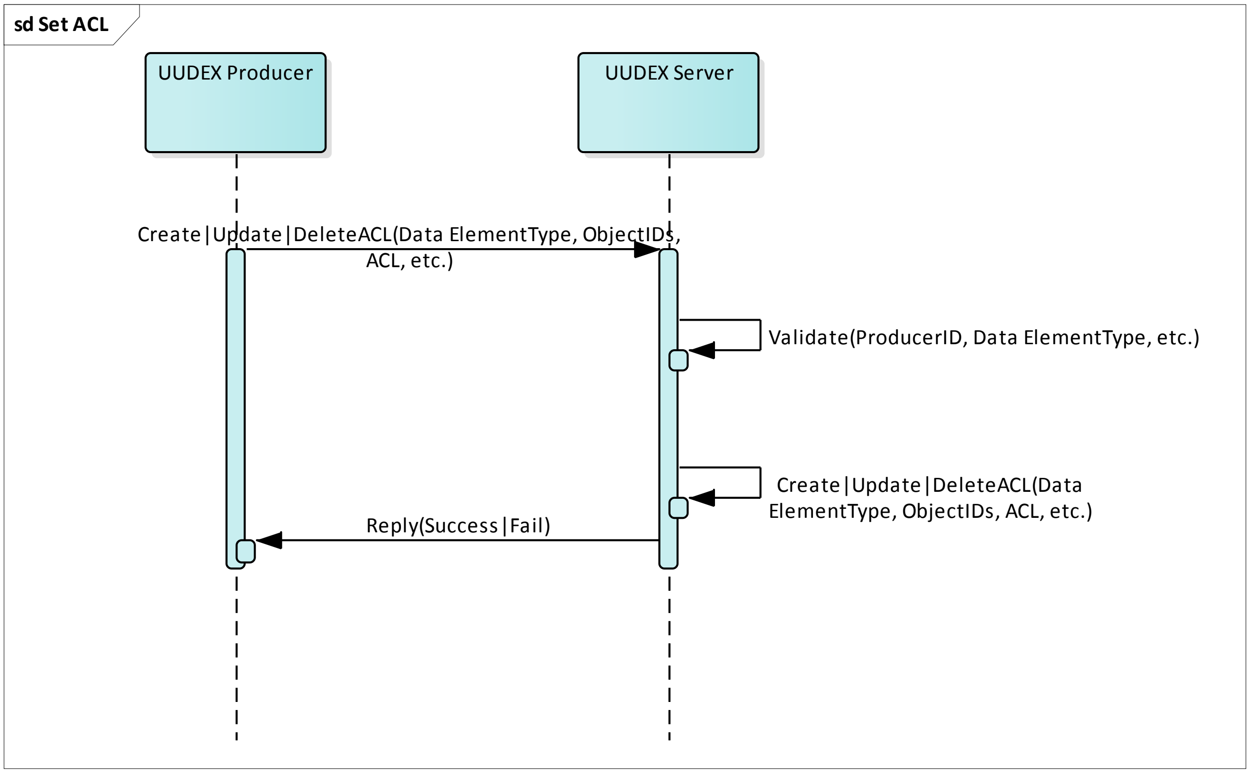
sd Set ACL
UUDEX Producer
UUDEX Server
Create|Update|DeleteACL(Data ElementType, ObjectIDs, ACL, etc.)
Validate(ProducerID, Data ElementType, etc.)
Create|Update|DeleteACL(Data ElementType, ObjectIDs, ACL, etc.)
Reply(Success|Fail)

Figure 20: Set ACL

6.5 Subscribe

The sequence diagram shown in Figure 21 illustrates a UUDEX Client subscribing for a given type of UUDEX Data Element. The subscription is managed by the UUDEX Server, which acts as an intermediary for information exchanges.

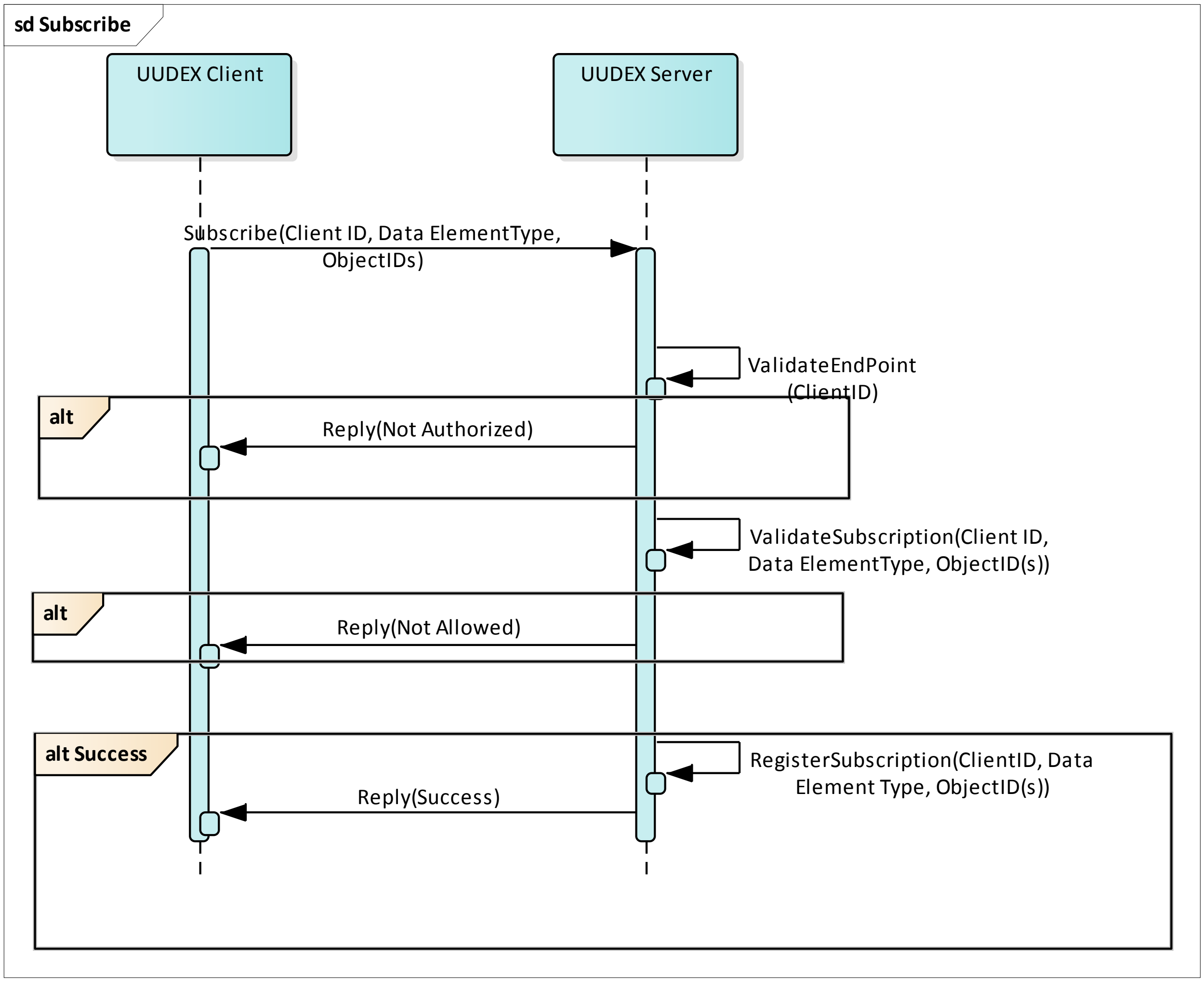
sd Subscribe
UUDEX Client
UUDEX Server
Subscribe(Client ID, Data ElementType, ObjectIDs)
ValidateEndPoint (ClientID)
alt
Reply(Not Authorized)
ValidateSubscription(Client ID, Data ElementType, ObjectID(s))
alt
Reply(Not Allowed)
alt Success
RegisterSubscription(ClientID, Data Element Type, ObjectID(s))
Reply(Success)

Figure 21: Subscribe

7.0 UUDEX Security Considerations

This section considers the security aspects of the UUDEX Framework. In particular, it identifies security risks and security trust relationships within UUDEX.

7.1 Security Risks

This section identifies key security risks within the UUDEX Framework. A security risk includes both risks imposed by security threats as well as concerns with security implications regardless of whether or not they result from malicious actors. This list is not exhaustive and other risks might exist within specific contexts. However, the list below identifies common, significant risks that any UUDEX deployment will face.

7.1.1 Information Disclosure

Many types of UUDEX Data Elements exchanged over UUDEX are sensitive and disclosure to unauthorized parties can result in damage to an organization or disruption of services. Examples of sensitive data include:

- Details about the configuration of IT assets, particularly details about IT security systems that could be used by malicious parties to plan cyberattacks.
- Disclosures of compromise of IT assets that could impact an organization’s reputation and be used by malicious parties to plan cyberattacks.
- Market-sensitive data that could give competitors, customers, or suppliers an unfair market advantage.
- Data from third parties to whom the recipient has a contractual obligation to secure the data from disclosure. Possible consequences of the disclosure of these data include loss of future access to the data or financial penalties.

To guard against unauthorized disclosure, data must be protected against disclosure both when at rest and at transit. In addition, all interactions with the data need to be constrained by ACLs to ensure that only authorized parties are able to view the data.

Note that it is not only the data that require protection from disclosure. Metadata linked to a UUDEX Data Element must also be carefully managed to prevent unauthorized disclosure. For example:

- The fact that a given entity created a cyber incident report would indicate that entity experienced and detected a cyberattack, even if the incident report could not be read. This could have reputational impact on the report creator and could also tip off the attacker that their efforts had been discovered.
- The UUDEX ACL Object of market-sensitive data could reveal an entity’s commercial partners in an economic transaction, even if the data were not readable. This could give a competitor unfair leverage in the market.
- In some cases, the mere existence of some types of UUDEX Data Elements might reveal sensitive information about an entity or the state of the grid. For example, certain classes of alerts might be sensitive because they direct operational changes and the issuance of such directives reveal a lot about the overall state of the power grid in ways that adversaries could use to their advantage.

For these reasons, interactions with data cannot leak information about associated metadata. In some cases, there needs to be no way to distinguish between a request that fails because access to the UUDEX Data Element was denied and a request that fails because the requested UUDEX Data Element does not exist. These and other error conditions need to be reviewed to ensure that they do not expose information to unauthorized parties.

7.1.2 Information Corruption

UUDEX does not support the exchange of control instructions, that is commands that directly manipulate the behavior of devices in the infrastructure. However, system operators and components will still be using information transported over UUDEX to make critical decisions that impact the functioning of their software and physical assets. For these reasons, it is critical that information stored in UUDEX Repositories and transported over UUDEX Tunnels be protected against corruption, both deliberate and accidental.

Information corruption covers both changes to a UUDEX Data Element's meaning (such as attackers changing an off indicator to an on indicator) as well as changes that render the data unintelligible. These different situations are likely to have different causes (e.g., the first is likely due to deliberate malicious activity, while the latter might be accidental), but both types of corruption can be damaging and must be addressed.

Data need to be protected from corruption both when at rest (i.e., when stored in a UUDEX Repository on a UUDEX Server) and when in motion (i.e., when transmitted from one UUDEX Participant to another using a UUDEX Tunnel). Ideally, such protections will prevent the corruption from occurring in the first place. At the very least, protections need to exist such that any data corruption will be detectable. This is necessary because small data corruptions, which could have significant operational impact, might not be detectable by all UUDEX Participants (e.g., a false 10-degree voltage phase angle shift reported by PMU data).

Another aspect of defending against information corruption involves ensuring that all parties modifying data through UUDEX-defined mechanisms are authorized to perform the actions they are taking.

Protection of the data includes protection of metadata surrounding the data in the UUDEX Data Envelope. The metadata associated with a UUDEX Data Element impacts how data are discovered, organized, identified, and accessed. As such, corruption of the UUDEX Data Envelope could result in multiple issues, including information leakage (if corrupted information allows broader access or discovery than intended) or denial of data (if corrupted information means the data can no longer be found). For these reasons, the UUDEX Data Envelope needs to receive the same protections against corruption as does its associated UUDEX Data Element.

7.1.3 Denial of Service

Entities that use UUDEX services will rely on it for critical information and communications central to their operations. As such, loss of these services, through accident or malice, could disrupt those operations. Hence, UUDEX will need to include mechanisms to reduce the chance that it can be used to deny participants necessary services.

One of the key services that UUDEX provides is support for UUDEX Clients retrieving information from UUDEX Servers. Denial of these services could come in many ways including, but not limited to:

- Loss of or congestion in the UUDEX Tunnel between the UUDEX Client and UUDEX Server.
- Rendering the UUDEX Server unavailable or unable to adequately respond to UUDEX Client requests.
- Corrupting or deleting the UUDEX Data Element the UUDEX Client is requesting, either on the UUDEX Server or when it is in transit to the UUDEX Client.
- Altering access rights on the UUDEX Data Element, UUDEX Data Envelope, or UUDEX Repository so the UUDEX Server's access control mechanisms prevent access.
- Corruption or deletion of UUDEX Consumer subscriptions established by the UUDEX Client on the UUDEX Server such that the UUDEX Client is not alerted to the publication of relevant UUDEX Data Elements.
- Corruption of the UUDEX Server's indexing or search functionality, causing requests by the UUDEX Client to fail to find necessary UUDEX Data Element.

Implementation of software products that serve UUDEX Roles will need mechanisms to mitigate the chance that access to necessary UUDEX Data Elements will be denied due to any of these circumstances.

Similarly, UUDEX Clients will depend on UUDEX Servers to deliver UUDEX Data Elements the UUDEX Client supplies to the appropriate UUDEX Consumers. Specifically, the message needs to be sent to a UUDEX Server from a UUDEX Producer and then made available to the appropriate UUDEX Consumers. This service could be denied by events similar to those listed above:

- Loss of or congestion in the UUDEX Tunnel between the UUDEX Publisher and UUDEX Server or loss of connection between the UUDEX Server and one or more valid UUDEX Consumers for the given data.
- Rendering the UUDEX Server unavailable or unable to adequately receive new UUDEX Data Elements or to respond to UUDEX Client requests.
- Corruption or deletion of the UUDEX Data Elements, either on the server or in transit from the UUDEX Publisher or to the UUDEX Consumer.
- Altering the access rights on the UUDEX Repository to prevent the UUDEX Publishers from posting to the appropriate UUDEX Repository.
- Altering access rights on the UUDEX Data Element or UUDEX Repository so the UUDEX Server's access control mechanisms prevent access by legitimate UUDEX Consumers.
- Corruption or deletion of subscriptions from UUDEX Consumers on the UUDEX Server such that those UUDEX Consumers are not alerted to the publication of the UUDEX Data Element.
- Corruption of the UUDEX Server's indexing or search functionality, causing requests by the UUDEX Consumers to fail to find the posted UUDEX Data Element.

Again, implementations of UUDEX Roles will need to be able to mitigate these types of threats to minimize the chance that necessary services are lost.

7.1.4 Identity Spoofing

Key to any access control strategy is the ability to authenticate actors so that requested actions can be compared against controls. If one entity can masquerade as another then access controls can be circumvented. This, in turn, could lead to many of the aforementioned issues, including, but not limited to, information disclosure, unauthorized data manipulation, insertion of false data under the identities of trusted parties or services, and denial of service.

There are several ways identity information could be spoofed. These include, but are not limited to:

- Inadequate vetting of parties requesting UUDEX Identity Objects. This occurs when a party provides false information when it requests a UUDEX Identity Object. It might claim to be a different party or might claim roles or attributes that it should not be granted in its UUDEX Identity Object.
- Falsifying identity evidence in messages. Some types of network attacks might allow valid credentials from one entity to be copied and used by another entity to pose as the former.
- Stolen private identity information. Identity evidence is often supported by calculations that only the valid holder of that identity could perform. However, it attackers were able to steal the secrets that allowed those calculations, they could pose as the entity.

Depending on the specific mechanisms UUDEX uses for managing identities, other attacks might also be possible. UUDEX implementations will need to include mechanisms to protect against these threats.

There may be situations in which suppliers of UUDEX information need to remain anonymous. For example, reporters of attack details often wish not to be identified. The need to support anonymity, at least anonymity relative to certain other UUDEX Participants, does not negate the need for strong identity controls. Mechanisms will be needed so participants are anonymous with regard to certain UUDEX Participants, but can still be identified by other, trusted parties.

7.2 UUDEX Trust Relationships

Trust relationships exist where parties must rely on others to conform to certain behaviors in the absence of any way to enforce those behaviors. This section outlines the key trust relationships in UUDEX.

7.2.1 UUDEX Servers

UUDEX Servers receive and act upon instructions from UUDEX Clients. They also are responsible for storing UUDEX Data Elements and delivering them as appropriate. UUDEX Servers are trusted as follows:

- To enforce security policies on UUDEX Data Elements with regard to requests to post, read, modify, and delete UUDEX Data Elements. This includes being trusted to enforce their own access rights to the UUDEX Data Elements (i.e., the UUDEX Server is trusted not to read UUDEX Data Elements to which it does not have read access).
- To accurately process queries for UUDEX Data Elements. This means they must correctly identify matching UUDEX Data Elements to which a requesting party has access and accurately respond to the requestor based on this information.

- To accurately maintain and serve subscriptions established by UUDEX Clients.
- Not to add, modify, or delete UUDEX Data Elements except at the direct instruction of an authorized UUDEX Client. This is the case even if the UUDEX Server is granted access rights to perform these activities.
- To execute commands from authorized UUDEX Clients (e.g., if a UUDEX Server is instructed to delete a UUDEX Data Element by an authorized UUDEX Client, the server is trusted to perform that action).
- To accurately report its status (e.g., whether its services are currently degraded).
- To conform to behaviors dictated by prioritization policies.

7.2.2 UUDEX Clients

UUDEX Clients issue commands to UUDEX Servers to post, modify, retrieve, and delete UUDEX Data Elements. UUDEX Clients are trusted as follows:

- To adequately protect UUDEX Data Elements they retrieve from UUDEX Servers. In particular, they are trusted not to disclose the UUDEX Data Element (intentionally or unintentionally) to parties that are not authorized to view the UUDEX Data Element.
- Not to send false information in UUDEX Data Elements.
- Not to create undue communications load by sending excessively large amounts of UUDEX Data Elements to UUDEX Servers.
- Not to create undue processing loads on UUDEX Servers by making excessive UUDEX Query or UUDEX Subscribe requests.

7.3 UUDEX Access Control Overview

This section is intended to provide an overview of the role and scope of access control within the UUDEX Framework.

- Establishing UUDEX Tunnels – All UUDEX Tunnels are required to be mutually authenticated. UUDEX Participants MAY control access at this stage. In this case, a UUDEX Participant would have a list of identities that might legitimately establish a UUDEX Tunnel with it and automatically deny any request to establish a UUDEX Tunnel by a UUDEX Participant not on the list. Note that this list would only constrain inbound connection attempts. A given UUDEX Participant could establish a UUDEX Tunnel with another UUDEX Participant even if the latter would not be permitted to establish an inbound connection. Figure 22 summarizes connection-based access control behaviors.

  This said, it is envisioned that access control at network connection would be a minority situation (with the exception of the policy to automatically deny tunnel requests from parties without a valid UUDEX Identity Object), and most access control would be at the data level.
- A UUDEX Server MAY limit the UUDEX Producers that are allowed to publish to its UUDEX Repositories. Moreover, if the UUDEX Server hosts multiple UUDEX Repositories, it MAY specify that some UUDEX Producers are allowed to publish to some UUDEX Repositories but not to others.

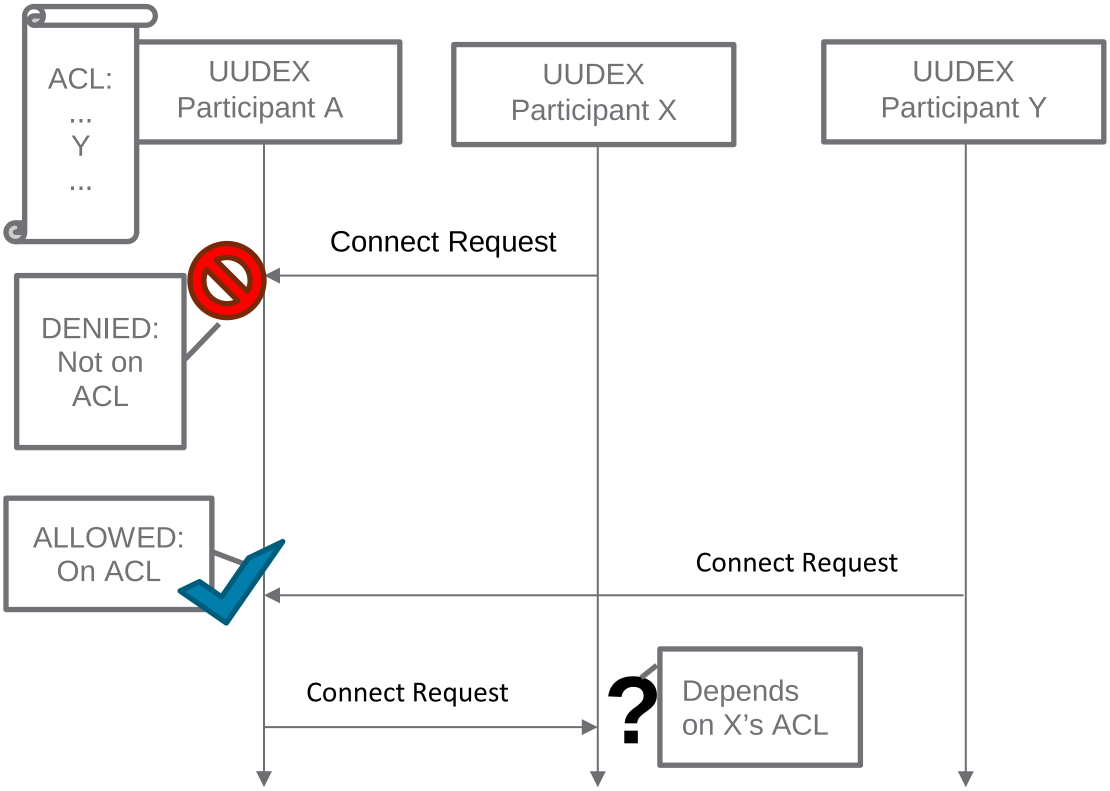

Figure 22: Access Control Overview

- When a UUDEX Producer sends a UUDEX Data Element to a UUDEX Server, they MAY include an associated UUDEX ACL Object with that UUDEX Data Element.
  - If no UUDEX ACL Object is included, the UUDEX Server receiving the UUDEX Data Element will dictate the distribution policy.
  - A UUDEX Server MAY choose to reject UUDEX Data Elements that have associated UUDEX ACL Objects (i.e., the UUDEX Server does not allow UUDEX Producers to dictate redistribution policies and instead requires the UUDEX Producer to delegate that role to itself). Note that, in this case, the UUDEX Server MUST NOT accept the UUDEX Data Elements and just ignore the UUDEX ACL Objects. It needs to reject the submission entirely.
  - If the UUDEX Producer includes UUDEX ACL Objects with the UUDEX Data Element, and the UUDEX Server accepts the submission, then the UUDEX Server is required to honor the UUDEX ACL Objects when redistributing the UUDEX Data Element. In most cases, it will be desirable to grant the UUDEX Server the right to read the content to allow the UUDEX Data Element to be indexed for delivery.
- UUDEX Data Elements in a UUDEX Repository can be controlled individually or by UUDEX Repository (e.g., one could have access to individual UUDEX Data Elements or to all UUDEX Data Elements in a UUDEX Repository).
- Access to UUDEX Data Elements can be by individual, role, or a combination of the two elements. Both individual identities and associated roles are included in UUDEX Identity Objects. Boolean combinations of identities, attributes, and Boolean expressions determine the final list of parties allowed to access information.

8.0 References

Use Chicago Author-Date format and the “reference” Word style. Right-click and choose open hyperlink to view the style guide.

Verify references are still needed and referenced in the document

Internet Engineering Task Force Requests for Comment:

RFC 959, “File Transfer Protocol”

RFC 2597, “Assured Forwarding Per-Hop Behavior Group”

RFC 3246, “An Expedited Forwarding Per-Hop Behavior)

RFC 3986, “Uniform Resource Identifier: Generic Syntax”

RFC 4180, “Common Format and MIME Type for Comma-Separated Values Files”

RFC 4122, “A Universally Unique Identifier URN Namespace”

RFC 5321, “Simple Mail Transfer Protocol”

RFC 7158, “The JavaScript Object Notation (JSON) Data Interchange Format”

International Electrotechnical Commission documents

IEC 61970, “Common Information Model / Energy Management”

IEC 61968, “Common Information Model / Distribution Management”

IEC 60870-6, “Telecontrol equipment and systems - Part 6: Telecontrol protocols compatible with ISO standards and ITU-T recommendations”. Also known as Inter-Control Center Communications Protocol or Telecontrol Application Service Element 2 (TASE.2).

Institute of Electrical and Electronics Engineers

IEEE Std C37.111, “IEEE Standard Common Format for Transient Data Exchange (COMTRADE) for Power Systems

IEEE Std C37.118.2, “IEEE Standard for Synchrophasor Data Transfer for Power Systems”

IEEE Std P2664, Proposed “IEEE Standard for Streaming Telemetry Transport Protocol”

North American Electric Reliability Corporation Standards

EOP-004, “Event Reporting”

CIP-008, “Cyber Security — Incident Reporting and Response Planning”

International Standards Organization Standards

ISO/IEC 7498, “Open Systems Interconnection”

ISO 32000, “Document management -- Portable document format (PDF)”

ISO 9506, “Manufacturing Message Specification (MMS)”

ISO 8601, “Data elements and interchange formats – Information interchange – Representation of dates and times”

ISO/IEC 10918, “Information technology -- Digital compression and coding of continuous-tone still images (JPEG)”

World Wide Web Consortium

“Simple Object Access Protocol (SOAP)

“eXtensible Markup Language (XML)”

European Computer Manufacturers Association

ECMA 404, “The JSON (JavaScript Object Notation) Data Interchange Syntax”

U.S. Department of Energy

“Electric Emergency Incident and Disturbance Report (Form OE-417)”

Organization for the Advancement of Structured Information Standards

“Security Assertion Markup Language (SAML)”

OTHER

“Structured Threat Information eXpression (STIX™)”

“Trusted Automated eXchange of Indicator Information (TAXII™)”

“Packet Capture”

OpenAuth

OpenID Connect

AzureAD

AWS IAM with Multi-factor Authentication

“Traffic Light Protocol”

Appendix A – Data Characteristics

	ICCP Data	RCIS	Power System Models	PMU	OE-417 Report	Files (COMTRADE, others)	Market Data	Asset Management	Cyber Incident Reporting (IDS/IPS logs, PCAP, etc.)	Indicator of Compromise sharing	Guidance (firewall, configuration, etc.)	Patch notification	Vulnerability or Threat notification (STIX, DOE, NERC, DHS) (Non-public)	Vulnerability or Threat notification (CVE, STIX, etc.) (public)
Additional security	Org role, ACL	Org role, ACL	Org role, ACL			File type, org role, ACL	Yes	org role, ACL	Yes – confidentiality, destination authentication	Yes – Integrity, confidentiality, mutual authentication	Yes – integrity, source authentication	Yes – integrity, source authentication	Yes – confidentiality, mutual authentication, integrity	Yes – integrity
Permissible delay between transmission and reception of the data set	Near R/T	Near R/T	minutes	Near R/T	minutes	Seconds-minutes	Near R/T	Minutes	Minutes	Minutes	Minutes	Minutes	Minutes	Minutes
Durability		Yes	Yes		Yes	Yes	Yes	Yes						
Retention or expiration	Ephemeral	Persistent	Persistent	Ephemeral except post event	Persistent	Persistent	Varied	Persistent	Persistent	Persistent	Persistent (short term)	Temporary	Persistent (short term)	Persistent (short term)
Core, desired or optional?	Core	Core	Desired	Core	Core	Optional	Optional	Optional	Desired	Desired	Desired	Desired	Core	Desired
Frequency of transmission	2 seconds+	Ad-hoc	Ad-hoc, periodic	stream	Ad-hoc	Ad-hoc, daily+	hourly, daily, transaction	Ad-hoc	Ad-hoc (weekly?)	Periodic (daily); ad-hoc for high-priority	Periodic (weekly); ad-hoc for high-priority	Periodic (weekly); ad-hoc for high-priority	ad-hoc	Periodic (weekly); ad-hoc

	ICCP Data	RCIS	Power System Models	PMU	OE-417 Report	Files (COMTRADE, others)	Market Data	Asset Management	Cyber Incident Reporting (IDS/IPS logs, PCAP, etc.)	Indicator of Compromise sharing	Guidance (firewall, configuration, etc.)	Patch notification	Vulnerability or Threat notification (STIX, DOE, NERC, DHS) (Non-public)	Vulnerability or Threat notification (CVE, STIX, etc.) (public)
Subscribable?	By point, point type, publisher	Yes	By model profile, by publisher	By point	Yes	By file type, by publisher	By object type, by publisher	By asset type, by publisher	Reported to designated authority	Yes (by multiple parameters)	Yes – by device covered	Yes – by device covered	Yes (by multiple parameters)	Yes (by multiple parameters)
Persistent publisher connection required?	Yes	No?	No	Yes	No	No	Yes?	No	No	No	No	No	No	No
Persistent subscriber connection required?	Yes	No?	No	Yes	No	No	Yes?	No	No	No	No	No	No	No
Stored for future use?	Recent history	Yes	Yes	Selectively; Recent snapshots	Yes	Yes	Recent history	Yes	Yes	Yes	Yes (short term)	Unlikely	Possibly	Possibly
Intermediate processing (e.g., down sample)	Down sample, Periodic snapshot, significant change, event detect	Store and forward	Store and notify	Down sample, Periodic snapshot, significant event detect	Store and forward	Store and notify	Store and notify	Store and notify	Probably: store, augment, anonymize, redistribute	Possibly: store, augment	Unlikely	Unlikely	Possibly – store, augment, anonymize, redistribute	Possibly – store, augment, anonymize, redistribute
Sender down sample (e.g., for degraded link performance)	Yes	No	Directory, change log	Yes	Directory, log	Directory, log	Directory, log	Change log	Unlikely	Unlikely	No	No	No	No
Priority of Message				Event-dependent										

	ICCP Data	RCIS	Power System Models	PMU	OE-417 Report	Files (COMTRADE, others)	Market Data	Asset Management	Cyber Incident Reporting (IDS/IPS logs, PCAP, etc.)	Indicator of Compromise sharing	Guidance (firewall, configuration, etc.)	Patch notification	Vulnerability or Threat notification (STIX, DOE, NERC, DHS) (Non-public)	Vulnerability or Threat notification (CVE, STIX, etc.) (public)
QOS (Traffic Prioritization)	High (?)	High		Event-dependent	High				Med	Med (periodic) or high (ad-hoc)	Med	Med	High	Med
Logical Topography	P2P, Pub/Sub	Pub/Sub	Pub/Sub	Peer-to-Peer, Pub/Sub	Pub/Sub	Pub/Sub	Pub/Sub	Pub/Sub	Hub-spoke	Pub/Sub	Pub/Sub	Pub/Sub	Hub-spoke	Pub/Sub
Minimum expected bandwidth (?)									MBs	MBs	MBs	MBs	MBs	MBs
Size of logical message / “data set”	Continuous, MBs	<1k	Large	Continuous	Small / medium	Large	Varied	Varied	10K-100MB	10K-100MB	10K-10MB	1MB-100MB	10K-100MB	10K-100MB
Originator Control required?	Yes			Yes					Yes – sensitive info included	Yes	Maybe	No	Yes	No
Recipient designations	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL; BAs, RCs	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL	Subscriptions as permitted by role or ACL	Yes (to authority)	Subscriptions as permitted by role or ACL	Subscriptions	Subscription	Subscriptions as permitted by role or ACL	Subscription
Special Handling Instructions					Yes				Yes	Yes	No	No	Yes	No
Sensitivity marking	Yes?	Yes	Yes		Yes	By file type	By object type		Yes	Yes	Maybe (unlikely)	No	Yes	No

	ICCP Data	RCIS	Power System Models	PMU	OE-417 Report	Files (COMTRADE, others)	Market Data	Asset Management	Cyber Incident Reporting (IDS/IPS logs, PCAP, etc.)	Indicator of Compromise sharing	Guidance (firewall, configuration, etc.)	Patch notification	Vulnerability or Threat notification (STIX, DOE, NERC, DHS) (Non-public)	Vulnerability or Threat notification (CVE, STIX, etc.) (public)
(Maximum) Sensitivity level	Different point types could have different sensitivity					By file type	By object type		Major org/ op impact	Private	Private	Public	Major org/ op impact	Public
Integrity sensitivity	High	Moderate	Moderate (can be independently validated before use)	Could have minor organizational/ operational impact; Timing data must be intact	Moderate	High	High	Moderate	Moderate to low	Moderate	High	High	High	Moderate
Reasonable number of recipients	0-20	>100	>100	0-10	>100	>100	>100	>100	0-20 (at least initially)	>>100	>>100	>>100	>>100	>>100
Trigger for sending	Periodic, on change	On create	On change or run	Periodic, stream; upon request	On create	On create, on update	On create	On create, on update	Automated or manual creation	All	All	All	All	All
Trigger for requesting			Notification receipt	A disturbance or event		Notification receipt	Notification receipt	Notification receipt	N/A (not requested from source)	Manual, periodic, or subscription	Manual, periodic, or subscription	Manual, periodic, or subscription	Manual, periodic, or subscription	All

Pacific Northwest National Laboratory

902 Battelle Boulevard
P.O. Box 999
Richland, WA 99352
1-888-375-PNNL (7665)

www.pnnl.gov